applied sciences



Special Issue Reprint

Engineering Applications of Hybrid Artificial Intelligence Tools

Edited by
Zbigniew Gomółka, Damian Kordos and Ewa Dudek-Dyduch

mdpi.com/journal/applsci

MDPI

Engineering Applications of Hybrid Artificial Intelligence Tools

Engineering Applications of Hybrid Artificial Intelligence Tools

Guest Editors

Zbigniew Gomółka
Damian Kordos
Ewa Dudek-Dyduch

MDPI

Basel • Beijing • Wuhan • Barcelona • Belgrade • Novi Sad • Cluj • Manchester

Guest Editors

Zbigniew Gomółka
Faculty of Exact and Technical Sciences
University of Rzeszow
Rzeszow
Poland

Damian Kordos
Faculty of Mechanical Engineering and Aeronautics
Rzeszow University of Technology
Rzeszow
Poland

Ewa Dudek-Dyduch
Faculty of Electrical Engineering, Automatics, Computer Science, and Biomedical Engineering
AGH University of Science and Technology in Krakow
Krakow
Poland

Editorial Office
MDPI AG
Grosspeteranlage 5
4052 Basel, Switzerland

This is a reprint of the Special Issue, published open access by the journal *Applied Sciences* (ISSN 2076-3417), freely accessible at: https://www.mdpi.com/journal/applsci/special_issues/00F11J2BKE.

For citation purposes, cite each article independently as indicated on the article page online and as indicated below:

Lastname, A.A.; Lastname, B.B. Article Title. *Journal Name* **Year**, *Volume Number*, Page Range.

ISBN 978-3-7258-9273-0 (Hbk)
ISBN 978-3-7258-9274-7 (PDF)
https://doi.org/10.3390/books978-3-7258-9274-7

Cover image courtesy of Zbigniew Gomółka

© 2026 by the authors. Articles in this reprint are Open Access and distributed under the Creative Commons Attribution (CC BY) license. The reprint as a whole is distributed by MDPI under the terms and conditions of the Creative Commons Attribution-NonCommercial-NoDerivs (CC BY-NC-ND) license (https://creativecommons.org/licenses/by-nc-nd/4.0/).

Contents

About the Editors . . . . . **vii**

Preface . . . . . **ix**

Zbigniew Gomolka, Damian Kordos and Ewa Dudek-Dyduch
From Models to Hybrid Intelligence: Emerging AI Approaches for Engineering Applications
Reprinted from: *Appl. Sci.* **2026**, *16*, 8864, https://doi.org/10.3390/app16178864 . . . . . **1**

Vasileios Charilogis, Ioannis G. Tsoulos, Anna Maria Gianni and DimitriosTsalikakis
ARQ: A Cohesive Optimization Design for Stable Performance on NoisyLandscapes
Reprinted from: *Appl. Sci.* **2025**, *15*, 12180, https://doi.org/10.3390/app152212180 . . . . . **11**

Aleksander Wojtowicz, Wiesław Paja and Urszula Bentkowska
Feature Importance Ranking Using Interval-Valued Methods and Aggregation Functions for Machine Learning Applications
Reprinted from: *Appl. Sci.* **2025**, *15*, 12130, https://doi.org/10.3390/app152212130 . . . . . **48**

Rodrigue G. , Thanda Shwe, Israel Mendonça and Masayoshi Aritsugi
Credit Evaluation Through Integration of Supervised and Unsupervised Machine Learning: Empirical Improvement and Unsupervised Component Analysis
Reprinted from: *Appl. Sci.* **2025**, *15*, 13020, https://doi.org/10.3390/app152413020 . . . . . **83**

Piotr Lasek
Extracting Behavioral Rules from Health Survey Data with Interpretable Models
Reprinted from: *Appl. Sci.* **2026**, *16*, 6146, https://doi.org/10.3390/app16126146 . . . . . **118**

Zbigniew Gomolka, Ewa Zeslawska and Lukasz Olbrot
Using Hybrid LSTM Neural Networks to Detect Anomalies in the Fiber Tube Manufacturing Process
Reprinted from: *Appl. Sci.* **2025**, *15*, 1383, https://doi.org/10.3390/app15031383 . . . . . **139**

Alexander Sieradzki, Jakub Bednarek, Albina Jegorowa and Jarosław Kurek
Explainable AI (XAI) Techniques for Convolutional Neural Network-Based Classification of Drilled Holes in Melamine Faced Chipboard
Reprinted from: *Appl. Sci.* **2024**, *14*, 7462, https://doi.org/10.3390/app14177462 . . . . . **161**

Yunfei Zhong, Yumei Pu, Xiaoxuan Li, Wenxuan Zhou, Hongjian He, Yuyang Chen, Lang Zhong and Danfei Liu
Deep Learning-Based Layout Analysis Method for Complex Layout Image Elements
Reprinted from: *Appl. Sci.* **2025**, *15*, 7797, https://doi.org/10.3390/app15147797 . . . . . **193**

Petra Radočaj, Goran Martinović and Dorijan Radočaj
Adaptive Multi-Scale Feature Learning Module for Pediatric Pneumonia Recognition in Chest X-Rays
Reprinted from: *Appl. Sci.* **2025**, *15*, 11824, https://doi.org/10.3390/app152111824 . . . . . **208**

Tomasz Les, Tomasz Markiewicz, Malgorzata Lorent, Miroslaw Dziekiewicz and Krzysztof Siwek
Deep Learning Based Correction Algorithms for 3D Medical Reconstruction in Computed Tomography and Macroscopic Imaging
Reprinted from: *Appl. Sci.* **2026**, *16*, 1954, https://doi.org/10.3390/app16041954 . . . . . **231**

Adam Szczur, Jan G. Bazan, Urszula Bentkowska, Piotr Kruczek and Stanislawa Bazan-Socha
Identifying and Predicting Changes in Behavioral Patterns for Temporal Data in Treatment of Neonatal Respiratory Failure
Reprinted from: *Appl. Sci.* **2025**, *15*, 12133, https://doi.org/10.3390/app152212133 . . . . . **253**

Marek Andryszczyk, Izabela Rojek, Tomasz Bednarek and Dariusz Mikołajewski
Possibilities of Reflecting the Mechanical Properties of Non-Absordable Surgical Meshes in an AI-Based Model in the Context of Industry 4.0/5.0
Reprinted from: *Appl. Sci.* **2025**, *15*, 12894, https://doi.org/10.3390/app152412894 . . . . . **274**

About the Editors

Zbigniew Gomółka

Zbigniew Gomółka is with the Institute of Computer Science, Faculty of Exact and Technical Sciences, University of Rzeszow (Poland). His research focuses on eye-tracking, image-processing, and neural networks, with applications in pilot attention analysis, control, and state estimation. He has led projects on gaze-based instrumentation awareness and contributed to hybrid AI methods in engineering contexts, in addition to co-editing Special Issues on applied AI and attention analytics.

Damian Kordos

Damian Kordos works in the Faculty of Mechanical Engineering and Aeronautics, Rzeszow University of Technology (Poland). His interests include aircraft systems, vision systems, flight simulators, and HMI. He has worked on integrating eye-tracking with simulator platforms to study operator workload and decision-making, and he has co-organized editorial projects bridging control, image-processing, and neural network techniques for aerospace applications.

Ewa Dudek-Dyduch

Ewa Dudek-Dyduch works in the Faculty of Electrical Engineering, Automatics, Computer Science, and Biomedical Engineering, AGH University of Science and Technology, in Krakow (Poland). Her work covers the scheduling of discrete processes, control algorithms, neural networks, and knowledge-based decision systems. She has advanced methods for attention modelling and 3D scene analysis and has co-edited initiatives on engineering applications of hybrid AI tools.

Preface

Artificial intelligence is increasingly becoming an integral component of modern engineering systems, supporting data analysis, prediction, optimization, decision making, and autonomous operation. At the same time, the complexity of real-world engineering problems often exceeds the capabilities of isolated computational models and encourages the integration of complementary AI methods, domain knowledge, optimization techniques, and interpretable decision mechanisms. This Reprint brings together research contributions that reflect this transition toward hybrid and integrated intelligence across a broad spectrum of applications. The presented studies demonstrate how machine learning, deep learning, explainable AI, optimization, and knowledge-driven approaches can be combined or adapted to address practical challenges in industrial systems, computer vision, biomedical analysis, behavioral modelling, and intelligent decision support. By bringing together different methodological perspectives and application domains, this Reprint provides researchers, engineers, and practitioners with a representative view of current developments in hybrid artificial intelligence and aims to stimulate further research toward reliable, interpretable, and application-oriented intelligent systems.

Zbigniew Gomółka, Damian Kordos, and Ewa Dudek-Dyduch
Guest Editors

applied sciences

MDPI

Editorial

From Models to Hybrid Intelligence: Emerging AI Approaches for Engineering Applications

Zbigniew Gomolka [1,*], Damian Kordos [2] and Ewa Dudek-Dyduch [3]

[1] Faculty of Exact and Technical Sciences, University of Rzeszow, Pigonia St. 1, 35-959 Rzeszow, Poland
[2] Faculty of Mechanical Engineering and Aeronautics, Rzeszow University of Technology, 35-959 Rzeszow, Poland; d.kordos@prz.edu.pl
[3] Faculty of Electrical Engineering, Automatics, Computer Science, and Biomedical Engineering, AGH University of Science and Technology in Krakow, 30-059 Krakow, Poland; edd@agh.edu.pl
* Correspondence: zgomolka@ur.edu.pl; Tel.: +48-17-851-8755

1. Introduction

Artificial intelligence has become an important component of contemporary engineering systems, supporting tasks ranging from optimization, prediction, and anomaly detection to computer vision, medical image analysis, and decision support. Advances in machine and deep learning have substantially increased the ability of computational models to extract complex representations from high-dimensional, heterogeneous, and temporally structured data. At the same time, real-world engineering problems increasingly expose the limitations of isolated data-driven models. Robust operation under uncertainty and noise, limited or imperfect training data, physical and operational constraints, interpretability, and the incorporation of expert knowledge frequently require capabilities that cannot be provided effectively by a single learning paradigm. Consequently, current research is increasingly moving toward hybrid artificial intelligence approaches in which complementary computational mechanisms, knowledge representations, and learning strategies are integrated within unified models, architectures, or decision-making pipelines [1–3].

An important direction within this transition is the integration of data-driven learning with explicit knowledge representations. Knowledge-driven and neuro-symbolic approaches combine representation learning with structured knowledge, logical constraints, semantic relationships, and reasoning mechanisms, thereby extending learning beyond purely statistical inference [3–6]. Recent developments include validated knowledge-graph completion, rule-enhanced temporal knowledge-graph reasoning, and the integration of large language models with structured temporal knowledge [7–9]. These approaches illustrate a broader shift toward architectures in which learning and reasoning operate as complementary components. From an engineering perspective, such integration is particularly relevant when prior knowledge about system structure, admissible states, constraints, or relationships can be used directly to guide the learning process.

A related direction concerns the incorporation of physical, geometric, and domain-specific knowledge into machine learning. Physics-informed learning provides a representative example by introducing governing equations, structural constraints, or mechanistic representations into model training [10]. Recent applications include electrical impedance tomography, simulation of physical and mechanical processes, learning on geometrically irregular domains, and uncertainty-aware dynamic force identification [11–14]. Physics-informed reinforcement learning extends this principle to sequential learning and decision-making [15]. These developments reflect a fundamental principle of engineering

AI: knowledge already available through physical laws, geometric relationships, or operational constraints can be explicitly incorporated to constrain or complement data-driven learning rather than being inferred entirely from data.

Hybridization also encompasses combinations of learning models, ensemble strategies, optimization mechanisms, multi-stage processing architectures, and explainability techniques. Ensemble learning has demonstrated that combining models with different properties can improve generalization and robustness [16,17], while recent approaches increasingly integrate predictive models with mechanisms that expose or interpret their decisions. Examples include explainable fuzzy–machine-learning frameworks, hybrid architectures for intelligent fault diagnosis, interpretable ensemble-tree models, and explainable ensemble systems for biomedical classification [18–21]. Explainability is also becoming an active component of learning workflows, for example by supporting query selection and human intervention in active learning [22]. Its role therefore extends beyond post-hoc visualization toward supporting the design and operation of intelligent systems in which model outputs interact with physical processes or human decision makers.

These methodological developments are accompanied by broader changes in the criteria used to evaluate AI systems. Predictive accuracy remains important, but robustness, uncertainty handling, computational efficiency, interpretability, traceability, and compatibility with operational workflows are increasingly relevant design objectives, particularly in industrial, biomedical, and other high-stakes applications [23–27]. Knowledge–data dual-driven systems illustrate this transition by distributing reasoning, retrieval, prediction, and decision-support functions among complementary intelligent components [28]. In this broader perspective, hybrid AI can be understood as the purposeful integration of computational and knowledge-based mechanisms according to the structure and requirements of the problem being solved. When human expertise, feedback, and decision-making are additionally incorporated, this development connects with the concept of hybrid intelligence, in which human and artificial capabilities constitute complementary resources [1,2].

Against this background, the Special Issue "Engineering Applications of Hybrid Artificial Intelligence Tools" was conceived to reflect the methodological diversity of contemporary hybrid AI and its relevance to practical engineering problems. The contributions span evolutionary optimization, feature ranking, supervised and unsupervised learning, interpretable rule extraction, industrial anomaly detection, explainable computer vision, multi-stage deep-learning pipelines, medical image analysis, multimodal anatomical reconstruction, temporal clinical prediction, and data-driven modeling of biomedical materials. Despite their methodological and application diversity, the studies share a tendency to move beyond isolated models by combining complementary learning strategies, optimization mechanisms, multi-scale representations, explainability techniques, domain knowledge, geometric constraints, or experimentally derived information. This spectrum illustrates that hybridization is not a single architectural principle, but a family of strategies through which different computational capabilities can be coordinated according to the requirements of a particular engineering problem.

In curating this Special Issue, our aims were to: highlight approaches that integrate complementary AI mechanisms, demonstrate how learning, optimization, interpretability, and domain-specific information can be combined within application-oriented computational frameworks and finally present practical studies illustrating their use in industrial, biomedical, and decision-support environments. Across the contributions, recurring themes include robustness and meaningful representation of heterogeneous data, integration of complementary models and processing stages, explainability and domain knowledge, and evaluation under realistic operational or experimental conditions. Taken together, the studies illustrate a broader progression from individual computational models toward

https://doi.org/10.3390/app16178864

integrated intelligent systems in which learning, optimization, knowledge, and interpretation operate within a common framework. The remainder of this Editorial provides an overview of the eleven contributions, emphasizing their methodological innovations, experimental validation, and relevance to the continuing development of hybrid artificial intelligence for engineering applications.

2. An Overview of Published Articles

In the study by Charilogis et al. (contribution 1), the authors address reliable continuous optimization in noisy and rugged search landscapes, where evolutionary algorithms must balance exploration, exploitation, and robustness to misleading fitness evaluations [29,30]. Building on recent adaptive approaches to differential evolution [31,32], they introduce Adaptive RTR with Quarantine (ARQ), a hybrid evolutionary method combining pbest differential evolution, success-history parameter adaptation, restricted tournament replacement, outlier quarantine, and targeted micro-restarts. The latter mechanisms redirect poorly performing individuals and restore population diversity during search stagnation. Evaluation on a heterogeneous benchmark suite against established evolutionary optimizers demonstrated competitive and stable performance, with ARQ achieving the best overall mean rank across the considered problems.

In the study by Wojtowicz et al. (contribution 2), the authors address reliable feature selection and ranking in machine learning, where feature-importance estimates may vary across data subsets, affecting model stability and interpretability [25,33,34]. They investigate the Interval-Valued Weighted Feature Ranking (IVWFR) method, which represents feature-importance estimates obtained across multiple data folds as intervals and combines them using weighted interval-valued aggregation functions. The resulting score reflects both feature relevance and its stability, thereby incorporating uncertainty directly into the ranking process. Evaluation on ten synthetic and five real-world datasets against established feature-selection methods showed that IVWFR achieved the best or among the best classification results across the considered datasets while maintaining high feature-selection stability and competitive computational efficiency.

In the study by Atteba et al. (contribution 3), the authors investigate a hybrid machine learning framework for credit evaluation that integrates supervised classification, unsupervised clustering, and consensus-based decision making [35]. The proposed methodology combines several classifiers with k-means, Agglomerative, and BIRCH clustering, while a Self-Organizing Map (SOM) is employed as a consensus mechanism for combining classifier outputs. The authors additionally analyze how cluster quality and stability, evaluated using the silhouette score and Adjusted Rand Index, relate to subsequent classification performance. The framework is assessed through five experimental routes on multiple credit datasets and is further extended to multiclass credit scoring. The results indicate that the coordinated use of unsupervised structure discovery, supervised learning, and SOM-based consensus can improve the robustness of credit evaluation. This contribution demonstrates how complementary learning paradigms can be integrated within a unified framework to combine data structure discovery with predictive decision making.

In the study by Lasek (contribution 4), interpretable machine learning is applied to behavioral and sociodemographic health survey data to identify human-readable patterns associated with diabetes. Rather than prioritizing predictive accuracy alone, the study emphasizes transparent models whose decision mechanisms can be directly examined, an important consideration in high-stakes applications of artificial intelligence [24,25]. A depth-constrained decision tree was trained on data from 16,824 respondents and 38 retained features from the Canadian Community Health Survey, producing 32 explicit IF–THEN rules. Feature importance and cross-validation analyses identified age, alcohol

https://doi.org/10.3390/app16178864

consumption patterns, and daily energy expenditure as the dominant and stable factors associated with diabetes status. Although the model achieved only moderate predictive performance and did not significantly outperform the majority-class baseline, sensitivity analyses demonstrated that the principal rules and feature rankings were stable across data folds and encoding strategies. The contribution therefore illustrates the value of interpretable AI as a knowledge-discovery mechanism, where transparent and reproducible behavioral patterns can provide useful population-level insights even when predictive performance is not the primary objective.

In the study by Gomolka et al. (contribution 5), a hybrid deep-learning architecture is proposed for anomaly detection in a real industrial process involving the production of fiber-optic cables. Detecting anomalies in multivariate industrial time series remains challenging because of high-dimensional sensor data, temporal dependencies, process-specific disturbances, and the limited availability of accurately labelled anomalous samples [36–39]. To address these challenges, the authors combine multi-resolution data preprocessing and K-means clustering with a three-channel LSTM architecture, where clustering supports the construction of training data when precise manual labelling of anomalous process segments is technologically difficult. The resulting representations are processed by three parallel LSTM predictors and integrated by an aggregation module to classify the monitored process into a normal state and four anomaly classes. Evaluation on industrial data collected from 232 measurement points yielded approximately $R^2 = 0.89$, with the proposed model outperforming Random Forest, SVM, and conventional RNN benchmarks in the reported anomaly-detection metrics. The system was also test-integrated into the industrial process monitoring environment, where its application was associated with a 66.3% reduction in defective production during the analyzed implementation period. The study demonstrates the practical integration of unsupervised data structuring, temporal deep learning, and decision-level aggregation for industrial process monitoring and quality control.

In the study by Sieradzki et al. (contribution 6), the authors investigate explainable artificial intelligence for automated tool condition monitoring in the machining of melamine-faced chipboard. Deep learning increasingly supports industrial visual inspection and quality control [40], while its deployment in decision-support systems creates a corresponding need for transparent and interpretable predictions [25,41]. The authors combine CNN-based classification with two complementary XAI techniques, Local Interpretable Model-Agnostic Explanations (LIME) and Gradient-weighted Class Activation Mapping (Grad-CAM), to examine the visual evidence underlying the classification of drilled holes. Three pretrained architectures, VGG16, VGG19, and ResNet101, were evaluated on 8526 images representing three drill-condition classes using five-fold cross-validation. VGG19 achieved the highest average accuracy of 67.03% and the lowest critical error rate and was subsequently selected for explainability analysis. LIME provided localized explanations of individual predictions, whereas Grad-CAM identified broader influential image regions, with the two methods providing complementary levels of interpretation. The study thus combines predictive deep learning with post-hoc explainability to support more transparent AI-assisted industrial quality monitoring.

In the study by Zhong et al. (contribution 7), the authors propose a two-stage deep-learning framework for automated analysis of complex graphic layouts. An improved DeepLabv3+ model first segments theme, text, and graphic regions, after which GoogLeNet classifies their spatial organization into predefined layout categories. The segmentation architecture incorporates MobileNetV3-small and a modified Atrous Spatial Pyramid Pooling module to improve efficiency and multi-scale feature extraction. Evaluation on 2300 annotated movie posters yielded an MIoU of 75.60%, outperforming FCN, PSPNet, DeepLabv3, and the original DeepLabv3+ while reducing training time relative to sev-

https://doi.org/10.3390/app16178864

eral of these architectures. The study illustrates how complementary segmentation and classification stages can connect pixel-level recognition with higher-level interpretation of complex graphical structures.

In the study by Radočaj et al. (contribution 8), the authors address automated pediatric pneumonia recognition from chest X-ray images, an area in which deep learning has become an important tool for computer-assisted radiographic analysis [42–45]. They propose an adaptive multi-scale feature learning module (IncMB) combining Inception-based multi-branch feature extraction, Mish activation, and Batch Normalization. IncMB was integrated with four transfer-learning architectures - InceptionV3, InceptionResNetV2, MobileNetV2, and DenseNet201 - and evaluated on 5856 pediatric chest X-ray images. Across all four architectures, the module improved classification performance while reducing the number of model parameters, with InceptionResNetV2 + IncMB achieving the best results at an accuracy of 0.9812 and an F1-score of 0.9761. Grad-CAM analysis further indicated that the enhanced models focused more consistently on clinically relevant pulmonary regions. The study demonstrates how adaptive multi-scale feature learning can improve both the diagnostic performance and computational efficiency of deep-learning models for medical image analysis.

In the study by Les et al. (contribution 9), the authors propose a hybrid two-stage framework for three-dimensional reconstruction of kidney anatomy from macroscopic images using CT-derived models as the anatomical reference. The approach combines deterministic Optimal Cross-section Matching (OCM), which performs constrained global geometric alignment, with a lightweight deep-learning network that learns residual local deformations associated with tissue distortion and shrinkage. Hough-transform-based metric calibration and Bézier-curve contour smoothing further support the physical consistency of the reconstruction. Evaluation on clinical data from 40 patients using patient-level five-fold cross-validation showed that the hybrid OCM + DL approach outperformed both OCM-only and DL-only configurations, achieving a Dice coefficient of 0.90 and HD95 of 1.9 mm, with a statistically significant improvement in Dice over DL alone ($p < 0.001$). The study provides a clear example of hybrid AI in which explicit geometric knowledge and deterministic optimization constrain data-driven learning, combining their complementary strengths in multimodal anatomical reconstruction.

In the study by Szczur et al. (contribution 10), the authors propose a machine-learning methodology for identifying and predicting changes in behavioral patterns in temporal data, illustrated using the treatment of respiratory failure in premature neonates [46,47]. A distinctive feature of the approach is the integration of domain knowledge with data-driven prediction: medical experts define clinically meaningful behavioral patterns through logical formulas and threshold values, while machine learning is used to predict transitions between these patterns from preceding temporal observations. The methodology transforms patient histories into pairs of adjacent time windows, derives aggregated temporal features, performs feature selection, and applies classification to predict the behavioral pattern of the subsequent window. The experiments were conducted using data from 173 neonates, yielding 7186 temporal-window pairs, with patients separated between training and test sets to prevent data leakage. Among nine evaluated classifiers, XGBClassifier achieved the highest macro F1-score of 0.970, increasing to 0.979 after feature selection. Further analysis showed high sensitivity not only for stable behavioral patterns but also for clinically relevant changes, including deterioration and improvement of the patient's condition. This contribution illustrates how expert-defined knowledge representations and machine learning can be combined to move beyond recognition of current states toward prediction of behavioral transitions in complex temporal systems.

https://doi.org/10.3390/app16178864

In the study by Andryszczyk et al. (contribution 11), the authors investigate the integration of experimental mechanical characterization and machine learning for modeling non-absorbable surgical meshes within the context of Industry 4.0 and Industry 5.0, where industrial AI increasingly supports intelligent, adaptive, and human-centric manufacturing systems [48–50]. Mechanical tests on 156 samples examined the effects of loading direction and suture integration under monotonic and cyclic tensile loading, showing that suturing reduces mechanical strength and increases deformation, while cyclic loading reveals progressive stiffness degradation and direction-dependent fatigue behavior. The resulting experimental data were used to develop two machine-learning models: one predicting the strength of the complete mesh–tissue connection and another supporting thread selection for a required connection strength. The best-performing regression models achieved reported predictive scores of 84.12% and 82.47%, respectively. Although the machine-learning component is presented as a proof-of-concept requiring broader datasets and further validation, the study demonstrates a pathway from experimental characterization toward data-driven modeling and future optimization of biomedical materials.

3. Conclusions

The contributions collected in this Special Issue demonstrate that contemporary engineering applications of artificial intelligence increasingly extend beyond the use of isolated computational models. Across optimization, feature analysis, credit evaluation, industrial monitoring, computer vision, medical imaging, temporal clinical data analysis, and biomedical material modeling, the presented studies show different ways in which complementary computational mechanisms can be combined to address the complexity of real-world problems. These combinations include adaptive evolutionary strategies, uncertainty-aware feature evaluation, supervised and unsupervised learning, multi-model aggregation, multi-scale feature extraction, explainability mechanisms, deterministic optimization coupled with deep learning, and the integration of expert-defined knowledge with data-driven prediction.

An important observation emerging from these studies is that hybridization does not represent a single architectural principle. In some contributions, it is implemented directly at the algorithmic level by combining complementary learning or optimization mechanisms. In others, hybridization occurs at the level of the processing pipeline, where different models perform specialized and mutually supporting tasks. Particularly relevant from an engineering perspective are approaches in which data-driven learning is complemented by explicit domain knowledge, geometric constraints, interpretable representations, or experimentally derived physical information. Such solutions indicate that the effectiveness of artificial intelligence in complex applications may depend not only on increasing model capacity, but also on determining which elements of a problem should be learned from data and which can be represented more effectively through prior knowledge, physical constraints, or expert-defined structures.

The contributions also emphasize that predictive performance alone is not sufficient to characterize the value of AI-based engineering systems. Interpretability, robustness, computational efficiency, stability under noisy conditions, meaningful feature representation, and the ability to integrate models with real operational or clinical workflows are becoming increasingly important design objectives. This is particularly evident in high-stakes and application-oriented domains, where understanding model behavior and maintaining consistency with domain knowledge may be as important as improvements in conventional performance metrics. These observations are consistent with the broader development of explainable, human-centered, and hybrid intelligence paradigms [1,2,23,25].

Taken together, the studies presented in this Special Issue suggest a gradual transition from individual AI models toward integrated intelligent systems in which learning

https://doi.org/10.3390/app16178864

algorithms operate alongside optimization methods, interpretable mechanisms, domain knowledge, physical models, and human expertise. This transition should not be understood as a replacement of established machine learning or deep learning techniques, but rather as their incorporation into broader computational frameworks designed around the structure and requirements of specific engineering problems. Future research is therefore expected to focus increasingly on adaptive integration of heterogeneous AI components, stronger incorporation of expert and physical knowledge, improved explainability and reliability, and systematic validation in real-world environments. In this perspective, the progression from models to hybrid intelligence represents not merely an increase in algorithmic complexity, but a shift toward more coherent, transparent, and application-aware artificial intelligence systems capable of supporting complex engineering decision-making.

Author Contributions: Conceptualization, Z.G.; writing—original draft preparation, Z.G., D.K. and E.D.-D.; writing—review and editing, Z.G., D.K. and E.D.-D.; supervision, Z.G. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflicts of interest.

List of Contributions

1. Charilogis, V.; Tsoulos, I.G.; Gianni, A.M.; Tsalikakis, D. ARQ: A Cohesive Optimization Design for Stable Performance on Noisy Landscapes. *Appl. Sci.* **2025**, *15*, 12180. https://doi.org/10.3390/app152212180.
2. Wojtowicz, A.; Paja, W.; Bentkowska, U. Feature Importance Ranking Using Interval-Valued Methods and Aggregation Functions for Machine Learning Applications. *Appl. Sci.* **2025**, *15*, 12130. https://doi.org/10.3390/app152212130.
3. Atteba, R.G.; Shwe, T.; Mendonça, I.; Aritsugi, M. Credit Evaluation Through Integration of Supervised and Unsupervised Machine Learning: Empirical Improvement and Unsupervised Component Analysis. *Appl. Sci.* **2025**, *15*, 13020. https://doi.org/10.3390/app152413020.
4. Lasek, P. Extracting Behavioral Rules from Health Survey Data with Interpretable Models. *Appl. Sci.* **2026**, *16*, 6146. https://doi.org/10.3390/app16126146.
5. Gomolka, Z.; Zeslawska, E.; Olbrot, L. Using Hybrid LSTM Neural Networks to Detect Anomalies in the Fiber Tube Manufacturing Process. *Appl. Sci.* **2025**, *15*, 1383. https://doi.org/10.3390/app15031383.
6. Sieradzki, A.; Bednarek, J.; Jegorowa, A.; Kurek, J. Explainable AI (XAI) Techniques for Convolutional Neural Network-Based Classification of Drilled Holes in Melamine Faced Chipboard. *Appl. Sci.* **2024**, *14*, 7462. https://doi.org/10.3390/app14177462.
7. Zhong, Y.; Pu, Y.; Li, X.; Zhou, W.; He, H.; Chen, Y.; Zhong, L.; Liu, D. Deep Learning-Based Layout Analysis Method for Complex Layout Image Elements. *Appl. Sci.* **2025**, *15*, 7797. https://doi.org/10.3390/app15147797.
8. Radočaj, P.; Martinović, G.; Radočaj, D. Adaptive Multi-Scale Feature Learning Module for Pediatric Pneumonia Recognition in Chest X-Rays. *Appl. Sci.* **2025**, *15*, 11824. https://doi.org/10.3390/app152111824.
9. Les, T.; Markiewicz, T.; Lorent, M.; Dziekiewicz, M.; Siwek, K. Deep Learning Based Correction Algorithms for 3D Medical Reconstruction in Computed Tomography and Macroscopic Imaging. *Appl. Sci.* **2026**, *16*, 1954. https://doi.org/10.3390/app16041954.
10. Szczur, A.; Bazan, J.G.; Bentkowska, U.; Kruczek, P.; Bazan-Socha, S. Identifying and Predicting Changes in Behavioral Patterns for Temporal Data in Treatment of

Neonatal Respiratory Failure. *Appl. Sci.* **2025**, *15*, 12133. https://doi.org/10.3390/app152212133.

11. Andryszczyk, M.; Rojek, I.; Bednarek, T.; Mikołajewski, D. Possibilities of Reflecting the Mechanical Properties of Non-Absorbable Surgical Meshes in an AI-Based Model in the Context of Industry 4.0/5.0. *Appl. Sci.* **2025**, *15*, 12894. https://doi.org/10.3390/app152412894.

References

1. Akata, Z.; Balliet, D.; de Rijke, M.; Dignum, F.; Dignum, V.; Eiben, G.; Fokkens, A.; Grossi, D.; Hindriks, K.; Hoos, H.; et al. A research agenda for hybrid intelligence: Augmenting human intellect with collaborative, adaptive, responsible, and explainable artificial intelligence. *Computer* **2020**, *53*, 18–28. [CrossRef]
2. Dellermann, D.; Ebel, P.; Söllner, M.; Leimeister, J.M. Hybrid intelligence. *Bus. Inf. Syst. Eng.* **2019**, *61*, 637–643. [CrossRef]
3. Benkirane, F.E.; Crombez, N.; Hilaire, V.; Ruichek, Y. Knowledge-driven deep learning approaches for computer vision tasks: A survey. *Knowl.-Based Syst.* **2025**, *330*, 114645. [CrossRef]
4. Sarker, M.K.; Zhou, L.; Eberhart, A.; Hitzler, P. Neuro-symbolic artificial intelligence: Current trends. *AI Commun.* **2021**, *34*, 197–209. [CrossRef]
5. Hitzler, P.; Eberhart, A.; Ebrahimi, M.; Sarker, M.K.; Zhou, L. Neuro-symbolic approaches in artificial intelligence. *Natl. Sci. Rev.* **2022**, *9*, nwac035. [CrossRef] [PubMed]
6. Bhuyan, B.P.; Ramdane-Cherif, A.; Tomar, R.; Singh, T.P. Neuro-symbolic artificial intelligence: A survey. *Neural Comput. Appl.* **2024**, *36*, 12809–12844. [CrossRef]
7. Purohit, D.; Chudasama, Y.; Vidal, M.-E. VANILLA: Validated knowledge graph completion—A Normalization-based framework for Integrity, Link prediction, and Logical Accuracy. *Knowl.-Based Syst.* **2025**, *325*, 113939. [CrossRef]
8. Pan, Q.; Yao, L.; Shen, G.; Han, X.; Chen, Y.; Kong, X. Leveraging temporal validity of rules via LLMs for enhanced temporal knowledge graph reasoning. *Knowl.-Based Syst.* **2025**, *327*, 114094. [CrossRef]
9. Bai, H.; Kang, S.; Zheng, J.; Qin, J.; Zhou, X. Learning from surprise: Fusing LLM-guided test-time adaptation for temporal knowledge graphs. *Knowl.-Based Syst.* **2026**, *342*, 115867. [CrossRef]
10. Karniadakis, G.E.; Kevrekidis, I.G.; Lu, L.; Perdikaris, P.; Wang, S.; Yang, L. Physics-informed machine learning. *Nat. Rev. Phys.* **2021**, *3*, 422–440. [CrossRef]
11. Wang, L.; Cheng, L.; Liu, Y. Uncertainty-oriented physics-informed long short-term memory (UOPI-LSTM) network framework for dynamic force identification with interval uncertainties. *Expert Syst. Appl.* **2025**, *274*, 127067. [CrossRef]
12. Smyl, D.; Tallman, T.N.; Homa, L.; Flournoy, C.; Hamilton, S.J.; Wertz, J. Physics Informed Neural Networks for Electrical Impedance Tomography. *Neural Netw.* **2025**, *188*, 107410. [CrossRef] [PubMed]
13. Guo, Y.; Fu, Z.; Min, J.; Lin, S.; Liu, X.; Rashed, Y.F.; Zhuang, X. Long-term simulation of physical and mechanical behaviors using curriculum-transfer-learning based physics-informed neural networks. *Neural Netw.* **2025**, *191*, 107825. [CrossRef] [PubMed]
14. Ryu, J.; Park, S.; Hwang, H.J. A Physics-Informed Neural Network framework for solving PDEs on point clouds via surface reconstruction. *Neural Netw.* **2025**, *192*, 107928. [CrossRef] [PubMed]
15. Banerjee, C.; Nguyen, K.; Fookes, C.; Raissi, M. A survey on physics informed reinforcement learning: Review and open problems. *Expert Syst. Appl.* **2025**, *287*, 128166. [CrossRef]
16. Sagi, O.; Rokach, L. Ensemble learning: A survey. *WIREs Data Min. Knowl. Discov.* **2018**, *8*, e1249. [CrossRef]
17. Yang, Y.; Lv, H.; Chen, N. A survey on ensemble learning under the era of deep learning. *Artif. Intell. Rev.* **2023**, *56*, 5545–5589. [CrossRef]
18. Akram, M.; Batool, I.; Kahraman, C. An explainable hybrid fuzzy-machine learning framework for multi-crop suitability classification using agro-environmental data. *Knowl.-Based Syst.* **2026**, *333*, 115008. [CrossRef]
19. Abboush, M.; Ghannoum, E.; Rausch, A. An explainable hybrid deep learning-enabled intelligent fault detection and diagnosis approach for automotive software systems validation. *Knowl.-Based Syst.* **2026**, *334*, 114922. [CrossRef]
20. Lee, Y.-S.; Yen, S.-J.; Jiang, W.; Chen, J.; Chang, C.-Y. Illuminating the black box: An interpretable machine learning based on ensemble trees. *Expert Syst. Appl.* **2025**, *272*, 126720. [CrossRef]
21. Elgendy, I.A.; Hosny, M.; Albashrawi, M.A.; Alsenan, S. Dual-stage explainable ensemble learning model for diabetes diagnosis. *Expert Syst. Appl.* **2025**, *274*, 126899. [CrossRef]
22. Shim, J.; Kang, S. Why does this query need to be labeled?: Enhancing active learning through explanation-based interventions in query selection. *Expert Syst. Appl.* **2025**, *290*, 128443. [CrossRef]
23. Shneiderman, B. Human-centered artificial intelligence: Reliable, safe and trustworthy. *Int. J. Hum.-Comput. Interact.* **2020**, *36*, 495–504. [CrossRef]

24. Rudin, C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nat. Mach. Intell.* **2019**, *1*, 206–215. [CrossRef] [PubMed]
25. Arrieta, A.B.; Díaz-Rodríguez, N.; Del Ser, J.; Bennetot, A.; Tabik, S.; Barbado, A.; García, S.; Gil-López, S.; Molina, D.; Benjamins, R.; et al. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Inf. Fusion* **2020**, *58*, 82–115. [CrossRef]
26. Minh, D.; Wang, H.X.; Li, Y.F.; Nguyen, T.N. Explainable artificial intelligence: A comprehensive review. *Artif. Intell. Rev.* **2022**, *55*, 3503–3568. [CrossRef]
27. Islam, M.R.; Ahmed, M.U.; Barua, S.; Begum, S. A systematic review of explainable artificial intelligence in terms of different application domains and tasks. *Appl. Sci.* **2022**, *12*, 1353. [CrossRef]
28. Xiao, Z.; Liu, H.; Zhou, X.; Cong, S.; Ju, Y. Power ops agent: A knowledge and data dual-driven multi-agent framework for substation operation and maintenance. *Expert Syst. Appl.* **2026**, *318*, 131965. [CrossRef]
29. Rakshit, P.; Konar, A.; Das, S. Noisy evolutionary optimization algorithms—A comprehensive survey. *Swarm Evol. Comput.* **2017**, *33*, 18–45. [CrossRef]
30. Tang, K.; Jin, Y.; Yao, X.; Zhou, Z.-H. On the effectiveness of sampling for evolutionary optimization in noisy environments. *Evol. Comput.* **2018**, *26*, 237–267. [CrossRef] [PubMed]
31. Zhang, S.X.; Liu, Y.H.; Zheng, L.M.; Zheng, S.Y. Differential evolution with collective ensemble learning. *Swarm Evol. Comput.* **2024**, *87*, 101521. [CrossRef]
32. Zhang, S.X.; Hu, X.R.; Zheng, S.Y. Differential evolution with evolutionary scale adaptation. *Swarm Evol. Comput.* **2024**, *85*, 101481. [CrossRef]
33. Rengasamy, D.; Mase, J.M.; Kumar, A.; Rothwell, B.; Torres, M.T.; Alexander, M.R.; Winkler, D.A.; Figueredo, G.P. Feature importance in machine learning models: A fuzzy information fusion approach. *Neurocomputing* **2022**, *511*, 163–174. [CrossRef]
34. Villa-Blanco, C.; Bielza, C.; Larrañaga, P. Feature subset selection for data and feature streams: A review. *Artif. Intell. Rev.* **2023**, *56*, 1011–1062. [CrossRef]
35. Zhang, X.; Yu, L. Consumer credit risk assessment: A review from the state-of-the-art classification algorithms, data traits, and learning methods. *Expert Syst. Appl.* **2024**, *237*, 121484. [CrossRef]
36. Zamanzadeh Darban, Z.; Webb, G.I.; Pan, S.; Aggarwal, C.C.; Salehi, M. Deep learning for time series anomaly detection: A survey. *ACM Comput. Surv.* **2024**, *57*, 15. [CrossRef]
37. Correia, L.; Goos, J.C.; Klein, P.; Bäck, T.; Kononova, A.V. Online model-based anomaly detection in multivariate time series: Taxonomy, survey, research challenges and future directions. *Eng. Appl. Artif. Intell.* **2024**, *138*, 109323. [CrossRef]
38. Wang, F.; Jiang, Y.; Zhang, R.; Wei, A.; Xie, J.; Pang, X. A survey of deep anomaly detection in multivariate time series: Taxonomy, applications, and directions. *Sensors* **2025**, *25*, 190. [CrossRef] [PubMed]
39. Rodríguez, M.; Tobón, D.P.; Múnera, D. Anomaly classification in industrial Internet of Things: A review. *Intell. Syst. Appl.* **2023**, *18*, 200232. [CrossRef]
40. Ameri, R.; Hsu, C.-C.; Band, S.S. A systematic review of deep learning approaches for surface defect detection in industrial applications. *Eng. Appl. Artif. Intell.* **2024**, *130*, 107717. [CrossRef]
41. Gupta, S.; Thakur, K.; Kumar, M. A systematic review of Explainable Artificial Intelligence models and applications: Recent developments and future trends. *Decis. Anal. J.* **2023**, *7*, 100230. [CrossRef]
42. van der Velden, B.H.M.; Kuijf, H.J.; Gilhuijs, K.G.A.; Viergever, M.A. Explainable artificial intelligence (XAI) in deep learning-based medical image analysis. *Med. Image Anal.* **2022**, *79*, 102470. [CrossRef] [PubMed]
43. Nazir, S.; Dickson, D.M.; Akram, M.U. Survey of explainable artificial intelligence techniques for biomedical imaging with deep neural networks. *Comput. Biol. Med.* **2023**, *156*, 106668. [CrossRef] [PubMed]
44. Çallı, E.; Sogancioglu, E.; van Ginneken, B.; van Leeuwen, K.G.; Murphy, K. Deep learning for chest X-ray analysis: A survey. *Med. Image Anal.* **2021**, *72*, 102125. [CrossRef] [PubMed]
45. Meedeniya, D.; Kumarasinghe, H.; Kolonne, S.; Fernando, C.; De la Torre Díez, I.; Marques, G. Chest X-ray analysis empowered with deep learning: A systematic review. *Appl. Soft Comput.* **2022**, *126*, 109319. [CrossRef] [PubMed]
46. De Roock, E.; Martin, N. Process mining in healthcare—An updated perspective on the state of the art. *J. Biomed. Inform.* **2022**, *127*, 103995. [CrossRef] [PubMed]
47. Guzzo, A.; Rullo, A.; Vocaturo, E. Process mining applications in the healthcare domain: A comprehensive review. *WIREs Data Min. Knowl. Discov.* **2022**, *12*, e1442. [CrossRef]
48. Leng, J.; Zhu, X.; Huang, Z.; Li, X.; Zheng, P.; Zhou, X.; Mourtzis, D.; Wang, B.; Qi, Q.; Shao, H.; et al. Unlocking the power of industrial artificial intelligence towards Industry 5.0: Insights, pathways, and challenges. *J. Manuf. Syst.* **2024**, *73*, 349–363. [CrossRef]
49. Zhang, C.; Wang, Z.; Zhou, G.; Chang, F.; Ma, D.; Jing, Y.; Cheng, W.; Ding, K.; Zhao, D. Towards new-generation human-centric smart manufacturing in Industry 5.0: A systematic review. *Adv. Eng. Inform.* **2023**, *57*, 102121. [CrossRef]

https://doi.org/10.3390/app16178864

50. Passalacqua, M.; Pellerin, R.; Magnani, F.; Doyon-Poulin, P.; Del-Aguila, L.; Boasen, J.; Léger, P.M. Human-centred AI in Industry 5.0: A systematic review. *Int. J. Prod. Res.* **2025**, *63*, 2638–2669. [CrossRef]

Short Biography of Authors



Zbigniew Gomolka is a researcher in artificial intelligence and industrial informatics at the University of Rzeszow. His research focuses on hybrid artificial intelligence, machine and deep learning, fractional-order neural networks, and intelligent decision-support systems, with particular emphasis on engineering applications. His work includes AI-based anomaly detection and process monitoring in industrial environments, as well as the development of data-driven and hybrid computational methods for complex manufacturing systems. He also conducts interdisciplinary research on eye-tracking and multimodal sensing for the analysis of human cognitive and decision-making processes. His research interests further include algebraic-logical meta-modeling (ALMM), intelligent scheduling and decision systems for UAV fleets, and autonomous systems operating under dynamic constraints. His current work explores the integration of learning algorithms, domain knowledge, intelligent sensing, and interpretable computational mechanisms toward more adaptive and human-centered AI systems. He serves as a reviewer for international journals including *Expert Systems with Applications*, *Knowledge-Based Systems*, *Neural Networks*, *Sensors*, *Drones*, and *Applied Sciences*. ORCID: 0000-0001-7023-1646.



Damian Kordos is an Associate Professor in the Department of Avionics and Control Systems at the Faculty of Mechanical Engineering and Aeronautics, Rzeszow University of Technology. His research interests span mechanical engineering, automation and control, electronics, electrical engineering, and space technologies, with particular emphasis on vision systems and their applications in aviation and industrial environments. His work also covers control systems, wireless communication, and human–machine interfaces. He has extensive experience in applied research and technology development and has served as a principal investigator or researcher in several dozen research, development, and implementation projects conducted in cooperation with the aviation, medical, tooling, and heavy-industry sectors. His research combines sensing, control, communication, and intelligent system technologies, with a strong focus on their practical implementation in complex engineering environments. He is a member of the Polish Society of Aeronautics and Astronautics (PSAA). ORCID: 0000-0003-4764-3102.



Ewa Dudek-Dyduch is a Professor whose research activities span computer science, automation and robotics, artificial intelligence, soft computing, parallel computing, and information systems. Her scientific work has focused particularly on formal methods for modeling and analyzing discrete processes, computational intelligence, and intelligent approaches to complex decision and control problems. Her research has contributed to the development of methods combining formal process representations with computational and artificial intelligence techniques for engineering applications. She has authored more than 100 scientific publications and has supervised several doctoral dissertations. During her academic career, she has held senior academic and administrative positions at AGH University of Science and Technology in Krakow, including Head of postgraduate programs and Vice-Dean of the Faculty of Electrical Engineering, Automatics, Computer Science and Electronics. She has also served as a visiting professor at numerous universities and research institutions in the United States, Germany, China, France, Denmark, and the United Kingdom. Her extensive international and interdisciplinary activity has supported research at the intersection of artificial intelligence, soft computing, formal modeling, and engineering systems. She has also been actively involved in scientific committees and professional organizations in the fields of automation, robotics, computer science, and neural networks. ORCID: 0000-0003-3263-1299.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

https://doi.org/10.3390/app16178864

applied sciences

MDPI

Article

ARQ: A Cohesive Optimization Design for Stable Performance on Noisy Landscapes

Vasileios Charilogis [1], Ioannis G. Tsoulos [1,*], Anna Maria Gianni [1] and Dimitrios Tsalikakis [2]

[1] Department of Informatics and Telecommunications, University of Ioannina, 47150 Kostaki Artas, Greece; v.charilog@uoi.gr (V.C.); am.gianni@uoi.gr (A.M.G.)
[2] Department of Engineering Informatics and Telecommunications, University of Western Macedonia, 50100 Kozani, Greece; dtsalikakis@uowm.gr
* Correspondence: itsoulos@uoi.gr

Abstract: The proposed Adaptive RTR with Quarantine (ARQ) method integrates, within a single evolutionary scheme for continuous optimization, three mature ideas of pbest differential evolution with an archive, success-history parameter adaptation, and restricted tournament replacement (RTR) and extends them with a novel outlier quarantine mechanism. At the heart of ARQ is a combination of the following complementary mechanisms: (1) an event-driven outlier-quarantine loop that triggers on robustly detected tail behavior, (2) a robust center from the best half of the population to which quarantined candidates are gently repaired under feasibility projections, (3) local RTR-based replacement that preserves spatial diversity and avoids premature collapse, (4) archive-guided trial generation that blends current and archived differences while steering toward strong exemplars, and (5) success-history adaptation that self-regulates search from recent successes and reduces manual fine-tuning. Together, these parts sustain focused progress while periodically renewing diversity. Search pressure remains focused yet diversity is steadily replenished through micro-restarts when progress stalls, producing smooth and reliable improvement on noisy or rugged landscapes. In a comprehensive benchmark campaign spanning separable, ill-conditioned, multimodal, hybrid, and composition problems, ARQ was compared against leading state-of-the-art baselines, including top entrants and winners from CEC competitions under identical evaluation budgets and rigorous protocols. Across these settings, ARQ delivered competitive peak results while maintaining favorable average behaviour, thereby narrowing the gap between best and typical outcomes. Overall, this design positions ARQ as a robust choice for practical performance and consistency, providing a dependable tool that can meaningfully strengthen the methodological repertoire of the research community.

Keywords: differential evolution; metaheuristics; regenerative computing; hybridization; evolutionary algorithms; global optimization; mutation strategies

1. Introduction

We studied continuous minimization over a rectangulardomain

$$\min_{\mathbf{x}\in\Omega} f(\mathbf{x}), \qquad \Omega := [\ell_1, u_1] \times \cdots \times [\ell_D, u_D] \subset \mathbb{R}^D, \tag{1}$$

where $f : \Omega \to \mathbb{R}$ is accessed via point wise evaluations (derivatives may be unavailable).

Feasible set and constraints. If constraints are present,

$$\mathcal{F} = \{\mathbf{x} \in \Omega : \; g_i(\mathbf{x}) \leq 0, \; i = 1, \ldots, m, \;\; h_k(\mathbf{x}) = 0, \; k = 1, \ldots, p\}, \tag{2}$$

and the task becomes $\min_{\mathbf{x} \in \mathcal{F}} f(\mathbf{x})$. Box feasibility is enforced by the projection

$$\big(\Pi_\Omega(\mathbf{z})\big)_j = \min\{u_j, \max\{\ell_j, z_j\}\}, \quad j = 1, \ldots, D. \tag{3}$$

General constraints can be handled with a penalty functional:

$$\varphi_\rho(\mathbf{x}) = f(\mathbf{x}) + \rho\left(\sum_{i=1}^{m} [g_i(\mathbf{x})]_+^\nu + \sum_{k=1}^{p} |h_k(\mathbf{x})|^\nu\right), \qquad [a]_+ := \max\{a, 0\}, \; \rho > 0, \; \nu \geq 1. \tag{4}$$

Best-so-far and termination. Given an evaluation budget N_{fe}, an algorithm produces iterates $\{\mathbf{x}_t\}_{t=1}^{N_{\text{fe}}}$ and tracks the best-so-far value

$$f_{\min}(t) := \min_{1 \leq s \leq t} f(\mathbf{x}_s). \tag{5}$$

Termination occurs when either $t = N_{\text{fe}}$ or a target quality f_{target} is reached.

Performance reporting (best and mean). Over R independent runs, let $f_{\min}^{(r)} := f_{\min}(N_{\text{fe}})$ denote the per-run best-so-far value at budget. We report

$$\textbf{Best} = \min_{1 \leq r \leq R} f_{\min}^{(r)}, \qquad \textbf{Mean} = \frac{1}{R} \sum_{r=1}^{R} f_{\min}^{(r)}, \tag{6}$$

optionally complemented by average rank across a problem suite.

Following the problem statement, we position our contribution within the main families of continuous optimization. When smoothness and dependable directional information are available, Gradient Descent (GD) remains a natural choice and modern momentum or learning rate schedules extend its reach, although the dependence on differentiability and well behaved curvature can limit robustness on rugged, high dimensional, or strongly multimodal objectives [1]. Newton Raphson can achieve rapid local convergence by exploiting curvature provided stable Hessian information or proper regularization is available [2]. Deterministic derivative free local search such as the Nelder and Mead Simplex Method offers geometry aware progress without gradients and is effective in low to moderate dimensions, yet performance typically degrades as dimensionality and multimodality grow [3]. Stochastic search complements these tools. Monte Carlo simulation legitimizes randomized proposals and statistical summary to probe complex landscapes [4], while simulated annealing regulates uphill acceptance via a temperature schedule to promote principled escapes from local minima [5].

Population based methods expand these ideas with distributed exploration and information sharing. Genetic algorithms model recombination and selection over evolving candidate sets [6]. Particle Swarm Optimization coordinates velocity and position updates to disseminate useful search directions and has inspired stability oriented and diversity oriented refinements, for example CLPSO and modern parallel or optimized variants [7–10]. Ant Colony Optimization and Artificial Bee Colony bias sampling toward historically promising regions while preserving exploratory motion [11–13].

Within this ecosystem, Differential Evolution (DE) has become a central tool for real parameter search due to expressive trial generators and strong empirical performance across heterogeneous test beds [14]. Recent DE progress converges on several effective motifs. Memory based parameter adaptation as in SHADE reduces manual tuning [15]. Population size schedul-

ing as in L-SHADE improves efficiency on diverse suites [16]. Updates in the refined success history as in jSO further stabilize performance under tight evaluation budgets [17]. Multi-population flows, elite regeneration, and adaptive mutation or selection policies add resilience [18,19]. Orthogonal advances include coordinate system learning and eigen informed operators that mitigate ill conditioning and non-separability [20,21], while parallel implementations extend practicality under limited budgets [22]. Targeted DE modifications and implementation guidelines also systematize robust choices in scaling, crossover, and selection [23]. Beyond these families, recent DE advances report improved mutation and parameter control mechanisms [24], and a multi-operator ensemble L-SHADE with restart and local search for single-objective optimization [25], further broadening the toolkit. Additional ensemble/restart-style enhancements likewise expand the L-SHADE lineage [16].

ARQ is motivated by two converging lines: multi-strategy/co-evolutionary DE that strengthens the exploration-exploitation balance across heterogeneous landscapes [26], and robust-statistics principles where trimmed/median location and robust scale safeguard estimates under heavy tails [27]. Adjacent domains (e.g., systems leveraging contextual knowledge to stabilize signals under domain noise) point to the same practical need: preserve a reliable core while injecting measured diversity [28,29]. Within this perspective, we aim for a lightweight yet effective mechanism that turns extremes from destabilizing noise into controlled exploration stimuli.

We propose ARQ, a method aimed at dependable best-so-far progress and competitive mean performance under finite budgets. ARQ integrates pbest DE with an external archive in the JADE or SHADE lineage of archive assisted search and success history adaptation [15–17,30]. It applies success history updates of the scale and crossover rates through gain weighted statistics [15,16]. It enforces a neighborhood-aware replacement rule inspired by restricted tournament reasoning to preserve structured diversity [31]. Our contribution is an explicitly triggered tail-quarantine coupled with a trimmed center and micro-restarts, so that stabilization coexists with a steady, low-intensity injection of diversity. The key innovation is an outlier quarantine mechanism. Individuals identified in the extreme tail of the fitness distribution relative to a robust threshold are gently attracted toward a robust population center computed from the better half of the population, with mild perturbation and box projection $x' = \Pi_{\Omega}(c + \varepsilon)$. When a repaired candidate improves, it is accepted, and the previous one is archived. By trimming distribution tails and re-aligning weak individuals with promising regions, quarantine stabilizes average progress while maintaining the ability to intensify around incumbents. Complementary micro-restarts refresh a small fraction of the worst solutions around the current elite to provide targeted escapes without global resets, while mini-batch updates modulate per-step cost and support scalability [22]. We recommend ARQ in scenarios where one seeks a robust optimizer that consistently achieves strong best-so-far values and reliable mean performance across independent runs. By combining selective attraction, self tuning, neighborhood-aware selection, and principled repair and regeneration, ARQ addresses common failure modes such as premature convergence, tail accumulation of poor individuals, and inefficient restarts, and offers a dependable and broadly applicable tool for continuous optimization.

In relation to recent DE variants, multi-strategy and cooperative/co-evolutionary schemes show that combining complementary operators improves resilience on heterogeneous problems [26]. Applied strands in consumer and industrial settings also report restart- and scheduling-oriented enhancements, such as NSGA-III delay-recovery pipelines and AHMQDE-ACO co-optimization [32,33]. In parallel, micro/partial restarts provide controlled diversity without dissolving the incumbent core [34]. ARQ differs by making tail isolation event-driven (triggered only upon tail inflation), defining a practical robust center as the mean of the top-50%, and systematically coupling quarantine with micro-restarts. The result is a retain-refresh loop that improves convergence consistency with negligible overhead [27,34]. Evidence from business-

process control corroborates robustness under operational constraints [35], while e-commerce pricing studies highlight stability in fast-changing demand environments [36]. In transportation analytics, arrival-time prediction pipelines benefit from resilient learning components [37], and distributed privacy-robust learning demonstrates complementary stability in federated settings [38]. Finally, adjacent optimization tasks including trajectory generation and macro-analytics illustrate portability beyond canonical benchmarks [39,40].

ARQ introduces an explicitly event-driven outlier-quarantine loop triggered by tail behavior using a robust cutoff $\theta = Q3 + \alpha \cdot IQR$. Flagged samples are quarantined up to a proportion ρ and repaired around a robust center (mean of the best 50%), in tandem with a low-overhead micro-restart. Unlike success-history-only DE variants, this mechanism turns extremes from destabilizing noise into controlled exploration stimuli, yielding a smaller best–mean gap without sacrificing peak performance.

Finally, the remainder of this article is organized as follows: Section 2 details the ARQ method its control flow, parameter-policy with success-history updates, pbest/1/bin trial construction with archive support, neighborhood-aware RTR selection, and the quarantine and micro-restart mechanisms linking the pseudo-code routines to their roles in the overall algorithm. Section 3 describes the experimental setup and real-world benchmarks, the evaluation protocol, and reporting conventions, it then presents the parameter sensitivity analysis (Section 3.3), the complexity analysis with respect to dimension (Section 3.4), and a comparative performance study against strong baselines (Section 3.5). The article concludes with a discussion of findings and implications for robust black-box optimization (Section 4).

2. The ARQ Method

2.1. The ARQ Method with Its Mechanisms

Scope and references to preliminaries. We adopt the continuous minimization setting and feasibility operators introduced in Equations (1)–(4), which define the objective over the box domain, the projection operator onto the box, the optional penalty model for general constraints, and the base distance or normalization metric used throughout.

State and mini-batch size. At iteration t, the algorithm maintains a population P_t of size N and an external archive A_t, together with the incumbent best

$$x_{best}(t) \in \arg\min_{x \in P_0 \cup \cdots \cup P_t} f(x), \qquad f_{best}(t) = f\big(x_{best}(t)\big). \tag{7}$$

A mini-batch size m is used per iteration

$$m = \left\lceil \text{agent}_{fraction} \cdot N \right\rceil. \tag{8}$$

Trial generation. For a selected agent $x \in P_t$, the mutant is formed with a pbest/1/bin scheme with archive support

$$v = x + F\,(x_{pbest} - x) + F\,(r_1 - r_2), \quad r_1 \in P_t,\ r_2 \in P_t \cup A_t,\ r_1 \neq r_2 \neq x. \tag{9}$$

Then, binomial crossover with index guard produces u

$$u_j = \begin{cases} v_j, & \text{if rand}() < CR \text{ or } j = j_{rand}, \\ x_j, & \text{otherwise,} \end{cases} \qquad j_{rand} \sim \text{Unif}\{1, \ldots, D\}, \tag{10}$$

and the candidate is projected to the box, $y = \Pi_\Omega(u)$, using the projection already defined in Equation (2).

Success history parameter adaptation. On sampling calls the control rates are drawn around running means and clipped to bounds

$$F \sim \text{Cauchy}(\mu_F, \gamma) \text{ clipped to } [F_{lo}, F_{hi}], \qquad CR \sim \mathcal{N}(\mu_{CR}, \sigma^2) \text{ clipped to } [0, 1]. \tag{11}$$

Let $\mathcal{S}$ index successful trials within the current iteration, with gains $g_i = f(\text{parent}_i) - f(\text{child}_i) > 0$ and normalized weights $w_i = g_i / \sum_{k \in \mathcal{S}} g_k$.

The running means are updated by

$$\mu_F \leftarrow \frac{\sum_{i \in \mathcal{S}} w_i F_i^2}{\sum_{i \in \mathcal{S}} w_i F_i}, \qquad \mu_{CR} \leftarrow \sum_{i \in \mathcal{S}} w_i CR_i, \tag{12}$$

and remain unchanged if $\mathcal{S} = \emptyset$.

Local replacement via RTR. Let $d_{BN}(\cdot, \cdot)$ denote the bounds normalized Euclidean distance introduced in the preliminaries, cf. Equation (4). The replacement rule is

$$\text{if } f(y) < f(x): \quad x \leftarrow y, \qquad \mathsf{Arch} \leftarrow \mathsf{Arch} \cup \{x_{\text{old}}\}. \tag{13}$$

$$\text{else draw } Q \subset P_t \text{ with } |Q| = \text{rtr_pool}, \quad q_{star} \in \arg\min_{q \in Q} d_{BN}(y, q). \tag{14}$$

$$\text{if } f(y) < f(q_{star}): \quad q_{star} \leftarrow y, \qquad \mathsf{Arch} \leftarrow \mathsf{Arch} \cup \{q_{star}^{\text{old}}\}. \tag{15}$$

Outlier quarantine. Using the fitness values of the current population, compute quartiles Q_1, Q_3 and the interquartile range $\text{IQR} = Q_3 - Q_1$ as in the preliminaries. Define the robust threshold

$$\theta = Q_3 + \alpha \cdot \text{IQR}, \qquad O = \{x \in P_t \mid f(x) \geq \theta\}. \tag{16}$$

Select a subset O_ρ with $|O_\rho| = \lfloor \rho \cdot |O| \rfloor$ and a robust center c as the mean of the best fifty percent of P_t. For each $x \in O_\rho$ propose a repair

$$\begin{aligned} x' &= \Pi_\Omega(c + \epsilon), \qquad \epsilon \sim \text{small zero-mean perturbation}, \\ &\text{accept if } f(x') < f(x), \quad \text{archive the displaced point.} \end{aligned} \tag{17}$$

Targeted micro-restart. Let W be the set containing a fraction w of the worst individuals. For each $x \in W$ propose a restart around the incumbent

$$x' = \Pi_\Omega(x_{best} + \eta), \qquad \eta \sim \mathcal{N}(0,\, \sigma^2 \cdot (u - \ell)^2), \tag{18}$$

and accept if $f(x') < f(x)$, archiving the displaced point. Activation of this mechanism is controlled by the dedicated stagnation trigger.

Termination and reporting. The loop continues while $\textit{iteration} < T$. Upon exit the method returns

$$x_{best}, \qquad f_{best} = f(x_{best}), \tag{19}$$

and, across R independent runs, we summarize the end-of-budget best-so-far via Equation (6).

Building on the preliminaries and problem conventions, we present ARQ from the overall control flow to its constituent mechanisms. We began with the overall control flow (Algorithm 1 and Figure 1), detailing initialization, the iterative cycle with mini-batch agent updates, success-history parameter adaptation, neighborhood-aware RTR selection, outlier quarantine, and the targeted micro-restart, terminating with the incumbent best. We then analyzed the four core subroutines that implement ARQ's key building blocks:

ParameterPolicy (Algorithm 2 and Figure 2) for sampling and gain-weighted updates of μ_F and μ_{CR}. TrialGeneration (Algorithm 3 and Figure 3) for pbest/1/bin trial construction with archive support and box projection. SelectionRTR (Algorithm 4 and Figure 4) for local replacement via restricted tournament. PopulationMaintenance (Algorithm 5 and Figure 5), which introduces the outlier-quarantine innovation together with the micro-restart mechanism. We concluded by mapping pseudo-code parameters to implementation controls and highlighting how these components interact to ensure stability and strong performance across diverse search regimes.

Algorithm 1 ARQ main pseudo-code

INPUT
- f: objective function
- Ω: box domain [ℓ1,u1] … [ℓD,uD]
- N: population size
- T: max evaluations
- p: pbest fraction
- $agent_{fraction}$: fraction updated per step
- μF: mean scale factor
- μCR: mean crossover rate
- α: quarantine threshold coefficient
- ρ: repaired outliers fraction
- w: worst fraction for micro-restart

OUTPUT
- x_{best}, f_{best}

INITIALIZATION
- Initialize P with N random solutions in Ω and evaluate f
- Set archive $\leftarrow \varnothing$ and means μF, μCR
- Extract x_{best} f_{best} from the best member of P
- Set $evals \leftarrow N$

ARQ main pseudo-code

01 While $evals < T$ do
02 Compute $m \leftarrow ceil(agent_{fraction} \cdot N)$
03 Draw random subset $A \subset P$ with $|A| = m$
04 Clear S_F, S_{CR}, S_{gain}
05 For each $x \in A$ do
06 Obtain F, CR from ParameterPolicy in sampling mode with input (μF, μCR)
07 Obtain y from TrialGeneration with input (x, P, $archive$, p, F, CR, Ω)
08 Set $fy \leftarrow f(y)$
09 Set $evals \leftarrow evals + 1$
10 Obtain x_{new}, fx_{new} from SelectionRTR with input (x, y, fy, P)
11 If $fx_{new} < f(x)$ then
12 Append F to S_F
13 Append CR to S_{CR}
14 Append $f(x) - fx_{new}$ to S_{gain}
15 Endif
16 If $fx_{new} < f_{best}$ then
17 Set $x_{best} \leftarrow x_{new}$
18 Set $f_{best} \leftarrow fx_{new}$
19 Endif
20 Endfor
21 Update (μF, μCR) $\leftarrow$ via ParameterPolicy. update (S_F, S_{CR}, S_{gain}, μCR)
22 Apply PopulationMaintenance with input (P, $archive$, Ω, x_{best}, α, ρ, w)
23 Endwhile
24 Return x_{best}, f_{best}

Algorithm 1 together with Figure 1 summarizes the control flow of ARQ. The process starts with population and archive initialization and with current means for F and CR. At each iteration, the termination condition iteration < T is assessed. If budget remains, a mini-batch of agents is selected and a compact improvement cycle is executed. This cycle samples parameters from the success history policy, constructs a pbest/1/bin trial with archive support and box projection, evaluates the trial, and applies neighborhood-aware RTR selection. Successful updates are recorded so that they inform the subsequent mean update. After the mini-batch is processed, the parameter means are updated using gain weighted statistics, outlier quarantine is applied to robustly handle poor solutions, and a targeted micro-restart is performed only when needed around the current incumbent. When the termination condition fails, the algorithm explicitly returns x_{best} and f_{best}. This design couples parameter adaptivity, local selection pressure, and controlled population refresh to deliver steady progress while avoiding derailment by outliers or premature convergence.

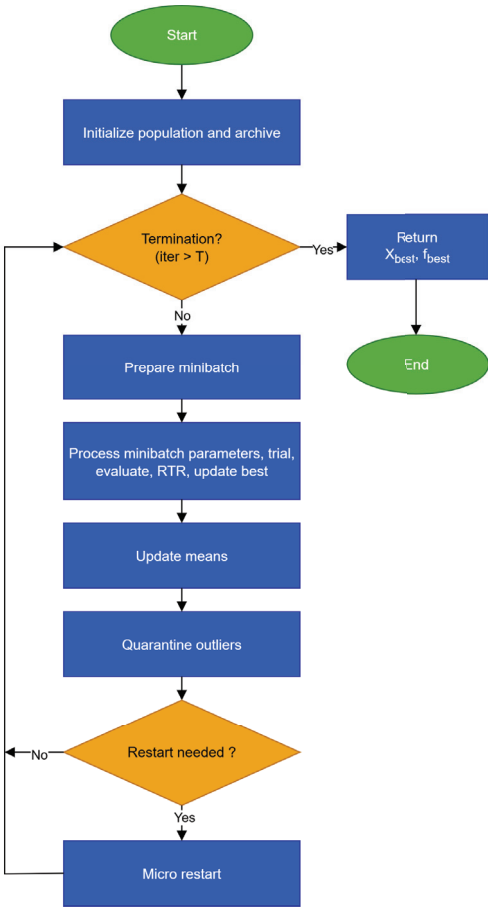


Figure 1. The ARQ method, an accelerated evolutionary optimization method with success-history adaptation, RTR selection, outlier quarantine, and targeted micro-restart.

Algorithm 2 and Figure 2 define the policy of ARQ's parameter. The routine has two clear modes. In sampling mode, it draws F and CR around their current means and clips them to preset bounds, supplying the improvement cycle with a controlled blend of exploration and exploitation. In update mode, it gathers the successful trials of the current cycle, forms gains as weights, and computes new means, using a weighted Lehmer mean for F and a weighted arithmetic mean for CR. This rewards parameters that delivered substantive improvement while damping isolated lucky events. If no successes are recorded, the means are kept unchanged to preserve stability. The result is fast adaptation when prof-

itable directions are coherent and conservative updates when improvements are scattered, aligning the learning pace with the search landscape and avoiding excessive oscillations.

Algorithm 2 Subroutine ParameterPolicy sampling and update of F, CR

INPUT $(\mu_F, \mu_{CR}, S_F, S_{CR}, S_{gain}, mode)$
OUTPUT $sampling \rightarrow (F, CR)$ $update \rightarrow (\mu_F, \mu_{CR})$
01 If mode = sampling then
02 Draw F from Cauchy centered at μ_F and clip to $0 < F \leq 1$
03 Draw CR from Normal with mean μ_{CR} and clip to $0 \leq CR \leq 1$
04 Return (F, CR)
05 Endif
06 If mode = update then
07 If S_{gain} empty then
08 Return (μ_F, μ_{CR})
10 Endif
11 Normalize weights w from S_{gain} so that sum $w = 1$
12 Set $\mu_F \leftarrow (sumw \cdot S_F^2)$ divided by (sum $w \cdot S_F$)
13 Set $\mu_{CR} \leftarrow$ sum $w \cdot S_{CR}$
14 Return (μ_F, μ_{CR})
15 Endif

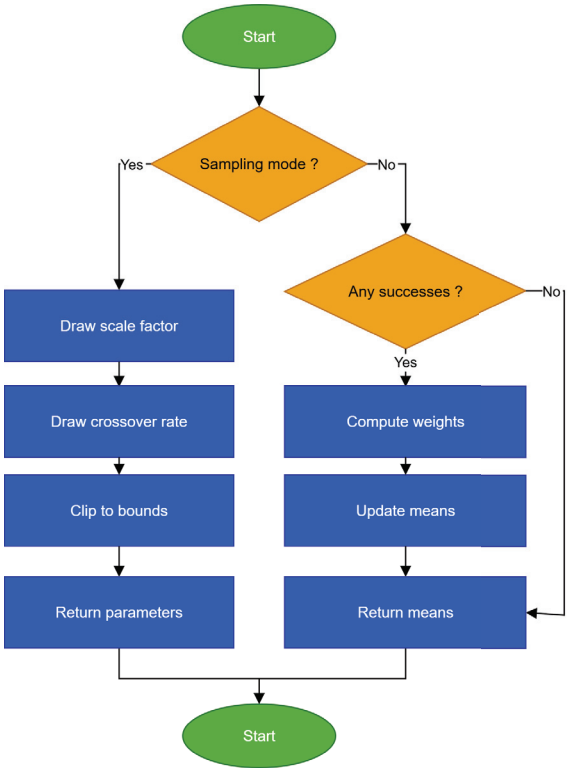


Figure 2. ParameterPolicy: sampling of control rates and success-history update of running means (fast self-tuning for coherent gains, conservative for scattered ones).

Algorithm 3 and Figure 3 show details of the trial construction pipeline that feeds ARQ's local improvement. A guide is first drawn from the top fraction of the population so that movement is biased toward promising regions. Two additional guides are then selected, one from the current population and one from either the population or the archive, injecting diverse directions. On these anchors, a pbest/1/bin style mutation is formed that blends an attraction toward the elite guide with a difference of two solutions, striking a balance between exploitation and exploration. Next, component-wise binomial crossover is applied

with a guarantee that at least one component originates from the mutant, preventing stagnation. Finally, the candidate is projected back to the box to enforce feasibility, which stabilizes behavior across variable scales. The archive broadens directional cues when progress stalls, and the projection keeps the process constraint compliant without sacrificing trial diversity.

Algorithm 3 Subroutine TrialGeneration pbest/1/bin with archive and projection

INPUT $(x, P, archive, p, F, CR, \Omega)$
OUTPUT (y)
01 Choose x_{pbest} from the top p fraction of P
02 Choose $r_1 \in P$ with $r_1 \neq x$
03 Choose $r_2 \in (P \cup archive)$ with $r_2 \neq x$ and $r_2 \neq r_1$ if feasible
04 Compute $v \leftarrow x + F \cdot (x_{pbest} - x) + F \cdot (r_1 - r_2)$
05 Sample j_{rand} uniformly from {1,2,...,D}
06 For each coordinate j do
07 If $rand() < CR$ or $j = j_{rand}$ then
08 set $u_j \leftarrow v_j$
09 else
10 set $u_j \leftarrow x_j$
11 Endif
12 Endfor
13 Project u to Ω componentwise to obtain y
14 Return (y)

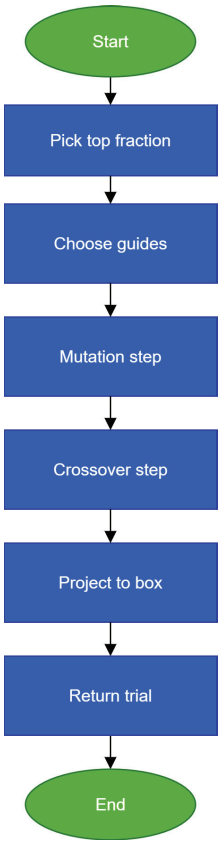


Figure 3. TrialGeneration: pbest/1/bin trial construction with archive support and box projection (targeted drift toward strong exemplars).

The selection routine in Algorithm 4 and Figure 4 implements a restricted tournament to keep replacements local and meaningful. The trial is first tested against its parent and,

if superior, it directly replaces the parent while the old solution is archived. Otherwise, a small candidate pool is drawn to maintain computational efficiency. From this pool, the nearest neighbor under bounds normalized distance is identified, making the locality metric scale invariant. The trial is compared only to this neighbor and replaces it if better, with the displaced solution pushed to the archive. If not, the parent is retained. This mechanism concentrates selection pressure where it yields real gains, preserves the spatial structure of the population, mitigates premature convergence, and sustains diversity without incurring substantial overhead.

Algorithm 4 Subroutine SelectionRTR local replacement with restricted tournament

INPUT (x, y, f_y, P)
OUTPUT $(x_{out}, f_{x_{out}})$
01 If $f_y < f(x)$ then
02 Replace x by y in P and archive old x
03 Return (y, f_y)
04 Endif
05 Draw fixed size pool $Q \subset P$
06 Find $q^{star} \in Q$ with minimum bounds normalized distance to y
07 If $f_y < f(q^{star})$ then
08 Replace q^{star} by y and archive oldq^{star}
09 Return (y, f_y)
10 Else
11 Return $(x, f(x))$
12 Endif

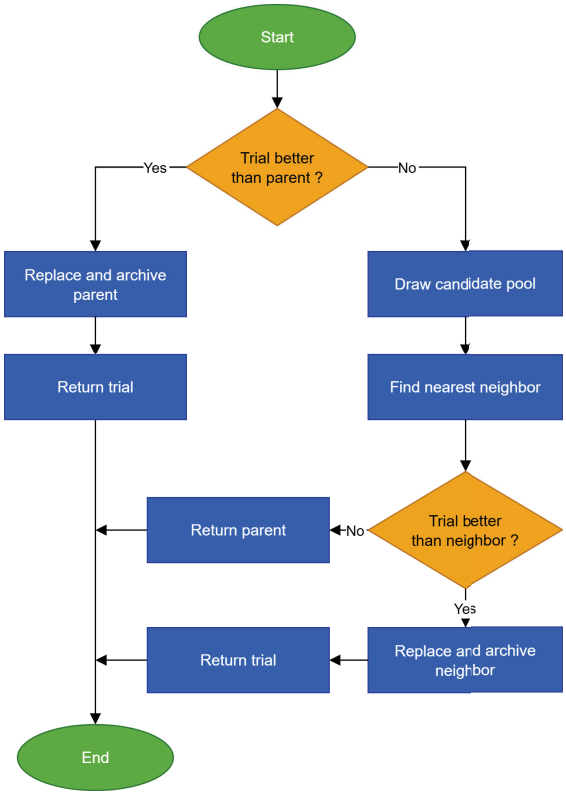


Figure 4. SelectionRTR: local replacement via restricted tournament with nearest-neighbor check (archiving displaced solutions for future diversity).

Unlike the previous routines, Algorithm 5 in Figure 5 explicitly shapes the population. The first phase identifies outliers using the threshold $\theta \leftarrow Q_3 + \alpha \cdot IQR$ and selects a subset

for repair. Each repair proposes a candidate near a robust center computed from the best half of the population and is accepted only if it yields a measurable improvement, with the displaced solution archived. The second phase targets a fraction of the worst individuals for a micro-restart around the current incumbent, using mild, box-scaled perturbations, replacements are again conditional and archived. Together, outlier quarantine and micro-restart prevent the accumulation of toxic points, continually steer the population geometry toward productive regions, and provide a controlled recovery mechanism when progress slows.

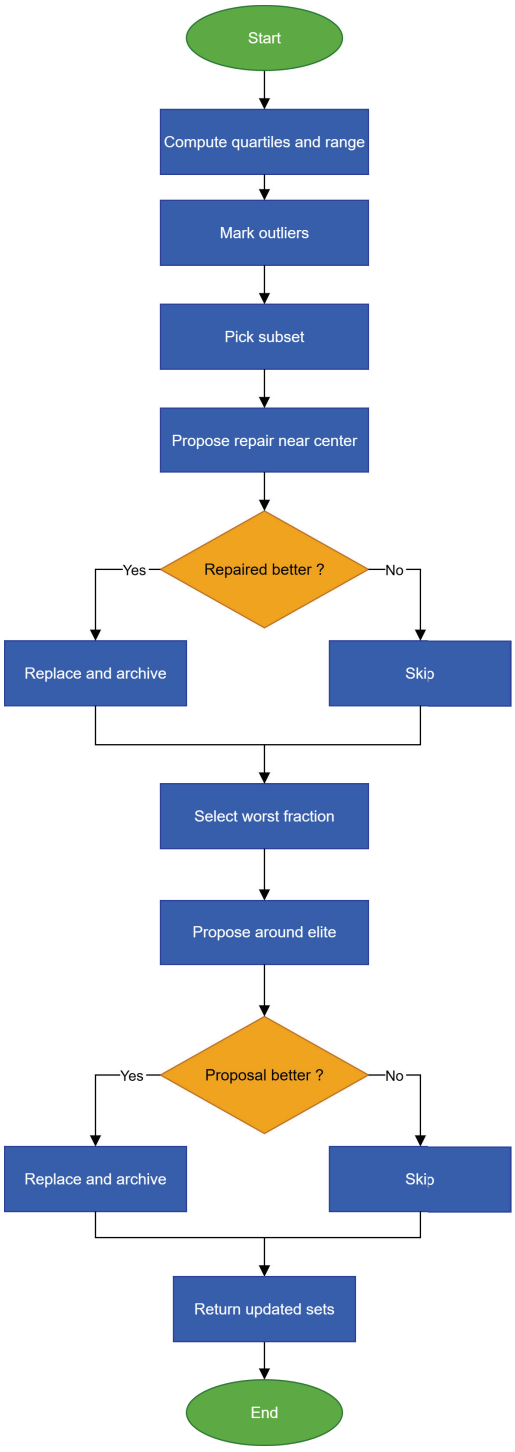


Figure 5. PopulationMaintenance: outlier quarantine and targeted micro-restart (controlled re-injection/refresh).

Algorithm 5 Subroutine PopulationMaintenance outlier quarantine and micro-restart

INPUT (P, *archive*, Ω, x_{best}, α, ρ, w)
OUTPUT (P, *archive*)
01 Compute Q_1, Q_3, IQR from fitness values and set $\theta \leftarrow Q_3 + \alpha \cdot IQR$
02 Set $O \leftarrow \{\, x \in P$ with $f(x) \geq \theta \,\}$
03 Compute center c as mean of the best fifty percent of P
04 Choose random subset O_ρ with $|O_\rho| = floor(\rho \cdot |O|)$
05 For each $x \in O_\rho$ do
06 Set $x_{new} \leftarrow c + \epsilon$ and project to Ω
07 If $f(x_{new}) < f(x)$ then
08 replace and *archive* old one
09 Endif
10 Endfor
11 Choose fraction w of the worst individuals of P
12 For each x in that fraction do
13 Set $x_{new} \leftarrow x_{best}$ + Normal$(0, \sigma \cdot (u - \ell))$ and project to Ω
14 If $f(x_{new}) < f(x)$ then
15 replace and *archive* old one
16 Endif
17 Endfor
18 Return (P, *archive*)

2.2. Design and Parameterization

The outlier quarantine mechanism is introduced to stabilize the search whenever a few very distant samples stretch the solution cloud and weaken the convergence signals. In this situation, standard means and spreads become brittle because the tails dominate. Quarantine temporarily isolates the most extreme portion of the population so that the core can update a reliable location reference and then reinsert the isolated points with a mild, controlled perturbation.

The quarantine proportion ρ is chosen to provide a simple and stable balance between robustness and diversity. We use a small-to-moderate ρ so that only the far tail is removed while the effective population remains intact. In practice, ρ adapts to population size and current dispersion: when the cloud is tight, a very small ρ suffices because outliers are rare, when tails visibly inflate, a slightly larger ρ removes disruptive noise without drying out exploration. This mirrors the logic of a trimmed mean, where clipping a small top slice stabilizes the location estimate.

The perturbation scale q_{sigma} determines how strongly quarantined points are reintroduced. We set it relative to the cloud's current geometry, using robust dispersion measures such as the median absolute deviation or a bounds-normalized per-coordinate standard deviation. When the distribution is compact, a small q_{sigma} yields fine local search around the center. When the distribution is scattered or the search is in early phases, a larger q_{sigma} helps recapture underrepresented regions. Consequently, $x' = \Pi_\Omega(c + \varepsilon)$ with zero-mean ε always scales to the landscape and avoids unnecessarily large jumps.

We define the robust center as the simple mean of the top half of individuals by fitness because this estimate remains sensitive to improvement and resistant to outliers. The top fifty percent is large enough to reduce random noise and clean enough to exclude low-quality or extreme points. Since the subset is formed by fitness ranking, it concentrates near the promising basin, its plain mean becomes a practical, stable location reference without requiring sophisticated robust estimators.

Synergy with micro-restart is direct and functional. Quarantine suppresses the heavy tail and protects the center estimate from jitter, allowing exploitative operators to move steadily toward the active basin. Micro-restart re-injects small but meaningful diversity near

the robust center or along uncertain directions, without dissolving the core. Alternating the two makes the search operate in a retain-refresh cycle: first the tail is smoothed, then the center is updated reliably, and finally exploration resumes with a measured step.

Parameter sensitivity clarifies each effect. Increasing ρ stabilizes references faster but can reduce diversity unless backed by micro-restart. Increasing q_{sigma} helps rediscover neighborhoods that would otherwise be missed, but if too large, it weakens local improvement. Using fifty percent to compute the center proved a dependable balance because it keeps statistical efficiency high while remaining outlier-resistant, small deviations around this level do not change qualitative behavior, making it a safe and portable default.

Operationally, the mechanism triggers periodically or when tail inflation is detected via bounds-normalized distances from the center and fitness ranking. Quarantined points are removed, resampled around the center at scale q_{sigma}, and reinserted under an acceptance rule that respects the objective while archiving displaced points. In this way we convert extremes from a source of destabilization into controlled exploration stimuli.

The method's novelty lies in tying three ideas into a light yet effective loop: it makes tail isolation explicit and actionable at the moment it harms estimation, it defines a practical robust center via a simple trimmed average aligned with improvement, and it systematically couples this stabilization with micro-restarts to maintain a steady trickle of fresh variability. The result is a retain-refresh process that improves convergence consistency without heavy adaptation or hyperparameter overhead and that integrates cleanly with existing evolutionary operators. See Figure 6 for the concise flow of quarantine, repair, and reinsertion.

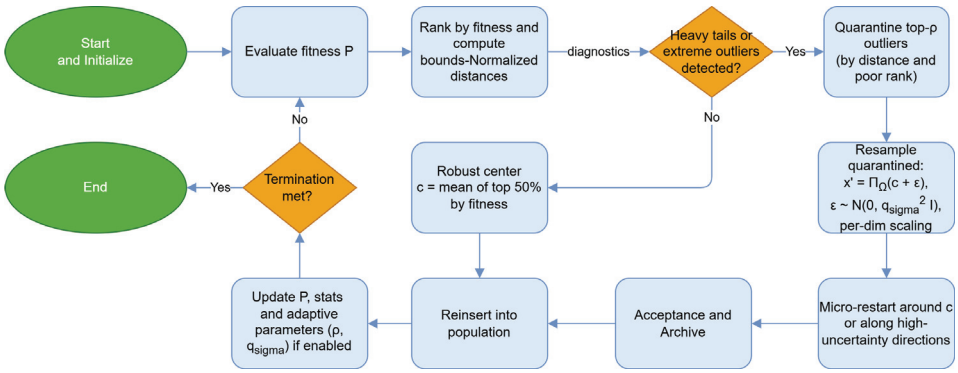


Figure 6. ARQ flow-detect tail, isolate, resample, and reinsert.

3. Experimental Setup and Benchmark Results

3.1. Setup

The following Tables 1 and 2 summarize all experimental settings and how results are reported.

Table 1. ARQ parameters in the pseudo-code.

Name	Value	Description
N	100	Population size
T	150,000	Maximum evaluations
p	0.12	Top fraction for pbest
$agent_{fraction}$	0.60	Agents updated per step
μ_F	0.6	Mean scale factor
μ_{CR}	0.85	Mean crossover rate.

Table 1. *Cont.*

Name	Value	Description
α	1.0	Outlier threshold coeff in $\theta = Q3 + \alpha \cdot IQR$
ρ	0.08	Outliers repaired fraction
w	0.08	Worst frac for micro-restart
sh_c	0.10	SH learning rate
F_{lo}	0.05	F sample min
F_{hi}	1.40	F sample max
CR_{low}	0.0	CR sample min
CR_{hi}	1.0	CR sample max
$archive_{rate}$	1.5	Archive capacity
rtr_k	7	RTR neighborhood size
rtr_{pool}	14	RTR pool size
rtr_{min}	0.0	RTR min gain
q_{sigma}	0.10	Quarantine perturb scale
$stagnation_{trigger}$	24	No-improve trigger
r_{sigma}	0.18	Restart noise stdev

Table 2. Parameters of other methods.

Name	Value	Description
N	100	Population size for all methods
	JSO	
NP_{factor}	18	Initial population size $NP_0 = NP_{factor} \times Dim$
H	20	Memory size for success-history means MF/MCR
$pbest$	0.11	Fraction for p-best selection (top-p set).
$archive_{rate}$	1	Archive size as multiple of NP0 (i.e., archiveMax = archive_rate × NP0).
	CLPSO	
cl_p	0.3	Comprehensive learning probability
$cognitive_{weight}$	1.49445	Cognitive weight
$inertia_{weight}$	0.729	Inertia weight
$mutation_{rate}$	0.01	Mutation rate
$social_{weight}$	1.49445	Social weight
	CMA-ES	
N_{CMA-ES}	$4 + \lfloor 3 \cdot \log(dim) \rfloor$	Population size
	EA4Eig	
$archive_{size}$	100	Archive size for JADE-style mutation
$eig_{interval}$	5	Recompute eigenbasis every k iterations
CR_{max}	1	Upper bound for CR
F_{max}	1	Upper bound for F

Table 2. *Cont.*

Name	Value	Description
CR_{min}	0	Lower bound for CR
F_{min}	0.1	Lower bound for F
p_{best}	0.2	pbest fraction (current-to-pbest/1/bin)
CR_{tau}	0.1	Self-adaptation prob. for CR
F_{tau}	0.1	Self-adaptation prob. for F
		mLSHADE_RL
$archive_{size}$	500	Archive size
$Memory_{size}$	10	Success-history memory size
$Population_{min}$	4	Minimum population size
$p_{best_{max}}$	0.2	Maximum pbest fraction
$p_{best_{min}}$	0.05	Minimum pbest fraction
		SaDE
CR_{sigma}	0.1	Std for CR sampling
F_{gamma}	0.1	Scale for Cauchy F sampling
CR_{init}	0.5	Initial CR mean
F_{init}	0.7	Initial F mean
$learning_{period}$	25	Iterations per adaptation window
		UDE3
$Population_{min}$	4	Minimum population size.
$Memory_{size}$	10	Success-history memory size
$archive_{size}$	100	Archive size
$p_{best_{min}}$	0.05	Minimum pbest fraction
$p_{best_{max}}$	0.2	Maximum pbest fraction.
TRIDENT-DE. The parameters for this method are mentioned in Table in [41].		

Protocol and configurations. Table 2 enumerates the hyper-parameters of all competing methods to enable strict reproducibility under a common budget. Population size is fixed at $N = 100$. A single termination rule is enforced function evaluations (FEs) and every solver runs under the same budget T. Method-specific controls (e.g., CLPSO comprehensive-learning probability, CMA-ES population and coefficients, EA4Eig JADE-style parameters, mLSHADE_RL and UDE3 [42] success-history memories and pbest ranges, SaDE adaptation schemes) follow the literature and public reference implementations. Complete settings appear in Tables 1 and 2.

Implementation and environment. All algorithms including the proposed method and baselines were implemented in optimized ANSI C++ and integrated into the open-source OPTIMUS framework [43]. Source code: https://github.com/itsoulos/GLOBALOPTIMUS (accessed on 15 October 2025). Builds used Debian 12.12 with GCC 13.4.

Hardware. Experiments ran on a high-performance node with an AMD Ryzen 9 5950X (16 cores, 32 threads) and 128 GB DDR4 memory, under Debian Linux.

Evaluation protocol. Each benchmark function was executed 30 independent runs with distinct random seeds to capture stochastic variability. With a fixed FE budget, comparisons are made at the same for all solvers.

Metrics and reporting. The primary outcomes are the best and mean objective values at T across 30 runs per test function. In the ranking tables, 1st place entries are highlighted green and 2nd place entries blue. Parameter choices exactly follow Tables 1 and 2.

3.2. Benchmark Functions

Table 3 compiles the real-world optimization problems used in our evaluation. For each case, we report a brief description, the dimensionality and variable types (continuous/mixed-integer), the nature and count of constraints (inequalities/equalities), salient landscape properties (nonconvexity and multi-modality), as well as the evaluation budget and comparison criteria. The set spans, indicatively, mechanical design, energy scheduling, process optimization, and parameter estimation with black-box simulators, ensuring that conclusions extend beyond synthetic test functions. Where applicable, we also note any normalizations or constraint reformulations adopted for fair comparison.

Table 3. The real world benchmark functions used in the conducted experiments.

Problem	Formula	Dim	Constraints/Bounds
Parameter Estimation for Frequency-Modulated Sound Waves [44–46]	$\min_{x\in[-6.4,6.35]^6} f(x) = \frac{1}{N}\sum_{n=1}^{N}\left\lvert y(n;x) - y_{\text{target}}(n)\right\rvert^2$ $y(n;x) = x_0 \sin(x_1 n + x_2 \sin(x_3 n + x_4 \sin(x_5 n)))$	6	$x_i \in [-6.4, 6.35]$
Lennard-Jones Potential (atoms: 10, 13, 38) [47]	$\min_{x\in\mathbb{R}^{3N-6}} f(x) = 4\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\left[\left(\frac{1}{r_{ij}}\right)^{12} - \left(\frac{1}{r_{ij}}\right)^{6}\right]$	24, 43, 108	$x_0 \in (0,0,0)$ $x_1, x_2 \in [0,4]$ $x_3 \in [0,\pi]$ x_{3k-3} x_{3k-2} $x_i \in [-b_k, +b_k]$
Bifunctional Catalyst Blend Optimal Control [48–50]	$\frac{dx_1}{dt} = -k_1 x_1$, $\frac{dx_2}{dt} = k_1 x_1 - k_2 x_2 + k_3 x_2 + k_4 x_3$, $\frac{dx_3}{dt} = k_2 x_2$, $\frac{dx_4}{dt} = -k_4 x_4 + k_5 x_5$, $\frac{dx_5}{dt} = -k_3 x_2 + k_6 x_4 - k_5 x_5 + k_7 x_6 + k_8 x_7 + k_9 x_5 + k_{10} x_7$ $\frac{dx_6}{dt} = k_8 x_5 - k_7 x_6$, $\frac{dx_7}{dt} = k_9 x_5 - k_{10} x_7$ $k_i(u) = c_{i1} + c_{i2} u + c_{i3} u^2 + c_{i4} u^3$	1	$u \in [0.6, 0.9]$
Optimal Control of a Non-Linear Stirred Tank Reactor [51–53]	$J(u) = \int_0^{0.72}\left[x_1(t)^2 + x_2(t)^2 + 0.1u^2\right]dt$ $\frac{dx_1}{dt} = -2x_1 + x_2 + 1.25u + 0.5\exp\left(\frac{x_1}{x_1+2}\right)$ $\frac{dx_2}{dt} = -x_2 + 0.5\exp\left(\frac{x_1}{x_1+2}\right)$ $x_1(0) = 0.9, \quad x_2(0) = 0.09, t \in [0, 0.72]$	1	$u \in [0, 5]$
Tersoff Potential for Model Si (B) [54,55]	$\min_{\mathbf{x}\in\Omega} f(\mathbf{x}) = \sum_{i=1}^{N} E(\mathbf{x}_i)$ $E(\mathbf{x}_i) = \frac{1}{2}\sum_{j\neq i} f_c(r_{ij})\left[V_R(r_{ij}) - B_{ij} V_A(r_{ij})\right]$ where $r_{ij} = \lVert\mathbf{x}_i - \mathbf{x}_j\rVert$, $V_R(r) = A\exp(-\lambda_1 r)$ $V_A(r) = B\exp(-\lambda_2 r)$ $f_c(r)$: cutoff function with $f_c(r)$: angle parameter	30	$x_1 \in [0,4]$ $x_2 \in [0,4]$ $x_3 \in [0,\pi]$ $x_i \in \left[\frac{4(i-3)}{4}, 4\right]$
Tersoff Potential for model Si (C) [54,55]	$\min_{\mathbf{x}} V(\mathbf{x}) = \sum_{i=1}^{N}\sum_{j>i}^{N} f_C(r_{ij})\left[a_{ij} f_R(r_{ij}) + b_{ij} f_A(r_{ij})\right]$ $f_C(r) = \begin{cases} 1, & r < R - D \\ \frac{1}{2} + \frac{1}{2}\cos\left(\frac{\pi(r-R+D)}{2D}\right), & \lvert r - R\rvert \le D \\ 0, & r > R + D \end{cases}$ $f_R(r) = A\exp(-\lambda_1 r)$ $f_A(r) = -B\exp(-\lambda_2 r)$ $b_{ij} = \left[1 + (\beta^n)\zeta_{ij}^n\right]^{-1/(2n)}$ $\sum_{k\neq i,j} f_C(r_{ik}) g(\theta_{ijk}) \exp\left[\lambda_3^3 (r_{ij} - r_{ik})^3\right]$	30	$x_1 \in [0,4]$ $x_2 \in [0,4]$ $x_3 \in [0,\pi]$ $x_i \in \left[\frac{4(i-3)}{4}, 4\right]$

Table 3. *Cont.*

Problem	Formula	Dim	Constraints/Bounds
Spread Spectrum Radar Polyphase Code Design [56]	$\min_{x\in X}\ f(x)=\max\{\lvert\varphi_1(x)\rvert,\lvert\varphi_2(x)\rvert,\ldots,\lvert\varphi_m(x)\rvert\}$ $X=\{x\in\mathbb{R}^n \mid 0\le x_j\le 2\pi,\ j=1,\ldots,n\}m=2n-1$ $\varphi_j(x)=\begin{cases}\sum_{k=1}^{n-j}\cos(x_k-x_{k+j}) & \text{for } j=1,\ldots,n-1\\ n & \text{for } j=n\\ \varphi_{2n-j}(x) & \text{for } j=n+1,\ldots,2n-1\end{cases}$ $\varphi_j(x)=\sum_{k=1}^{n-j}\cos(x_k-x_{k+j}),\quad j=1,\ldots,n-1$ $\varphi_n(x)=n,\ \varphi_{n+\ell}(x)=\varphi_{n-\ell}(x),\quad \ell=1,\ldots,n-1$	20	$x_j\in[0,2\pi]$
Transmission Network Expansion Planning [57]	$\min\sum_{l\in\Omega}c_l n_l+W_1\sum_{l\in OL}\lvert f_l-\bar{f}_l\rvert+W_2\sum_{l\in\Omega}\max(0,n_l-\bar{n}_l)$ $Sf=g-d$ $f_l=\gamma_l n_l\Delta\theta_l,\quad\forall l\in\Omega$ $\lvert f_l\rvert\le\bar{f}_l n_l,\quad\forall l\in\Omega$ $0\le n_l\le\bar{n}_l,\quad n_l\in\mathbb{Z},\quad\forall l\in\Omega$	7	$0\le n_i\le\bar{n}_l$ $n_i\in\mathbb{Z}$
Electricity Transmission Pricing [58]	$\min_x\ f(x)=\sum_{i=1}^{N_g}\left(\frac{C_i^{gen}}{P_i^{gen}}-R_i^{gen}\right)^2+\sum_{j=1}^{N_d}\left(\frac{C_j^{load}}{P_j^{load}}-R_j^{load}\right)^2$ $\sum_j GD_{i,j}+\sum_j BT_{i,j}=P_i^{gen},\quad\forall i$ $\sum_i GD_{i,j}+\sum_i BT_{i,j}=P_j^{load},\quad\forall j$ $GD_{i,j}^{max}=\min(P_i^{gen}-BT_{i,j},\ P_j^{load}-BT_{i,j})$	126	$GD_{i,j}\in[0,GD_{i,j}^{max}]$
Circular Antenna Array Design [59]	$\min_{r_1,\ldots,r_6,\varphi_1,\ldots,\varphi_6}\quad f(\mathbf{x})=\max_{\theta\in\Omega}AF(\mathbf{x},\theta)$ $AF(\mathbf{x},\theta)=\left\lvert\sum_{k=1}^{6}\exp\left(j\left[2\pi r_k\cos(\theta-\theta_k)+\varphi_k\frac{\pi}{180}\right]\right)\right\rvert$	12	$r_k\in[0.2,1]$ $\varphi_k\in[-180,180]$
Cassini 2: Spacecraft Trajectory Optimization Problem [60]	$x=\left[t_0,\ \Delta t_1,\ldots,\Delta t_5,\ p_1,\ldots,p_4\right]^\top,$ $\min_x J(x)=\Delta v_{\text{launch}}+\sum_{\ell=1}^{4}\Delta v_\ell^{\text{DSM}}+\Delta v_{\text{arrival}},$ $E\ \to\ V\ \to\ E\ \to\ V\ \to\ J\ \to\ S$ $t_0\in[850,\ 2000],\qquad\Delta t_\ell\in[20,\ 1200],\qquad p_\ell\in[1.0,\ 10.0].$	10	$r_{p,\ell}\ge R_{P_\ell}+h_{\min,P_\ell},$ $\delta_\ell\le\delta_{\max,P_\ell},$ $t_0^{\min}\le t_0\le t_0^{\max},$ $\Delta t_\ell^{\min}\le\Delta t_\ell\le\Delta t_\ell^{\max}.$
Wireless Coverage Antenna Placement [61,62]	$x_i,\ y_i\in\mathbb{R}\quad(\text{position}),P_i\in\mathbb{R}_{\ge0}\quad(\text{transmission power}).$ $S_{iu}(x_i,y_i,P_i)\ =\ \frac{G\,P_i}{\left((x_i-x_u)^2+(y_i-y_u)^2\right)^{\alpha/2}+\varepsilon},$ $\min_{x,y,P}\ \underbrace{\sum_{u\in\mathcal{U}}\left[\tau-S_u(x,P)\right]_+}_{\text{coverage deficiency}}+\lambda_P\sum_{i=1}^{N}P_i+,$ $\lambda_O\sum_{u\in\mathcal{U}}\sum_{i<j}\left[\min\{S_{iu},S_{ju}\}-\kappa\right]_+$ $S_u(x,P)\ =\ \max_{i=1,\ldots,N}S_{iu}(x_i,y_i,P_i).$	30	$0\le x_i\le W,$ $0\le y_i\le H,$ $0\le P_i\le P_{\max}$
Dynamic Economic Dispatch 1 [63]	$\min_{\mathbf{P}}\quad f(\mathbf{P})=\sum_{t=1}^{24}\sum_{i=1}^{5}\left(a_iP_{i,t}^2+b_iP_{i,t}+c_i\right)$ $P_i^{\min}\le P_{i,t}\le P_i^{\max},\quad\forall i=1,\ldots,5,\ t=1,\ldots,24$ $\sum_{i=1}^{5}P_{i,t}=D_t,\quad\forall t=1,\ldots,24$ $P_{\min}=[10,20,30,40,50]$ $P_{\max}=[75,125,175,250,300]$	120	$P_i^{\min}\le P_{i,t}\le P_i^{\max}$
Dynamic Economic Dispatch 2 [63]	$\min_{\mathbf{P}}\quad f(\mathbf{P})=\sum_{t=1}^{24}\sum_{i=1}^{9}\left(a_iP_{i,t}^2+b_iP_{i,t}+c_i\right)$ $P_i^{\min}\le P_{i,t}\le P_i^{\max},\quad\forall i=1,\ldots,5,\ t=1,\ldots,24$ $\sum_{i=1}^{5}P_{i,t}=D_t,\quad\forall t=1,\ldots,24$ $P_{\min}=[150,135,73,60,73,57,20,47,20]$ $P_{\max}=[470,460,340,300,243,160,130,120,80]$	216	$P_i^{\min}\le P_{i,t}\le P_i^{\max}$
Static Economic Load Dispatch (1, 2, 3, 4, 5) [63]	$\min_{P_1,\ldots,P_{N_G}}F=\sum_{i=1}^{N_G}f_i(P_i)$ $f_i(P_i)=a_iP_i^2+b_iP_i+c_i,\quad i=1,2,\ldots,N_G$ $f_i(P_i)=a_iP_i^2+b_iP_i+c_i+\lvert e_i\sin(f_i(P_i^{\min}-P_i))\rvert$ $P_i^{\min}\le P_i\le P_i^{\max},\quad i=1,2,\ldots,N_G$ $\sum_{i=1}^{N_G}P_i=P_D+P_L$ $P_L=\sum_{i=1}^{N_G}\sum_{j=1}^{N_G}P_iB_{ij}P_j+\sum_{i=1}^{N_G}B_{0i}P_i+B_{00}$ $P_i-P_i^0\le UR_i\quad P_i^0-P_i\le DR_i$	6 13 15 40 140	See Technical Report of CEC2011

3.3. Parameter Sensitivity Analysis of ARQ

Following Lee et al.'s [64] parameter-sensitivity methodology, we constructed a structured analysis to quantify responsiveness to parameter changes and the preservation of reliability across diverse operating regimes.

For the static economic load dispatch one problem, the dominant finding of the sensitivity analysis is that p exhibits by far the largest main effect on mean best (range ≈ 0.937), making it the primary regulator of the exploitation–exploration balance, the trend is clearly decreasing, as increasing p from 0.05 to 0.30 is accompanied by an almost monotonic drop in mean best, with 0.05 emerging as the best setting, indicating that exploitation pressure should remain modest to preserve diversity and avoid population alignment that harms the mean, see Table 4 for the exact main-effect ranges and the per-setting summaries. The main effects and stability ranges are visualized in Figure 7.

Table 4. Sensitivity analysis of the method parameters for the static economic load dispatch one problem.

Parameter	Value	Mean	Min	Max	Iters	Main Effect Range
α	0.5	2978.9797	2967.2491	3169.5417	9600	0.1594
	1	2978.9852	2967.2491	3169.5417	9600	
	1.5	2979.0918	2967.2491	3169.5417	9600	
	2	2978.9324	2967.2491	3166.8100	9600	
	2.5	2978.9569	2967.2491	3166.8100	9600	
rtr_k	3	2978.9785	2967.2491	3169.5417	9600	0.0748
	5	2979.0218	2967.2491	3169.5417	9600	
	7	2978.9469	2967.2491	3169.5417	9600	
	10	2979.0129	2967.2491	3169.5417	9600	
	14	2978.9860	2967.2491	3166.8100	9600	
sh_c	0.1	2979.0735	2967.2491	3169.5417	12,000	0.2084
	0.3	2978.8650	2967.2491	3169.5417	12,000	
	0.5	2979.0562	2967.2491	3169.5417	12,000	
	0.7	2979.0562	2967.2491	3166.8100	12,000	
p	0.05	2979.4978	2967.2491	3169.5417	12,000	0.9368
	0.1	2979.0781	2967.2491	3169.5417	12,000	
	0.2	2978.8201	2967.2491	3167.0430	12,000	
	0.3	2978.5609	2967.2491	3169.5417	12,000	
$stagnation_{trigger}$	10	2979.0096	2967.2491	3169.5417	12,000	0.0990
	20	2978.9588	2967.2491	3166.8100	12,000	
	30	2979.0438	2967.2491	3169.5417	12,000	
	50	2978.9447	2967.2491	3169.5417	12,000	

The second most influential factor is the success-history memory rate sh_c (range ≈ 0.208). Values 0.1 and 0.5 deliver the strongest averages, whereas 0.3 and 0.7 underperform. This matches the expected trade-off: too fast forgetting induces oscillations in F/CR, while too slow memory anchors prematurely in suboptimal ranges. The data suggest two sweet spots, a more agile setting at 0.1 and a more conservative one at 0.5, implying that problem attributes likely modulate the preferable regime, an adaptive schedule that moves from 0.5 to 0.1 when progress resumes would be reasonable.

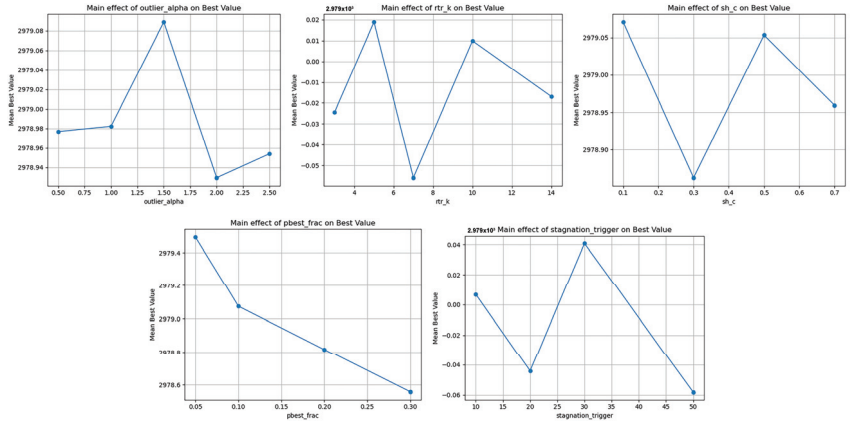


Figure 7. Sensitivity to selection controllers, main effects, and stability range. Graphical representation of outlier_alpha (α), rtr_k (rtr_k), sh_c (sh_c), pbest_frac (p) end stagnation_trigger ($stagnation_{trigger}$) for the static economic load dispatch one problem.

The α parameter in quarantine/repair shows a moderate main effect ($\approx$0.159), with a peak around 1.5 and only minor differences nearby. Values far below 1.0 or above 2.0 do not improve mean best in this sample. A mid-level aggressiveness for tail definition appears optimal sufficient to trim sporadic failures without crushing diversity supporting the view of quarantine as a noise regulator rather than a hard filter.

The RTR geometry parameter rtr_k has the smallest main effect in the tested band ($\approx$0.075), with shallow maxima at 5–10 neighbors. This suggests robustness to modest neighborhood changes within the examined range, extremes would likely matter more. A mid-range choice around 5–10 preserves local structure without trapping the population.

The $stagnation_{trigger}$ exhibits a small-to-moderate effect ($\approx$0.099), with a best value near 30. Short windows (10–20) do not help mean best, presumably due to premature micro-restarts that cut off promising paths, while very long windows (50) delay exiting genuine stagnation. A trigger around 30 strikes the best balance for the present budget.

Putting this together, the effect ranking is unambiguous: $p \gg sh_c \gtrsim \alpha > stagnation_{trigger} > rtr_k$. A consolidated view of these effects and recommended settings is reported in Table 4. The data-driven recommendation is to fix p at a low level around 0.05, set sh_c to either 0.1 or 0.5 depending on algorithmic phase, place α near 1.5, choose $stagnation_{trigger}$ around 30 generations, and keep rtr_k in a mid-range of 5–10. While the main-effect signals are clear, confirmatory runs should check for adverse interactions at extreme combinations, especially between p and sh_c, and between α and $stagnation_{trigger}$.

For dynamic economic dispatch one (ELD1), sensitivity is dominated by p and outlier_alpha with main-effect ranges of $\approx$0.311 and $\approx$0.295, whereas rtr_k, sh_c, and $stagnation_{trigger}$ play a much smaller role. See Table 5 for the exact main-effect ranges and the per-setting summaries. Unlike the static ELD1, increasing p from 0.05 to 0.30 consistently raises the mean best (peaking at 0.30), indicating that this problem benefits from stronger exploitation, the intertemporal coupling and ramp constraints reward elite-guided moves and reduce the cost of exploratory drift. Higher α values (2.0–2.5) further improve the mean best, suggesting that a more permissive tail threshold is advantageous: over-aggressive trimming of outliers can strip away useful diversity needed for smooth cross-period transitions. Changes in sh_c are tiny with a slight edge at 0.7, implying that a heavier memory stabilizes F/CR adaptation without suppressing peaks, and rtr_k is marginally better around $k = 3$ with negligible differences across the tested band. $stagnation_{trigger}$ is essentially neutral with a mild optimum near 30 generations. Overall, the effect ordering is $p \gtrsim \alpha \gg rtr_k \approx sh_c > stagnation_{trigger}$, and the configuration profile

emerging for the dynamic case favors a higher p (~0.30), elevated α (≈2.0–2.5), slightly slowersh_c (~0.7), small rtr_k (≈3), and $stagnation_{trigger}$ around 30 choices that strengthen guided exploitation while avoiding over-sterilization of diversity across time-coupled. The corresponding graphical summary is given in Figure 8.

Table 5. Sensitivity analysis of the method parameters for the dynamic economic dispatch one problem.

Parameter	Value	Mean	Min	Max	Iters	Main Effect Range
α	0.5	130,643.6614	130,642.7267	130,668.1727	9600	0.2952
	1	130,643.6546	130,642.7276	130,667.1070	9600	
	1.5	130,643.7393	130,642.7131	130,662.7681	9600	
	2	130,643.9058	130,642.7216	130,709.2564	9600	
	2.5	130,643.9498	130,642.7275	130,702.0175	9600	
rtr_k	3	130,643.7943	130,642.7253	130,709.2564	9600	0.0309
	5	130,643.7775	130,642.7275	130,702.0175	9600	
	7	130,643.7889	130,642.7233	130,684.7783	9600	
	10	130,643.7634	130,642.7282	130,662.7681	9600	
	14	130,643.7869	130,642.7131	130,667.5166	9600	
sh_c	0.1	130,643.7862	130,642.7282	130,680.0939	12,000	0.0295
	0.3	130,643.7630	130,642.7168	130,680.9305	12,000	
	0.5	130,643.7870	130,642.7131	130,709.2564	12,000	
	0.7	130,643.7925	130,642.7253	130,702.0175	12,000	
p	0.05	130,643.6902	130,642.7131	130,709.2564	12,000	0.3111
	0.1	130,643.6697	130,642.7259	130,702.0175	12,000	
	0.2	130,643.7879	130,642.7393	130,684.7783	12,000	
	0.3	130,643.9809	130,642.7459	130,682.8584	12,000	
$stagnation_{trigger}$	10	130,643.7790	130,642.7233	130,709.2564	12,000	0.0164
	20	130,643.7830	130,642.7294	130,684.7783	12,000	
	30	130,643.7915	130,642.7253	130,702.0175	12,000	
	50	130,643.7751	130,642.7131	130,680.0939	12,000	

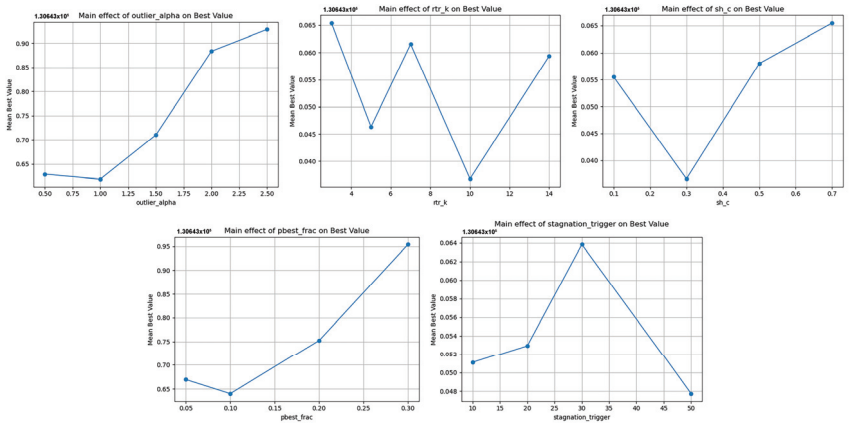


Figure 8. Sensitivity to selection controllers, main effects, and stability range. Graphical representation of outlier_alpha (α), rtr_k (rtr_k), sh_c (sh_c), pbest_frac (p) end stagnation_trigger ($stagnation_{trigger}$) for the dynamic economic dispatch one problem.

For Lennard-Jones (13 atoms, 43D) the dominant driver of mean best is sh_c (main-effect range ≈ 0.991): heavier success-history memory around 0.7 yields more negative (better) means, improving from -39.33 at 0.1 to -40.32 at 0.7. See Table 6 for the exact main-effect ranges and the per-setting summaries. This indicates that the highly multimodal, deceptive LJ-13 landscape benefits from slower, stabilizing adaptation of F, CR, which dampens oscillations and sustains coherent progress into deeper energy wells. p shows a moderate effect ($\approx$0.202) with a clear preference for low values, increasing it from 0.05 to 0.30 degrades the mean from -40.02 to -39.82, so restrained exploitation and preserved exploration are essential to avoid alignment around shallow basins. α exhibits a modest, favorable trend ($\approx$0.195) as it rises toward 2.0–2.5, a more permissive outlier threshold makes quarantine gentler, which here helps retain useful diversity in challenging basins. rtr_k and $stagnation_{trigger}$ have small effects ($\approx$0.040 and $\approx$0.014): performance is marginally better near $k \approx 10$ and $trigger \approx 50$, but differences are minor within the tested band. Overall, LJ-13 favors $sh_c \approx 0.7$, $p \approx 0.05$, $\alpha \approx 2.0$–2.5, with rtr_k mid-range and a slightly larger $stagnation_{trigger}$ a configuration that promotes steady, directed exploration without overly sterilizing the tails. Notably, the min consistently reaches about -44.33 across settings, suggesting that global-level depths are sporadically discovered regardless, while the recommended choices compress the best–mean gap and systematically improve average quality. See Figure 9 for the main-effect curves.

Table 6. Sensitivity analysis of the method parameters for problem entitled Lennard-Jones Potential, Atoms 13, Dim: 43.

Parameter	Value	Mean	Min	Max	Iters	Main Effect Range
α	0.5	−39.8220	−44.32680	−35.8189	9600	0.1952
	1	−39.9118	−44.32680	−35.1252	9600	
	1.5	−39.9657	−44.32680	−35.3909	9600	
	2	−39.9857	−44.32680	−36.0638	9600	
	2.5	−40.0173	−44.32680	−35.3308	9600	
rtr_k	3	−39.9465	−44.32680	−35.1252	9600	0.0404
	5	−39.9392	−44.32680	−35.3909	9600	
	7	−39.9172	−44.32680	−35.3308	9600	
	10	−39.9577	−44.32680	−35.4352	9600	
	14	−39.9419	−44.32680	−35.8104	9600	
sh_c	0.1	−39.3275	−44.32680	−35.3909	12,000	0.9911
	0.3	−39.9237	−44.32680	−35.5921	12,000	
	0.5	−40.1921	−44.32680	−35.1252	12,000	
	0.7	−40.3186	−44.32680	−36.0487	12,000	
p	0.05	−40.0216	−44.32680	−35.1252	12,000	0.2021
	0.1	−40.0054	−44.32680	−35.3909	12,000	
	0.2	−39.9157	−44.32680	−35.3308	12,000	
	0.3	−39.8194	−44.32680	−35.4769	12,000	
$stagnation_{trigger}$	10	−39.9334	−44.32680	−35.3308	12,000	0.0141
	20	−39.9398	−44.32680	−35.8387	12,000	
	30	−39.9412	−44.32680	−35.7413	12,000	
	50	−39.94763	−44.32680	−35.1252	12,000	

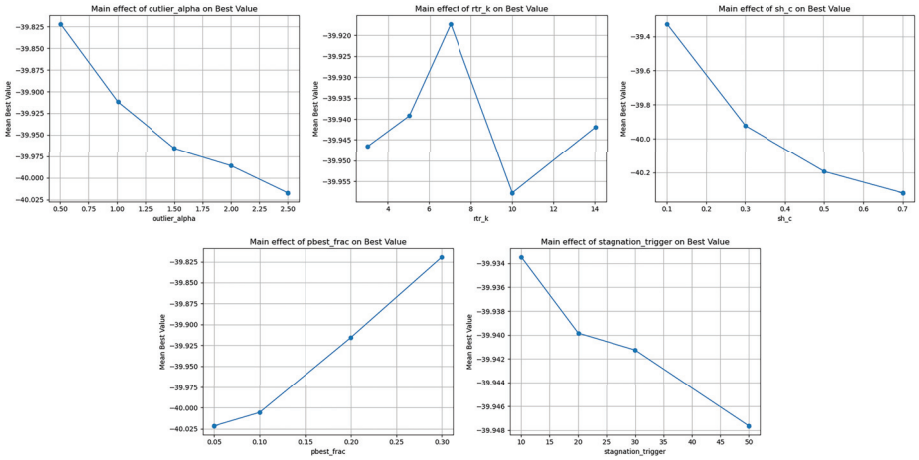


Figure 9. Sensitivity to selection controllers, main effects, and stability range. Graphical representation of outlier_alpha (α), rtr_k (rtr_k), sh_c (sh_c), pbest_frac (p) end stagnation_trigger ($stagnation_{trigger}$) for problem of Lennard-Jones Potential, Atoms 13, Dim: 43.

3.4. Analysis of Complexity of ARQ

Below, we present three real-world problems with brief descriptions shell-and-tube heat-exchanger design, a Brayton-type gas-cycle surrogate, and the TANDEM multi-gravity-assist space-trajectory surrogate. In the experiments, we varied only the problem dimension while keeping all modeling and algorithmic settings fixed, so that the observed effects reflect purely the impact of dimensionality. This protocol lets us examine how the proposed optimization method scales as the number of decision variables grows, without confounding changes to constraints, penalties, or hyperparameters. For these measurements, the termination rule was 500 iterations, which proved sufficient to reach the optimal value and thus no additional iterations were required.

3.4.1. Shell-And-Tube Heat Exchanger (Surrogate) [65–67]

Vars: $\mathbf{x} = [D_s, D_t, L, p_r, B, c_b]^\top$. **Bounds:** $D_s \in [0.30, 1.50]$ m, $D_t \in [0.010, 0.050]$ m, $L \in [1, 6]$ m, $p_r \in [1.25, 2.00]$, $B \in [0.10, 0.60]$ m, $c_b \in [0.15, 0.45]$.

$$\min_{\mathbf{x}} J(\mathbf{x}) = C_{\text{area}} A + w_{\text{pump}} P_{\text{pump}} t_y + \Pi_{UA} + \Pi_{\text{soft}}.$$

Heat transfer: $A = \pi D_t L N_t \kappa_f$, $U = (1/h_i + R_w + 1/h_o + R_f)^{-1}$, $UA = U\,A$.

$$h_i = a_t\, v_t^{0.8} D_i^{-0.2}, \quad h_o = a_s\, v_s^{0.8} D_s^{-0.2}, \quad R_w = t_w / k_{\text{wall}}.$$

Requirement/penalties: $UA_{\text{req}} = Q_{\text{req}}/\text{LMTD}$, $\Pi_{UA} = \lambda_h \left[\frac{(UA_{\text{req}} - UA)_+}{UA_{\text{req}}}\right]^2$, $\Pi_{\text{soft}} = \lambda_s\big([v_t - 3]_+^2 + 0.5[v_s - 5]_+^2 + [0.2D_s - B]_+^2\big)$.

Hydraulics: $P_{\text{pump}} = (\Delta P_{\text{tube}} \dot{V}_c + \Delta P_{\text{shell}} \dot{V}_h)/\eta_{\text{pump}}$, $\Delta P_{\text{tube}} = k_t L v_t^2 / D_i$, $\Delta P_{\text{shell}} = k_s (D_s/B) v_s^2$.

3.4.2. Gas Cycle (Brayton-like Surrogate) [68]

Vars: $\mathbf{x} = [T_1, T_3, P_1, P_3]^\top$. **Bounds:** $T_1 \in [300, 1500]$ K, $T_3 \in [1200, 2000]$ K, $P_1, P_3 \in [1, 20]$ bar.

$$\min_{\mathbf{x}} f(\mathbf{x}) = -\eta(\mathbf{x}), \qquad \eta(\mathbf{x}) = 1 - r^{-(\gamma-1)/\gamma} \frac{T_1}{T_3}, \qquad r = \frac{P_3}{P_1},\ \gamma = 1.4.$$

3.4.3. Space Trajectory: TANDEM (MGA-1DSM Surrogate) [69–71]

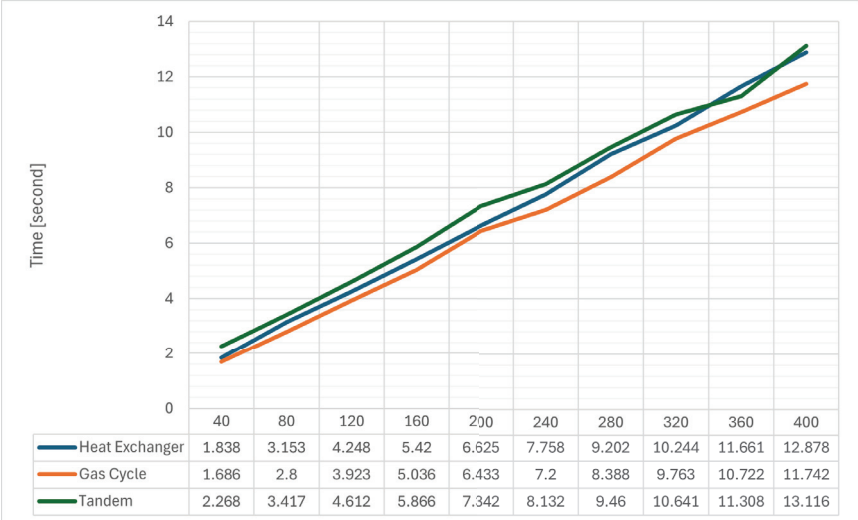
Vars ($D = 18$): $\mathbf{x} = [t_0, T_1, T_2, T_3, T_4, T_{5A}, T_{5B}, s_1, s_2, s_3, s_4, s_{5A}, s_{5B}, r_p, k_{A1}, k_{A2}, k_{B1}, k_{B2}]^\top$. **Bounds:** $t_0 \in [7000, 10{,}000]$ (MJD2000 d), $T_1 \in [30, 500]$, $T_2 \in [30, 600]$, $T_3 \in [30, 1200]$, $T_4 \in [30, 1600]$, $T_{5A}, T_{5B} \in [30, 2000]$ d, $s_\bullet, r_p, k_\bullet \in [0, 1]$.

$$\min_{\mathbf{x}} \Delta V_{\text{tot}} = \Delta V_{\text{launch}}(T_1) + \sum_{i=1}^{4} \Delta V_{\text{leg}}(T_i) + \Delta V_{\text{branch}}(T_{5A}, s_{5A}, k_{A1}, k_{A2}) +$$

$$\Delta V_{\text{branch}}(T_{5B}, s_{5B}, k_{B1}, k_{B2}) \quad + \sum_{i=1}^{4} \Delta V_{\text{DSM}}(s_i) - G_{\text{GA}}(T_1, T_2, T_3) - G_{\text{J}}(T_4) + \Pi_{\text{ToF}} + \Pi_{\text{bounds}}.$$

Compact component forms (surrogate): $\Delta V_{\text{leg}}(T) = \ell_s / (1 + 2\,\text{clip}(T/t_r, 0.2, 4))$, $\Delta V_{\text{DSM}}(s) = c_{\text{DSM}}(0.25 + 0.75s)$, $\Delta V_{\text{branch}}(\cdot)$ adds DSM shaping terms with $r_p, k_\cdot$, $G_{\text{GA}}, G_{\text{J}}$ are decreasing functions of leg times, $\Pi_{\text{ToF}} = \beta\,[(T_1+T_2+T_3+T_4+\frac{1}{2}(T_{5A}+T_{5B})) - T_{\text{soft}}]_+$.

Across the three problems, runtime grows smoothly with dimension from 40 to 400 and the trend is very close to linear (Figure 10). Heat exchanger rises from 1.838 s at 40 D to 12.878 s at 400 D, gas cycle from 1.686 s to 11.742 s, and tandem from 2.268 s to 13.116 s. Doubling the dimensionality typically increases runtime by roughly a factor of 1.7–1.9, indicating linear growth with a small fixed overhead at low dimensions. Gas cycle is consistently the fastest, heat exchanger sits in the middle, and Tandem is the slowest, largely due to a higher constant setup cost per evaluation even though its growth rate with dimension is similar to the others. Averaged over the grid, each additional 40 variables adds about 1.1–1.3 s, which remains stable across the range. Overall, the evidence points to an approximately linear time complexity in dimension for the proposed method, with differences between problems attributable mainly to per-evaluation overhead rather than to divergent asymptotic behavior.



	40	80	120	160	200	240	280	320	360	400
Heat Exchanger	1.838	3.153	4.248	5.42	6.525	7.758	9.202	10.244	11.661	12.878
Gas Cycle	1.686	2.8	3.923	5.036	6.433	7.2	8.388	9.763	10.722	11.742
Tandem	2.268	3.417	4.612	5.866	7.342	8.132	9.46	10.641	11.308	13.116

Figure 10. Runtime vs. dimension (40–400) for ARQ on three real-world problems (500-iteration budget).

3.5. Comparative Performance Analysis of ARQ

Our comparator suite JSO, TRIDENT-DE, UDE3, EA4Eig, mLSHADE_RL, SaDE, CMA-ES, jDE, and CLPSO is a deliberate, sharp, and diverse set that enables fair and generalizable assessment under a common protocol with uniform end-of-budget best-so-far and mean metrics. It blends state-of-the-art DE variants with canonical non-DE references, prioritizing

excellence and representativeness (strong CEC and real-world records), so the proposed method is tested against genuinely formidable opponents.

The suite spans the mechanisms that govern the best vs. mean trade-off: aggressive exploitation (pbest/1/bin, current-to-best/1/bin, adaptive F/CR), diversity preservation (self-adaptation, and mutation ensembles), stagnation handling (restarts and re-initialization), and learning-driven exploration (policy selection). JSO and mLSHADE_RL serve as state-of-the-art DE with robust adaptation/learning and consistent means, TRIDENT-DE is a practitioner-grade baseline known for strong best-so-far, UDE3 probes robustness via ensemble strategies, EA4Eig leverages eigenspace/second-order cues for ill-conditioned landscapes, SaDE tests whether endogenous strategy learning suffices, CMA-ES is the non-DE gold standard learning local geometry via covariance, jDE provides a lightweight self-adaptation lower bound, and CLPSO adds cooperative PSO strength on multimodal/noisy tasks. Together, this mix stresses the proposed method across peak best-so-far, stable mean, and overall robustness, making any observed gains substantive and credible.

Table 7 provides a CEC-style descriptive suite-level summary (best/mean). Statistical tests and significance markers are reported per problem in Tables 8–10.

Table 7 provides the end-of-budget snapshot, reporting for each real-world task the end-of-budget best-so-far alongside the mean over multiple independent runs. This dual perspective separates peak-attainment from run-to-run consistency, revealing not only who reaches high but also who does so reliably. Within this lens, ARQ exhibits the profile it was designed for: strong best-so-far on rugged or noisy landscapes without sacrificing mean stability. The pattern is mechanistically grounded rather than accidental. Success-history parameter adaptation quickly locks F/CR into profitable ranges, curbing wasteful oscillations that typically inflate variance. Neighborhood-aware RTR replacement focuses selection where it matters while preserving population geometry and diversity, thus avoiding brittle premature convergence. The quarantine mechanism, paired with targeted micro-restarts around the incumbent, trims toxic tails by gently pulling a small fraction of worst individuals toward a robust center and accepting repairs only when they yield genuine improvement. The net effect is a tighter best–mean gap high peaks without bleeding the average.

Reading across problem families reinforces the same message. On geometry-sensitive or ill-conditioned settings, where eigenspace-aware or covariance-adapting methods traditionally shine, ARQ maintains a competitive mean precisely because tail control suppresses rare catastrophic runs that would otherwise drag the average down. On energy and network planning tasks, micro-restarts provide inexpensive, targeted escapes from stagnation that accumulate into steady gains in both best and mean. On classic multimodal benchmarks, the combination of pbest/1/bin with an archive and success-history keeps exploration directed late into the budget, which shows up as robust best-so-far without a collapse in mean.

The deltas between best and mean act as a health indicator of the performance distribution. Where ARQ ties or narrowly trails specialized competitors in best, it often compensates with a superior mean, signaling run-level resilience. Where it attains the top best, the mean remains stable, indicating not a solitary spike but a cohesive cloud of good solutions around it. Because all solvers run under a harmonized protocol, identical budgets, and multiple independent trials, this advantage cannot be dismissed as a tuning artifact. Table 7 therefore suggests that ARQ achieves the hard balance between peak best-so-far and preserved mean through a cohesive TRIDENT-DE: gain-weighted parameter learning, neighborhood-sensitive replacement that protects geometry and diversity, and tail control via quarantine and micro-restarts. This sets the stage for the rankings that follow, in which superior peaks are not purchased at the expense of stability.

Table 7. Comparison Based on Best and Mean after 1.5×10^5 FEs.

Problem	ARQ Best Mean	JSO Best Mean	TRIDENT-DE Best Mean	UDE3 Best Mean	EA4Eig Best Mean	mLSHADE_RL Best Mean	SaDE Best Mean	CMA-ES Best Mean	jDE Best Mean	CLPSO Best Mean
Lennard-Jones Potential (10 atoms)	−28.42253189	−27.68965427	−28.42253189	−17.60964115	−22.47842596	−28.42252711	−22.86077544	−28.42253189	−15.91366007	−16.59269921
	−27.62259837	−26.43980674	−27.19833972	−16.33758634	−19.48663385	−23.77189104	−21.22806787	−27.52754934	−13.75563015	−13.55503647
Lennard-Jones Potential (13 atoms)	−44.12080311	−36.56391435	−41.39220729	−21.90945922	−28.01101572	−40.6992486	−29.31393114	−44.32680142	−18.77073675	−18.00250514
	−38.97673056	−35.18151099	−39.14390841	−19.49372047	−24.58810405	−30.67706246	−27.48874237	−41.44245617	−15.4243072	−15.86691988
Lennard-Jones Potential (38 atoms)	−157.3383508	−73.79663557	−125.1886792	−2.015038465	−55.05415428	−120.0680452	−68.91313107	−167.7369019	9186.25096	330.9934848
	−125.0267732	−67.48713618	−107.1751456	140.2211284	−3.350181902	−73.95117658	−45.80265994	−163.6091673	9186.25096	1087.035295
Tersoff Potential for model Si (B)	−29.30289228	−28.73373322	−28.93480467	−25.43447342	−26.90746941	−28.12281867	−26.65687822	−28.33045002	−24.75133772	−22.67387001
	−28.279056	−28.0408539	−27.70615763	−23.30318979	−24.69059932	−25.49977206	−25.27422603	−27.38991233	−22.94168766	−21.21150428
Tersoff Potential for model Si (C)	−33.78825497	−33.24194738	−33.8820283	−29.30227462	−30.88865174	−31.70444684	−30.94469385	−32.50963421	−29.44789882	−26.88039528
	−32.71055796	−32.47083022	−31.91749393	−27.53891341	−29.0199918	−29.44303263	−29.70029831	−31.53772845	−29.44789882	−24.653633
Parameter Estimation for Frequency-ModulatedSound Waves	5.37×10^{-20}	2.40×10^{-5}	0.116157535	0	0.15272453	0.116157535	0.148007602	0.210122687	3.21×10^{-25}	0.131483748
	0.122436815	0.114197358	0.134324544	0.103406319	0.213099692	0.208210846	0.148007602	0.267329914	0.132539923	0.212498169
Circular Antenna Array Design	0.006809638	0.006809638	0.006809638	0.006809665	0.006809638	0.006809662	0.006814682	0.007253731	0.00681715	0.006933401
	0.006810417	0.006819653	0.006809683	0.006817385	0.006809638	0.006825338	0.00790701	0.008755359	0.006835764	0.051815518
Spread Spectrum Radar Polyphase Code Design	0.006902215	0.331062321	0.014426836	0.953872709	0.601567824	0.074552911	0.550837019	0.062519409	1.005739785	0.860294378
	0.351872002	0.44224577	0.26254527	1.206577385	0.869257599	0.535028919	0.803527605	0.197713522	1.331416957	1.273200439
Cassini 2: Spacecraft Trajectory Optimization Problem	1.42×10^{-13}	1.42×10^{-13}	4.54×10^{-10}	0.000926598	1.42×10^{-13}	1.76×10^{-12}	0.039231105	2.84×10^{-13}	2.70×10^{-5}	1.22633022
	2.90×10^{-7}	1.42×10^{-13}	1.12×10^{-5}	0.008206106	5.96×10^{-14}	1.73×10^{-5}	0.070230417	5.929143722	0.000285411	3.639687905
Wireless Coverage Antenna Placement	0.946350736	0.946350736	0.946350736	0.946350736	0.946655032	0.946350736	0.946361987	1.18939375	0.946350736	0.946365969
	0.946371023	0.94636088	0.94662124	0.946502884	0.946659575	0.946875107	0.946688757	1.190699803	0.946401452	0.946727502
Transmission Network Expansion Planning	4.485292926	4.485292926	4.485292926	4.485295106	4.485292926	4.485292926	4.485299525	4.485292926	4.485292926	4.486699087
	4.485292926	4.485292926	4.485304003	4.485304003	4.485292926	4.485292926	4.485311924	4.485292948	4.485292926	4.495857336
Dynamic Economic Dispatch 1	130,642.8565	130,661.111	130,850.0389	130,693.5423	130,694.29	130,882.0646	131,010.8769	130,650.9354	131,225.2453	131,834.4235
	130,643.4825	130,681.9224	130,931.1074	130,717.6052	130,862.9893	130,955.331	131,099.1959	130,654.2758	131,225.2453	132,151.5397
Dynamic Economic Dispatch 2	165,304.7824	171,554.4678	165,980.9574	164,946.164	172,067.4426	167,519.3281	167,908.0605	165,847.1092	186,121.2812	177,120.3822
	165,662.994	172,600.3374	166,478.1534	165,614.9256	172,931.6964	168,275.5429	168,495.9731	166,233.691	190,793.5621	178,190.4198

Table 7. *Cont.*

Problem	ARQ Best Mean	JSO Best Mean	TRIDENT-DE Best Mean	UDE3 Best Mean	EA4Eig Best Mean	mLSHADE_RL Best Mean	SaDE Best Mean	CMA-ES Best Mean	jDE Best Mean	CLPSO Best Mean
Static Economic Load Dispatch 1	2967.249196	2967.249196	2967.249196	2967.249196	2979.803369	2967.249196	2967.249196	2967.249586	2967.249196	2967.249197
	2978.966424	2973.649521	2975.721343	2976.057657	2979.803369	2976.956221	2976.139816	3108.931917	2967.659992	2973.33009
Static Economic Load Dispatch 2	17,866.8974	17,864.04428	17,879.73679	17,864.69687	18,006.65976	17,882.28384	17,892.38129	17,960.84734	17,867.57447	17,910.47794
	17,915.76463	17,864.65039	17,928.65603	17,890.83413	18,063.12085	17,950.15892	17,958.94204	18,077.58006	17,992.09486	18,089.83526
Static Economic Load Dispatch 3	32,367.57735	32,367.57735	32,367.57735	32,367.57735	32,415.80367	32,367.57765	32,376.02197	32,645.34102	32,391.64981	32,384.85409
	32,539.37396	32,367.57735	32,440.87752	32,400.86337	32,573.03572	32,491.57235	32,491.28971	32,867.1729	32,476.58405	32,532.19143
Static Economic Load Dispatch 4	121,093.4505	121,078.7215	121,071.4654	121,066.9247	121,197.2468	121,085.9922	121,195.4656	122,350.1013	121,234.0466	121,328.7006
	121,331.4674	508,807.9108	121,422.9908	121,197.2468	121,545.1315	121,308.5801	121,517.4918	122,957.6217	121,526.7414	121,541.4182
Static Economic Load Dispatch 5	508,614.1309	508,807.9108	508,663.8176	508,661.3113	508,872.6908	508,851.668	508,985.9092	508,717.1467	511,174.5326	509,025.7426
	508,614.8964	508,858.0755	508,703.0424	508,676.4938	508,986.6092	508,988.5396	509,125.2079	508,770.1661	562,012.6548	509,080.3439

Table 8. Detailed Ranking of Algorithms Based on Best after 1.5×10^5 FEs. (1 = green, 2 = blue).

Problem	ARQ	JSO	TRIDENT-DE	UDE3	EA4Eig	mLSHADERL	SaDE	CMA-ES	jDE	CLPSO
Lennard-Jones Potential (10 atoms)	1	4	3	8	7	5	6	1	9	10
Lennard-Jones Potential (13 atoms)	3	4	2	8	7	5	6	1	10	9
Lennard-Jones Potential (38 atoms)	2	5	3	8	7	4	6	1	10	9
Tersoff Potential for model Si (B)	1	2	3	8	7	5	6	4	9	10
Tersoff Potential for model Si (C)	1	2	3	9	8	7	5	4	6	10
Parameter Estimation for Frequency-Modulated Sound Waves	3	2	5	1	9	7	6	10	4	8
Circular Antenna Array Design	3	5	2	4	1	6	8	9	7	10
Spread Spectrum Radar Polyphase Code Design	3	4	2	8	7	5	6	1	10	9
Cassini 2: Spacecraft Trajectory Optimization Problem	3	1	4	7	1	5	8	10	6	9
Wireless Coverage Antenna Placement	2	1	5	4	6	9	7	10	3	8
Transmission Network Expansion Planning	1	1	7	7	1	1	9	6	1	10
Dynamic Economic Dispatch 1	1	3	6	4	5	7	8	2	9	10
Dynamic Economic Dispatch 2	2	7	4	1	8	5	6	3	10	9
Static Economic Load Dispatch 1	8	3	4	5	9	7	6	10	1	2
Static Economic Load Dispatch 2	3	1	4	2	8	5	6	9	7	10
Static Economic Load Dispatch 3	8	1	3	2	9	6	5	10	4	7
Static Economic Load Dispatch 4	3	10	4	1	8	2	5	9	6	7
Static Economic Load Dispatch 5	1	5	3	2	6	7	9	4	10	8

Table 8 converts the raw outcomes of Table 7 into per-problem ranks and aggregate indicators, making it explicit whether a solver wins consistently or only sporadically. The pattern for ARQ is a high density of top 1/2 finishes across a broad portion of the suite, accompanied by a low average rank and a tight spread. This indicates breadth and stability rather than a handful of outlier peaks.

The juxtaposition of best-rank and mean-rank indicates a balanced profile. When ARQ achieves a top end-of-budget best-so-far, it does not incur a collapse in mean-rank, this suggests that tail control and micro-restarts keep the population concentrated around the most promising basin. Conversely, in cases where highly specialized competitors hold a slight edge in best-rank, ARQ often compensates with a superior mean-rank, evidencing run-level resilience and an ability to avoid degenerative trajectories.

The geography of ranks across problem families reinforces this reading. On ill-conditioned or geometry-sensitive tasks, where methods that learn search-space geometry traditionally excel, ARQ remains near the front with limited variance, consistent with neighborhood-aware replacement preserving structure and diversity. On multimodal landscapes, where occasional spectacular hits can inflate best-rank while harming mean behavior, Table 8 shows a small gap between ARQ's two ranking columns, in line with directed exploration sustained late into the budget. On energy and network design applications, micro-restarts around the incumbent accumulate incremental gains that lower average rank and increase the frequency of top-2 placements.

Head-to-head comparisons against recognized state-of-the-art references highlight that ARQ competes not only within the DE family but also against non-DE geometry-learning approaches, reducing the likelihood that its advantage is a tuning artifact. The overall rank and the accompanying aggregates distill this behavior into a single index, yielding clear evidence of generalizability: ARQ does not win only here and there, but stays consistently near the top across heterogeneous classes of problems.

In Figure 11, we report one-sided paired Wilcoxon signed-rank tests (hypothesis ARQ < Other) across all datasets, with Benjamini–Hochberg (FDR) adjustment. ARQ achieves lower ranks on average and is significantly better than most competitors after FDR correction specifically, ARQ vs. UDE3, mLSHADE_RL, CMA-ES, EA4Eig, SaDE, jDE, and CLPSO are significant, whereas ARQ vs. JSO and ARQ vs. TRIDENT-DE remain non-significant. Overall, ARQ shows consistent improvements in rank relative to most alternatives, while being statistically indistinguishable from the two strongest baselines (JSO and TRIDENT-DE). Consistent with the per-problem best outcomes, Figure 12 presents the aggregate mean ranking, condensing averages. Consistent with the per-problem best outcomes, Figure 13 highlights ARQ's frequent top-tier placements.

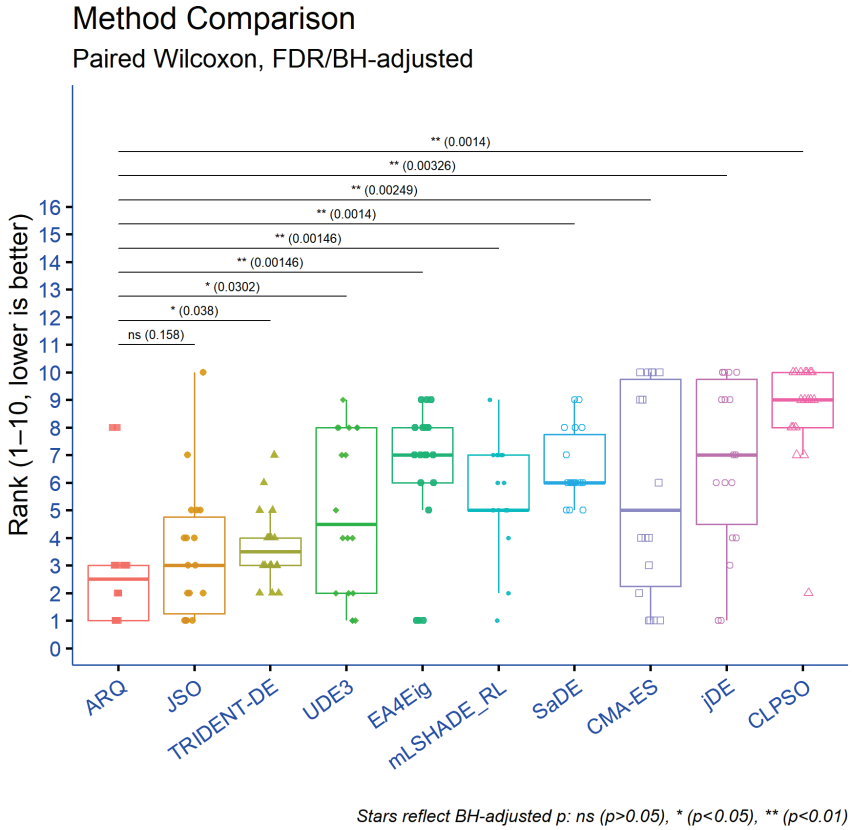


Figure 11. Statistical comparison of optimization methods by best-so-far over 30 runs (evaluation budget 150,000 FEs).

Table 9 acts as a distillation of the evidence: per-problem ranks are aggregated into global indicators that capture overall behavior. ARQ's position on these summary measures reflects a solver that excels not by isolated flashes, but by maintaining persistently strong showings across the board. Totals and average ranks remain low, with a tight dispersion around the central tendency, in practice, this means frequent proximity to the top with limited variability from task to task.

The alignment between summary metrics derived from best and from mean confirms the balance observed previously. When ARQ signals strength at the end-of-budget best-so-far, the mean-based summary does not erode, indicating that the cloud of solutions is cohesive rather than driven by rare spikes. Even where narrowly specialized competitors gain a local edge, the cumulative ranking over the whole suite shifts the advantage back to ARQ, because small, steady improvements add up.

What emerges from Table 9 is not merely many first places, but a combination of high top-2 frequency with a scarcity of poor outcomes. ARQ exhibits a compact performance profile: the tails of weak runs are curtailed while strong results appear repeatedly, not accidentally. This distributional shape explains why the global indicators favor the method. In practical terms, Table 9 certifies that the proposed approach preserves a competitive margin as problem geometry, conditioning, and multimodality vary. Consistency across these shifts is precisely what converts scattered victories into an overall lead.

In Figure 14, ARQ secures a clearly lower average rank and leaves most rivals behind even after multiple-comparison control. The strongest pairwise gains are against UDE3, mLSHADE_RL, CMA-ES, EA4Eig, SaDE, jDE, and CLPSO, whereas its gaps to JSO and TRIDENT-DE do not rise to statistical significance. In short, ARQ consistently ranks better than the bulk of competing methods while being on par with the two top baselines (JSO, TRIDENT-DE). Figure 12 presents the aggregate mean ranking, condensing averages. Figure 13 highlights ARQ's frequent top-tier placements.e performance into a single comparative view.

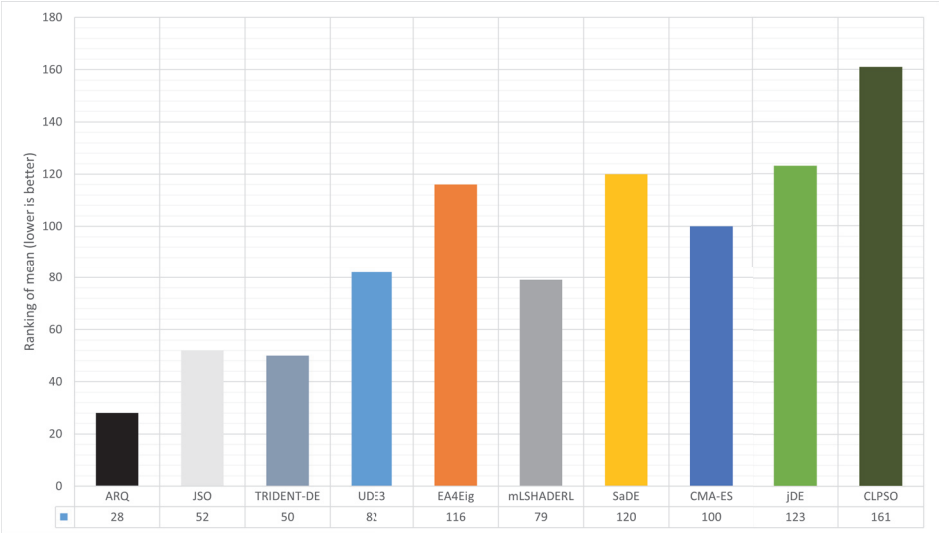


Figure 12. Aggregate mean ranking across ten algorithms.

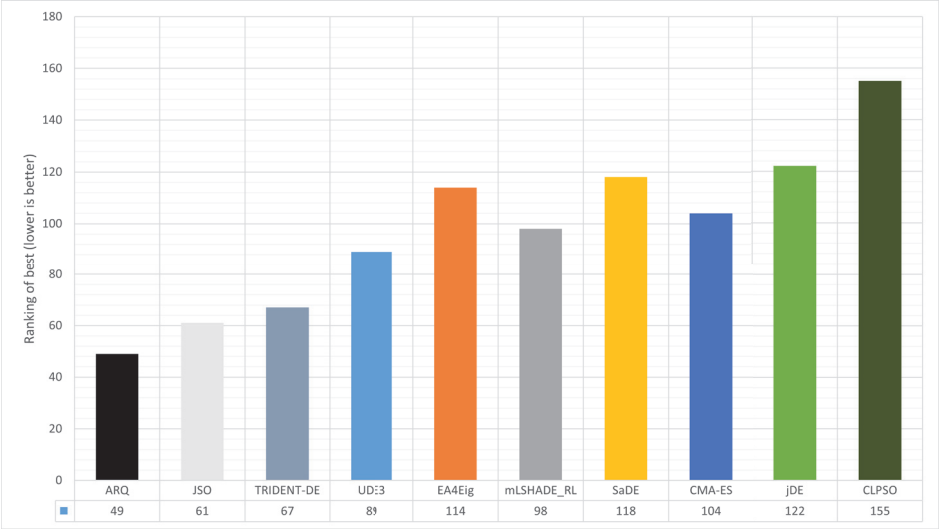


Figure 13. Aggregate best ranking across ten algorithms.

Table 9. Detailed ranking of algorithms based on mean after 1.5×10^5 FEs. (1 = green, 2 = blue).

Problem	ARQ	JSO	TRIDENT-DE	UDE3	EA4Eig	mLSHADERL	SaDE	CMA-ES	jDE	CLPSO
Lennard-Jones Potential (10 atoms)	2	5	3	8	7	4	6	1	9	10
Lennard-Jones Potential (13 atoms)	2	5	3	8	7	4	6	1	10	9
Lennard-Jones Potential (38 atoms)	1	3	2	8	6	5	7	4	9	10
Tersoff Potential for model Si (B)	2	3	1	9	7	5	6	4	8	10
Tersoff Potential for model Si (C)	1	4	5	1	9	5	8	10	1	7
Parameter Estimation for Frequency-Modulated Sound Waves	1	1	1	6	1	5	7	10	8	9
Circular Antenna Array Design	1	5	2	9	7	4	6	3	10	8
Spread Spectrum Radar Polyphase Code Design	1	1	6	8	1	5	9	1	7	10
Cassini 2: Spacecraft Trajectory Optimization Problem	1	1	1	1	9	1	7	10	1	8
Wireless Coverage Antenna Placement	1	1	1	8	1	1	9	1	1	10
Transmission Network Expansion Planning	1	3	6	4	5	7	8	2	9	10
Dynamic Economic Dispatch 1	2	7	4	1	8	5	6	3	10	9
Dynamic Economic Dispatch 2	1	1	1	1	10	1	1	9	1	8
Static Economic Load Dispatch 1	3	1	5	2	10	6	7	9	4	8
Static Economic Load Dispatch 2	1	1	1	1	9	5	6	10	8	7
Static Economic Load Dispatch 3	5	3	2	1	7	4	6	10	8	9
Static Economic Load Dispatch 4	1	5	3	2	7	6	8	4	10	9
Static Economic Load Dispatch 5	1	2	3	4	5	6	7	8	9	10

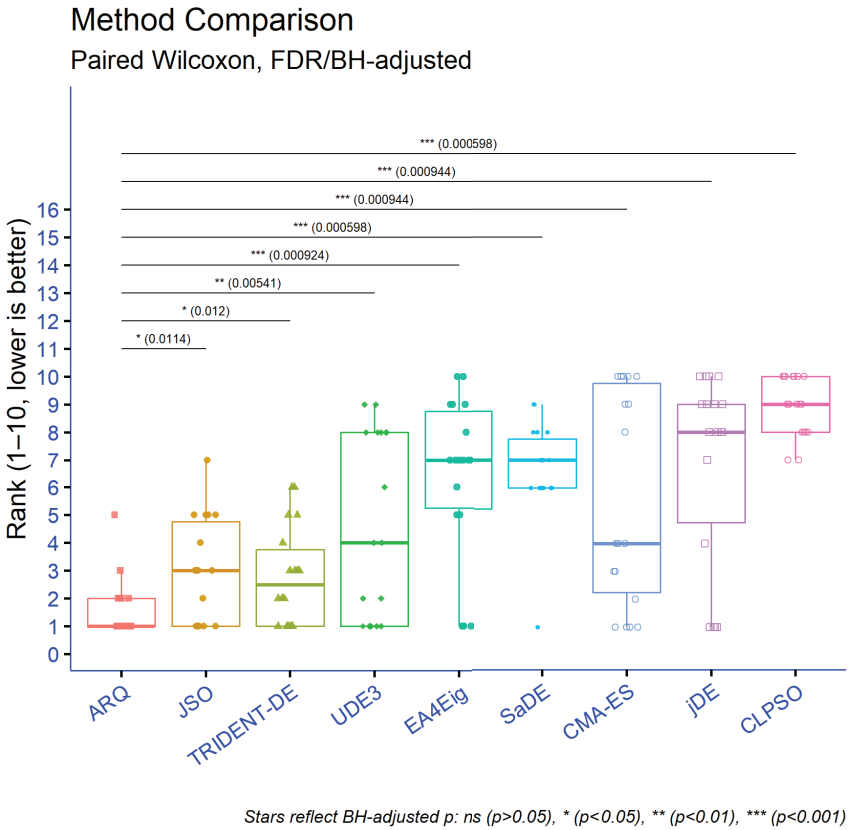


Figure 14. Statistical comparison of optimization methods by mean best-so-far over 30 runs (evaluation budget 150,000 FEs).

Table 10. Comparison of algorithms and final Ranking. (1 = green, 2 = blue).

Algorithm	Best Total Rank	Mean Total Rank	Overall Rank Sum	Average Rank
ARQ	**49**	**28**	**77**	**2.139**
JSO	**61**	**52**	**113**	**3.139**
TRIDENT-DE	**67**	**50**	**117**	**3.250**
UDE3	**89**	**82**	**171**	**4.750**
mLSHADE_RL	**98**	**79**	**177**	**4.917**
CMA-ES	**104**	**100**	**204**	**5.666**
EA4Eig	**114**	**116**	**230**	**6.389**
SaDE	**118**	**120**	**238**	**6.611**
jDE	**122**	**123**	**245**	**6.806**
CLPSO	**155**	**161**	**316**	**8.778**

Table 10 consolidates the picture established earlier: ARQ attains the lowest total rank on both best and mean 49 and 28, respectively for a combined sum of 77, corresponding to an average rank of 2.139. The next best competitor sits noticeably higher (e.g., JSO: 61 + 52 = 113), and others trail further behind, indicating that ARQ's advantage stems from consistent top placements across the suite rather than isolated spikes. This result is aligned with the harmonized experimental protocol and the carefully matched competitor settings, which lends credibility to the comparison.

The pattern strong totals in best without erosion in mean is consistent with ARQ's design. Success-history parameter adaptation stabilizes profitable ranges for F and CR early on, neighborhood-aware RTR preserves structure and diversity via local replacement, and the outlier-quarantine mechanism with targeted micro-restarts trims tail failures. Together, these components compress the gap between peak and average performance, a behavior visible in Table 7 and now distilled by the aggregates in Table 10. In practical terms, ARQ achieves high end-of-budget best-so-far without paying for it with a degraded mean, and this translates directly into low overall ranks.

In Figure 15, we compare ten algorithms over 36 problems using average ranks and assess pairwise differences with the Nemenyi test ($\alpha = 0.05$). The pattern is clear: ARQ attains the best mean rank and is significantly better than UDE3 ($p \approx 0.0096$), mLSHADE_RL ($p \approx 0.0039$), CMA-ES ($p \approx 3.4 \times 10^{-5}$), EA4Eig ($p \approx 1.2 \times 10^{-7}$), SaDE ($p \approx 1.7 \times 10^{-8}$), jDE ($p \approx 2.8 \times 10^{-9}$) and CLPSO ($p \approx 1.0 \times 10^{-13}$), while its differences with JSO ($p \approx 0.93$) and TRIDENT-DE ($p \approx 0.87$) are not significant. JSO and TRIDENT-DE form the next tier: each is significantly better than CMA-ES ($p \approx 0.015$ and$p \approx 4.6 \times 10^{-4}$, respectively) and markedly better than EA4Eig, SaDE, jDE and CLPSO ($p \leq 10^{-3}$), but neither differs from ARQ and they do not differ from each other. The mid-ranked group (UDE3, mLSHADE_RL, CMA-ES) shows mostly non-significant pairwise differences, except that CMA-ES is worse than JSO and TRIDENT-DE as noted above. CLPSO is the worst on average and is significantly worse than ARQ, JSO, TRIDENT-DE, UDE3, mLSHADE_RL, CMA-ES and EA4Eig ($p \leq 0.03$), but its gaps to SaDE and jDE are not significant at $\alpha = 0.05$ ($p \approx 0.073$ and $p \approx 0.15$, respectively). Overall, the global ranking differences are strong, with ARQ leading; JSO and TRIDENT-DE are competitive with ARQ, whereas the remaining methods particularly CMA-ES, EA4Eig, SaDE, jDE and CLPSO trail with multiple significant deficits.

Figure 16 reports the combined (best + mean) ranking as a single bar chart, offering a holistic view that aligns with the consolidated statistics in Table 10.

An additional point is breadth. The per-problem rankings show competitiveness on geometry-sensitive and ill-conditioned landscapes as well as on multimodal and application-driven tasks in energy and network planning, where incremental gains must accumulate reliably. That ARQ remains first overall despite strong specialists CMA-ES in highly correlated settings or JSO/mLSHADE_RL for mean stability suggests that its mechanisms address the common failure modes of stochastic optimization: premature convergence, accumulation of weak individuals in the tails, and coarse-grained restarts. Given the aligned configurations and transparent implementation, the evidence supports a substantive rather than accidental advantage.

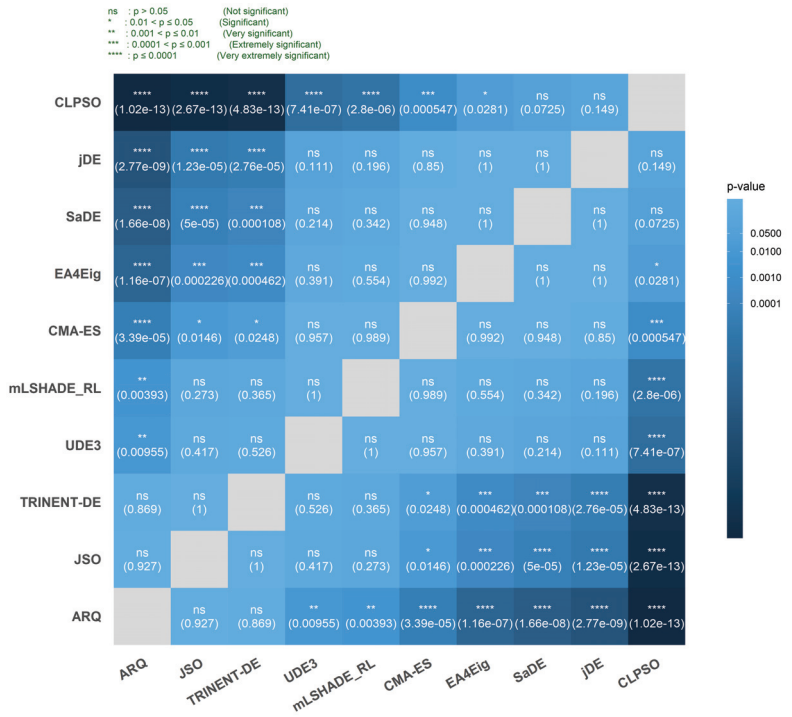


Figure 15. Per-method significance heatmap: ARQ vs. competitors (Wilcoxon signed-rank with Holm correction; effect direction encoded by color).

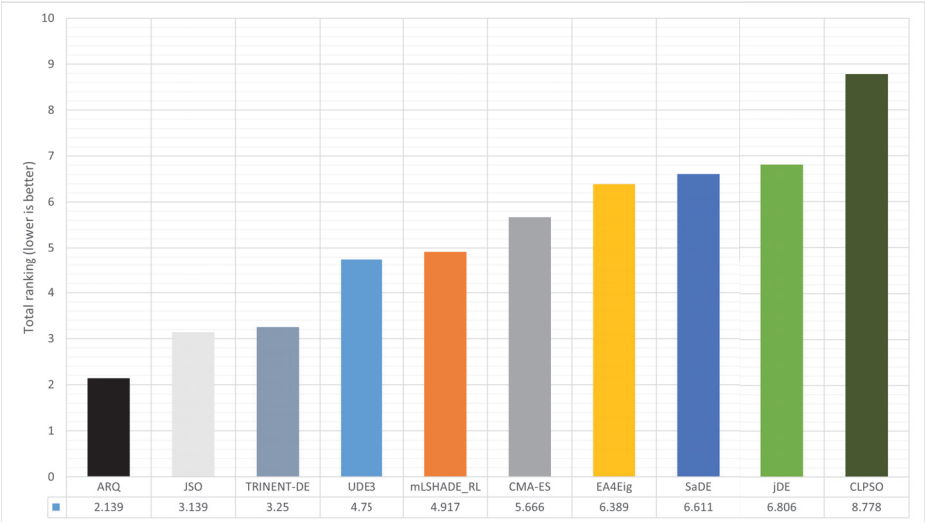


Figure 16. Total ranking combining best and mean across ten algorithms.

4. Conclusions

This study demonstrates that ARQ achieves the intended balance between strong end-of-budget best-so-far and preserved mean performance under a strictly harmonized protocol and a demanding comparator suite. Population size, evaluation budget, and opponent configurations are aligned to minimize confounders and ensure reproducibility.

The comparator set spans state-of-the-art DE variants and canonical non-DE references with complementary strengths self-adaptation, policy learning, eigenspace/second-order cues, and swarm strategies so conclusions are not confined to a single operator family.

In terms of metrics, Table 10 consolidates ARQ's advantage: it attains the lowest total rank on best and mean (49 and 28, sum 77, average rank 2.139), with the next best solver noticeably higher, indicating breadth rather than isolated spikes.

Table 9 clarifies the distributional shape behind this outcome a compact profile with frequent top placements and few poor runs while Table 8 shows that peaks do not come at the expense of stability, when ARQ narrowly trails in best, it typically compensates with a superior mean.

Mechanistically, the behavior is consistent with ARQ's cohesive design. Success-history adaptation quickly locks F/CR into profitable ranges, neighborhood-aware RTR preserves geometry and diversity while channeling selection pressure locally, outlier quarantine together with targeted micro-restarts trims tail failures and provides controlled recovery when progress stalls. The net effect is a persistently small best–mean gap that appears at the per-problem level and crystallizes in the aggregate ranks.

In the present study we targeted α, rtr_k, sh_c, p, and $stagnation_{trigger}$ because they map to ARQ's four core mechanisms (TrialGeneration, ParameterPolicy, SelectionRTR, and PopulationMaintenance with micro-restart) and they exhibit material, measured main effects in Tables 4–6 and Figures 7–9, thereby covering the decisive levers for mean and best performance as well as stability. In a forthcoming updated release of ARQ, we will also include principled techniques for parameter setting such as fractional–factorial screening and response–surface tuning together with automated controllers such as success-history adaptation and auto-tuning of the restart trigger, so that recommended defaults emerge from data rather than manual selection.

The contribution thus lies not in a single trick but in a principled combination of mature ideas that interact constructively, explaining why ARQ competes strongly against both DE and non-DE geometry-learning methods on ill-conditioned or correlated landscapes.

Validity is reinforced by three factors. First, strict resource alignment reduces the chance of artificial gains.

Second, the heterogeneous set of real-world tasks from physico-chemical potentials and code design to energy/network planning and interplanetary trajectories supports generalization.

Third, the explicit exposition of subroutines and controls makes the link between design choices and observed behavior transparent.

There are limitations. Results are reported for a 1.5×10^5 evaluation budget and a specific population size, while parameter sensitivity follows a structured methodology, a deeper map of the interactions among α, ρ, and w, together with the micro-restart trigger, would clarify edge-case trade-offs.

Moreover, although the benchmark suite is diverse, scenarios with time-varying noise or large-scale mixed-integer constraints remain under-explored.

These observations motivate several research avenues. Adaptive control of quarantine intensity via data-driven tuning of $\theta = Q3 + \alpha \times IQR$ and the repair fraction ρ, informed by tail statistics and diversity indices, could balance tail-cutting with exploration more finely.

Policy-learning to orchestrate the handover between RTR replacement and restart regimes may further compress the best–mean gap on stagnation-prone landscapes. Geometry-aware differences and archive schemes that periodically estimate principal directions could improve progress on highly correlated objectives without sacrificing DE's simplicity. Asynchronous mini-batch updates and population-fraction scheduling deserve attention for better hardware utilization. Finally, coupling ARQ with active resampling on noisy objectives, and profiling per-subroutine complexity relative to f-call costs, would sharpen guidance for tight-budget or expensive-evaluation settings.

What distinguishes ARQ. The tail-quarantine loop (with α for a robust threshold and ρ for controlled repair intensity), coupled with the robust center and micro-restarts, acts as a diversity-stabilizing mechanism that cooperates with success-history parameter adaptation. In practice, this delivers consistent mean performance together with strong best results, avoiding the large volatility often observed with adaptation-only DE settings.

Overall, the evidence supports a clear conclusion: a cohesive blend of parameter self-adaptation, neighborhood-aware selection, tail control, and targeted regeneration can deliver top-tier best-so-far while keeping the mean stable. Given resource alignment and implementation transparency, ARQ's advantage appears substantive and practically relevant, providing a dependable optimization tool for heterogeneous, challenging real-world landscapes.

Author Contributions: Conceptualization, I.G.T.; Software, V.C.; Validation, A.M.G. and D.T.; Visualization, A.M.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research has been financed by the European Union: Next Generation EU through the Program Greece 2.0 National Recovery and Resilience Plan, under the call RESEARCH-CREATE-INNOVATE, project name "iCREW: Intelligent small craft simulator for advanced crew training using Virtual Reality techniques" (project code: TAEDK-06195).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

greenReferences

1. Tapkin, A. A Comprehensive Overview of Gradient Descent and its Optimization Algorithms. *Int. Adv. Res. J. Sci. Eng. Technol.* **2023**, *10*, 37–45. [CrossRef]
2. Cawade, S.; Kudtarkar, A.; Sawant, S.; Wadekar, H. The Newton-Raphson Method: A Detailed Analysis. *Int. J. Res. Appl. Sci. Eng. (IJRASET)* **2024**, *12*, 729–734. [CrossRef]
3. Nelder, J.A.; Mead, R. A simplex method for function minimization. *Comput. J.* **1965**, *7*, 308–313. [CrossRef]
4. Bonate, P.L. A Brief Introduction to Monte Carlo Simulation. *Clin. Pharmacokinet.* **2001**, *40*, 15–22. [CrossRef] [PubMed]
5. Eglese, R.W. Simulated annealing: A tool for operational research. *Eur. J. Oper. Res.* **1990**, *46*, 271–281. [CrossRef]
6. Holland, J.H. *Adaptation in Natural and Artificial Systems*; University of Michigan Press: Ann Arbor, MI, USA, 1975.
7. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the ICNN'95-International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948. [CrossRef]
8. Liang, J.J.; Qin, A.K.; Suganthan, P.N.; Baskar, S. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE Trans. Evol. Comput.* **2006**, *10*, 281–295. [CrossRef]
9. Charilogis, V.; Tsoulos, I.G. Toward an Ideal Particle Swarm Optimizer for Multidimensional Functions. *Information* **2022**, *13*, 217. [CrossRef]
10. Charilogis, V.; Tsoulos, I.G.; Tzallas, A. An Improved Parallel Particle Swarm Optimization. *SN Comput. Sci.* **2023**, *4*, 766. [CrossRef]

11. Dorigo, M.; Di Caro, G. Ant Colony Optimization. In Proceedings of the 1999 Congress on Evolutionary Computation-CEC99, Washington, DC, USA, 6–9 July 1999; Volume 2, pp. 1470–1477. [CrossRef]
12. Karaboga, D. An idea based on honey bee swarm for numerical optimization: Artificial bee colony (ABC) algorithm. *J. Glob. Optim.* **2005**, *39*, 459–471. [CrossRef]
13. Kyrou, G.; Charilogis, V.; Tsoulos, I.G. Improving the Giant-Armadillo Optimization Method. *Analytics* **2024**, *3*, 225–240. [CrossRef]
14. Pant, M.; Zaheer, H.; Garcia-Hernandez, L.; Abraham, A. Differential Evolution: A review of more than two decades of research. *Eng. Appl. Artif. Intell.* **2020**, *90*, 103479. [CrossRef]
15. Tanabe, R.; Fukunaga, A. Success-History Based Parameter Adaptation for Differential Evolution (SHADE). In Proceedings of the 2013 IEEE Congress on Evolutionary Computation (CEC), Cancún, Mexico, 20–23 June 2013; pp. 71–78. [CrossRef]
16. Tanabe, R.; Fukunaga, A.S. Improving the search performance of SHADE using linear population size reduction (L-SHADE). In Proceedings of the 2014 IEEE Congress on Evolutionary Computation (CEC), Beijing, China, 6–11 July 2014; pp. 1658–1665. [CrossRef]
17. Brest, J.; Maučec, M.S.; Boskovic, B. jSO: An advanced differential evolution algorithm using success-history and linear population size reduction. In Proceedings of the 2017 IEEE Congress on Evolutionary Computation (CEC), Donostia/San Sebastián, Spain, 5–8 June 2017; pp. 1995–2002. [CrossRef]
18. Cao, Y.; Luan, J. A novel differential evolution algorithm with multi-population and elites regeneration. *PLoS ONE* **2024**, *19*, e0302207. [CrossRef]
19. Sun, Y.; Wu, Y.; Liu, Z. An improved differential evolution with adaptive population allocation and mutation selection. *Expert Syst. Appl.* **2024**, *258*, 125130. [CrossRef]
20. Hansen, N.; Ostermeier, A. Completely derandomized self-adaptation in evolution strategies. *Evol. Comput.* **2001**, *9*, 159–195. [CrossRef]
21. Bujok, P.; Kolenovský, P. Eigen crossover in cooperative model of evolutionary algorithms applied to CEC 2022 single objective numerical optimisation. In Proceedings of the 2022 IEEE Congress on Evolutionary Computation (CEC), Padua, Italy, 18–23 July 2022; pp. 1–8. [CrossRef]
22. Charilogis, V.; Tsoulos, I.G. Parallel Implementation of the Differential Evolution Method. *Analytics* **2023**, *2*, 17–30. [CrossRef]
23. Charilogis, V.; Tsoulos, I.G.; Tzallas, A.; Karvounis, E. Modifications for the Differential Evolution Algorithm. *Symmetry* **2022**, *14*, 447. [CrossRef]
24. Deng, W.; Shang, S.; Cai, X.; Zhao, H.; Song, Y.; Xu, J. An improved differential evolution algorithm and its application in optimization problem. *Soft Comput.* **2021**, *25*, 5277–5298. [CrossRef]
25. Chauhan, D.; Trivedi, A.; Shivani. A Multi-operator Ensemble LSHADE with Restart and Local Search Mechanisms for Single-objective Optimization. *arXiv* **2024**, arXiv:2409.15994. [CrossRef]
26. Deng, W.; Shang, S.; Zhang, L.; Lin, Y.; Huang, C.; Zhao, H.; Ran, X.; Zhou, X.; Chen, H. Multi-strategy quantum differential evolution algorithm with cooperative co-evolution and hybrid search for capacitated vehicle routing. *IEEE Trans. Intell. Transp. Syst.* **2025**, *26* , 18460–18470. [CrossRef]
27. Huber, P.J.; Ronchetti, E.M. *Robust Statistics*, 2nd ed.; Wiley: Hoboken, NJ, USA, 2009. [CrossRef]
28. Guo, D.; Zhang, S.; Yang, B.; Lin, Y.; Li, J. Exploring contextual knowledge-enhanced speech recognition in air traffic control communication: A comparative study. *IEEE Trans. Neural Netw. Learn. Syst.* **2025**, *36*, 16085–16099. [CrossRef] [PubMed]
29. Guo, D.; Zhang, S.; Lin, Y. Multi-modal intelligent situation awareness in real-time air traffic control: Control intent understanding and flight trajectory prediction. *Chin. J. Aeronaut.* **2024**, *37*, 103376. [CrossRef]
30. Zhang, J.; Sanderson, A.C. JADE: Adaptive differential evolution with optional external archive. *IEEE Trans. Evol. Comput.* **2009**, *13*, 945–958. [CrossRef]
31. Lima, C.F.; Lobo, F.G.; Goldberg, D.E. Investigating restricted tournament replacement in ECGA for stationary and non-stationary optimization. In Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation (GECCO '08), Atlanta, GA, USA, 12–16 July 2008; pp. 1–8. [CrossRef]
32. Deng, W.; Li, X.; Zhao, H.; Xu, J. PSO-K-means clustering-based NSGA-III for delay recovery. *IEEE Trans. Consum. Electron.* 2025, *Advance online publication*. [CrossRef]
33. Zhao, H.; Deng, W.; Lin, Y. Joint optimization scheduling using AHMQDE-ACO for key resources in smart operations. *IEEE Trans. Consum. Electron.* 2025, *Advance online publication*. [CrossRef]
34. Shen, Y.; Zhang, N.; Wang, Z.; Wang, X. Dual-performance multi-subpopulation adaptive restart differential evolution (DPR-MGDE). *Symmetry* **2025**, *17*, 223. [CrossRef]
35. Horita, H. Optimizing runtime business processes with fair workload distribution. *J. Compr. Bus. Adm. Res.* **2025**, *2*, 162–173. [CrossRef]

36. Sun, J.; Wang, Z.; Qiao, Z.; Li, X. Dynamic pricing model for e-commerce products based on DDQN. *J. Compr. Bus. Adm. Res.* **2024**, *1*, 171–178. [CrossRef]
37. Deng, W.; Li, K.; Zhao, H. A flight arrival time prediction method based on cluster clustering-based modular with deep neural network. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 6238–6247. [CrossRef]
38. Li, X.; Zhao, H.; Xu, J.; Deng, W. APDPFL: Anti-poisoning attack decentralized privacy-enhanced federated learning scheme for flight operation data sharing. *IEEE Trans. Wirel. Commun.* **2024**, *23*, 19098–19109. [CrossRef]
39. Ran, X.; Suyaroj, N.; Tepsan, W.; Lei, M.; Ma, H.; Zhou, X.; Deng, W. A novel fuzzy system-based genetic algorithm for trajectory segment generation in urban GPS. *J. Adv. Res.* 2025, *in press*. [CrossRef]
40. Lopatin, A. Intelligent system of estimation of total factor productivity (TFP) and investment efficiency in the economy with external technology gaps. *J. Compr. Bus. Adm. Res.* **2023**, *1*, 160–170. [CrossRef]
41. Charilogis, V.; Tsoulos, I.G.; Gianni, A.M. TRIDENT-DE: Triple-Operator Differential Evolution with Adaptive Restarts and Greedy Refinement. *Future Internet* **2025**, *17*, 488. [CrossRef]
42. Dehghani, M.; Trojovská, E.; Trojovský, P.; Malik, O.P. OOBO: A new metaheuristic algorithm for solving optimization problems. *Biomimetics* **2023**, *8*, 468. [CrossRef] [PubMed]
43. Tsoulos, I.G.; Charilogis, V.; Kyrou, G.; Stavrou, V.N.; Tzallas, A. OPTIMUS: A Multidimensional Global Optimization Package. *J. Open Source Softw.* **2025**, *10*, 7584. [CrossRef]
44. Das, S.; Abraham, A.; Chakraborty, U.K.; Konar, A. Differential evolution using a neighborhood-based mutation operator. *IEEE Trans. Evol. Comput.* **2009**, *13*, 526–553. [CrossRef]
45. Kluabwang, J.; Thomthong, T. Solving parameter identification of frequency modulation sounds problem by modified adaptive tabu search under management agent. *Procedia Eng.* **2012**, *31*, 1006–1011. [CrossRef]
46. Ahandan, M.A.; Alavi-Rad, H.; Jafari, N. Frequency modulation sound parameter identification using shuffled particle swarm optimization. *Int. J. Appl. Evol. Comput.* **2013**, *4*, 1–15. [CrossRef]
47. Lennard-Jones, J.E. On the determination of molecular fields. *Proc. R. Soc. A* **1924**, *106*, 463–477. [CrossRef]
48. Hofer, E.P. Optimization of bifunctional catalysts in tubular reactors. *J. Optim. Theory Appl.* **1976**, *18*, 379–393. [CrossRef]
49. Luus, R.; Dittrich, J.; Keil, F.J. Multiplicity of solutions in the optimization of a bifunctional catalyst blend in a tubular reactor. *Can. J. Chem. Eng.* **1992**, *70*, 780–785. [CrossRef]
50. Luus, R.; Bojkov, B. Global optimization of the bifunctional catalyst problem. *Can. J. Chem. Eng.* **1994**, *72*, 160–163. [CrossRef]
51. Javinsky, M.A.; Kadlec, R.H. Optimal control of a continuous flow stirred tank chemical reactor. *AIChE J.* **1970**, *16*, 916–924. [CrossRef]
52. Soukkou, A.; Khellaf, A.; Leulmi, S.; Boudeghdegh, K. Optimal control of a CSTR process. *Braz. J. Chem. Eng.* **2008**, *25*, 799–812. [CrossRef]
53. Pinheiro, C.I.C.; de Souza, M.B., Jr.; Lima, E.L. Model predictive control of reactor temperature in a CSTR with constraints. *Comput. Chem. Eng.* **1999**, *23*, 1553–1563. [CrossRef]
54. Tersoff, J. New empirical approach for the structure and energy of covalent systems. *Phys. Rev. B* **1988**, *37*, 6991–7000. [CrossRef] [PubMed]
55. Tersoff, J. Modeling solid-state chemistry: Interatomic potentials for multicomponent systems. *Phys. Rev. B* **1989**, *39*, 5566–5568. [CrossRef] [PubMed]
56. He, H.; Stoica, P.; Li, J. Designing unimodular sequence sets with good correlations-Including an application to MIMO radar. *IEEE Trans. Signal Process.* **2009**, *57*, 4391–4405. https://ieeexplore.ieee.org/document/5072243. [CrossRef]
57. Garver, L.L. Transmission network estimation using linear programming. *IEEE Trans. Power Appar. Syst.* **1970**, *PAS-89*, 1688–1697. [CrossRef]
58. Schweppe, F.C.; Caramanis, M.; Tabors, R.D.; Bohn, R.E. *Spot Pricing of Electricity*; Kluwer Academic Publishers: Dordrecht, The Netherlands, 1988. [CrossRef]
59. Balanis, C.A. *Antenna Theory: Analysis and Design*, 4th ed.; Wiley: Hoboken, NJ, USA, 2016.
60. Biscani, F.; Izzo, D.; Yam, C.H. Global Optimization for Space Trajectory Design (GTOP Database and Benchmarks). European Space Agency, Advanced Concepts Team. (GTOP Online Resource; See Also Related ACT Publications). 2010. Available online: https://www.esa.int/gsp/ACT/projects/gtop/ (accessed on 13 November 2025).
61. Calles-Esteban, F.; Olmedo, A.A.; Hellín, C.J.; Valledor, A.; Gómez, J.; Tayebi, A. Optimizing antenna positioning for enhanced wireless coverage: A genetic algorithm approach. *Sensors* **2024**, *24*, 2165. [CrossRef] [PubMed]
62. Li, J.; Chen, X.; Zhang, Y. Optimization of 5G base station coverage based on self-adaptive genetic algorithm. *Comput. Commun.* **2024**, *218*, 1–12. [CrossRef]
63. Das, S.; Suganthan, P.N. Problem Definitions and Evaluation Criteria for CEC 2011 Competition on Testing Evolutionary Algorithms on Real World Optimization Problems. Jadavpur University, Nanyang Technological University, Kolkata, 341–359.

2010. Available online: https://www.semanticscholar.org/paper/Problem-Definitions-and-Evaluation-Criteria-for-CEC-Das-Suganthan/d2f546248edd0c66d833c3e5f67e094e6922d262#citing-papers (accessed on 13 November 2025).
64. Lee, Y.; Filliben, J.; Micheals, R.; Phillips, J. *Sensitivity Analysis for Biometric Systems: A Methodology Based on Orthogonal Experiment Designs*; National Institute of Standards and Technology Gaithersburg (NISTIR): Gaithersburg, MD, USA, 2012. . [CrossRef]
65. Shah, R.K.; Sekulić, D.P. *Fundamentals of Heat Exchanger Design*; Wiley: Hoboken, NJ, USA, 2003. [CrossRef]
66. Serna, M.; Jiménez, A. A compact formulation of the Bell–Delaware method for heat exchanger design and optimization. *Chem. Eng. Res. Des.* **2005**, *83*, 539–550. [CrossRef]
67. Gonçalves, C.d.O.; Costa, A.L.H.; Bagajewicz, M.J. Linear method for the design of shell and tube heat exchangers using the Bell–Delaware method. *AIChE J.* **2019**, *65*, e16602. [CrossRef]
68. Moran, M.J.; Shapiro, H.N.; Boettner, D.D.; Bailey, M.B. *Fundamentals of Engineering Thermodynamics,* 9th ed.; Wiley: Hoboken, NJ, USA, 2019.
69. Coustenis, A.; Atreya, S.K.; Balint, T.; Brown, R.H.; Dougherty, M.; Dragonetti, Y.; Zarnecki, J.C. TandEM: Titan and Enceladus mission. *Exp. Astron.* **2009**, *23*, 893–946. [CrossRef]
70. Ceriotti, M. Global Optimisation of Multiple Gravity Assist Trajectories. Doctoral Dissertation, University of Glasgow, Glasgow, Scotland, 2010. Available online: https://theses.gla.ac.uk/2003/ (accessed on 13 November 2025).
71. Hinckley, D., Jr.; Parker, J.S. Global optimization of interplanetary trajectories using MGA-1DSM transcription (AAS 15-582). In Proceedings of the AAS/AIAA Astrodynamics Specialist Conference, Vail, CO, USA, 9–13 August 2015; NASA Technical Reports. Available online: https://ntrs.nasa.gov/api/citations/20150020817/downloads/20150020817.pdf (accessed on 13 November 2025).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

applied sciences

MDPI

Article

Feature Importance Ranking Using Interval-Valued Methods and Aggregation Functions for Machine Learning Applications

Aleksander Wojtowicz , Wiesław Paja * and Urszula Bentkowska

Faculty of Exact and Technical Sciences, University of Rzeszów, Al. Rejtana 16C, 35-959 Rzeszów, Poland; alwojtowicz@ur.edu.pl (A.W.); ubentkowska@ur.edu.pl (U.B.)
* Correspondence: wpaja@ur.edu.pl

Abstract: Feature selection is one of the key stages in the process of creating machine learning models and conducting data analysis. This paper presents the results of research related to the implementation of a new algorithm for feature selection and ranking based on weighted interval aggregations. It took into account interval importance values obtained from dividing the dataset into subsets. The algorithm was highly effective in identifying relevant features. The results of comparative studies with nine known methods of feature importance assessment are presented. Ten synthetic datasets and five real datasets were used for the experiments. The calculations also included tests of the relevance of the results obtained. In most experiments, the IVWFR algorithm proved to be the best, achieving the best classification results after identifying subsets of relevant features.

Keywords: feature ranking; feature selection; aggregation; interval-valued aggregation; machine learning; classification

1. Introduction

Feature selection is one of the key stages in the process of building machine learning models and conducting data analysis. The appropriate selection of a subset of input variables not only allows for dimensionality reduction and improved computational efficiency, but also increases the interpretability of models and reduces the risk of overfitting [1,2]. The literature on this subject describes many approaches to feature selection, including filtering, wrapper, and embedded methods [3,4]. Each of them differs in the mechanism of evaluating the usefulness of individual variables, as well as the trade-off between accuracy and computational costs.

One promising direction for feature selection is methods based on feature relevance aggregation using intervals. Their basic idea is to analyze the stability and repeatability of feature ranks by determining relevance measures in multiple repetitions or data divisions, and then grouping the results in the form of confidence intervals or rank aggregations [5–9]. This makes it possible not only to identify the most important features, but also to estimate the degree of certainty assigned to their position in the ranking.

The interval aggregation approach is used in situations where individual significance measures may be unstable, especially when there are a large number of features and a limited sample size [10]. The use of this type of method increases the resilience of the selection process to random data fluctuations and provides a more reliable picture of the significance of individual variables in the model [11,12].

The aim of this contribution is to carefully examine the method Interval-Valued Weighted Feature Ranking (IVWFR) proposed in [13]. IVWFR is a feature selection method that is built upon the feature importance derived from a classification model used in the method. The proposed algorithm aims to improve the stability and interpretability of feature selection by incorporating uncertainty (represented by the intervals using the idea of interval-valued fuzzy sets) and aggregating information across multiple data folds. In [13], only preliminary results on the new method were provided. Moreover, the synthetic datasets were the only ones applied in the performed experiments. In this contribution, thorough analysis of the IVWFR method and its performance on different datasets is presented. Both synthetic and real datasets are applied in the experiments. Statistical tests are performed to show the significant differences in the classification metric, balanced accuracy, in favor of the new IVWFR method. The results of the IVWFR method are compared with known feature selection methods: RFECV, RFE, ANOVA, Pearson correlation, Spearman correlation, Mutual Information, Random Forest, and XGBoost. In the experiments, RF denotes Random Forest trained on all features, while RF_Importance refers to a two-stage approach: first computing feature importance, then training a new Random Forest on selected features only. Furthermore, the number of features selected by each method is compared on each dataset and the Jaccard similarity index is computed. The results show that the IVWFR method outperforms many of the existing methods. IVWFR may have comparable results with respect to well-established methods like RFECV or RFE; however, its computational complexity is very competitive against time-consuming methods such as RFE or RFECV.

The structure of this paper is as follows: In Section 2, the basic notions connected with interval-valued calculus are recalled. In Section 3, details of the considered method are provided. Section 4 describes the datasets applied in the experiments. Finally, in Sections 5 and 6, the results of the experiments and the discussion are presented.

2. Interval Methods and Aggregation Functions

In the 1970s, Zadeh introduced interval-valued fuzzy sets (IVFSs) to represent ambiguity and uncertainty in practical applications [14]. Since then, IVFSs have been applied in various domains, including decision-making, pattern recognition, and control systems. Compared to conventional fuzzy sets, the use of interval-valued membership degrees allows for a more flexible and comprehensive representation of uncertainty and vagueness. For instance, due to an element's inherent ambiguity or variability, assigning a single membership degree may not be appropriate in certain applications. In such cases, an IVFS can be employed to capture the range of possible values or the degree of uncertainty associated with an element's membership.

One of the key advantages of IVFSs is their ability to handle imprecise and uncertain information more robustly than standard fuzzy sets. In real-world applications, IVFSs provide a more realistic representation of ambiguous and vague information, which can lead to more accurate and reliable decision-making. Furthermore, IVFSs can be effectively used to model complex and multidimensional uncertainty in situations where conventional fuzzy sets may be insufficient [15].

Interval Calculus is a mathematical framework designed for studying and analyzing real number intervals. The set L^I is the set of all subintervals of the unit interval $I = [0, 1]$, i.e., $L^I = \{[\underline{x}, \overline{x}], \underline{x} \leqslant \overline{x}, \underline{x}, \overline{x} \in [0, 1]\}$, where each interval can be represented by a pair of numbers $[\underline{x}, \overline{x}]$, such that $\underline{x} \leq \overline{x}$.

An interval-valued fuzzy set F in X is defined as a function $F : X \to L^I$, $F(x) = [\underline{F}(x), \overline{F}(x)] \in L^I$ for $x \in X$, where $\underline{F}(x), \overline{F}(x)$ denote fuzzy sets.

In our experiments, we represent the data with the use of intervals and we need to aggregate intervals. As a result, we use interval-valued aggregations.

Let $\mathbf{x} = [\underline{x}, \overline{x}], \mathbf{x} = [\underline{y}, \overline{y}] \in L^I$. $\mathcal{A} : (L^I)^n \to L^I$ is called an interval-valued aggregation function (cf. [16]) in L^I if it is increasing, i.e.,

$$\underset{\mathbf{x}_i, \mathbf{y}_i \in L^I}{\forall} \mathbf{x}_i \preceq \mathbf{y}_i \Rightarrow \mathcal{A}(\mathbf{x}_1, \ldots, \mathbf{x}_n) \preceq \mathcal{A}(\mathbf{y}_1, \ldots, \mathbf{y}_n)$$

and $\mathcal{A}(\underbrace{\mathbf{0}, ..., \mathbf{0}}_{n\times}) = \mathbf{0}$, $\mathcal{A}(\underbrace{\mathbf{1}, ..., \mathbf{1}}_{n\times}) = \mathbf{1}$, where $\mathbf{0} = [0, 0]$, $\mathbf{1} = [1, 1]$. The classical partial order for intervals is in the form $[\underline{x}, \overline{x}] \preceq [\underline{y}, \overline{y}] \Leftrightarrow \underline{x} \leqslant \underline{y}$, $\overline{x} \leqslant \overline{y}$; however, other orders (cf. [17]) or comparability relations for intervals may be used in the above monotonicity condition.

In our experiments, the interval-valued aggregation function based on the weighted arithmetic mean was used

$$\mathcal{A}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n) = \left[\sum_{i=1}^{n} \hat{w}_i \cdot \underline{x}_i, \sum_{i=1}^{n} \hat{w}_i \cdot \overline{x}_i \right] \tag{1}$$

where $\hat{w}_i$ denotes weights and $\sum_{i=1}^{n} \hat{w}_i = 1$; however, other weighted interval-valued aggregation functions may also be applied.

Below we present the methods of creating weights for aggregating n intervals, representing the importance of a feature F (obtained across n folds) denoted as

$$\mathbf{x}_i = [\underline{x}_i, \overline{x}_i] \quad \text{for } i = 1, 2, \ldots, n \tag{2}$$

where $\underline{x}_i$ and $\overline{x}_i$ represent the lower and upper bounds of the i-th interval, respectively. The goal is to combine these intervals into a single interval, where narrower intervals carry greater weight, reflecting their higher certainty. The strategy detailed below was adapted.

3. Algorithm and Experimental Details

3.1. Interval-Valued Weighted Feature Ranking

The Interval-Valued Weighted Feature Ranking (IVWFR) algorithm, proposed in [13], is a feature selection approach based on feature importance measures obtained from a classification model. In this study, the Random Forest classifier was employed for this purpose. The proposed algorithm aims to enhance the stability and interpretability of feature selection by incorporating uncertainty and aggregating importance information across multiple data folds. The IVWFR procedure is summarized in Algorithm 1.

This approach prioritizes robust feature selection and emphasizes the stability of feature importance while mitigating noise across data folds.

IVWFR is especially suitable for high-dimensional datasets. Comparing the computational efficiency of IVWFR to wrapper-based methods, we see that for a dataset with f features, n samples, and k cross-validation folds, the complexity of IVWFR is approximately $\mathcal{O}(k \cdot T_{\mathrm{RF}}(n, f))$, where $T_{\mathrm{RF}}(n, f)$ is the training time of a Random Forest classifier. In contrast, methods like RFE and RFECV perform repeated training on multiple feature subsets, resulting in a typical complexity of $\mathcal{O}(f \cdot T_{\mathrm{RF}}(n, f))$ or higher.

Algorithm 1: IVWFR: Interval-based Weighted Feature Ranking and Selection

Input:

X	Feature matrix (n samples, m features)
y	Target vector
k	Number of cross-validation splits
$Classifier$	Base model for importance estimation (e.g., RF)
aggregation_method	Strategy to combine importance intervals
selection_method	Strategy to select features
α, β	Weights for importance and uncertainty

Output: Selected feature subset
Trained classifier model on selected features

if *selection_method is 'contrast'* **then**
  Add a random 'contrast' feature to X;
end
Create k stratified folds from (X, y);
foreach *fold* $f \in \{1, \ldots, k\}$ **do**
  Train $Classifier_f$ on the training data of fold f;
  foreach *feature* $j \in \{1, \ldots, m\}$ **do**
    Compute importance mean $\mu_{j,f}$ and std $\sigma_{j,f}$ from $Classifier_f$;
    $I_{j,f} \leftarrow [\max(0, \mu_{j,f} - 1.96\sigma_{j,f}), \mu_{j,f} + 1.96\sigma_{j,f}]$;
  end
end
foreach *feature* $j \in \{1, \ldots, m\}$ **do**
  Aggregate intervals $\{I_{j,1}, \ldots, I_{j,k}\}$ into a single interval $\bar{I}_j$;
end
Normalize all aggregated intervals $\bar{I}_j$;
foreach *normalized interval* $\bar{I}_j$ **do**
  Calculate its center c_j and width w_j;
  Compute score $S_j \leftarrow \alpha \cdot c_j + \beta \cdot (1 - w_j)$;
end
Sort features by scores S in descending order;
Select final features X_{selected} based on the *selection_method*;
if *selection_method is 'contrast'* **then**
  Remove 'contrast' feature from X_{selected} if present;
end
Train final $Classifier$ on (X_{selected}, y);
return *Features in* $X_{selected}$ *and the final* $Classifier$;

3.2. Cross-Validation and Importance Interval Estimation

The input dataset is divided into several folds using stratified k-fold cross-validation, ensuring that each fold reflects the overall data distribution. The size of each training fold can be adjusted to a desired proportion of the total dataset. For each training fold, a Random Forest classifier is trained. Rather than relying on single-point estimates of feature importance, the algorithm computes confidence intervals for the importance of each feature. These intervals are derived from the distribution of importance values across individual trees within the Random Forest, providing a quantitative measure of the uncertainty associated with each feature's importance.

3.3. Interval-Valued Aggregation, Normalization, and Feature Scoring

The importance intervals for each feature obtained from different cross-validation folds are combined into a single representative interval using interval-valued aggregation functions. This aggregation process employs an interval version of the weighted arithmetic mean, where the weights are determined by the width of each interval and a selected representative point within it (such as the lower bound, upper bound, or midpoint). A small smoothing factor, denoted as ϵ, is incorporated to handle intervals with very small or zero widths. The aggregated intervals are then normalized to a common scale, ensuring comparability across features.

Based on the normalized intervals, a composite score is calculated for each feature. This score reflects a combination of the interval's central tendency (representing the average importance) and its width (representing the uncertainty). The relative contribution of the center and the width in the score calculation can be adjusted, and various types of averaging methods may be employed, such as arithmetic, geometric, or harmonic means. The details of the proposed procedure are summarized below.

Step 1: Calculate the width of each interval

For each interval x_i, the width is calculated as:

$$\text{width}_i = \overline{x}_i - \underline{x}_i \tag{3}$$

Step 2: Determine the representative value for each interval

A representative value r_i is chosen for each interval, which can be one of the following:

- Lower bound: $r_i = \underline{x}_i$;
- Upper bound: $r_i = \overline{x}_i$;
- Midpoint: $r_i = \frac{\underline{x}_i + \overline{x}_i}{2}$.

The midpoint is used as the default representative value in the presented method.

Step 3: Calculate the weight for each interval

The weight w_i for each interval is defined as the product of the inverse of the interval width and the representative value. This approach considers both the interval's precision and its magnitude:

$$w_i = \frac{r_i}{\text{width}_i + \epsilon} = \frac{r_i}{\overline{x}_i - \underline{x}_i + \epsilon} \tag{4}$$

where ϵ is a small positive constant (e.g., 10^{-5}) added to prevent division by zero when dealing with zero-width intervals.

Step 4: Normalize the weights

To ensure that the weights sum to 1, they are normalized as follows:

$$\hat{w}_i = \frac{w_i}{\sum_{j=1}^{n} w_j} \tag{5}$$

Step 5: Calculate the resulting interval

The final aggregated interval is computed according to Equation (1) as the weighted arithmetic mean of the $\mathbf{x}_i = [\underline{x}_i, \overline{x}_i]$ using the normalized weights determined as explained above.

3.4. Feature Selection and Final Model Training

Features are ranked in descending order based on their calculated scores. Given a feature F_k, we assume m intervals of importance $\mathbf{y}_k = [\underline{y}_k, \overline{y}_k]$ for $k = 1, 2, \ldots, m$ are available. The objective is to derive a score that links both the feature's importance (represented by the interval's center) and its stability (represented by the interval's width). This score will then be used to rank the features in terms of their overall significance.

Step 1: Interval Center and Width Calculation

For each interval $\mathbf{y}_k$,
calculate the center_k as the midpoint of the interval:

$$\text{center}_k = \frac{\underline{y}_k + \overline{y}_k}{2}. \tag{6}$$

Calculate the width_k of interval $\mathbf{y}_k = [\underline{y}_k, \overline{y}_k]$ as the difference between the upper and lower bounds (see Equation (3)), i.e., $\text{width}_k = \overline{y}_k - \underline{y}_k$.

Step 2: Normalization of Center and Width

To ensure comparability across features, we normalize the center and width values to the range $[0, 1]$:

Normalize the center:

$$\text{norm_center}_k = \frac{\text{center}_k - \min(\text{center})}{\max(\text{center}) - \min(\text{center})} \tag{7}$$

where $\min(\text{center})$ and $\max(\text{center})$ are the minimum and maximum center values across all intervals and all features, respectively.

Normalize the width:

$$\text{norm_width}_k = 1 - \frac{\text{width}_k - \min(\text{width})}{\max(\text{width}) - \min(\text{width})} \tag{8}$$

where $\min(\text{width})$ and $\max(\text{width})$ are the minimum and maximum width values across all intervals and all features, respectively. Note that we use $1 -$ normalized width so that smaller widths result in larger values, reflecting higher stability.

Step 3: Score Calculation

The score_k for each interval is computed as a weighted combination of the normalized center and the normalized width. The weights α and β dictate the relative influence of these two components. We investigate several averaging methods to combine these components:

Weighted arithmetic mean

$$\text{score}_k^{(A)} = \alpha \cdot \text{norm_center}_k + \beta \cdot \text{norm_width}_k \tag{9}$$

Weighted geometric mean

$$\text{score}_k^{(G)} = (\text{norm_center}_k)^{\alpha} \cdot (\text{norm_width}_k)^{\beta} \tag{10}$$

Weighted harmonic mean

$$\text{score}_k^{(H)} = \frac{\alpha + \beta}{\frac{\alpha}{\text{norm_center}_k + \epsilon} + \frac{\beta}{\text{norm_width}_k + \epsilon}} \tag{11}$$

where ϵ is a small positive constant (e.g., 10^{-5}) to prevent division by zero.

These averaging methods offer different sensitivities to the magnitude of the normalized center and width:

- Arithmetic mean—provides a linear balance between importance and stability.
- Geometric mean—exhibits higher sensitivity to smaller values (a near-zero value in either component will significantly lower the score).
- Harmonic mean—amplifies the impact of small values even further than the geometric mean, making it suitable when low stability or importance will drastically reduce the score.

Step 4: Feature Ranking

Finally, the features are ranked in descending order according to their score_k values. Features with higher scores are considered more important, as they exhibit both high relevance (a large central value) and high stability (a small width, resulting in a large normalized width). A detailed numerical example illustrating the proposed algorithm can be found in [13].

Subsequently, a subset of features is selected according to a chosen selection strategy. This strategy may involve choosing features whose scores exceed the average or median score. The algorithm also ensures that a minimum number of features are retained. Finally, a new IVWFR model is trained on the entire dataset using only the selected subset of features. This final model can then be used to perform predictions on previously unseen data.

3.5. Hyperparameters and Their Optimization Using Optuna

The IVWFR algorithm uses the following hyperparameters:

- `n_splits`—number of folds for stratified k-fold cross-validation.
- `fold_size`—desired proportion of the dataset used for each training fold.
- `random_state`—seed for the random number generator, ensuring reproducibility.
- `selection_method`—method for selecting the features mean and median.
- `alpha`—weight given to interval's center in feature score (average importance).
- `representative`—representative point in interval: left, right or center.
- `mean_type`—type of mean to combine center and width: arithmetic, geometric or harmonic.

The performance of the IVWFR algorithm depends on its hyperparameter configuration. To automate the search for optimal settings, the Optuna framework [18] is employed. At the core of the optimization process lies the objective function, which receives a set of hyperparameters as input and returns a performance score that Optuna seeks to maximize.

For each set of hyperparameters (a trial), Optuna uses its `suggest` methods to sample values for each parameter within predefined ranges. It then performs 3-fold stratified cross-validation on the input dataset. In each fold, the IVWFR model is trained on the training portion using the current hyperparameter values, and its feature ranking performance is evaluated on the test portion using a custom metric.

The custom metric is designed to assess how well the model ranks features according to their known types. It assigns a higher score when truly relevant features are ranked at the top, redundant features appear near the top, and irrelevant features are placed lower in the ranking. Specifically, the metric awards 1 point if a relevant feature appears among the top-ranked ones, 0.5 points if it is ranked slightly lower but still within the top set of relevant and redundant features, and 0.5 points if a redundant feature is included among the top-ranked relevant and redundant features (cf. [13]). The final score is normalized to the range $[0, 1]$, and the objective function computes the mean score across all three folds. Optuna then maximizes this average score during the optimization process.

After the optimization is complete, Optuna returns the hyperparameter combination that achieves the highest average metric value. These optimal hyperparameters are subsequently used to train the final IVWFR model for feature selection on the entire dataset.

3.6. Performance Metrics

To evaluate experiments, the following classification metrics are used: accuracy, balanced accuracy, recall, precision, and F1 score.

Accuracy [19] is calculated as the number of accurate predictions divided by the total amount of predictions and represents the overall accuracy of the model.

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

Balanced accuracy provides a more informative evaluation of model performance than conventional accuracy, particularly when dealing with imbalanced class distributions. It is defined as the arithmetic mean of the recall scores for each class [20]. This metric allows the performance on each class to be assessed independently while accounting for class imbalance, making it especially suitable for datasets with unequal class representation.

Let

$$\begin{gathered}\text{C—total number of classes,}\\ \text{i—each class's index,}\\ \text{TP}_i\text{—true positives for class } i,\\ \text{FN}_i\text{—false negatives for class } i,\\ \text{FP}_i\text{—false positives for class } i.\end{gathered}$$

The following formula is used for calculating multiclass balanced accuracy:

$$\text{Balanced Accuracy} = \frac{1}{\text{C}}\sum_{i=1}^{C}\frac{\text{TP}_i}{\text{TP}_i + \text{FN}_i}.$$

Recall [20] is defined as the ratio of true positive predictions to the total number of actual positive instances. It represents the proportion of positive cases that the model correctly identifies. Several variants of this metric exist for multiclass classification, including micro, macro, and weighted versions. In this study, we employ the weighted recall, which calculates the average recall across all classes by weighting each class according to its relative frequency in the dataset:

$$\text{Weighted Recall} = \sum_{i=1}^{C}\left(\frac{\text{TP}_i}{\text{TP}_i + \text{FN}_i}\cdot\frac{\text{Instances of class } i}{\text{Total instances}}\right).$$

Another metric used in the experiments is weighted precision. It is defined as the ratio of true positive predictions to the total number of true positive and false positive predictions. Precision [21] represents the proportion of positive predictions that are truly correct. The average precision across all classes is computed using weighted averaging, where the contribution of each class is weighted according to its relative frequency in the dataset:

$$\text{Weighted Precision} = \sum_{i=1}^{C}\left(\frac{\text{TP}_i}{\text{TP}_i + \text{FP}_i}\cdot\frac{\text{Instances of class } i}{\text{Total instances}}\right).$$

Finally, the F1 score [22] is defined as the harmonic mean of the model's precision and recall. Both precision and recall contribute equally to this metric. A model achieving a perfect F1 score demonstrates both perfect precision and perfect recall, meaning that it

makes no false positive or false negative predictions. In this study, we use the weighted F1 score, which accounts for class imbalance by computing the class-wise F1 scores and averaging them according to the number of samples in each class. The formula for the weighted F1 score is given below:

$$\text{Weighted F1 score} = \frac{2 \cdot \text{Weighted Precision} \cdot \text{Weighted Recall}}{\text{Weighted Precision} + \text{Weighted Recall}}.$$

4. Datasets

In the experiments, two types of datasets are used: real datasets and synthetic datasets. Characteristics of these datasets are presented in Table 1 and Table 2, respectively.

The synthetic datasets used in this study were obtained from the `SynthSelect` toolkit [23]. This toolkit provides a collection of ten benchmark datasets, each containing predefined relevant, redundant, and irrelevant features. These datasets exhibit varying levels of complexity, making them particularly suitable for evaluating feature selection algorithms. Among them, the Double Spiral and 5D XOR datasets are considered the most challenging in terms of correctly identifying feature types (see Table 3 and Figures 11 and 12 in [23]). All datasets provided by the `SynthSelect` toolkit were used in our experiments, and to avoid potential overfitting, cross-validation was applied during both feature selection and performance evaluation phases.

In addition, five real-world datasets were employed: Climate, Estimation of Obesity Levels, Iranian Churn, and Sonar from the UCI Machine Learning Repository, as well as the Water Potability dataset from Kaggle. Using both synthetic and real datasets enables a comprehensive assessment of the proposed method's effectiveness.

All experiments were conducted in Python 3.13 using open-source libraries, including NumPy 2.3.3, pandas 2.3.2, SciPy 1.16.2, scikit-learn 1.7.2, XGBoost 3.0.5, Optuna 4.5.0, and matplotlib 3.10.6.

Table 1. Summary of real datasets.

Dataset	Rows	Features	Classes (Ratios)
Climate	540	18	2 (0.09, 0.91)
Estimation of Obesity	2111	16	7 (0.13, 0.14, 0.17, 0.14, 0.15, 0.14, 0.14)
Iranian Churn	3150	13	2 (0.84, 0.16)
Sonar	208	60	2 (0.53, 0.47)
Water Potability	3276	9	2 (0.61, 0.39)

Table 2. Summary of synthetic datasets. In this case, Rel, Red, and Irr denote the ground truth number of relevant, redundant, and irrelevant features in the dataset, respectively.

Dataset	Rows	Features	Classes (Ratios)	Rel	Red	Irr
10Class Multicut	500	126	10 (0.09, 0.06, 0.04, 0.04, 0.14, 0.18, 0.14, 0.14, 0.09, 0.09)	6	20	100
4D AND	100	113	2 (0.56, 0.44)	4	9	100
5Class Multicut	500	126	5 (0.03, 0.05, 0.05, 0.17, 0.70)	6	20	100
5D XOR	100	115	2 (0.49, 0.51)	5	10	100
Cone	400	123	2 (0.54, 0.46)	3	20	100
Double Spiral	200	303	2 (0.50, 0.50)	3	37	263
Hypersphere 3D	400	123	2 (0.52, 0.48)	3	20	100
Trig	200	57	2 (0.56, 0.43)	2	5	50
Yinyang	601	62	2 (0.51, 0.49)	2	10	50
y = x	200	122	2 (0.49, 0.51)	2	20	100

5. Results

This section presents a comparative analysis of the performance of various feature selection models on both real-world and synthetic datasets. We begin with a visual examination of the models' performance, followed by a statistical evaluation assessing the significance of the obtained results.

5.1. Performance Analysis

Figures A1–A10 present the results for the ten synthetic datasets, which were designed to evaluate model behavior under controlled conditions. Figure 1 consolidates these results to illustrate overall performance trends.

Similarly, Figures A11–A15 show the performance of each model on the five real-world datasets, while Figure 2 summarizes the results obtained on this practical, non-ideal data.

The models are ranked in descending order according to their balanced accuracy. For most of the real-world datasets, the proposed IVWFR method achieved the best performance. This observation is further confirmed in Figure 2, which aggregates the results from all real datasets.

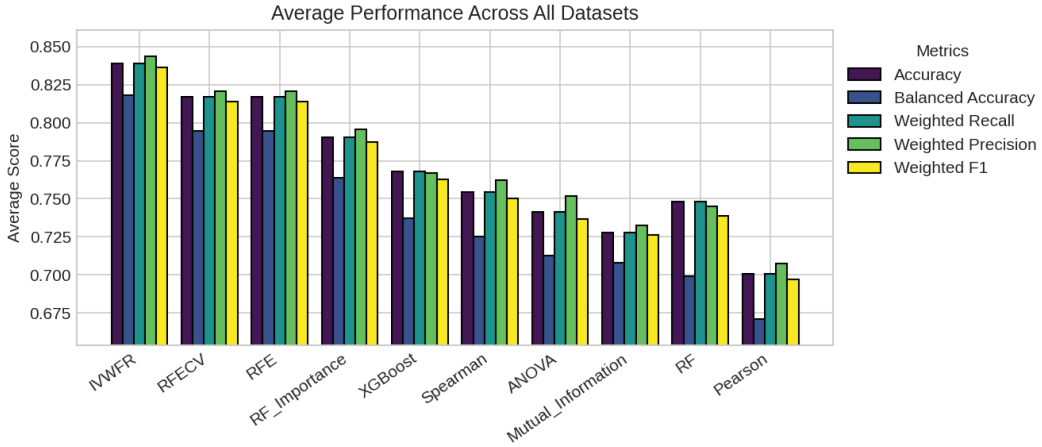


Figure 1. Summary of performance on synthetic datasets. This figure provides a consolidated view of the results across all synthetic datasets.

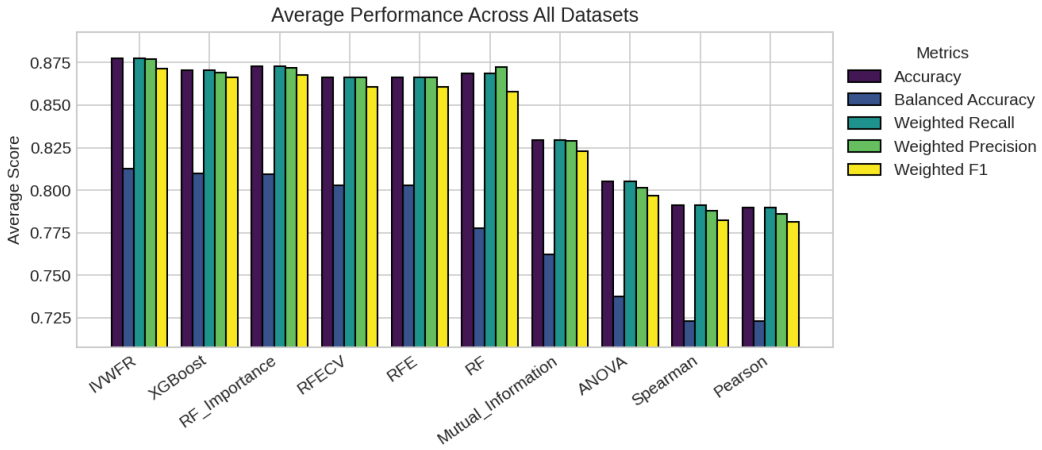


Figure 2. Summary of performance on real datasets. This figure provides a consolidated view of the results across all real datasets.

A similar trend is observed in the case of synthetic datasets. The proposed IVWFR method appears particularly effective for the Double Spiral and 5D XOR datasets, which-as

previously mentioned-are among the most challenging due to the difficulty of correctly identifying feature types. For these datasets, the performance differences in favor of IVWFR are the most pronounced. As shown in Figure 1, which consolidates all synthetic results, IVWFR consistently outperforms the competing methods.

5.2. Statistical Significance and Discussion

To assess the statistical significance of the performance differences, the Wilcoxon signed-rank test was conducted separately for the real and synthetic datasets. This non-parametric test compares pairs of models based on their balanced accuracy scores. The resulting p-values, corrected for multiple comparisons using the Bonferroni method, are reported in Tables 3 and 4.

For the real-world datasets (Table 4), performance varied across models; however, IVWFR demonstrated a statistically significant advantage over traditional methods such as ANOVA, Pearson, and Spearman correlation. Nonetheless, many models—including RFECV, RFE, and the baseline classifiers—did not differ significantly from one another, suggesting that for these datasets, the use of more sophisticated feature selection methods may not be crucial. The lack of significant differences between some methods on real data may result from reaching a "ceiling" in model quality, while IVWFR stands out with fewer selected features while maintaining the same effectiveness.

In contrast, the analysis of the synthetic datasets (Table 3) revealed more pronounced distinctions among the evaluated methods. In this case, the IVWFR model significantly outperformed nearly all competing approaches, indicating that the algorithm is particularly effective at identifying relevant features in datasets with complex and well-defined structures. As expected, the performance of RFE and RFECV was statistically indistinguishable, since RFE was configured to select the same number of features that RFECV determined as optimal.

Table 3. p-values for synthetic datasets from the Wilcoxon signed-rank test *.

	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF-Imp.	Mutual Inf.	RFE	RF	XGBoost
IVWFR	-	**0.041**	$\mathbf{2.2 \times 10^{-7}}$	$\mathbf{1.4 \times 10^{-7}}$	$\mathbf{1.3 \times 10^{-6}}$	$\mathbf{2.3 \times 10^{-4}}$	$\mathbf{2.1 \times 10^{-6}}$	**0.041**	$\mathbf{5.9 \times 10^{-8}}$	$\mathbf{2.1 \times 10^{-6}}$
RFECV	**0.041**	-	$\mathbf{5.3 \times 10^{-5}}$	$\mathbf{1.4 \times 10^{-5}}$	**0.008**	**0.038**	$\mathbf{6.1 \times 10^{-4}}$	1	$\mathbf{2.0 \times 10^{-6}}$	$\mathbf{1.6 \times 10^{-4}}$
ANOVA	$\mathbf{2.2 \times 10^{-7}}$	$\mathbf{5.3 \times 10^{-5}}$	-	0.217	1	0.423	1	$\mathbf{5.3 \times 10^{-5}}$	1	1
Pearson	$\mathbf{1.4 \times 10^{-7}}$	$\mathbf{1.4 \times 10^{-5}}$	0.217	-	**0.022**	$\mathbf{8.3 \times 10^{-5}}$	1	$\mathbf{1.4 \times 10^{-5}}$	1	0.259
Spearman	$\mathbf{1.3 \times 10^{-6}}$	**0.008**	1	**0.022**	-	1	1	**0.008**	1	1
RF-Imp.	$\mathbf{2.3 \times 10^{-4}}$	**0.038**	0.423	$\mathbf{8.3 \times 10^{-5}}$	1	-	**0.024**	**0.038**	**0.005**	0.198
Mutual Inf.	$\mathbf{2.1 \times 10^{-6}}$	$\mathbf{6.1 \times 10^{-4}}$	1	1	1	**0.024**	-	$\mathbf{6.1 \times 10^{-4}}$	1	1
RFE	**0.041**	1	$\mathbf{5.3 \times 10^{-5}}$	$\mathbf{1.4 \times 10^{-5}}$	**0.008**	**0.038**	$\mathbf{6.1 \times 10^{-4}}$	-	$\mathbf{2.0 \times 10^{-6}}$	$\mathbf{1.6 \times 10^{-4}}$
RF	$\mathbf{5.9 \times 10^{-8}}$	$\mathbf{2.0 \times 10^{-6}}$	1	1	1	**0.005**	1	$\mathbf{2.0 \times 10^{-6}}$	-	0.331
XGBoost	$\mathbf{2.1 \times 10^{-6}}$	$\mathbf{1.6 \times 10^{-4}}$	1	0.259	1	0.198	1	$\mathbf{1.6 \times 10^{-4}}$	0.331	-

* RF-Imp.: RF Importance; Mutual Inf.: Mutual Information. The table shows Bonferroni-corrected p-values. Values in bold indicate statistical significance ($p < 0.05$).

Table 4. p-values for real datasets from the Wilcoxon signed-rank test *.

	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF-Imp.	Mutual Inf.	RFE	RF	XGBoost
IVWFR	-	1	**0.010**	**0.010**	**0.005**	1	**0.021**	1	0.552	1
RFECV	1	-	0.114	0.114	0.292	1	0.321	1	1	1
ANOVA	**0.010**	0.114	-	1	1	**0.008**	1	0.114	1	0.064
Pearson	**0.010**	0.114	1	-	1	**0.008**	0.834	0.114	1	0.064
Spearman	**0.005**	0.292	1	1	-	**0.029**	0.999	0.292	1	**0.040**
RF-Imp.	1	1	**0.008**	**0.008**	**0.029**	-	0.052	1	1	1
Mutual Inf.	**0.021**	0.321	1	0.834	0.999	0.052	-	0.321	1	0.097
RFE	1	1	0.114	0.114	0.292	1	0.321	-	1	1
RF	0.552	1	1	1	1	1	1	1	-	1
XGBoost	1	1	0.064	0.064	**0.040**	1	0.097	1	1	-

* RF-Imp.: RF Importance; Mutual Inf.: Mutual Information. The table shows Bonferroni-corrected p-values. Values in bold indicate statistical significance ($p < 0.05$).

5.3. Number of Features Selected

Before analyzing feature overlap, we first examine the number of features selected by each method, as this provides essential context for interpreting the Jaccard similarity scores. Tables 5 and 6 summarize the feature counts for the real-world and synthetic datasets, respectively.

Several key patterns emerge from these results. By experimental design, most conventional methods (from RFECV to RFE) select the same number of features, as this count is fixed to the optimum determined by RFECV. In contrast, the IVWFR method frequently selects a different number of features, reflecting its distinct selection strategy. Finally, the RF and XGBoost models show the highest feature counts because they serve as baseline classifiers that utilize all available features. This variation in the number of selected features is an important factor influencing the subsequent overlap analysis. The number of features for comparative methods is set at the optimal level determined by RFECV to enable a reliable comparison with an identical feature space size. In order to assess sensitivity, we conduct an additional analysis in which the number of selected features is changed and the quality of the models is measured. The results of this analysis for individual datasets are presented in the graphs in Figures A16–A30 in Appendix A.

Table 5. Number of features selected for each synthetic dataset and each method. RF-Imp. is an abbreviation for RF Importance and Mutual Inf. for Mutual Information.

Dataset	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF Imp.	Mutual Inf.	RFE	RF	XGBoost
5Class Multicut	8	7	7	7	7	7	7	7	126	126
10Class Multicut	39	11	11	11	11	11	11	11	126	126
Trig	5	5	5	5	5	5	5	5	57	57
Hypersphere 3D	19	95	95	95	95	95	95	95	123	123
4D AND	25	4	4	4	4	4	4	4	113	113
Double Spiral	152	227	227	227	227	227	227	227	303	303
Cone	26	25	25	25	25	25	25	25	123	123
5D XOR	40	35	35	35	35	35	35	35	115	115
y = x	23	14	14	14	14	14	14	14	122	122
Yinyang	11	7	7	7	7	7	7	7	62	62

Table 6. Number of features selected for each real dataset and each method. RF-Imp. is an abbreviation for RF Importance and Mutual Inf. for Mutual Information.

Dataset	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF Imp.	Mutual Inf.	RFE	RF	XGBoost
estimation_of_obesity	5	2	2	2	2	2	2	2	16	16
water_potability	5	4	4	4	4	4	4	4	9	9
iranian_churn	13	11	11	11	11	11	11	11	13	13
sonar	30	47	47	47	47	47	47	47	60	60
climate	5	4	4	4	4	4	4	4	18	18

Simultaneously with the analysis of classification quality, in the case of synthetic data, an analysis of ground truth feature identification was performed. The TPRel, TPRed and TPIrr evaluation measures were used for this purpose, which stand for true positive relevant, true positive redundant, and true positive irrelevant [13]. In addition, the TCC measure, or total correct classification, also defined in [13], was used, combining TPRel and TPRed as the quality of identification of all relevant features in the set. The results of individual measures are presented in Figure 3, and a summary of these results for the six best feature selection methods is presented in Figure 4.

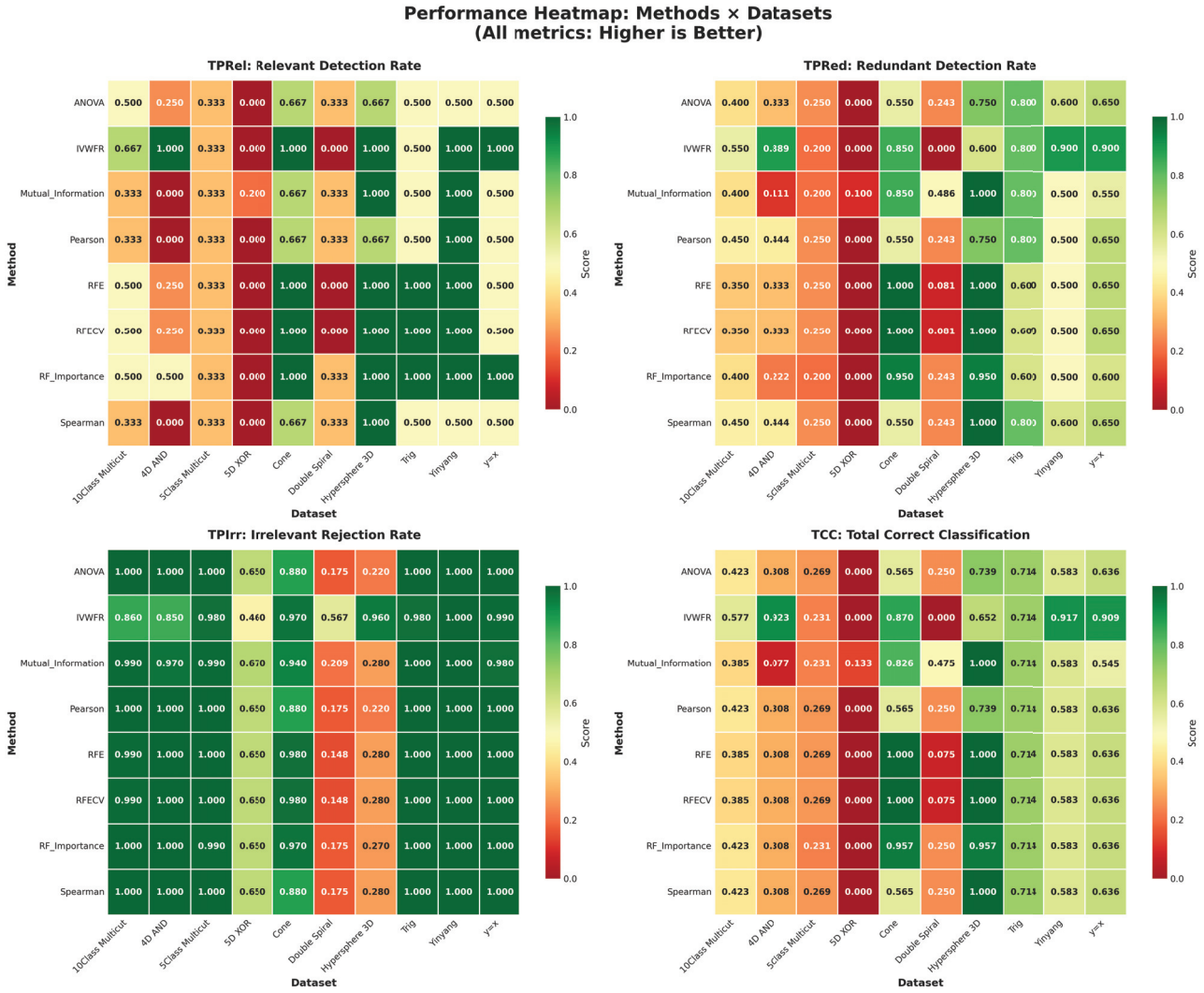


Figure 3. Results of ground truth feature identification for synthetic datasets in the context of the feature selection method used. Individual graphs show the identification of relevant (TPRel), redundant (TPRed), and irrelevant (TPIrr) features, and total correct classification (TCC), i.e., the correct identification of relevant and redundant features [13].

5.4. Feature Selection Overlap Analysis

Beyond performance metrics, we also examined the degree of agreement among feature selection models regarding which features they identified as important. The Jaccard similarity index was used to quantify the overlap between the feature subsets selected by each pair of models, where a value of 1.0 indicates identical sets and a value of 0.0 indicates no shared features.

5.4.1. Overlap on Synthetic Datasets

The analysis of the synthetic datasets (Tables 7–16) makes the algorithmic differences between the models more apparent. Notably, IVWFR often selects a different number of features compared to the other methods, which are constrained by the RFECV output. This confirms that IVWFR relies on a fundamentally different principle for identifying the optimal feature set. While certain groups of methods (e.g., statistical filters) still show some degree of agreement, the overall similarity is lower than that observed for the real-world datasets. Consequently, the complex structures of the synthetic data are more effective in revealing the distinctive behavior of each feature selection strategy.

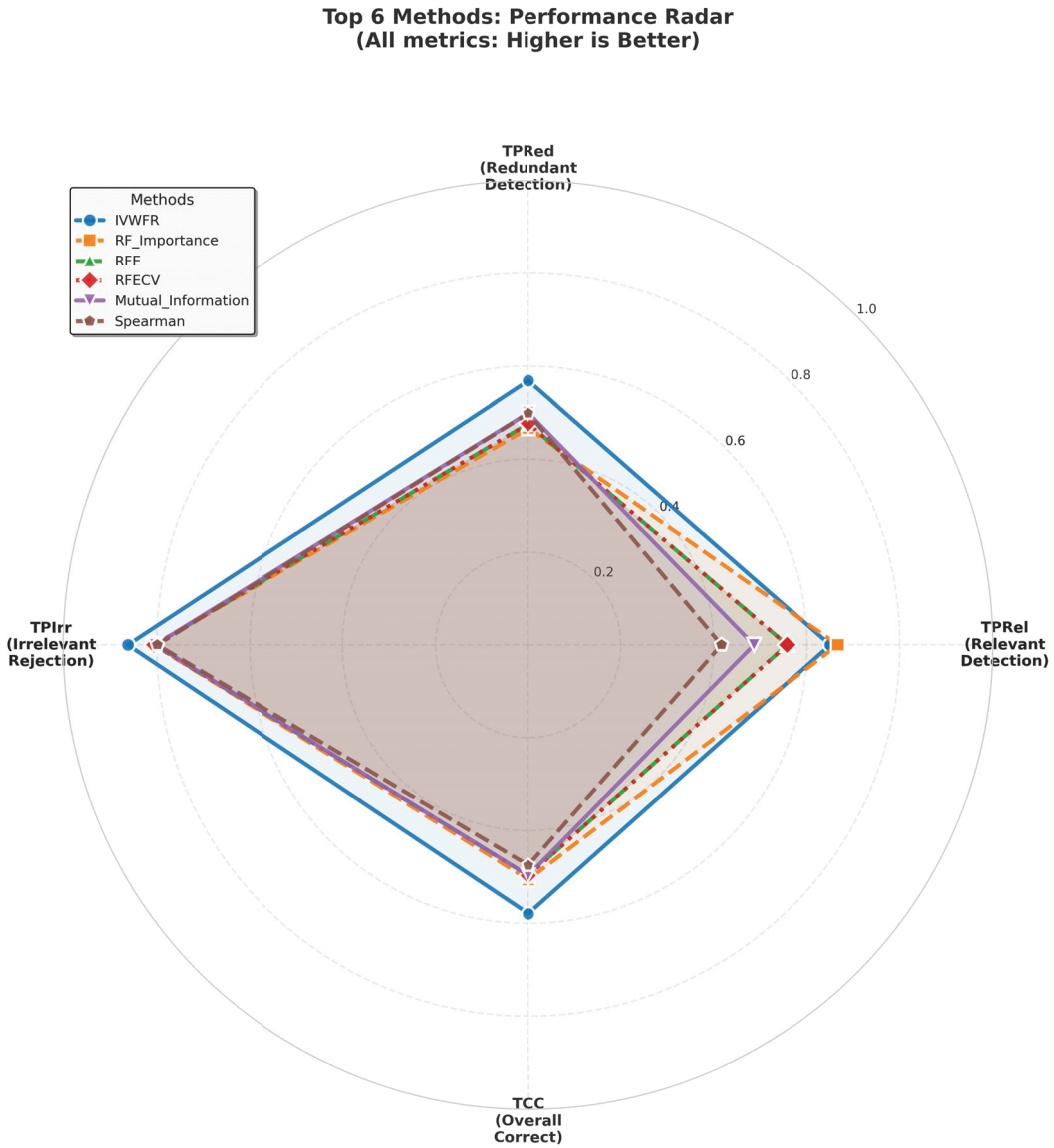


Figure 4. Summary results of ground truth feature identification for synthetic databases in the context of the six best feature selection methods used. The results for the RFE and RFECV methods are identical and overlap.

Table 7. Jaccard similarity on the 5Class Multicut dataset.

	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF-Imp.	Mutual Inf.	RFE	RF	XGBoost
IVWFR	1.000	0.667	0.667	0.154	0.667	0.667	0.667	0.667	0.063	0.063
RFECV	0.667	1.000	0.750	0.167	0.750	0.750	0.750	1.000	0.056	0.056
ANOVA	0.667	0.750	1.000	0.167	0.750	0.750	0.750	0.750	0.056	0.056
Pearson	0.154	0.167	0.167	1.000	0.273	0.167	0.167	0.167	0.056	0.056
Spearman	0.667	0.750	0.750	0.273	1.000	0.750	0.750	0.750	0.056	0.056
RF-Imp.	0.667	0.750	0.750	0.167	0.750	1.000	0.750	0.750	0.056	0.056
Mutual Inf.	0.667	0.750	0.750	0.167	0.750	0.750	1.000	0.750	0.056	0.056
RFE	0.667	1.000	0.750	0.167	0.750	0.750	0.750	1.000	0.056	0.056
RF	0.063	0.056	0.056	0.056	0.056	0.056	0.056	0.056	1.000	1.000
XGBoost	0.063	0.056	0.056	0.056	0.056	0.056	0.056	0.056	1.000	1.000

Table 8. Jaccard similarity on the 10Class Multicut dataset.

	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF-Imp.	Mutual Inf.	RFE	RF	XGBoost
IVWFR	1.000	0.282	0.250	0.250	0.250	0.250	0.220	0.282	0.310	0.310
RFECV	0.282	1.000	0.571	0.294	0.467	0.571	0.467	1.000	0.087	0.087
ANOVA	0.250	0.571	1.000	0.467	0.692	1.000	0.692	0.571	0.087	0.087
Pearson	0.250	0.294	0.467	1.000	0.692	0.467	0.467	0.294	0.087	0.087
Spearman	0.250	0.467	0.692	0.692	1.000	0.692	0.467	0.467	0.087	0.087
RF-Imp.	0.250	0.571	1.000	0.467	0.692	1.000	0.692	0.571	0.087	0.087
Mutual Inf.	0.220	0.467	0.692	0.467	0.467	0.692	1.000	0.467	0.087	0.087
RFE	0.282	1.000	0.571	0.294	0.467	0.571	0.467	1.000	0.087	0.087
RF	0.310	0.087	0.087	0.087	0.087	0.087	0.087	0.087	1.000	1.000
XGBoost	0.310	0.087	0.087	0.087	0.087	0.087	0.087	0.087	1.000	1.000

Table 9. Jaccard similarity on the Trig dataset.

	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF-Imp.	Mutual Inf.	RFE	RF	XGBoost
IVWFR	1.000	0.429	0.250	0.250	0.250	0.429	0.429	0.429	0.088	0.088
RFECV	0.429	1.000	0.429	0.429	0.429	0.667	0.667	1.000	0.088	0.088
ANOVA	0.250	0.429	1.000	1.000	1.000	0.429	0.429	0.429	0.088	0.088
Pearson	0.250	0.429	1.000	1.000	1.000	0.429	0.429	0.429	0.088	0.088
Spearman	0.250	0.429	1.000	1.000	1.000	0.429	0.429	0.429	0.088	0.088
RF-Imp.	0.429	0.667	0.429	0.429	0.429	1.000	0.429	0.667	0.088	0.088
Mutual Inf.	0.429	0.667	0.429	0.429	0.429	0.429	1.000	0.667	0.088	0.088
RFE	0.429	1.000	0.429	0.429	0.429	0.667	0.667	1.000	0.088	0.088
RF	0.088	0.088	0.088	0.088	0.088	0.088	0.088	0.088	1.000	1.000
XGBoost	0.088	0.088	0.088	0.088	0.088	0.088	0.088	0.088	1.000	1.000

Table 10. Jaccard similarity on the Hypersphere 3D dataset.

	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF-Imp.	Mutual Inf.	RFE	RF	XGBoost
IVWFR	1.000	0.188	0.118	0.118	0.200	0.200	0.188	0.188	0.154	0.154
RFECV	0.188	1.000	0.727	0.727	0.727	0.712	0.624	1.000	0.772	0.772
ANOVA	0.118	0.727	1.000	1.000	0.727	0.667	0.652	0.727	0.772	0.772
Pearson	0.118	0.727	1.000	1.000	0.727	0.667	0.652	0.727	0.772	0.772
Spearman	0.200	0.727	0.727	0.727	1.000	0.681	0.638	0.727	0.772	0.772
RF-Imp.	0.200	0.712	0.667	0.667	0.681	1.000	0.681	0.712	0.772	0.772
Mutual Inf.	0.188	0.624	0.652	0.652	0.638	0.681	1.000	0.624	0.772	0.772
RFE	0.188	1.000	0.727	0.727	0.727	0.712	0.624	1.000	0.772	0.772
RF	0.154	0.772	0.772	0.772	0.772	0.772	0.772	0.772	1.000	1.000
XGBoost	0.154	0.772	0.772	0.772	0.772	0.772	0.772	0.772	1.000	1.000

Table 11. Jaccard similarity on the 4D AND dataset.

	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF-Imp.	Mutual Inf.	RFE	RF	XGBoost
IVWFR	1.000	0.160	0.160	0.160	0.160	0.160	0.115	0.160	0.221	0.221
RFECV	0.160	1.000	0.333	0.143	0.333	0.143	0.143	1.000	0.035	0.035
ANOVA	0.160	0.333	1.000	0.143	0.333	0.333	0.143	0.333	0.035	0.035
Pearson	0.160	0.143	0.143	1.000	0.600	0.000	0.143	0.143	0.035	0.035
Spearman	0.160	0.333	0.333	0.600	1.000	0.000	0.143	0.333	0.035	0.035
RF-Imp.	0.160	0.143	0.333	0.000	0.000	1.000	0.000	0.143	0.035	0.035
Mutual Inf.	0.115	0.143	0.143	0.143	0.143	0.000	1.000	0.143	0.035	0.035
RFE	0.160	1.000	0.333	0.143	0.333	0.143	0.143	1.000	0.035	0.035
RF	0.221	0.035	0.035	0.035	0.035	0.035	0.035	0.035	1.000	1.000
XGBoost	0.221	0.035	0.035	0.035	0.035	0.035	0.035	0.035	1.000	1.000

Table 12. Jaccard similarity on the Double Spiral dataset.

	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF-Imp.	Mutual Inf.	RFE	RF	XGBoost
IVWFR	1.000	0.592	0.522	0.522	0.566	0.560	0.486	0.592	0.502	0.502
RFECV	0.592	1.000	0.739	0.739	0.733	0.733	0.645	1.000	0.749	0.749
ANOVA	0.522	0.739	1.000	1.000	0.823	0.694	0.621	0.739	0.749	0.749
Pearson	0.522	0.739	1.000	1.000	0.823	0.694	0.621	0.739	0.749	0.749
Spearman	0.566	0.733	0.823	0.823	1.000	0.700	0.610	0.733	0.749	0.749
RF-Imp.	0.560	0.733	0.694	0.694	0.700	1.000	0.633	0.733	0.749	0.749
Mutual Inf.	0.486	0.645	0.621	0.621	0.610	0.633	1.000	0.645	0.749	0.749
RFE	0.592	1.000	0.739	0.739	0.733	0.733	0.645	1.000	0.749	0.749
RF	0.502	0.749	0.749	0.749	0.749	0.749	0.749	0.749	1.000	1.000
XGBoost	0.502	0.749	0.749	0.749	0.749	0.749	0.749	0.749	1.000	1.000

Table 13. Jaccard similarity on the Cone dataset.

	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF-Imp.	Mutual Inf.	RFE	RF	XGBoost
IVWFR	1.000	0.821	0.342	0.342	0.378	0.821	0.545	0.821	0.211	0.211
RFECV	0.821	1.000	0.389	0.389	0.389	0.923	0.613	1.000	0.203	0.203
ANOVA	0.342	0.389	1.000	1.000	0.613	0.351	0.351	0.389	0.203	0.203
Pearson	0.342	0.389	1.000	1.000	0.613	0.351	0.351	0.389	0.203	0.203
Spearman	0.378	0.389	0.613	0.613	1.000	0.351	0.220	0.389	0.203	0.203
RF-Imp.	0.821	0.923	0.351	0.351	0.351	1.000	0.613	0.923	0.203	0.203
Mutual Inf.	0.545	0.613	0.351	0.351	0.220	0.613	1.000	0.613	0.203	0.203
RFE	0.821	1.000	0.389	0.389	0.389	0.923	0.613	1.000	0.203	0.203
RF	0.211	0.203	0.203	0.203	0.203	0.203	0.203	0.203	1.000	1.000
XGBoost	0.211	0.203	0.203	0.203	0.203	0.203	0.203	0.203	1.000	1.000

Table 14. Jaccard similarity on the 5D XOR dataset.

	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF-Imp.	Mutual Inf.	RFE	RF	XGBoost
IVWFR	1.000	0.442	0.630	0.630	0.630	0.442	0.172	0.442	0.348	0.348
RFECV	0.442	1.000	0.556	0.556	0.556	0.321	0.167	1.000	0.304	0.304
ANOVA	0.630	0.556	1.000	1.000	1.000	0.429	0.186	0.556	0.304	0.304
Pearson	0.630	0.556	1.000	1.000	1.000	0.429	0.186	0.556	0.304	0.304
Spearman	0.630	0.556	1.000	1.000	1.000	0.429	0.186	0.556	0.304	0.304
RF-Imp.	0.442	0.321	0.429	0.429	0.429	1.000	0.167	0.321	0.304	0.304
Mutual Inf.	0.172	0.167	0.186	0.186	0.186	0.167	1.000	0.167	0.304	0.304
RFE	0.442	1.000	0.556	0.556	0.556	0.321	0.167	1.000	0.304	0.304
RF	0.348	0.304	0.304	0.304	0.304	0.304	0.304	0.304	1.000	1.000
XGBoost	0.348	0.304	0.304	0.304	0.304	0.304	0.304	0.304	1.000	1.000

Table 15. Jaccard similarity on the y = x dataset.

	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF-Imp.	Mutual Inf.	RFE	RF	XGBoost
IVWFR	1.000	0.609	0.609	0.609	0.609	0.609	0.480	0.609	0.189	0.189
RFECV	0.609	1.000	0.647	0.750	0.647	0.647	0.647	1.000	0.115	0.115
ANOVA	0.609	0.647	1.000	0.750	0.867	0.750	0.750	0.647	0.115	0.115
Pearson	0.609	0.750	0.750	1.000	0.867	0.750	0.750	0.750	0.115	0.115
Spearman	0.609	0.647	0.867	0.867	1.000	0.750	0.750	0.647	0.115	0.115
RF-Imp.	0.609	0.647	0.750	0.750	0.750	1.000	0.750	0.647	0.115	0.115
Mutual Inf.	0.480	0.647	0.750	0.750	0.750	0.750	1.000	0.647	0.115	0.115
RFE	0.609	1.000	0.647	0.750	0.647	0.647	0.647	1.000	0.115	0.115
RF	0.189	0.115	0.115	0.115	0.115	0.115	0.115	0.115	1.000	1.000
XGBoost	0.189	0.115	0.115	0.115	0.115	0.115	0.115	0.115	1.000	1.000

Table 16. Jaccard similarity on the Yinyang dataset.

	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF-Imp.	Mutual Inf.	RFE	RF	XGBoost
IVWFR	1.000	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.177	0.177
RFECV	0.500	1.000	0.556	0.556	0.400	0.556	0.750	1.000	0.113	0.113
ANOVA	0.500	0.556	1.000	0.750	0.750	0.750	0.556	0.556	0.113	0.113
Pearson	0.500	0.556	0.750	1.000	0.750	1.000	0.750	0.556	0.113	0.113
Spearman	0.500	0.400	0.750	0.750	1.000	0.750	0.556	0.400	0.113	0.113
RF-Imp.	0.500	0.556	0.750	1.000	0.750	1.000	0.750	0.556	0.113	0.113
Mutual Inf.	0.500	0.750	0.556	0.750	0.556	0.750	1.000	0.750	0.113	0.113
RFE	0.500	1.000	0.556	0.556	0.400	0.556	0.750	1.000	0.113	0.113
RF	0.177	0.113	0.113	0.113	0.113	0.113	0.113	0.113	1.000	1.000
XGBoost	0.177	0.113	0.113	0.113	0.113	0.113	0.113	0.113	1.000	1.000

5.4.2. Overlap on Real Datasets

For the real-world datasets, the Jaccard similarity matrices (Tables 17–21) show that methods based on Random Forest feature importance (RFECV, RF-Imp., and RFE) often exhibit perfect or near-perfect similarity, as they rely on the same underlying feature ranking. Similarly, filter-based approaches such as ANOVA and Pearson correlation frequently produce comparable feature subsets. In contrast, the IVWFR model consistently demonstrates a more distinct selection pattern, underscoring its unique selection strategy. The lower agreement between the IVWFR and RFE/RFECV methods in feature selection results from a different selection philosophy—IVWFR focuses on stability and interval significance, thus selecting a smaller but more relevant set of features, which translates into classification efficiency.

Table 17. Jaccard similarity on the Climate dataset.

	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF-Imp.	Mutual Inf.	RFE	RF	XGBoost
IVWFR	1.000	0.800	0.800	0.800	0.800	0.800	0.800	0.800	0.278	0.278
RFECV	0.800	1.000	1.000	1.000	1.000	1.000	0.600	1.000	0.222	0.222
ANOVA	0.800	1.000	1.000	1.000	1.000	1.000	0.600	1.000	0.222	0.222
Pearson	0.800	1.000	1.000	1.000	1.000	1.000	0.600	1.000	0.222	0.222
Spearman	0.800	1.000	1.000	1.000	1.000	1.000	0.600	1.000	0.222	0.222
RF-Imp.	0.800	1.000	1.000	1.000	1.000	1.000	0.600	1.000	0.222	0.222
Mutual Inf.	0.800	0.600	0.600	0.600	0.600	0.600	1.000	0.600	0.222	0.222
RFE	0.800	1.000	1.000	1.000	1.000	1.000	0.600	1.000	0.222	0.222
RF	0.278	0.222	0.222	0.222	0.222	0.222	0.222	0.222	1.000	1.000
XGBoost	0.278	0.222	0.222	0.222	0.222	0.222	0.222	0.222	1.000	1.000

Table 18. Jaccard similarity on the Iranian Churn dataset.

	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF-Imp.	Mutual Inf.	RFE	RF	XGBoost
IVWFR	1.000	0.846	0.846	0.846	0.846	0.846	0.846	0.846	1.000	1.000
RFECV	0.846	1.000	0.692	0.692	0.692	1.000	0.833	1.000	0.846	0.846
ANOVA	0.846	0.692	1.000	1.000	0.833	0.692	0.833	0.692	0.846	0.846
Pearson	0.846	0.692	1.000	1.000	0.833	0.692	0.833	0.692	0.846	0.846
Spearman	0.846	0.692	0.833	0.833	1.000	0.692	0.692	0.692	0.846	0.846
RF-Imp.	0.846	1.000	0.692	0.692	0.692	1.000	0.833	1.000	0.846	0.846
Mutual Inf.	0.846	0.833	0.833	0.833	0.692	0.833	1.000	0.833	0.846	0.846
RFE	0.846	1.000	0.692	0.692	0.692	1.000	0.833	1.000	0.846	0.846
RF	1.000	0.846	0.846	0.846	0.846	0.846	0.846	0.846	1.000	1.000
XGBoost	1.000	0.846	0.846	0.846	0.846	0.846	0.846	0.846	1.000	1.000

Table 19. Jaccard similarity on the Sonar dataset.

	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF-Imp.	Mutual Inf.	RFE	RF	XGBoost
IVWFR	1.000	0.638	0.510	0.510	0.540	0.638	0.426	0.638	0.500	0.500
RFECV	0.638	1.000	0.741	0.741	0.741	0.774	0.621	1.000	0.783	0.783
ANOVA	0.510	0.741	1.000	1.000	0.918	0.709	0.679	0.741	0.783	0.783
Pearson	0.510	0.741	1.000	1.000	0.918	0.709	0.679	0.741	0.783	0.783
Spearman	0.540	0.741	0.918	0.918	1.000	0.741	0.649	0.741	0.783	0.783
RF-Imp.	0.638	0.774	0.709	0.709	0.741	1.000	0.649	0.774	0.783	0.783
Mutual Inf.	0.426	0.621	0.679	0.679	0.649	0.649	1.000	0.621	0.783	0.783
RFE	0.638	1.000	0.741	0.741	0.741	0.774	0.621	1.000	0.783	0.783
RF	0.500	0.783	0.783	0.783	0.783	0.783	0.783	0.783	1.000	1.000
XGBoost	0.500	0.783	0.783	0.783	0.783	0.783	0.783	0.783	1.000	1.000

Table 20. Jaccard similarity on the Water Potability dataset.

	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF-Imp.	Mutual Inf.	RFE	RF	XGBoost
IVWFR	1.000	0.800	0.500	0.500	0.500	0.800	0.500	0.800	0.556	0.556
RFECV	0.800	1.000	0.333	0.333	0.333	1.000	0.600	1.000	0.444	0.444
ANOVA	0.500	0.333	1.000	1.000	1.000	0.333	0.143	0.333	0.444	0.444
Pearson	0.500	0.333	1.000	1.000	1.000	0.333	0.143	0.333	0.444	0.444
Spearman	0.500	0.333	1.000	1.000	1.000	0.333	0.143	0.333	0.444	0.444
RF-Imp.	0.800	1.000	0.333	0.333	0.333	1.000	0.600	1.000	0.444	0.444
Mutual Inf.	0.500	0.600	0.143	0.143	0.143	0.600	1.000	0.600	0.444	0.444
RFE	0.800	1.000	0.333	0.333	0.333	1.000	0.600	1.000	0.444	0.444
RF	0.556	0.444	0.444	0.444	0.444	0.444	0.444	0.444	1.000	1.000
XGBoost	0.556	0.444	0.444	0.444	0.444	0.444	0.444	0.444	1.000	1.000

Table 21. Jaccard similarity on the Estimation of Obesity dataset.

	IVWFR	RFECV	ANOVA	Pearson	Spearman	RF-Imp.	Mutual Inf.	RFE	RF	XGBoost
IVWFR	1.000	0.400	0.167	0.167	0.167	0.400	0.400	0.400	0.312	0.312
RFECV	0.400	1.000	0.333	0.333	0.333	1.000	0.333	1.000	0.125	0.125
ANOVA	0.167	0.333	1.000	0.333	0.333	0.333	0.333	0.333	0.125	0.125
Pearson	0.167	0.333	0.333	1.000	1.000	0.333	0.333	0.333	0.125	0.125
Spearman	0.167	0.333	0.333	1.000	1.000	0.333	0.333	0.333	0.125	0.125
RF-Imp.	0.400	1.000	0.333	0.333	0.333	1.000	0.333	1.000	0.125	0.125
Mutual Inf.	0.400	0.333	0.333	0.333	0.333	0.333	1.000	0.333	0.125	0.125
RFE	0.400	1.000	0.333	0.333	0.333	1.000	0.333	1.000	0.125	0.125
RF	0.312	0.125	0.125	0.125	0.125	0.125	0.125	0.125	1.000	1.000
XGBoost	0.312	0.125	0.125	0.125	0.125	0.125	0.125	0.125	1.000	1.000

5.4.3. Feature Importance Stability Analysis

To evaluate the robustness and repeatability of the feature selection process, a stability analysis was performed across multiple data splits during 5-fold cross-validation. For each feature selection method, two complementary stability indices were computed to quantify the consistency of the selected feature subsets across different folds (see Figures A31 and A32 for synthetic datasets and Figures A33 and A34 for real-world datasets).

1. Jaccard Similarity Index (J):

$$J(A_i, A_j) = \frac{|A_i \cap A_j|}{|A_i \cup A_j|} \tag{12}$$

where A_i and A_j denote the sets of selected features obtained in two independent folds. The Jaccard index measures the proportion of common features between two selections and ranges from 0 (no overlap) to 1 (identical feature subsets).

2. Kuncheva Index (κ):

$$\kappa(A_i, A_j) = \frac{|A_i \cap A_j| \cdot N - d^2}{d \cdot (N - d)} \tag{13}$$

where N is the total number of available features and d is the number of features selected. The Kuncheva index corrects for random overlap between subsets, providing a more conservative estimate of stability.

Both indices were calculated for all pairs of folds and then averaged to obtain the overall stability of each feature selection method for synthetic (Figure 5) and real-world (Figure 6) datasets. Higher values of J and κ indicate greater robustness of the feature selection process to data perturbations and random partitioning.

Among the real-world datasets used, one set, Water Potability, contains approximately 4.86% missing data, which affects three of the nine features. Analysis of this issue (Figure A37) indicates that a 5% missing value rate for this database leads to a loss in classification quality from approximately 61.9% to only 61.7%, which confirms its applicability. We use the following strategy for imputing missing values: each missing feature value is replaced with the mean value of that feature. This strategy is applied before feature scaling and selection and is consistent across all feature selection methods. The summary results of the simulation analysis of missing values in the range of 5–20% for all sets are presented in Figure A37, while detailed values are presented in Tables A1–A5.

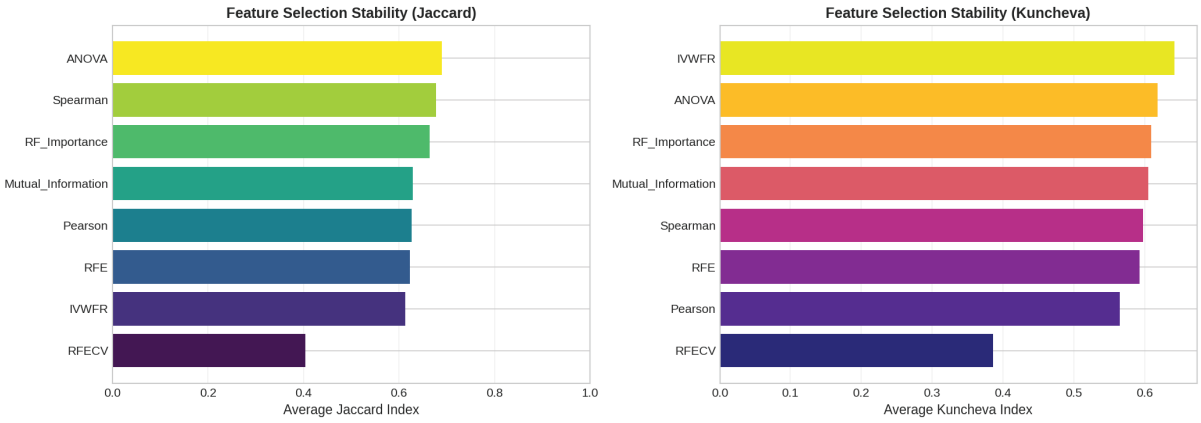


Figure 5. Average stability across synthetic datasets.

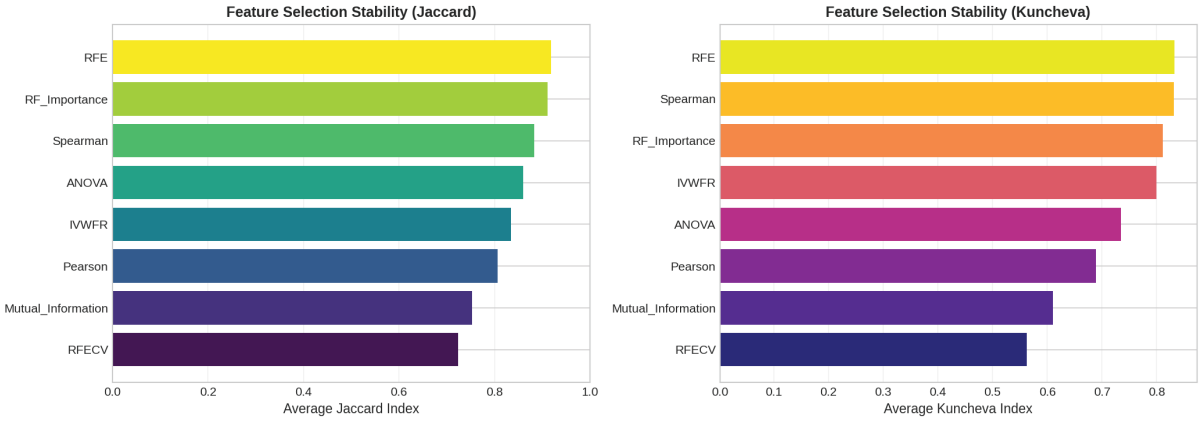


Figure 6. Average stability across real-world datasets.

6. Discussion

The research conclusions drawn from the results obtained are very optimistic. Analysis of synthetic data (Figures A1–A10) shows that the IVWFR algorithm achieves the best classification results among the selected methods, or ranks among the top results (e.g., Cone or Trig dataset). The summary classification chart (Figure 1) proves its effectiveness, where the results obtained by the IVWFR algorithm are significantly better than those of the other methods. Significant differences in the results achieved are noticeable, for example, for the 10Class Multicut dataset, for which the algorithm achieved the best classification results and, at the same time, identified 39 significant features compared to 11 features identified by the RFECV and RFE algorithms (see Table 5), which may indicate its sensitivity to less significant but important features that improve classification. A similarly sensitive result was obtained for the synthetic Double Spiral dataset, where the classification results were much better than those obtained using other methods (Figure A6). At the same time, the IVWFR algorithm identified only 152 significant features, while other methods identified between 227 and 303 features (Table 5). This may indicate the sensitivity of the algorithm in terms of removing irrelevant information noise from the dataset.

The results obtained for five real-world datasets are also very promising. The IVWFR algorithm achieved the best results or was among the best methods (Figures A11–A15). The summary results (Figure 2) also show that the IVWFR algorithm achieved the best classification results. At the same time, for example, for the Sonar dataset, it identified 30 significant features, while the other methods identified between 47 and 60 (Table 6) out of all 60 features in the dataset (Table 1), and the classification results were clearly better (Figure A14). Moreover, the p-values obtained for the IVWFR algorithm results also confirm its effectiveness for both synthetic data (Table 3) and real data (Table 4). The IVWFR method achieved comparable accuracy with significantly fewer features, confirming its effectiveness and ability to reduce complexity.

The results in Figures 5 and 6 clearly demonstrate that for the synthetic dataset the proposed IVWFR method achieves high feature selection stability among all compared approaches. The average Jaccard index reaches 0.61 and Kuncheva index reaches 0.64 (the best result) for synthetic datasets and 0.83 (Jaccard) and 0.80 (Kuncheva) for real-world datasets, indicating that IVWFR consistently selects a subset of most of the same features across different cross-validation folds.

However, simple statistical filters (ANOVA, Pearson, Spearman correlation) exhibit similar repeatability, suggesting the same sensitivity to data perturbations. In contrast, wrapper-based methods achieve moderate (RFE) or the worst (RFECV) stability but at the cost of considerably higher computational complexity (Figures A35 and A36).

The appropriate stability of IVWFR, both for synthetic and real-world data, can be attributed to its interval-valued aggregation mechanism, which smooths out the fluctuations in feature importance values obtained from multiple data partitions. By incorporating uncertainty intervals into the ranking process, the method effectively suppresses random noise and yields a more reproducible and interpretable feature set.

Future research directions will focus on developing the proposed approach, i.e., the use of innovative, adaptive methods for aggregating significance intervals; developing new weighting methods, nonlinear methods that take into account the influence of individual intervals; developing new methods for scoring significance, incorporating interval stability; and using optimization to determine feature contributions.

Author Contributions: Conceptualization, W.P. and U.B.; methodology, A.W. and U.B.; software, A.W.; validation, A.W. and W.P.; formal analysis, U.B.; investigation, A.W.; resources, A.W. and W.P.; data curation, A.W. and W.P.; writing—original draft preparation, A.W., W.P. and U.B.; writing—review and editing, U.B. and W.P.; visualization, A.W.; supervision, W.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data available in a publicly accessible repository.

Acknowledgments: This work was partially supported by the Centre for Innovation and Transfer of Natural Sciences and Engineering Knowledge of University of Rzeszów, Poland.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

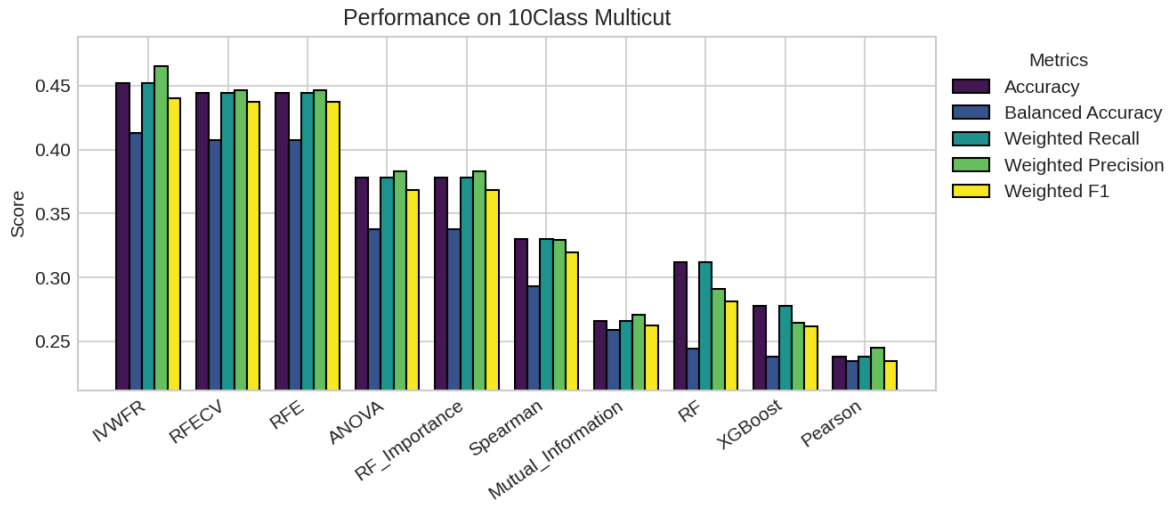


Figure A1. Performance on 10Class Multicut dataset.

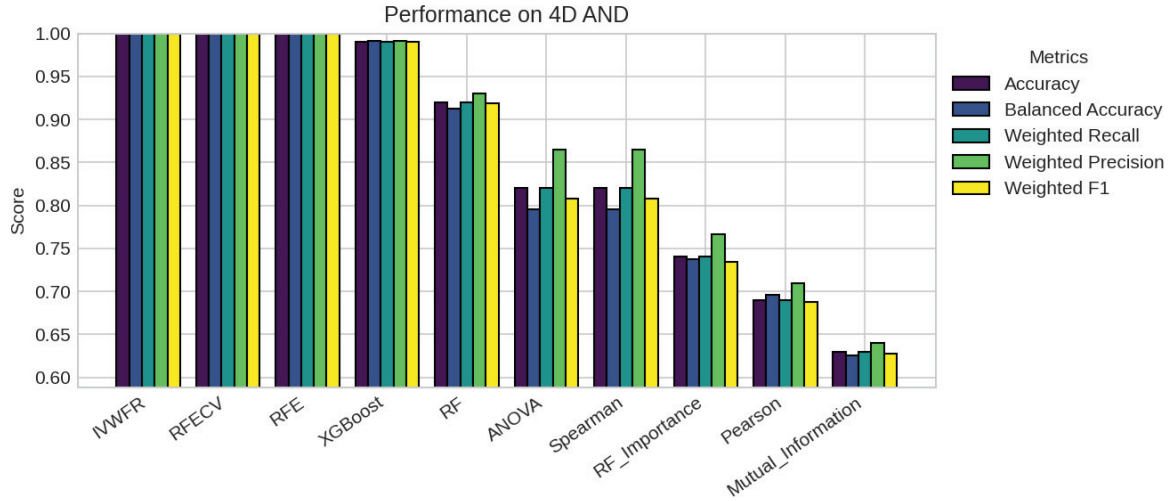


Figure A2. Performance on 4D AND dataset.

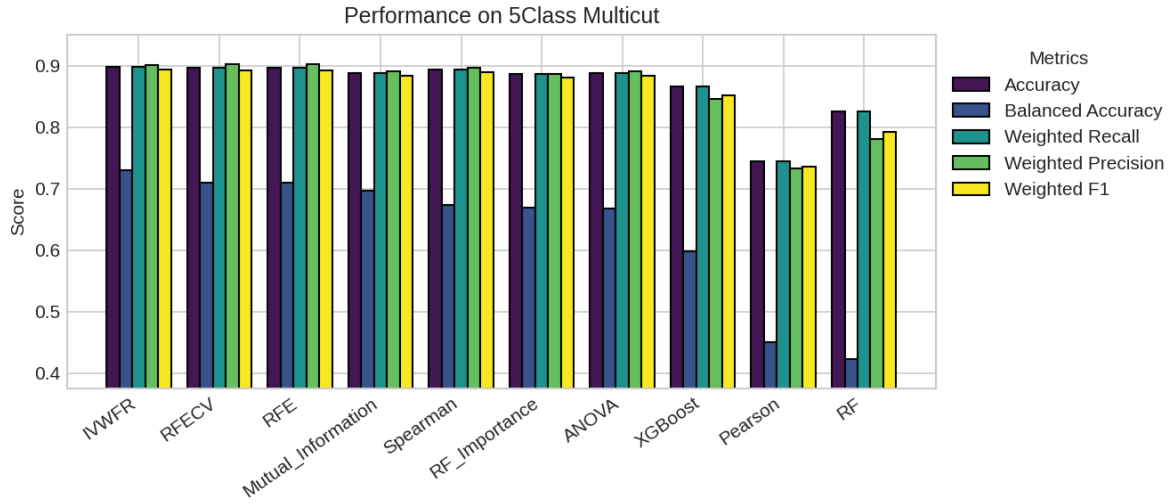


Figure A3. Performance on 5Class Multicut dataset.

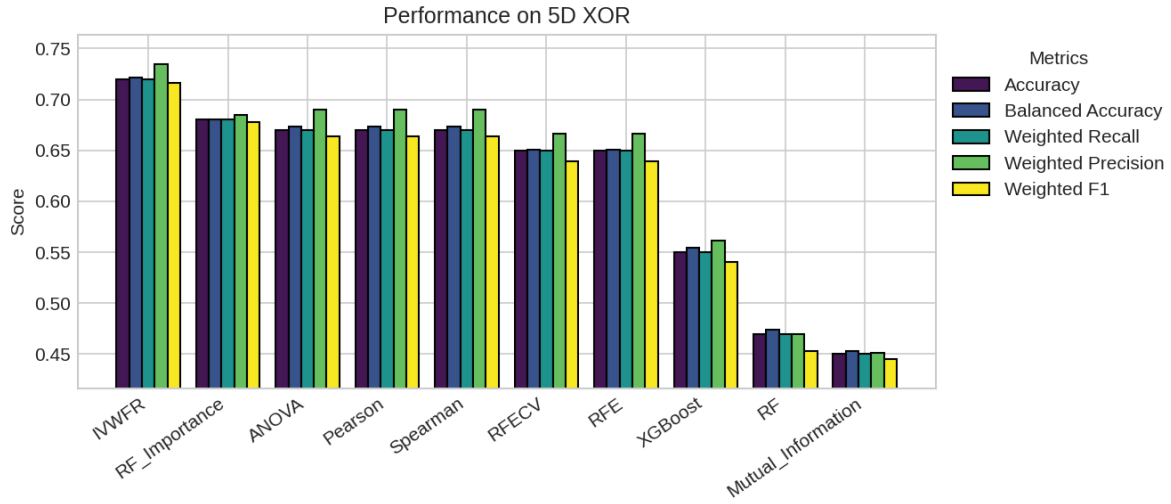


Figure A4. Performance on 5D XOR dataset.

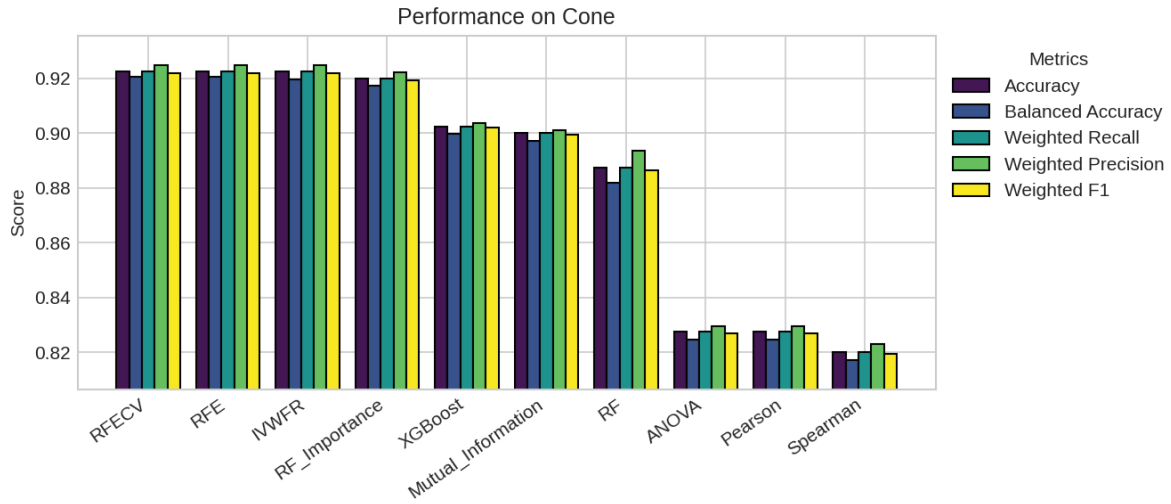


Figure A5. Performance on Cone dataset.

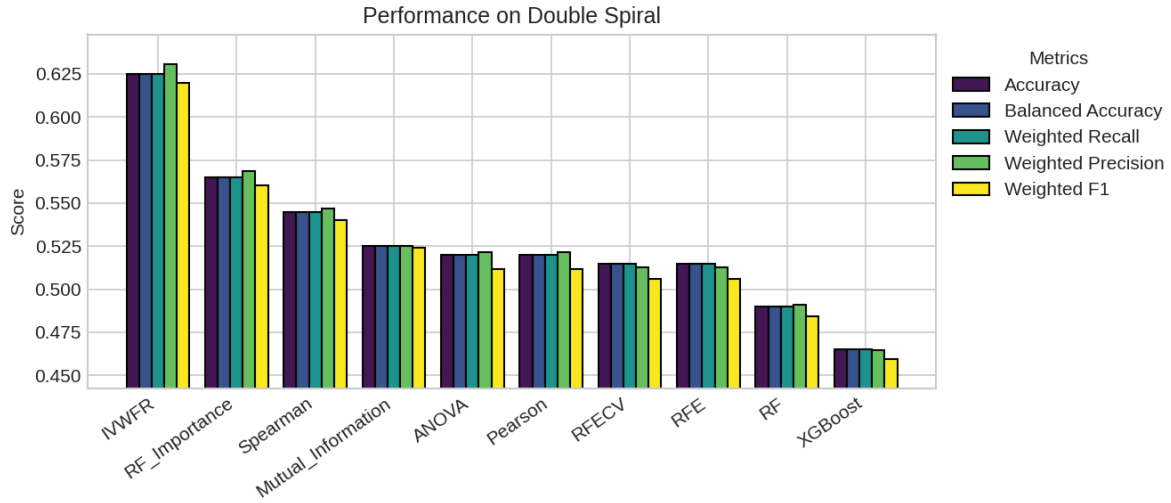


Figure A6. Performance on Double Spiral dataset.

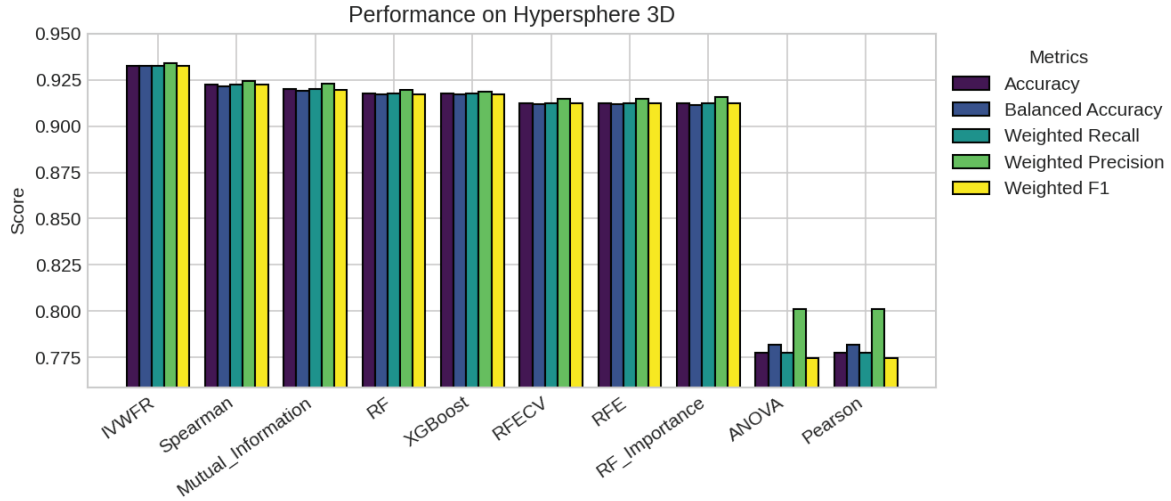


Figure A7. Performance on Hypersphere 3D dataset.

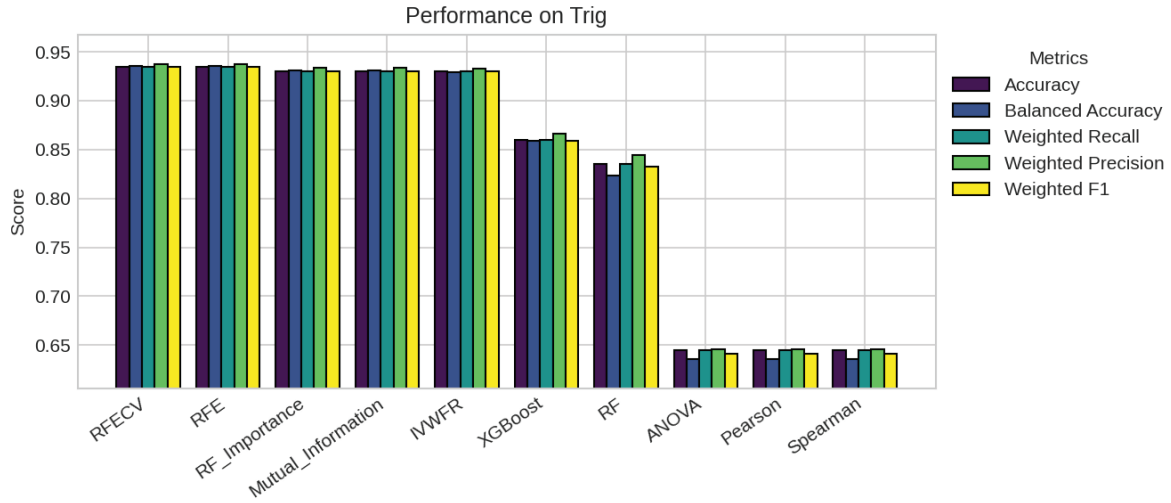


Figure A8. Performance on Trig dataset.

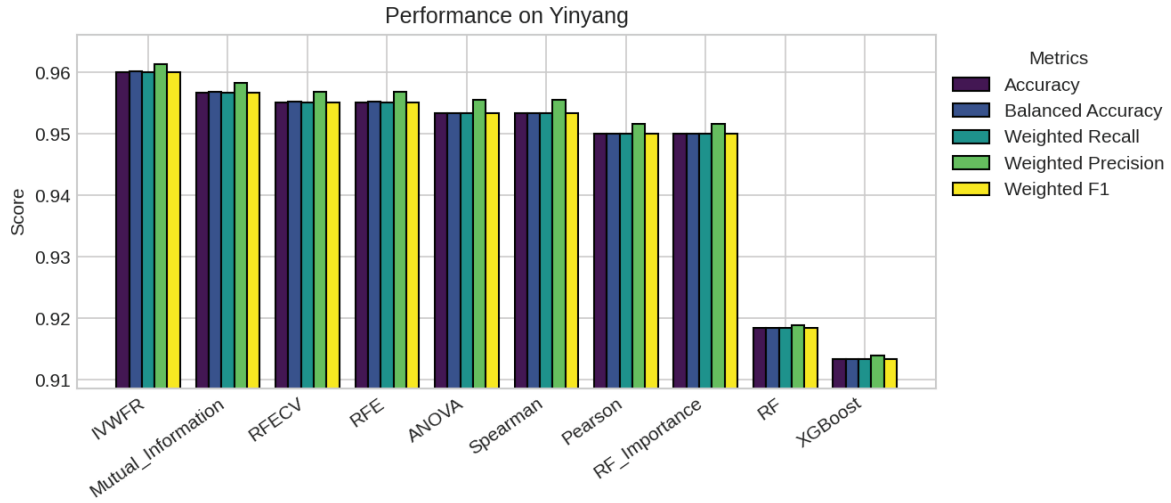


Figure A9. Performance on Yinyang dataset.

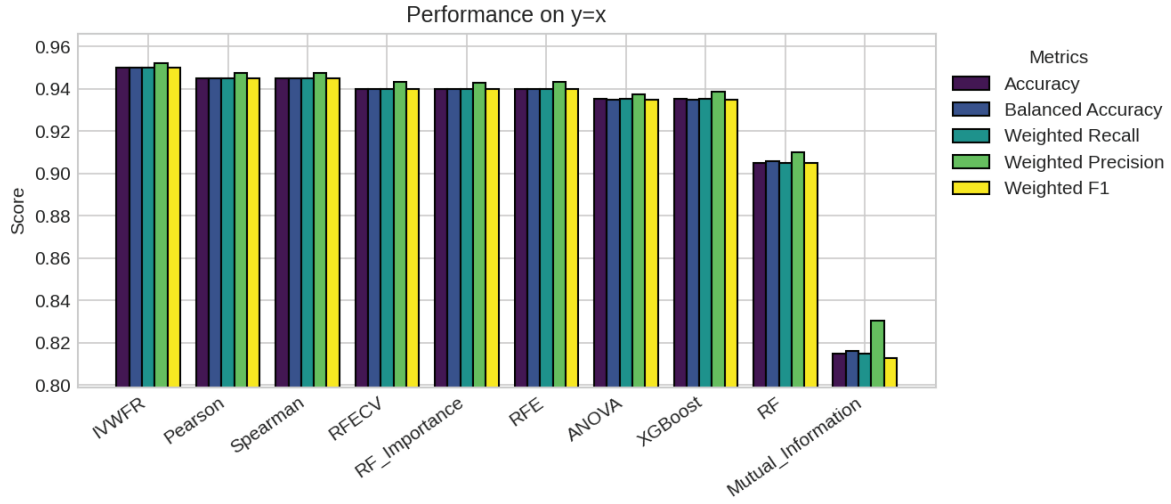


Figure A10. Performance on y = x dataset.

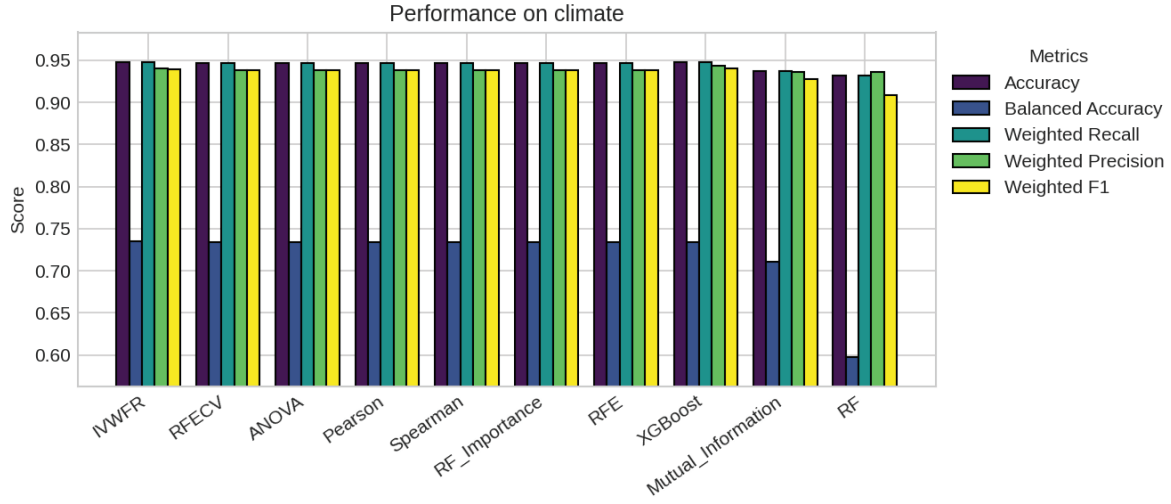


Figure A11. Performance on Climate dataset.

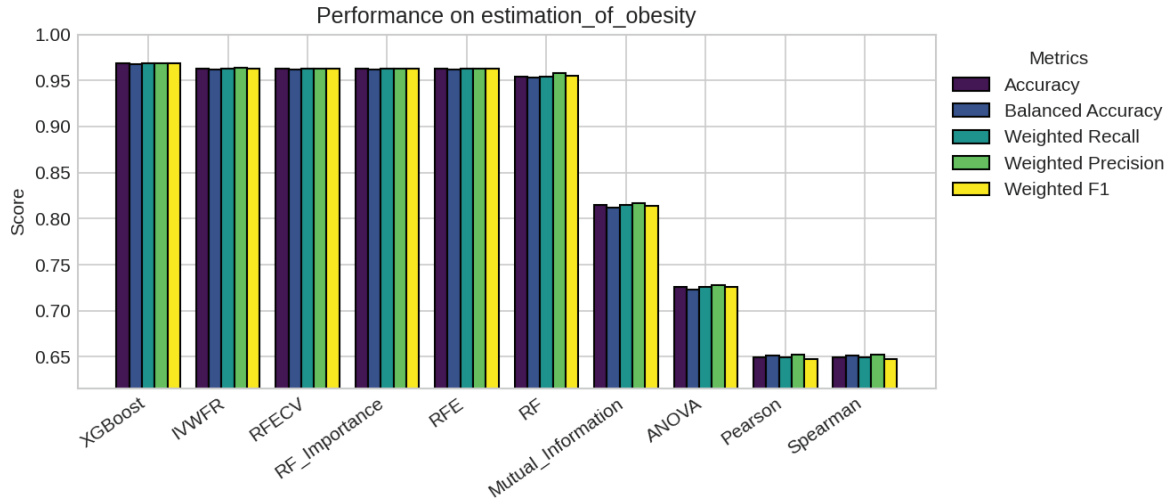


Figure A12. Performance on Estimation of Obesity Levels dataset.

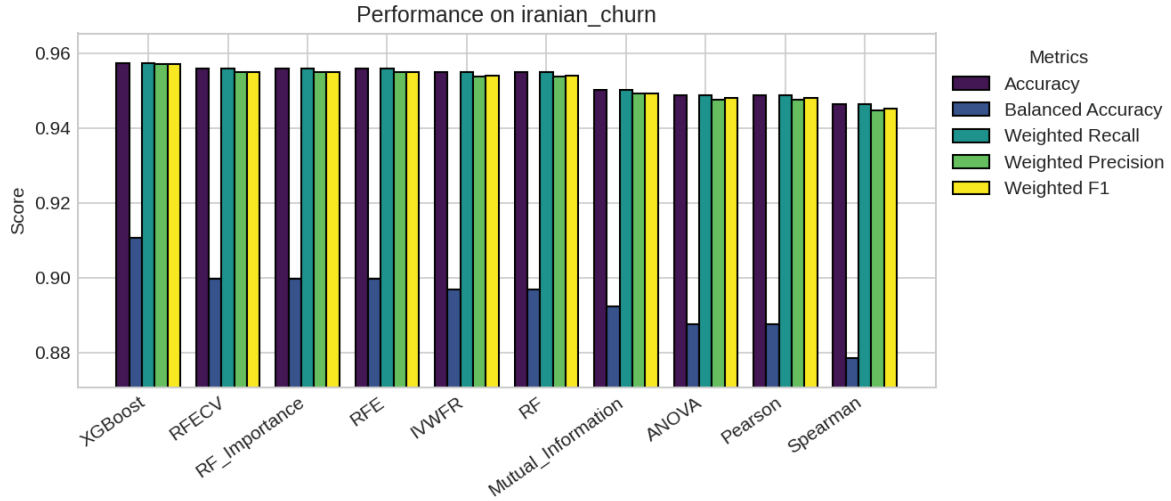


Figure A13. Performance on Iranian Churn dataset.

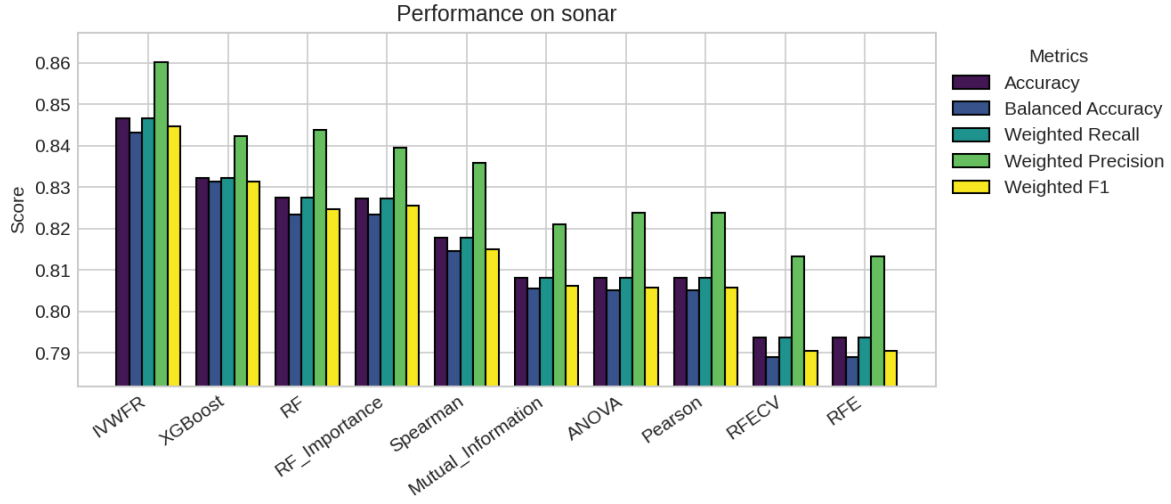


Figure A14. Performance on Sonar dataset.

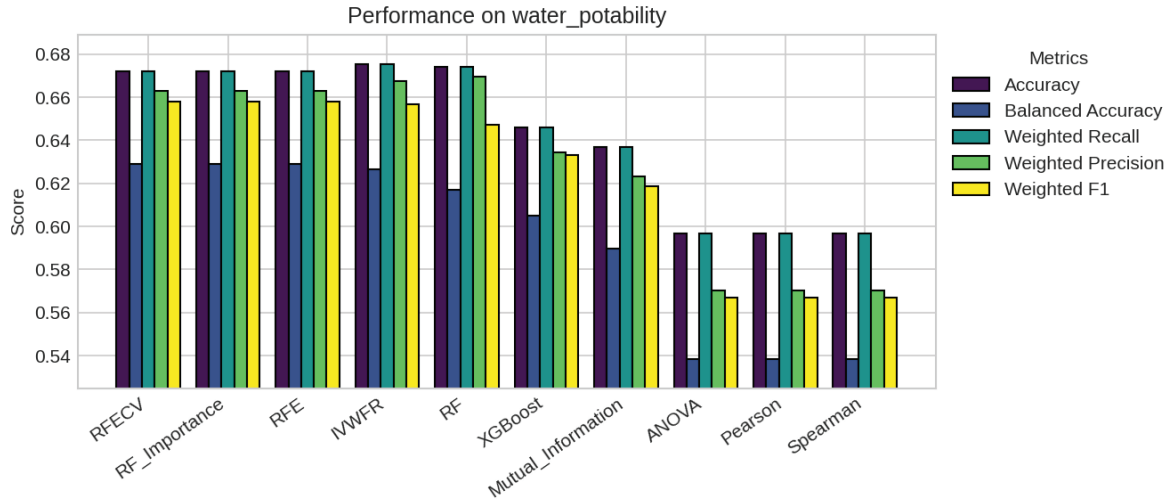


Figure A15. Performance on Water Potability dataset.

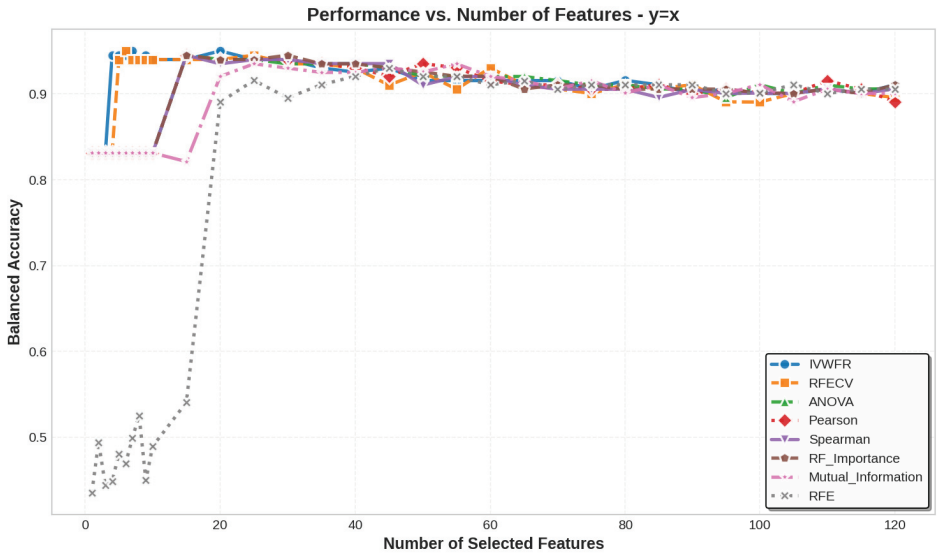


Figure A16. Performance compared to number of features using y = x dataset.

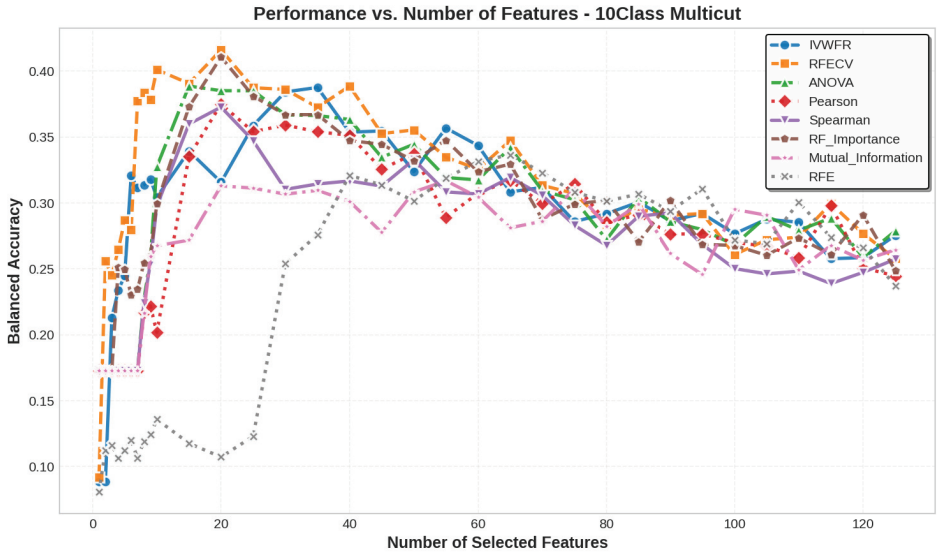


Figure A17. Performance compared to number of features using 10Class Multicut dataset.

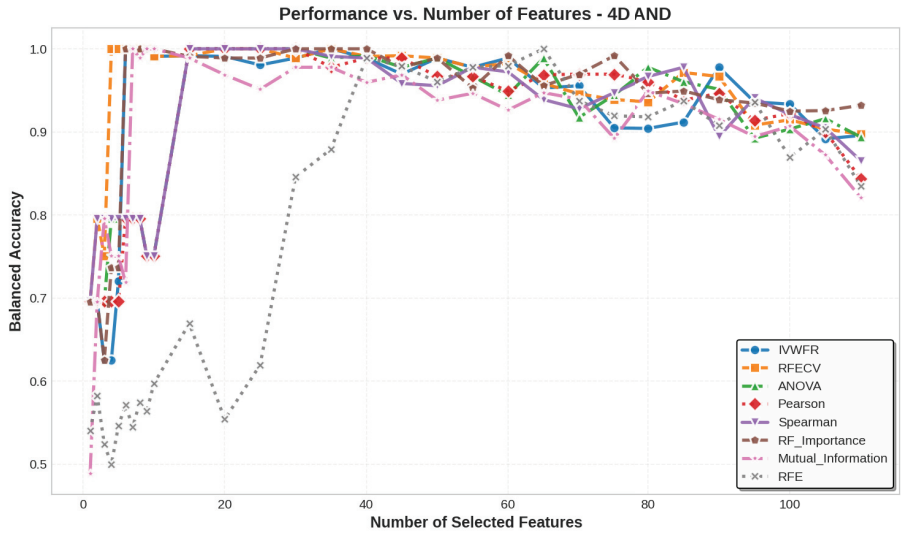


Figure A18. Performance compared to number of features using features_4D AND dataset.

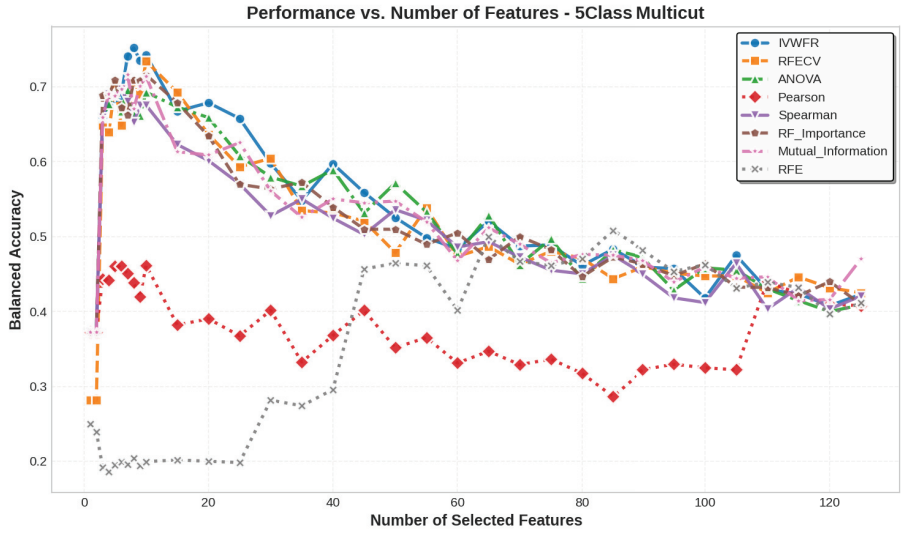


Figure A19. Performance compared to number of features using 5Class Multicut dataset.

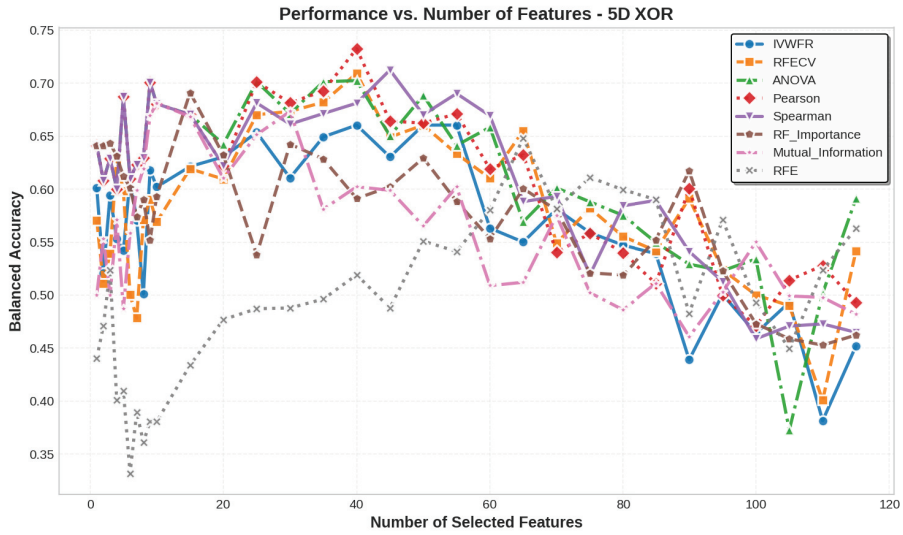


Figure A20. Performance compared to number of features using 5D XOR dataset.

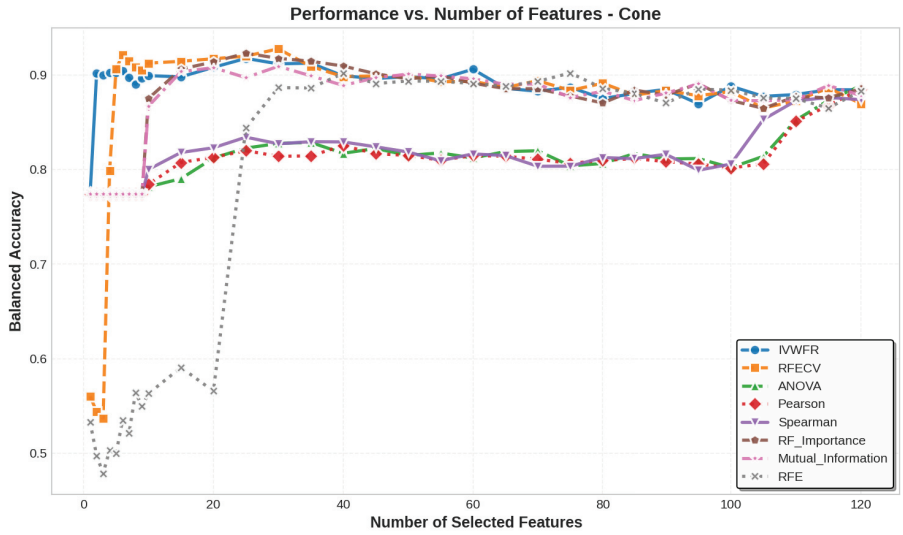


Figure A21. Performance compared to number of features using Cone dataset.

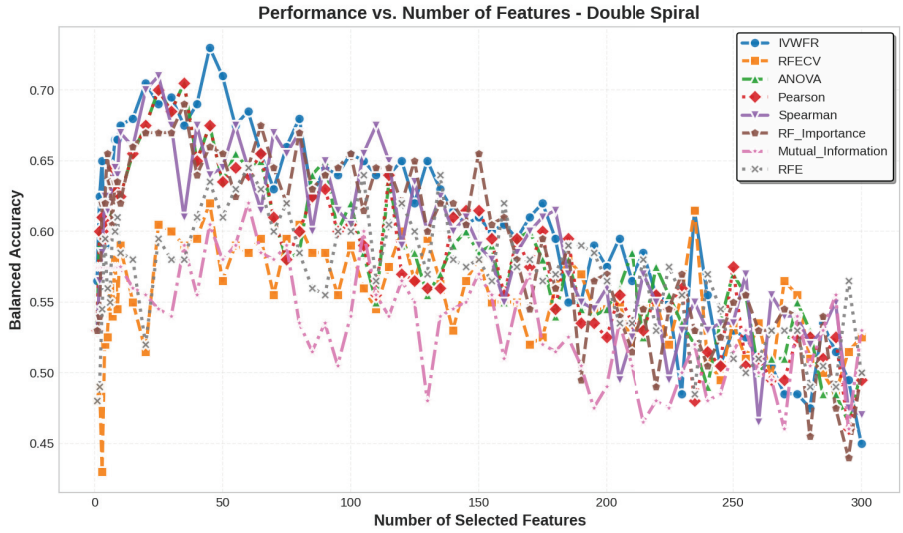


Figure A22. Performance compared to number of features using Double Spiral dataset.

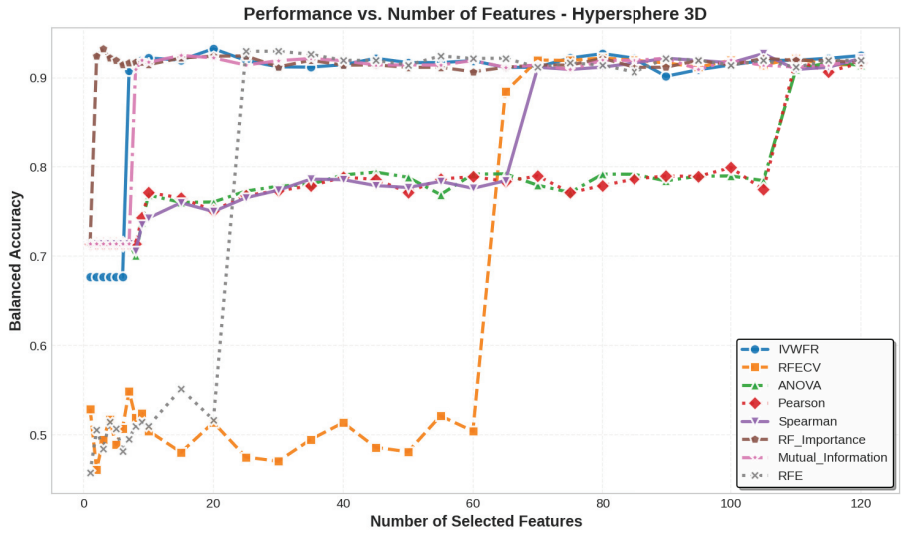


Figure A23. Performance compared to number of features using Hypersphere 3D dataset.

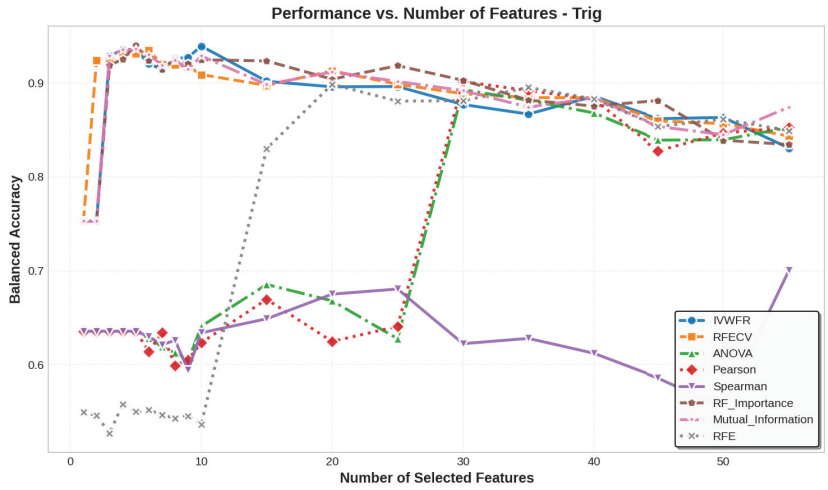


Figure A24. Performance compared to number of features using Trig dataset.

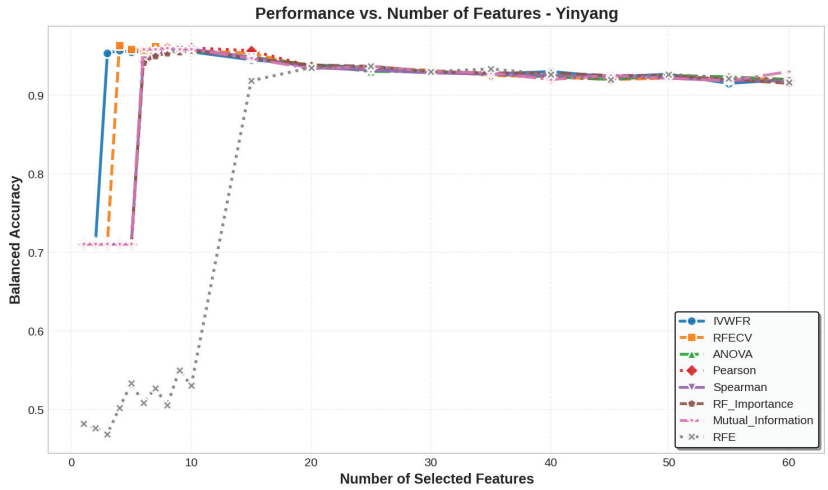


Figure A25. Performance compared to number of features using Yinyang dataset. The results for the Pearson, Spearman, RF_Importance and Mutual_Information methods and for the IVWFR and ANOVA methods are almost identical and overlap.

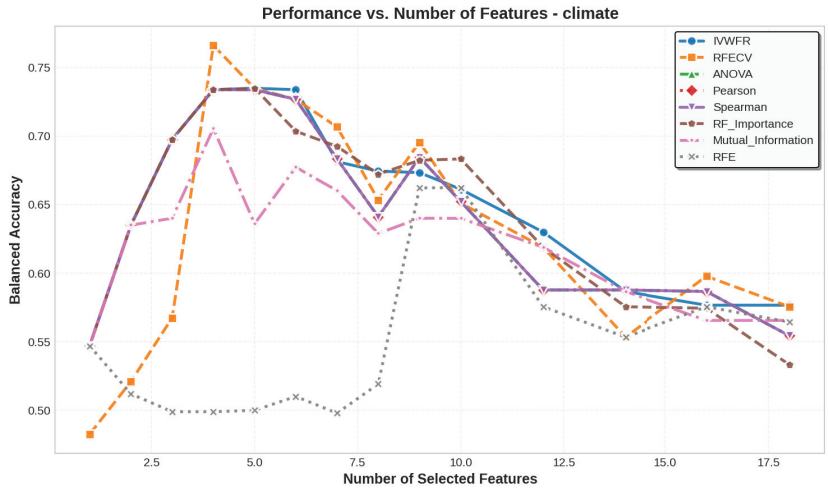


Figure A26. Performance compared to number of features using Climate dataset. The results for the ANOVA and IVWFR methods and for the Pearson and Spearman correlation methods are identical and overlap.

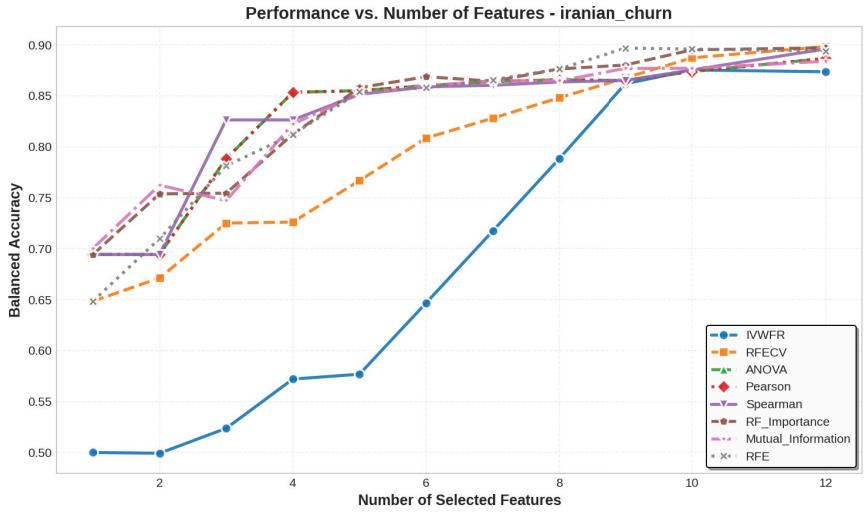


Figure A27. Performance compared to number of features using Iranian Churn dataset. The results for the ANOVA and Pearson correlation methods are identical and overlap.

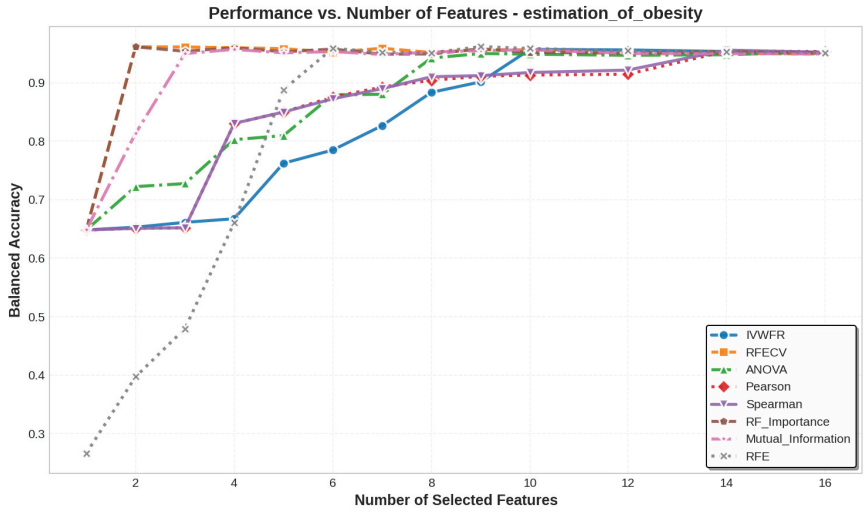


Figure A28. Performance compared to number of features using Estimation of Obesity dataset.

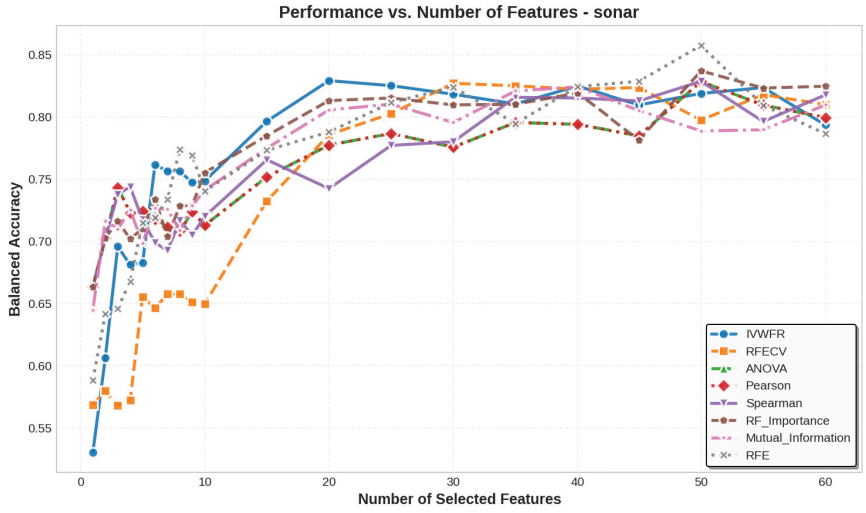


Figure A29. Performance compared to number of features using Sonar dataset. The results for the ANOVA and Pearson correlation methods are identical and overlap.

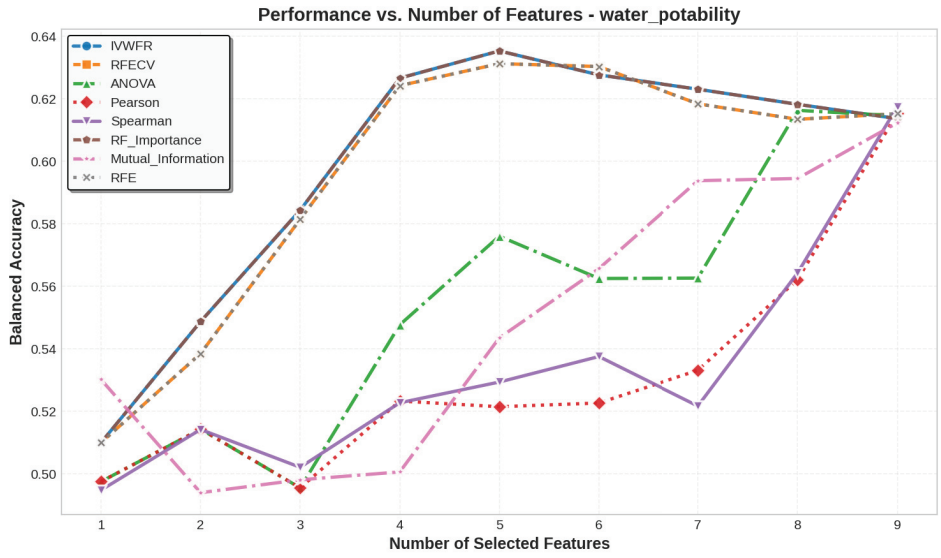


Figure A30. Performance compared to number of features using Water potability dataset. The results for the RF_Importance and IVWFR methods and for the RFE and RFECV methods are identical and overlap.

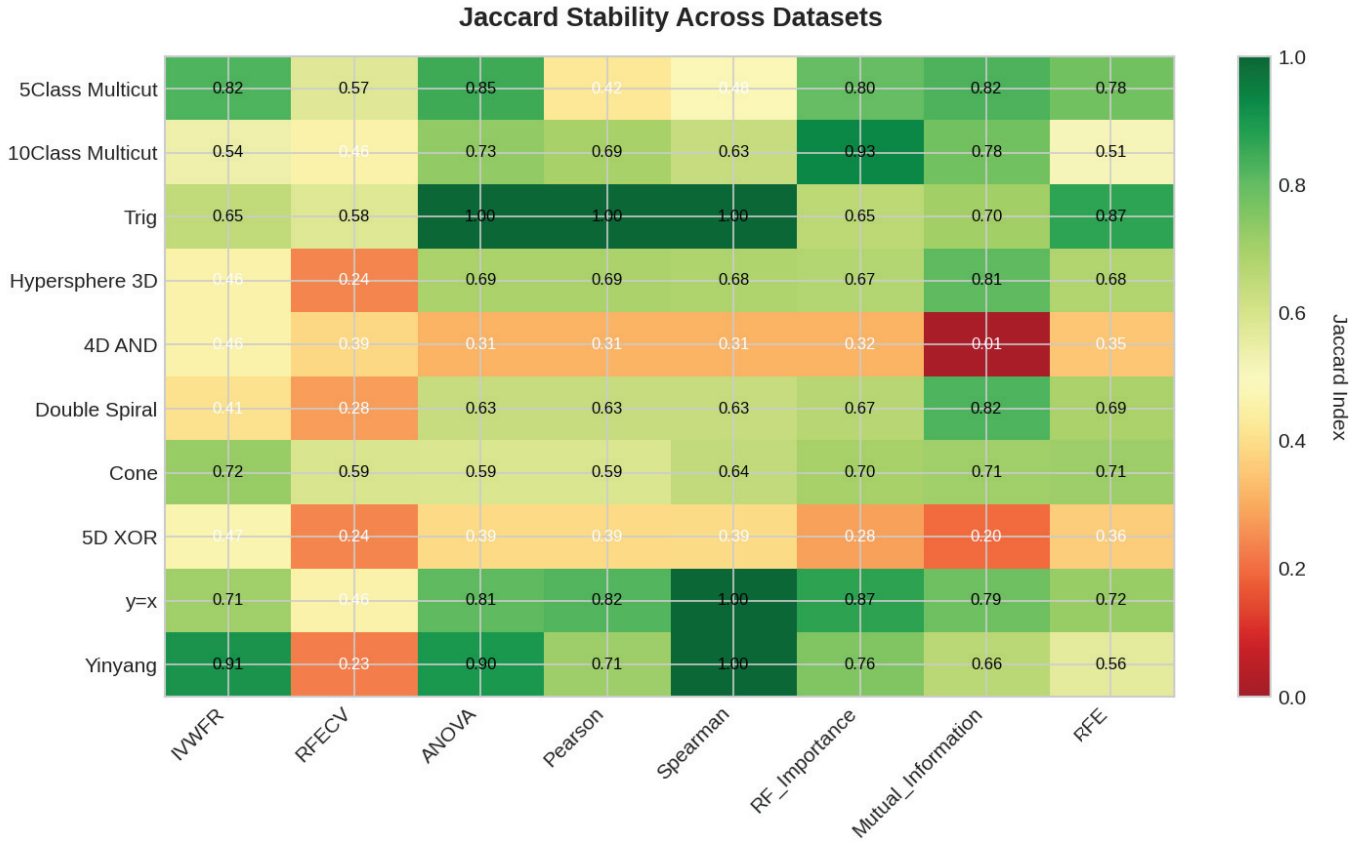


Figure A31. Jaccard stability across synthetic datasets.

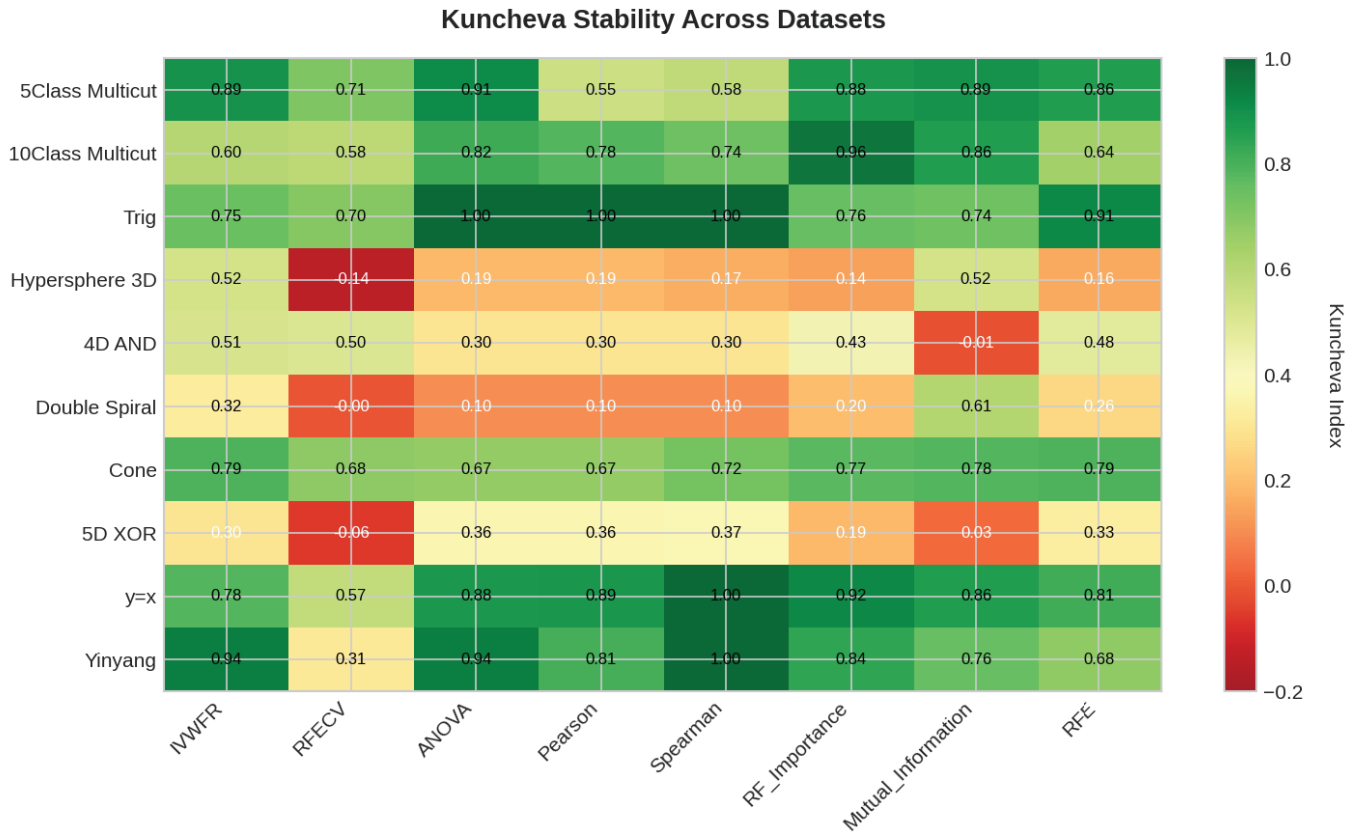


Figure A32. Kuncheva stability across synthetic datasets.

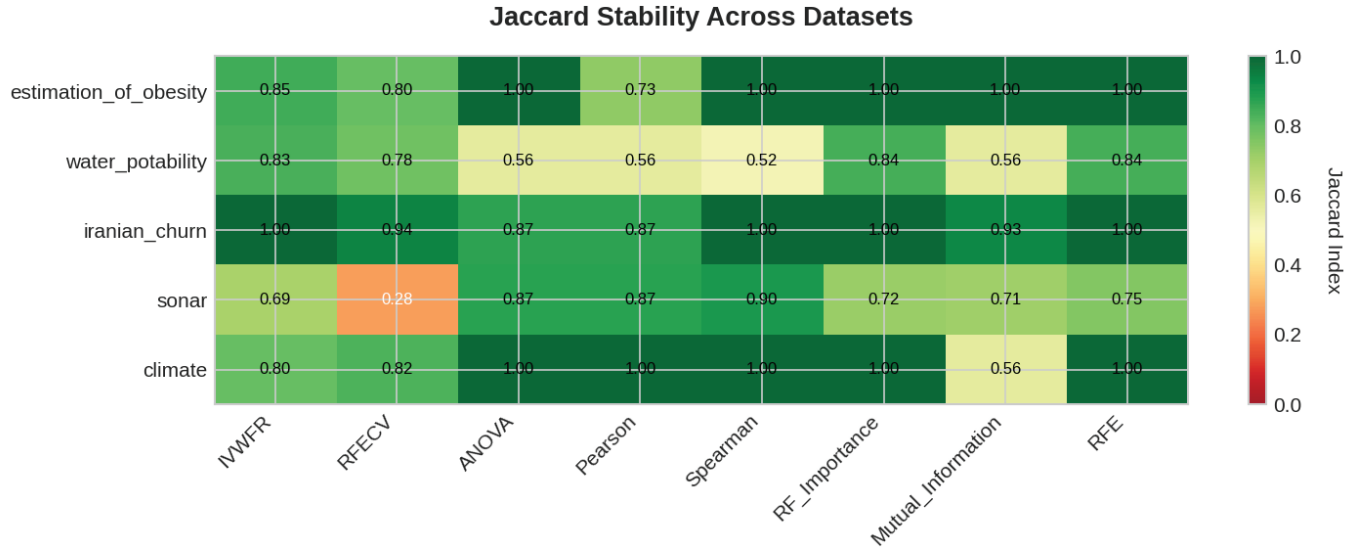


Figure A33. Jaccard stability across real-world datasets.

Table A1. Results of performance with artificial missing data for the Estimation of Obesity dataset.

Missing %	Mean Acc	Std Dev	Degradation	Avg Features
5%	0.9178	0.0049	0.0376 (3.9%)	8.0
10%	0.8912	0.0032	0.0643 (6.7%)	8.0
15%	0.8635	0.0032	0.0920 (9.6%)	8.0
20%	0.8352	0.0070	0.1203 (12.6%)	8.0

Table A2. Results of performance with artificial missing data for the Water Potability dataset.

Missing %	Mean Acc	Std Dev	Degradation	Avg Features
5%	0.6172	0.0055	0.0021 (0.3%)	5.0
10%	0.6113	0.0084	0.0080 (1.3%)	5.0
15%	0.5923	0.0087	0.0271 (4.4%)	5.0
20%	0.5948	0.0052	0.0246 (4.0%)	5.0

Table A3. Results of performance with artificial missing data for the Iranian Churn dataset.

Missing %	Mean Acc	Std Dev	Degradation	Avg Features
5%	0.8377	0.0156	0.0411 (4.7%)	7.0
10%	0.7991	0.0171	0.0797 (9.1%)	7.0
15%	0.7858	0.0046	0.0930 (10.6%)	7.0
20%	0.7724	0.0166	0.1064 (12.1%)	7.0

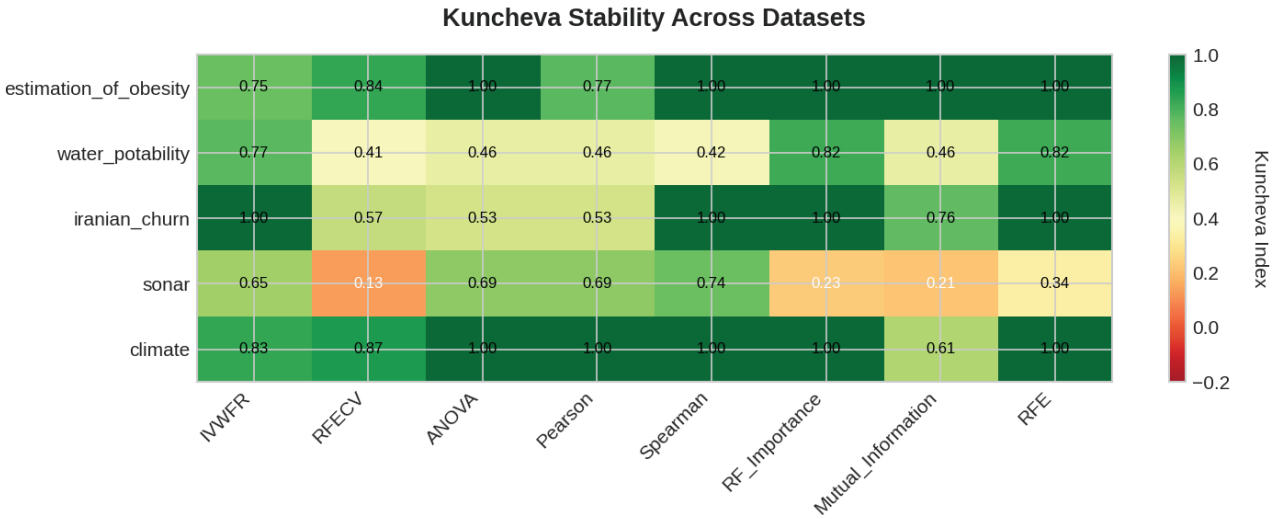


Figure A34. Kuncheva stability across real-world datasets.

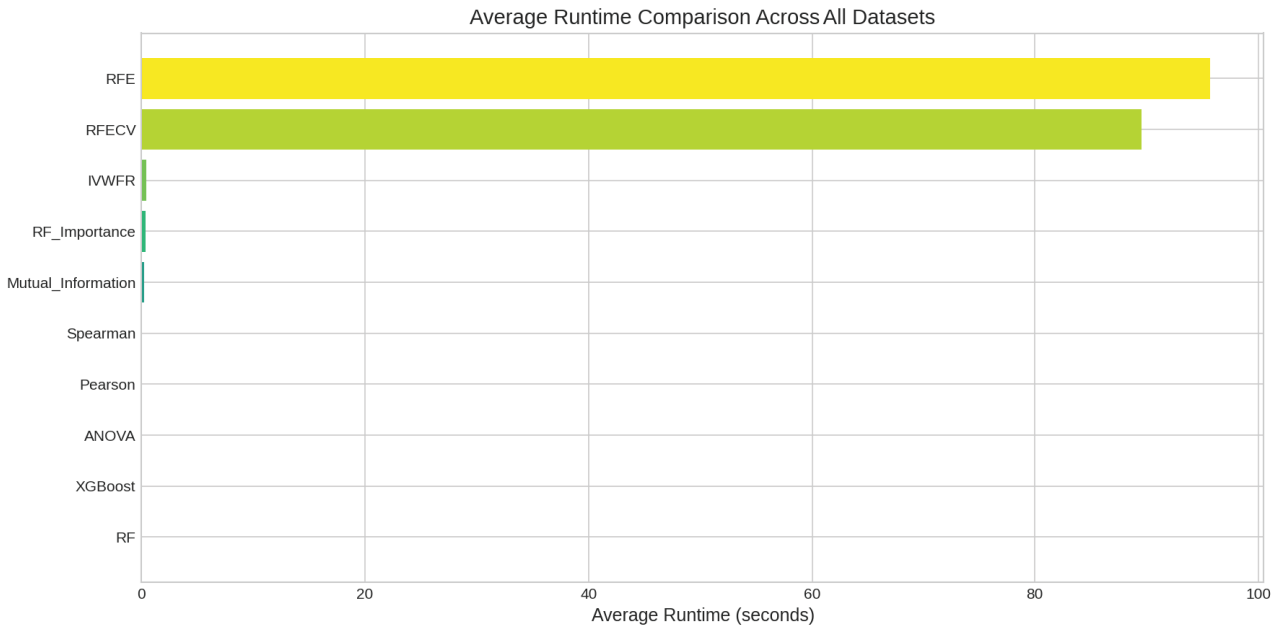


Figure A35. Average runtime across synthetic datasets.

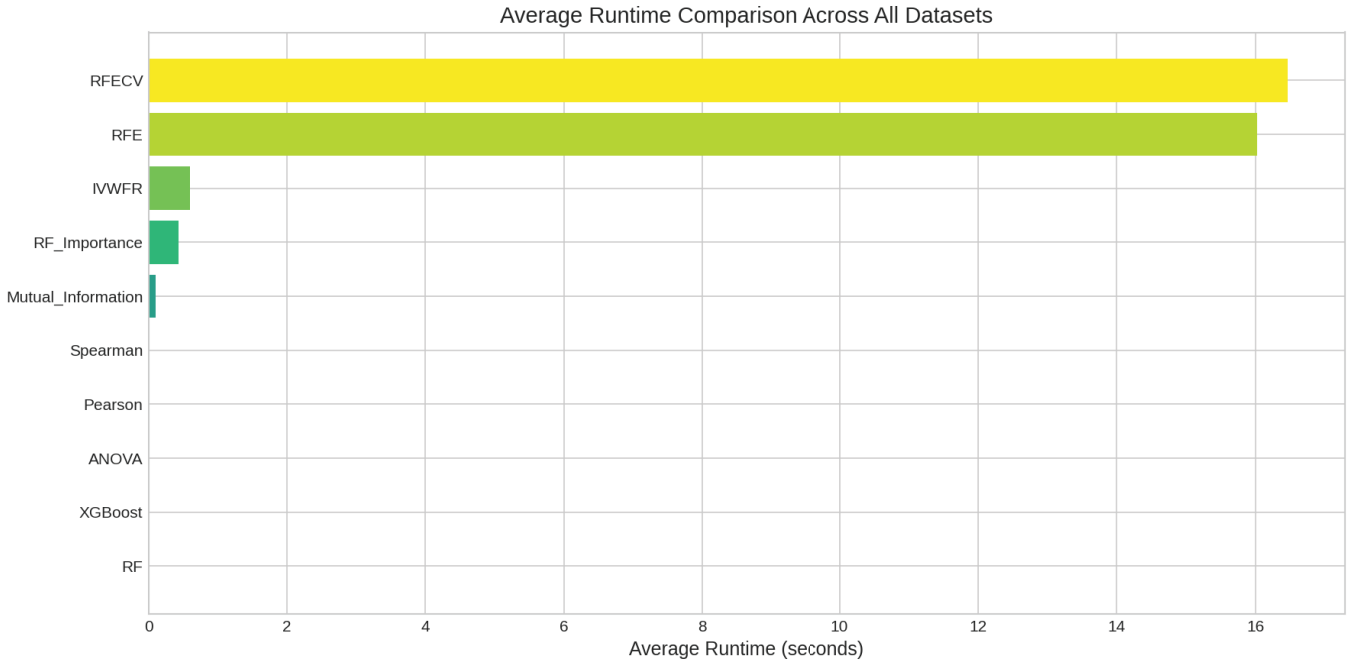


Figure A36. Average runtime across real-world datasets.

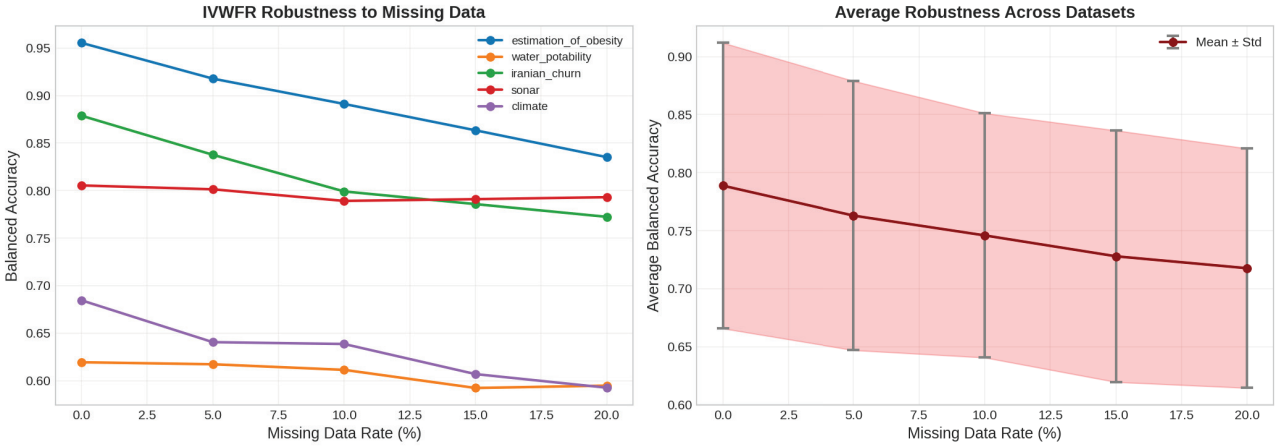


Figure A37. IVWFR algorithm robustness to missing data.

Table A4. Results of performance with artificial missing data for the Sonar dataset.

Missing %	Mean Acc	Std Dev	Degradation	Avg Features
5%	0.8014	0.0123	0.0041 (0.5%)	30.0
10%	0.7891	0.0207	0.0163 (2.0%)	30.0
15%	0.7910	0.0190	0.0144 (1.8%)	30.0
20%	0.7931	0.0219	0.0124 (1.5%)	30.0

Table A5. Results of performance with artificial missing data for the Climate dataset.

Missing %	Mean Acc	Std Dev	Degradation	Avg Features
5%	0.6405	0.0181	0.0440 (6.4%)	9.0
10%	0.6386	0.0365	0.0460 (6.7%)	9.0
15%	0.6069	0.0298	0.0777 (11.3%)	9.0
20%	0.5926	0.0232	0.0920 (13.4%)	9.0

References

1. Dash, M.; Liu, H. Feature selection for classification. *Intell. Data Anal.* **1997**, *1*, 131–156. [CrossRef]
2. Guyon, I.; Elisseeff, A. An introduction to variable and feature selection. *J. Mach. Learn. Res.* **2003**, *3*, 1157–1182.
3. Saeys, Y.; Inza, I.; Larrañaga, P. A review of feature selection techniques in bioinformatics. *Bioinformatics* **2007**, *23*, 2507–2517. [CrossRef] [PubMed]
4. Chandrashekar, G.; Sahin, F. A survey on feature selection methods. *Comput. Electr. Eng.* **2014**, *40*, 16–28. [CrossRef]
5. Kuncheva, L.I. A stability index for feature selection. In Proceedings of the 25th IASTED International Multi-Conference: Artificial Intelligence and Applications, Innsbruck, Austria, 12–14 February 2007; pp. 390–395.
6. Haury, A.C.; Gestraud, P.; Vert, J.P. The influence of feature selection methods on accuracy, stability and interpretability of molecular signatures. *PLoS ONE* **2011**, *6*, e28210. [CrossRef] [PubMed]
7. Nogueira, S.; Sechidis, K.; Brown, G. On the stability of feature selection algorithms. *J. Mach. Learn. Res.* **2018**, *18*, 1–54.
8. Cawley, G.C.; Talbot, N.L. On over-fitting in model selection and subsequent selection bias in performance evaluation. *J. Mach. Learn. Res.* **2010**, *11*, 2079–2107.
9. Ahmad, J.; Latif, S.; Khan, I.U.; Alshehri, M.S.; Khan, M.S.; Alasbali, N.; Jiang, W. An interpretable deep learning framework for intrusion detection in industrial Internet of Things. *Internet Things* **2025**, *33*, 101681. [CrossRef]
10. Strobl, C.; Boulesteix, A.L.; Kneib, T.; Augustin, T.; Zeileis, A. Conditional variable importance for random forests. *BMC Bioinform.* **2008**, *9*, 307. [CrossRef] [PubMed]
11. Meinshausen, N.; Bühlmann, P. Stability selection. *J. R. Stat. Soc. Ser. B (Stat. Methodol.)* **2010**, *72*, 417–473. [CrossRef]
12. Kalousis, A.; Prados, J.; Hilario, M. Stability of feature selection algorithms: A study on high-dimensional spaces. *Knowl. Inf. Syst.* **2007**, *12*, 95–116. [CrossRef]
13. Wojtowicz, A.; Bentkowska, U.; Paja, W. The interval-valued weighted feature ranking method based on aggregation functions. In Proceedings of the 2025 IEEE International Conference on Fuzzy Systems (FUZZ), Reims, France, 6–10 July 2025; pp. 1–7. [CrossRef]
14. Zadeh, L.A. The concept of a linguistic variable and its application to approximate reasoning—I. *Inf. Sci.* **1975**, *8*, 199–249. [CrossRef]
15. Dubois, D.; Prade, H. Internal-valued fuzzy sets, possibility theory and imprecise probability. In Proceedings of the 4th Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT 2005) and 11èmes Rencontres Francophones sur la Logique Floue et ses Applications (LFA 2005), Barcelona, Spain, 7–9 September 2005; pp. 314–319.
16. Komorníková, M.; Mesiar, R. Aggregation functions on bounded partially ordered sets and their classification. *Fuzzy Sets Syst.* **2011**, *175*, 48–56. [CrossRef]
17. Bustince, H.; Fernandez, J.; Kolesárová, A.; Mesiar, R. Generation of linear orders for intervals by means of aggregation functions. *Fuzzy Sets Syst.* **2013**, *220*, 69–77. [CrossRef]
18. Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. Optuna: A Next-generation Hyperparameter Optimization Framework. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 4–8 August 2019; pp. 2623–2631. [CrossRef]
19. Reference Page for the Accuracy Score Implementation in Scikit-Learn. Available online: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html#sklearn.metrics.accuracy_score (accessed on 13 June 2023).
20. Reference Page for the Recall Score Implementation in Scikit-Learn. Available online: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.recall_score.html (accessed on 13 June 2023).
21. Reference Page for the Precision Score Implementation in Scikit-Learn. Available online: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.precision_score.html (accessed on 13 June 2023).
22. Reference Page for the F1 Score Implementation in Scikit-Learn. Available online: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html (accessed on 13 June 2023).
23. Mitra, R.; Ali, E.; Varam, D.; Sulieman, H.; Kamalov, F. Variable Selection in Data Analysis: A Synthetic Data Toolkit. *Mathematics* **2024**, *12*, 570. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

applied sciences

MDPI

Article

Credit Evaluation Through Integration of Supervised and Unsupervised Machine Learning: Empirical Improvement and Unsupervised Component Analysis

Rodrigue G. Atteba [1], Thanda Shwe [2], Israel Mendonça [2] and Masayoshi Aritsugi [2,*]

[1] Graduate School of Science and Technology (GSST), Kumamoto University, Kumamoto 860-8555, Japan; 238d9348@st.kumamoto-u.ac.jp
[2] Faculty of Advanced Science and Technology, Kumamoto University, Kumamoto 860-8555, Japan; thandashwe@cs.kumamoto-u.ac.jp (T.S.); israel@cs.kumamoto-u.ac.jp (I.M.)
* Correspondence: aritsugi@cs.kumamoto-u.ac.jp

Abstract: In the financial sector, machine learning has become essential for credit risk assessment, often outperforming traditional statistical approaches, such as linear regression, discriminant analysis, or model-based expert judgment. Although machine learning technologies are increasingly being used, further research is needed to understand how they can be effectively combined and how different models interact during credit evaluation. This study proposes a technique that integrates hierarchical clustering, namely Agglomerative clustering and Balanced Iterative Reducing and Clustering using Hierarchies, along with individual supervised models and a self organizing map-based consensus model. This approach helps to better understand how different clustering algorithms influence model performance. To support this approach, we performed a detailed unsupervised component analysis using metrics such as the silhouette score and Adjusted Rand Index to assess cluster quality and its relationship with the classification results. The study was applied to multiple datasets, including a Taiwanese credit dataset. It was also extended to a multiclass classification scenario to evaluate its generalization ability. The results show that the quality metrics of the cluster correlate with the performance, highlighting the importance of combining unsupervised clustering and self organizing map consensus methods for improving credit evaluation.

Keywords: hybrid artificial intelligence; AI in engineering; machine learning; optimization algorithms; predictive modeling; AI-based decision making; ensemble methods; clustering methods; model integration; credit scoring

1. Introduction

Machine learning develops algorithms to identify and learn data patterns and thus predict future outcomes or make complex decisions automatically [1]. In lending scenarios, Machine Learning can be used for two purposes: (1) to use supervised learning techniques to predict an applicant's likelihood of defaulting on a loan, and (2) to employ unsupervised learning to find hidden patterns in the applicant database so that lenders can perform more thorough risk evaluation of applicants [2], for example, in predicting the probability of customer defaults, identifying fraudulent transactions, and facilitating other forms of decision-making [3]. Machine learning is increasingly being considered for integration into credit rating systems to identify the risk of customer default [4]. Credit risk models

use characteristics such as the credit history of the customer, level of income, existing debts, and demographic characteristics such as age, sex, and number of dependents [5]. These models produce a score indicating the probability of default. Numerous examples show that machine learning models offer superior accuracy and adaptability to previously used credit risk scoring techniques based on traditional statistical methods such as linear regressions, discriminant analysis, or expert opinions [6].

Currently, financial institutions heavily rely on automated decision-making tools to process credit applications. Recent advances in supervised learning have enabled these systems to classify applicant risks more accurately than traditional methods [7]. Credit scoring models are essential tools for financial institutions, as they enable them to make more informed decisions when granting loans. Even slight improvements in model performance can generate substantial benefits [8], including reduced analysis costs, faster decision-making, better debt collection, and improved risk management [9].

To improve these models, researchers have tested several machine learning techniques, moving from single models to combinations of models using consensus models. Similarly, they are investigating how to combine supervised and unsupervised learning to achieve more robust performance. Among these approaches is the reference study by Bao et al. [10], which compared several combinations of models: individual supervised models, a Self-Organizing Map (SOM) [11] as a consensus model, and clustering using k-means.

Although promising, this methodology can be improved for a better analysis. For example, although SOM was used as a consensus model, the Area Under the Curve (AUC) metric was not calculated because of the difficulty in extracting the probability scores. However, it is possible to calculate the AUC using the probabilistic predictions of individual models as input to the SOM. This approach allows the SOM to organise these predictions in a lower-dimensional space for meaningful evaluation. Further exploration of the clustering component is necessary to better understand its influence on performance. Previous studies have often limited clustering to k-means, which may not represent more complex or non-linear data structures, thus motivating the exploration of alternative clustering methods. Clustering methods such as Agglomerative clustering [12] or Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH) [13] can reveal alternative data structures that classifiers can exploit, potentially improving model performance. Therefore, it is useful to assess the clustering step more closely to learn more about the association between the nature of clustering, classifier learning, and performance.

The general objective of this study was to examine the implications of the clustering step on model performance. More specifically, it seeks to determine whether classifiers can learn better from clusters created using alternative methods to k-means clustering, such as Agglomerative and BIRCH clustering. Cluster quality was evaluated according to the silhouette score [14] and Adjusted Rand Index (ARI) [15]. This provides a better understanding of how different clustering structures affect supervised learning and the overall ensemble performance.

Before proceeding with the analysis, it was relevant to explain the methodology described by Bao et al. [10] using the same unbalanced datasets (Australian and German), and an additional dataset (Taiwanese), while also studying the case of multiclass classification to assess its generalizability. The methodology was implemented through a reproducible step-by-step pipeline, allowing future replication and offering meaningful contributions to the development of more robust credit scoring systems.

To summarize, first, we introduced a systematic integration of hierarchical clustering with SOM-based consensus. This integration offers a deeper understanding of the data structures relevant to predictive modeling, which may improve model accuracy. We further present a new robustness analysis that relates both unsupervised metrics (i.e., Silhouette and Adjusted Rand

Index (ARI)) to classification performance. A third contribution is our multi-route comparative framework (Routes 1–5), which enables researchers to comparatively evaluate multiple, distinct approaches (Supervised, clustering-based, clustering-based consensus), thus providing a comprehensive understanding of the relative advantages and disadvantages of each approach. Last, we demonstrated how this technique can be extended to multiclass credit scoring, showing its generalizability beyond binary classification settings.

The remainder of this paper is organized as follows: In Section 2, a review of related studies explains the main methodologies and techniques relevant to this study. Sections 3 and 4 describe the proposed framework and the motivation for its use in robust cluster analysis. Section 5 presents the evaluation metrics, datasets used, data preprocessing steps, feature selection, and hyperparameter tuning. Section 6 details the experimental process, including the choice of the number of clusters and the implementation of SOM consensus models based on hierarchical clustering. In Section 7, the results obtained from binary classification tasks on the German, Australian, and Taiwanese datasets are discussed. Section 8 presents a robust cluster analysis, including a comparison of clustering metrics and model performance in relation to these metrics. Section 9 addresses the case of multiclass classification, describes the dataset used, presents the experimental results, and compares them with the binary classification findings. Section 10 discusses the significance of the results and presents the limitations of this study. The practical contributions and implications and their significance to real-life applications are established in Section 11. Finally, Section 12 concludes the study, reviews the main findings and outlines possible ideas for future research.

2. Related Work

2.1. Consensus Models

Among machine learning techniques, consensus models are the most widely used and effective for improving prediction performance. Consensus models take advantage of the different strengths of multiple classifiers and improve predictions by combining classifiers to decrease bias and variance and enhance generalization ability. The fundamental concept is that aggregated predictions of multiple models generally outperform those of a single model. Common ensemble strategies include Bagging [16], Boosting [17], Stacking, and the Majority Voting (MV) models.

In MV, several independent models are trained independently, and their final prediction is made by voting: either a hard vote, where the class predicted by the majority is selected, or a soft vote, where predictions are made based on aggregated probabilities. This method is widely used for binary and multiclass classifications. Kuncheva and Whitaker [18] examined the role of classifying diversity using ensemble methods and demonstrated how diversity improves the effectiveness of MV strategies. In the context of credit scoring, the MV serves as a robust consensus strategy, effectively merging predictions from various models to achieve greater accuracy. Muniappan and Paruvachi Subramanian [19] studied the use of MV to improve the accuracy of credit scoring and demonstrated that combining model output through majority voting significantly reduced error rates and improved predictive results compared to individual classifiers. Thus, consensus models not only improve robustness and performance but also contribute to more stable and reliable decision-making systems in high-stakes applications such as credit evaluation.

In this work, we present Kohonen's Self-Organizing Maps (SOM) as an alternative consensus mechanism that is beneficial compared to standard ensemble techniques. Kohonen [20] laid the theoretical principles of SOM, applying it to classification and data analysis. SOM is an unsupervised neural network, which converts high-dimensional data to a lower-dimensional grid regarding the topological structure of the input space. This property allows SOM to

effectively group like predictions from ensemble classifiers, even when these classifiers have different error structures. This is important for the situation of credit scoring, in which a classifier is likely to make different types of errors. The property of SOM to cluster similar predictions is also an intuitive advantage for improving the credit score.

A significant advantage of SOM compared to conventional ensemble methods such as stacking or majority voting is its ability to smooth noise and errors [21]. Before engaging in majority voting on the resulting predictions, SOM will cluster the prediction vectors in a way that reduce the effect of any noisy or outlier predictions that would normally bias the results with a simple ensemble method.

SOM can accommodate the heterogeneity present in model predictions. SOM leads to a more stable and reliable consensus, particularly in high-variance environments such as credit rating. It is also beneficial in terms of interpretability, which is important in some contexts like credit scoring [22,23]. SOM offers a two-dimensional map that interprets how models agree or disagree, while also providing insight into the underlying relationships that exist among the classifiers. This explains why analysts can investigate the contribution each model had in ultimately coming up with the final decision. In this way, SOM acts as both a performance booster but is also growing in popularity as a means of making decisions more interpretable.

2.2. Integration of Supervised and Unsupervised Models

Traditional credit scoring mainly relies on a supervised model, such as an Artificial Neural Network (ANN), Support Vector Machine (SVM), Decision Tree (DT), or Gradient Boosting Decision Tree (GBDT) trained on labeled datasets to predict the probability of default. However, there are opportunities to improve these approaches, notably through unsupervised learning strategies, for complex unlabeled or structured data. Integrating supervised and unsupervised models has proven to be a promising strategy for improving the predictive accuracy of credit scoring by exploiting both the explicit patterns found in labeled data and hidden structures detectable in unlabeled datasets. For example, Hsieh [24] proposed a hybrid approach combining clustering techniques (self-organizing map (SOM) and k-means) with neural networks, demonstrating improved model performance through the synergy of these methods. Similarly, Arutjothi and Senthamarai [25] presented an integrated model that merges Logistic Regression (LR) with k-means, effectively capturing structured credit history and unstructured financial behaviors for optimized default probability modeling. Yu et al. [26] further introduced a two-stage hybrid system (TSC-SVM) that combines spectral clustering and a semi-supervised SVM to address unbalanced data issues in credit risk modeling, showing clear improvements over standard supervised approaches. Although many studies have focused on hybrid models involving supervised and unsupervised methods, consensus models involving multiple unsupervised techniques, such as clustering combined with SOM, remain underexplored.

Despite the established role of clustering in customer segmentation, the impact of these clusters on classifier performance has rarely been quantified, and comparative analyses of different clustering algorithms are often absent. This study addresses this gap by exploring the potential of combining multiple unsupervised methods in a consensus framework and evaluating their influence on the stability and performance of classifiers. Through this integration, the research not only evaluates various machine learning strategies but also tests the robustness of the baseline methodology to improve the accuracy and interpretability of credit scoring models.

2.3. Comparison with Alternative Machine Learning Paradigms

Numerous studies use different machine learning techniques to build credit evaluation prediction models. The following section compares credit scoring with ensemble-based ma-

chine learning methods and other Machine Learning paradigms (including reinforcement learning, rule-based, neuro-symbolic AI and neuromorphic engineering).

2.3.1. Reinforcement Learning

Reinforcement Learning (RL) refers to an approach whereby agents learn to maximize their cumulative reward based on interactions with the environment. Even though RL has potential for credit scoring, its application remains limited. This is because credit data is mainly historical and non-interactive, unlike the dynamic and interactive environments typical of RL applications. Some studies have used RL to evaluate portfolios and for credit recommendations, but it is unknown if RL is effective for the credit scoring process, since credit scoring uses historical data and is more reactive than interactive [27,28]. Dynamic decision-making environments such as portfolio optimization and trading systems are the ideal environment for using RL for developing a model due to the way agents will need to make sequential decisions and repeatedly interact with the environment over time [29]. In contrast, a static credit classification task does not allow for sequential decision making nor require an agent to continually interact with the environment. The use of RL in this case is not practical since credit scoring is performed in batch mode, with a fixed dataset and a predetermined outcome.

2.3.2. Rule-Based Learning

While traditional expert systems and rule-based learning methods provide strong model interpretability, they have limitations in terms of their capability to model complex nonlinear behaviors. For example, rule-based systems are typically used in situations where the importance of explainability takes precedence (such as credit scoring), so the models created by these systems are less able to identify and capture the complex nonlinear relationship of financial variables [30]. The hybrid method outlined in this article is to combine supervised learning and clustering together to identify nonlinear relationships, while still providing some level of interpretability because of the clustering structure.

2.3.3. Neuro-Symbolic AI

Neuro-symbolic AI aims to incorporate the statistical strengths of deep learning with the structured reasoning learned from symbolic techniques to enable complex tasks needing logic and the ability to analyze patterns. A problem thus far has been that most credit scoring databases are tabular and structured in nature, making it difficult to effectively utilize symbolic learning techniques. However, Neuro-symbolic AI could be a valuable avenue for future work since it could potentially utilize the model interpretability associated with symbolic AI, while providing the predictive capabilities of deep learning [31]. Currently, although there are several neuro-symbolic methodologies proven to allow companies to utilize its capabilities, these methodologies are not widely utilized in credit scoring systems at this time [32]. The methodology presented in this paper takes a different practical approach as it incorporates an ensemble of models to provide a robust performance.

2.3.4. Neuromorphic Engineering

Neuromorphic engineering techniques focus on creating hardware in a way similar to the brain and using this hardware to optimize the process of evaluating very complicated calculations. it remains fundamentally a hardware-oriented approach. As a result, there is little or no application of neuromorphic techniques in the analysis of credit reports using tabular, historical and static datasets. This is in large part because most of the applications of neuromorphic techniques are in the areas of online learning (requiring real-time processing

of incoming information). In comparison to the potential of neuromorphic engineering with regards to the application of the Internet of Things (IoT) or autonomous systems, the focus of this article does not encompass the effects that neuromorphic techniques would have within this application area [33]. Neuromorphic systems are designed specifically for hardware optimization and energy-efficient computing. They achieve optimized architecture and performance by emulating the characteristics of biological neural networks [34]. While relevant for large-scale deployment scenarios, this approach addresses implementation, and efficiency concerns rather than offering algorithmic improvements for tabular data classification. Our case focuses on improving prediction capability.

3. Overview of Proposed Framework

This section explains the methodological approach and helps understand the logical connections between different sections of the paper.

Figure 1 describes the framework process and the link between the C's five credit evaluation (Character, Conditions, Capital, Capacity, and Collateral) [35]. The framework begins with data preparation (Section 5) where three datasets (German, Australian, Taiwan) are described. The next section (Section 6) describes five experimental routes from the clustered data to the final model. Each route is a different combination of clustering methods with supervised classifiers and the use of Self Organizing Map (SOM) method to generate a consensus model from the classifiers. Route 1 establishes baseline performance using individual classifiers, while Route 2 introduces SOM consensus. Routes 3–5 also tested different clustering methods (k-means, Agglomerative, BIRCH) using the same three datasets. Route 5 introduced hierarchical clustering approach. All routes converge into the evaluation phase (Sections 7–9). This phase provides a performance analysis of each route for binary classification, clustering robustness, and multiclass extension. The results are synthesized to draw conclusions about the effectiveness of integrating unsupervised and supervised learning for credit evaluation.

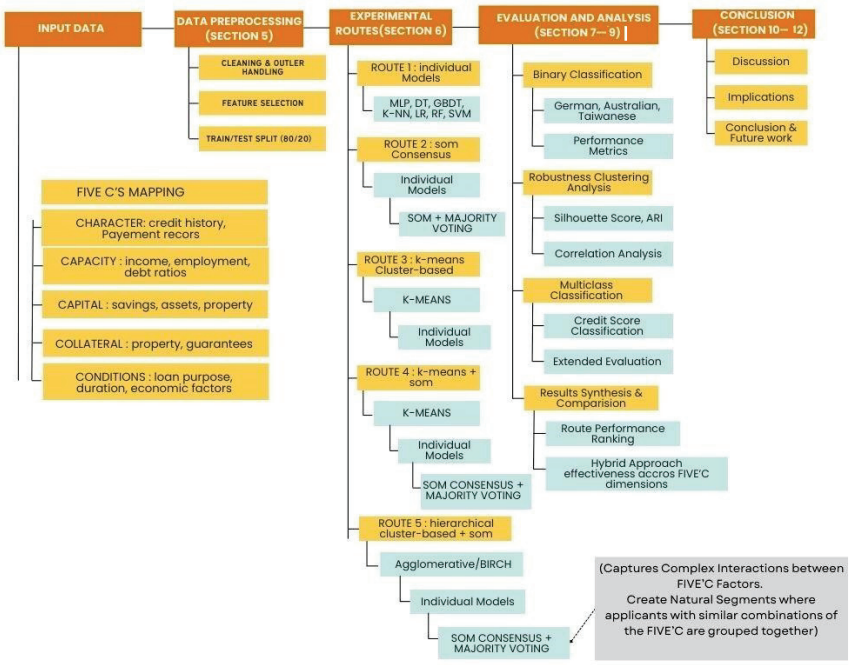


Figure 1. Study framework.

Problem definition: The study addresses the improvement of credit scoring models by integrating unsupervised clustering with supervised classifiers. It investigated how clustering quality influences the ability of classifiers to accurately predict creditworthiness using clusters generated with different clustering techniques (i.e., Agglomerative, BIRCH, k-means). The clusters were combined with supervised classification methods (e.g., ANN, SVM, RF) as well as a consensus mechanism based on SOM.

Inputs: The inputs are three publicly available credit datasets:

- German: 1000 samples, 20 features, binary class (70% good, 30% bad).
- Australian: 690 samples, 14 features, binary class (44.5% good, 55.5% bad).
- Taiwanese: 30,000 samples, 23 features, binary class (77.88% good, 22.12% bad).

Outputs: The outputs are classification performance metrics (MCC, AUC, Accuracy, Recall, Precision, Type I/II errors) for all five experimental routes (R1–R5) and resumed in Tables 4–9 (for binary classification) and Tables 22 and 23 (for multiclass classification). The results show the impact of clustering and consensus method on model performance.

4. Methodology

This section is divided into two sections. It presents the framework proposed in the reference paper, adding a new approach as well as the clustering analysis performed to evaluate the consistency and impact of different clustering techniques.

4.1. Proposed Framework

Figure 2 illustrates the methodological framework and presents five experimental Routes focused on quantifying the usefulness of integrating unsupervised learning into supervised credit scoring models. Route 1–4 were from Bao et al. [10]. The first experimental route establishes a baseline by evaluating each classifier (such as a multi-layer perceptron (MLP), DT, GBDT, LR, k-Nearest Neighbors (k-NN), Random Forest (RF), and SVM) before any integration is applied. Route 2 builds on the classifiers by applying a consensus model based on the SOM, where the predictions of each classifier are aggregated by majority voting, producing a more robust classification. Route 3 introduced k-means clustering as a preprocessing step, generating clusters and adding the clusters as a new variable to the dataset, and then individually training the classifiers to predict the corresponding label. Route 4 extends Route 3 by applying the SOM consensus to k-means-based models, following the same pattern as Route 2. Finally, Route 5 was added. It combines hierarchical clustering, individual classifiers, and consensus based on the SOM, allowing for a comparative examination of different clustering techniques.

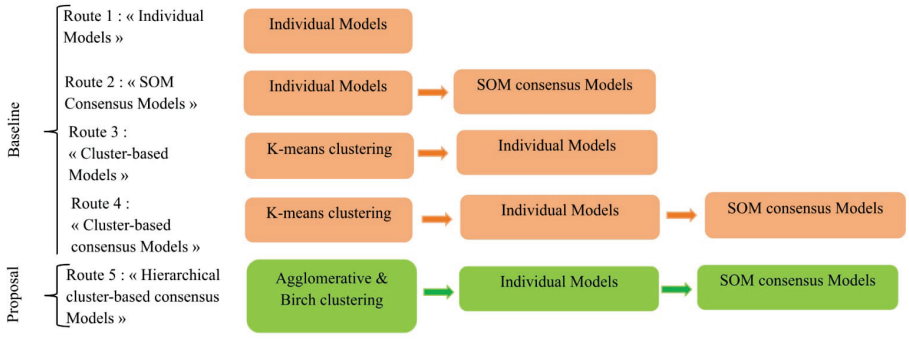


Figure 2. Baseline approach [1] and Proposal. [1] Source: [10].

4.2. Robustness Clustering Analysis

Following the model development and evaluation through the five experimental pathways, the next part of the methodology focuses on examining the robustness and influence of clustering on model performance. This evaluation is necessary to determine the effectiveness of clustering in improving the classification performance.

To determine robustness, we systematically compare the three clustering strategies in this study (k-means, Agglomerative, and BIRCH) in terms of their impact on subsequent classification. The quality and stability of the clustering results were assessed using common metrics such as the silhouette score and Adjusted Rand Index (ARI). These metrics will help evaluate the stability of the clusters produced in terms of cohesion and separation. They also provide insights into the qualitative depth and stability of clusters, which are necessary for the formation of meaningful clusters. We then compared the performance of the classifiers for each method to identify the cluster with the best synergy. Finally, we analyze the performance metrics of each route to draw conclusions regarding the effectiveness of the clustering methods in a hybrid credit scoring system. Overall, this two-dimensional analysis allows us to go beyond simple clustering directly to information about clusters or their stability, which allows us to better understand the conditions under which unsupervised learning can be most effective. This analysis focuses only on routes 4 and 5. The clustering analysis methodology is illustrated in Figure 3.

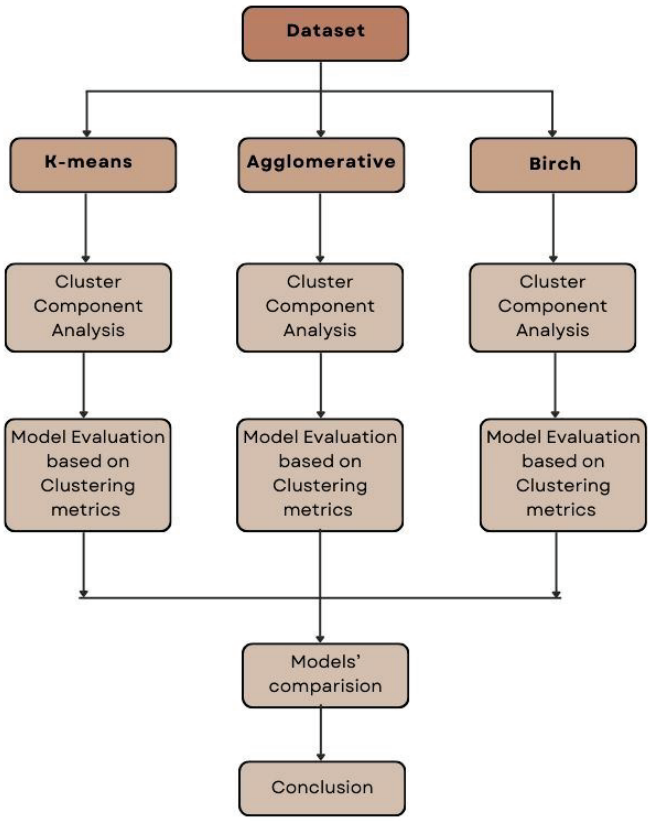


Figure 3. Proposed clustering analysis methodology.

4.3. Integration of the Five C's into the Proposed Approach

This section presents the link between our approach and the five C's of credit evaluation: Character, Capacity, Capital, Collateral, and Conditions. We explain how these concepts are integrated into our methodology (Figure 1).

The explanatory variables in the three datasets used in this study (German, Australian, Taiwanese) include the five C's, though their representation varies across datasets. Below, we explain how each 'C' is represented in the dataset characteristics [36]:

- Character (Credit History): This factor includes variables related to payment history, past defaults, and credit behavior. This information helps assess a borrower's reliability by capturing indicators such as late payments and the consistency of past repayments.
- Capacity (Repayment capacity): The ability to repay a loan is shown through characteristics like income, employment status, and debt-to-income ratios. These variables help determine if a borrower can handle new debt based on their income and current financial obligations.
- Capital (Financial reserves): Savings, assets held, and financial reserves are examples of financial indicators that represent this C. This data reveals a borrower's financial stability, which may have an impact on their capacity to repay in the event of problems.
- Collateral : Our datasets contain variables that indicate the existence of collateral, such as real estate or other property. When evaluating a loan's security in the event of borrower default, this information is essential.
- Conditions (Economic context): Factors pertaining to the loan's purpose, interest rates, and, if available, macroeconomic indicators are considered. These factors can have an impact on a borrower's capacity to repay. For instance, the borrower's financial stability and, consequently, their capacity to repay may be impacted by overall economic conditions.

Our method combines clustering and supervised classification techniques with features taken from the datasets to address these five dimensions. We use a SOM (Self-Organizing Map) consensus mechanism and clustering algorithms (like k-means, Agglomerative, and BIRCH) to find groups of borrowers with comparable profiles based on these five criteria. By incorporating these variables into our model, we can investigate the intricate relationships between these various elements as well as more precisely evaluate credit risk.

Based on similarity across all features, our clustering techniques (k-means, Agglomerative, and BIRCH) automatically group candidates. Classifiers can learn cluster-specific decision patterns because this effectively creates natural segments where applicants with similar combinations of the five C's are grouped together. While maintaining the topological connections between various credit profiles, the Self-Organizing Map consensus mechanism functions as an ensemble credit committee that synthesizes several specialized opinions (individual classifiers). This is like how human credit analysts combine data from all five C's to make a final decision. For instance, there are a total of twenty explanatory variables in the German dataset. The most pertinent ones for the five C's framework are covered in the Table 1 below. Because credit evaluation factors are interconnected, some variables (such as property and other_parties) contribute to more than one category. Since they don't directly fit into the conventional five C's categories, variables like "telephone" and "number_people_liable" were left out of this mapping.

Although our methodology is based on machine learning, it remains closely linked to the fundamental principles of credit evaluation (Figure 1 above), where the five "C's" are naturally integrated into the data analysis process. This reinforces the relevance and interpretability of our approach, while allowing us to draw conclusions about the effectiveness of integrating supervised and unsupervised learning techniques for credit assessment.

Table 1. Mapping of German credit dataset variables to the five C's of credit evaluation.

Five C's Category	German Dataset Variables	Description & Justification
Character	status, credit_history, purpose, personal_status	status: Status of existing checking account; credit_history: History of previous payments; purpose: Intended use of loan; personal_status: Personal status and sex.
Capacity	employment, installment_rate, other_installment_plans, existing_credits	employment: Present employment since; installment_rate: Installment rate as percentage of disposable income; other_installment_plans: Other installment plans; existing_credits: Number of existing credits.
Capital	savings, property, other_parties	savings: Savings accounts/bonds; property: Property ownership; other_parties: Other debtors/guarantors.
Collateral	property, other_parties, housing	property: Real estate or other assets; other_parties: Additional guarantors; housing: Home ownership status.
Conditions	job, foreign_worker, duration, age	job: Job type and stability; foreign_worker: Employment conditions; duration: Loan duration; age: Applicant's age and life stage.

5. Evaluation Metrics, Data Sources and Preprocessing

5.1. Evaluation Metrics

The test sets were evaluated for predictive performance using seven metrics: the Matthews Correlation Coefficient (MCC), Area Under the Curve (AUC), accuracy (ACC), Recall, Precision, Type I error, and Type II error (Table A1 in the Appendix B). MCC is particularly relevant when studying unbalanced datasets because it integrates four different classifications (i.e., true positive, false positive, true negative, and false negative). It is more reliable and balanced assessment of classification performance than accuracy when the class distribution is unbalanced. MCC measures the consistency between the predicted and actual results with a range from -1 (perfect disagreement) to 1 (perfect prediction); zero indicates random classification.

The AUC quantifies the area under the ROC curve and evaluates a classifier's ability to produce a ranking. However, the SOM, which is an unsupervised neural network, organizes data topologically without providing probabilities of class membership. Therefore, the AUC cannot be calculated directly from the SOM results. SOM is an unsupervised neural network that organizes instance data topologically, grouping similar instances close together on a two-dimensional grid. Therefore, SOM does not provide results in terms of probabilities; the AUC cannot therefore be calculated solely from these results. This limitation has also been noted in a previous study. That said, the AUC can be estimated indirectly using SOM in a hybrid or post-processing framework.

For example, in clustering approaches followed by supervised classification, the SOM is first trained to group similar data. After clustering, classifiers, such as LR or k-NN, predict class probabilities. Another technique, presented by Halkidi et al. [37], involves combining SOM with k-means clustering, in which the clusters formed by neurons are further processed to estimate the probability distributions per sample. This enabled the calculation of AUC based on the probabilities of the derived class.

In this study, the supervised models generated class probability outputs that were organized and clustered using SOM. The consensus model uses the predictions of individual classifiers to accumulate them by majority vote to evaluate the results of the aggregated predictions. This means that we can calculate the AUC scores, including the probabilistic output of the base classifiers, even with SOM. Although AUC is computed by feeding supervised model probabilities into SOM, the SOM itself does not generate probability distributions or necessarily preserve ranking properties needed for ROC metrics. Thus,

the AUC values attributed to 'SOM consensus' reflect the underlying supervised models, not the SOM. Figure 4 illustrates the complete process of aggregating predictions using the SOM consensus model, the steps involved in implementing majority voting, and the process of evaluating the performance of the outputs.

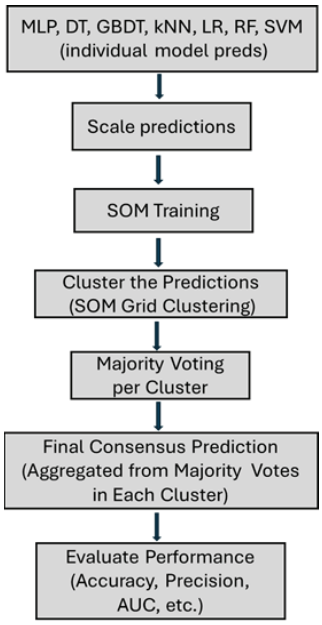


Figure 4. Majority Voting process.

5.2. Datasets

Here are descriptions of the datasets used:

- German dataset

The German bank dataset is available in the UCI repository [38] and contains 1000 observations and 21 variables, including 20 explanatory variables. The target variable is the credit score, with two classes: good (70%) and bad (30%). The set is unbalanced between classes with no missing values.

- Australian dataset

The Australian dataset, also available on the UCI repository [39], is linked to credit card applications and is also referenced as the "Credit Screening Database" and uses a binary variable to indicate whether a candidate is good or bad. Of the 690 samples containing 307 were good candidates (44.5%) and 383 were bad candidates (55.5%).

- Taiwan dataset

The Taiwanese dataset comes from the UCI repository and concerns credit card customer payment defaults in Taiwan, collected in October 2005 by a major credit card and cash credit issuing bank [40]; it includes a binary variable as a target, representing payment default (1 for "Yes", 0 for "No"), and contains 30,000 samples, including 6636 defaults (bad candidates) and 23,364 non-defaults (good candidates), with defaulting candidates representing 22.12% of samples and non-defaulters accounting for 77.88%.

The datasets used were class-imbalanced, which affects performance on minority classes. To overcome this issue, we chose majority voting (MV) as the consensus mechanism. This method has several advantages in an imbalanced data context, including reduced risk of overfitting on the majority class and improved robustness. Majority voting has the potential to increase model robustness while reducing the risk of overfitting to the majority

class and improving the overall accuracy by combining predictions from multiple models. Several studies have shown that this method, considered for imbalanced datasets, can help manage the impact of the minority class on the model performance [41,42]. Table 2 presents a summary of the dataset information in terms of the sample size and feature dimension.

Table 2. Datasets description.

Dataset	Samples	Features	Good Applicants (%)	Bad Applicants (%)
German	1000	20	70	30
Australian	690	14	44.5	55.5
Taiwan	30,000	23	77.88	22.12

5.3. Data Pre-Processing

Data cleaning involves removing some cases and imputing missing values to improve the quality. This process aims to eliminate errors and correct the inaccuracies that can affect the results. However, the data used in this study did not contain missing values or duplicates, making data cleaning unnecessary. We visualized the feature distributions using box plots to observe the distribution of the variables across different classes. Outliers are identified during this step. To handle outliers, we applied the interquartile range (IQR) and then modified the outliers by taking the mean or mode, depending on the type of variable. However, in the datasets, outliers did not appear to affect model performance.

5.4. Features Selection

Feature selection is an important step in building a machine learning model. This is the process of selecting the most appropriate and relevant features from a dataset that are likely to influence the predictive power of the model. To identify the most important features and eliminate irrelevant features, we applied a selection process based on the RF model using scikit-learn. We visualized the importance of the features, selected the most significant features, and analyzed the performance of the model before and after adding the feature selection to each dataset. The data were split into training (80%) and testing (20%) sets and saved as .csv files.

The training dataset was first loaded, and the RF model was trained on all features. The performance of the model was assessed, and the features were ranked according to their importance. Significant features were selected, and a new model was trained on this reduced set of features. The performance was re-evaluated and recorded.

- German dataset features selection:

The accuracy before feature selection was 0.7350. After selecting 15 features, it was increased to 0.7550. The model based on the 15 features outperformed the original dataset model. Thus, 15 features were selected for the final prediction.

- Australian dataset features selection:

The accuracy before feature selection was 0.8768, whereas after selecting 10 features, it decreased to 0.8551. A reduced feature set is not used. We used the original dataset for the final prediction.

- Taiwan dataset features selection:

The accuracy before feature selection was 0.8160. After selecting 15 features, it dropped to 0.7550, and only after selecting 18 or 20 features did the accuracy return to 0.8148. This slight variation suggests that most of the features contributed. Hence, we used the original dataset for the predictions. Table 3 summarizes the feature selection process.

Table 3. Feature selection summary.

Dataset	Original Features	Accuracy Before Selection	Selected Features	Accuracy After Selection	Selection Applied
German	20	0.7350	15	0.7550	Yes
Australian	14	0.8768	10	0.8551	No
Taiwan	23	0.8160	15 18 20	0.7550 0.8148 0.8148	No

5.5. Hyperparameter Tuning

To optimize the efficiency of each model, we tuned the hyperparameters using the GridSearchCV method combined with cross-validation (CV = 5). We used a large grid of hyperparameters, allowing the model to reach a wide range of possible configurations and then to find the optimal parameters for optimum performance. In addition, we increased the number of iterations to provide the models with more time to converge, thereby increasing the chances of reaching the most efficient solution. Once the best parameters were identified through a grid search, we re-tuned the final model on the entire training dataset using these optimal hyperparameters. This final step ensures that the model is retrained with the most suitable parameters, making the final version as accurate and robust as possible (see Table A2 in the Appendix B).

For Agglomerative, we chose to set n_clusters = 2 and linkage = 'average'. This configuration is commonly used for its ability to produce stable and interpretable results in hierarchical clustering contexts. The choice of linkage = 'average' allows us to consider the average distance between the points in each cluster, which yielded good results in our tests. For BIRCH, we used n_clusters = 2. For the other parameters, we used the default parameters threshold = 0.5, branching_factor = 50, max_iter = 100, compute_labels = True, copy = True, affinity = 'euclidean', random_state = None.

6. Experimental Process

In this section, we focus only on the proposed Route 5. For reference, we referred to the study by Bao et al. [10] and maintained the same experimental setup to ensure comparability. All experiments were performed in Python 3.10 in a server environment equipped with a core memory size of 130 GiB and a CPU AMD Ryzen 9 5950X 16-Core Processor. The consensus models analyzed followed the approaches detailed in Section 4.1 and a summary of Routes 1–4 is presented in Table 4.

Appendix C presents a general example illustrating the experimental process of Route 5 from a simple dataset, in order to provide a clear overview of the entire experimental process.

Table 4. Summary of Experimental Routes (1–4).

Route	Learning Type	Description	Goal
Route 1	Individual models	Individual models (MLP, DT, SVM, LR, k-NN, RF, GBDT) trained separately on the full dataset.	Establish baseline performance for each classifier independently.
Route 2	Individual models + SOM	Predictions (probabilities) of seven individual models are passed to a SOM. The SOM clusters similar outputs and applies majority voting for final consensus.	Combine strengths of base classifiers through SOM consensus.

Table 4. *Cont.*

Route	Learning Type	Description	Goal
Route 3	k-means + individual models	k-means clustering is applied first to split the dataset into clusters. Then, individual models are trained within each cluster. No consensus model is used.	Capture local patterns in the data and evaluate performance within clusters.
Route 4	k-means + individual models + SOM	Combines Route 3 with SOM. Predictions from individual models (trained in each k-means cluster) are passed to a SOM, which aggregates them into a final consensus using majority vote.	Combining cluster-level predictions using SOM.

6.1. Choice of the Number of Clusters k

The k-means method requires the value k to be pre-specified. k is the number of clusters into which data can be grouped. The elbow method was used to determine the optimal number of clusters. Visualization of the elbow method for the three datasets showed that $k = 2$ or 3 are good values for k (where the graph becomes more linear (Figure 5)). To select a suitable value for k, we computed the silhouette score [14] for each value of the k number. The silhouette score indicates separation between clusters. A low silhouette score (0 or negative) indicates some overlap between clusters, suggesting confusion in classification due to the lack of distinctiveness in the data points being classified. $k = 2$ provided the best possible scores for all the datasets. Principal Component Analysis (PCA) from the sklearn decomposition library was used to reduce the original dimensionality of the datasets to two dimensions (2D) for visualization (Figure 6). Visualizations based on the first two principal components (PCA) are used to illustrate the clusters. It is important to note that these 2D visualizations, produced by an PCA, are only illustrative and qualitative. They do not accurately represent the separations between clusters in the original high-dimensional space. In fact, all clustering decisions were made using the complete set of features in their original dimension. They allow us to observe general trends and apparent groupings in the data, but they should not be interpreted as accurate representations of the separation of clusters in the full feature space.

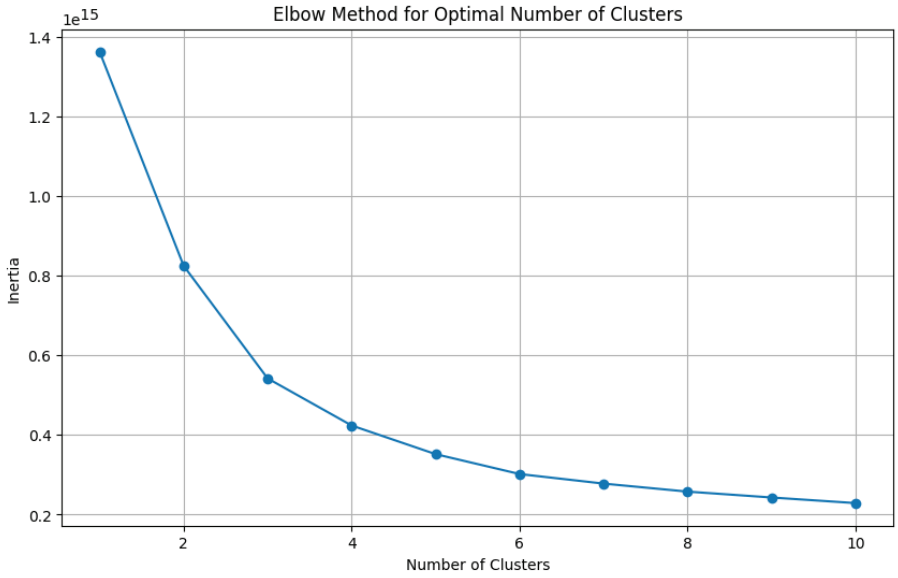


Figure 5. Elbow method for optimal number of clusters of Taiwan dataset.

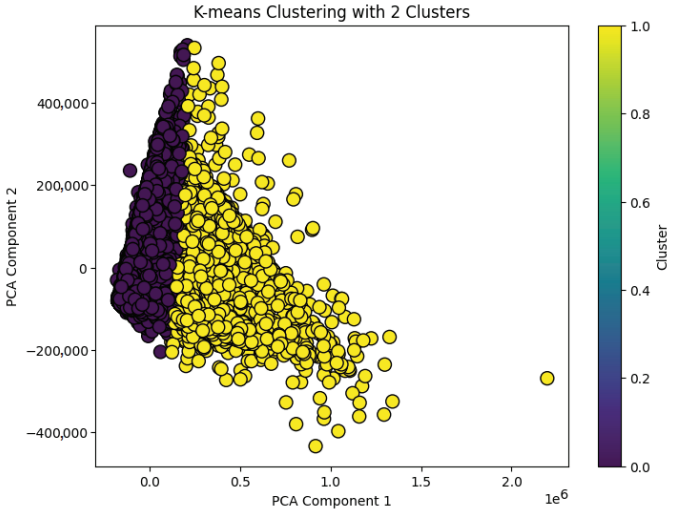


Figure 6. Clusters of Taiwan dataset.

6.2. ROUTE 5: SOM Consensus Model Based on Hierarchical Clustering

Figure 7 illustrates the experimental process for Route 5. This route utilizes a combination of hierarchical clustering techniques, supervised models, and a consensus model based on SOM to improve the prediction and generalization.

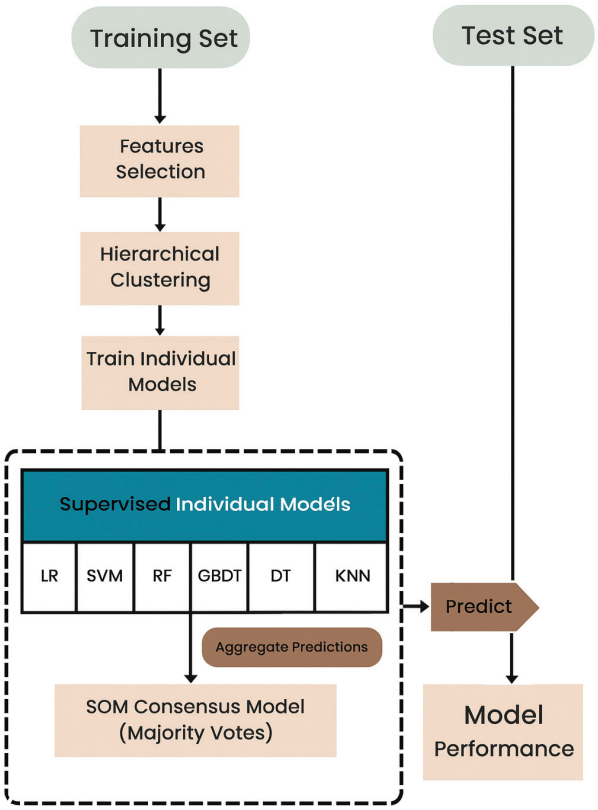


Figure 7. Route 5: SOM consensus model based on hierarchical clustering + individual models' experimental process.

First, hierarchical clustering was performed on the data to cluster similar points before supervision. In this study, we evaluated two hierarchical clustering algorithms: Agglomerative clustering (with linkage average) and BIRCH clustering. These methods were selected because they import structures into the data without actively controlling the shapes and sizes of clusters.

The SOM uses the prediction probabilities from each of the clusters and base models as inputs and organizes them topologically. The predictions from all individual models were combined using DataFrame (Figure 8). Each model (e.g., MLP and DT) has separate columns that represent their predictions. This combination, along with the final DataFrame, contains a column for each model, allowing for the collective analysis of predictions and the better application of ensemble methodologies. Consensus predictions are based on the predictions of individual classifiers. These predictions were scaled and integrated into the SOM model, which grouped the similar predictions.

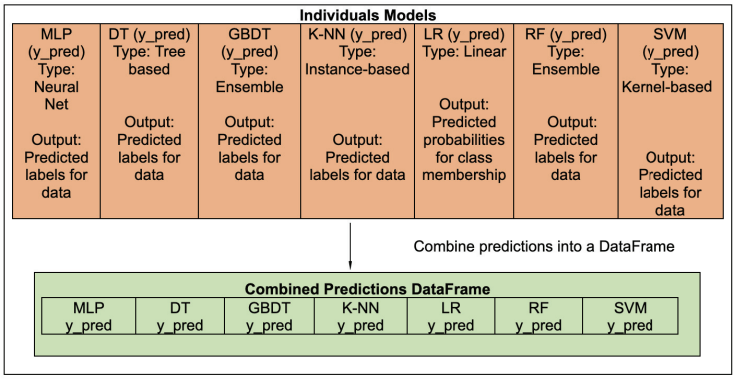


Figure 8. Combine predictions dataframe.

The final decision was made by a majority vote on the SOM grid. This additional SOM model assists in reconciling differences between cluster-based models and provides a global view of the dataset's patterns. This route retains the benefits of unsupervised learning for data structure (by hierarchical clustering), whereas supervised learning is used for prediction with SOM serving as a consensus mechanism to improve generalization and overall robustness of the final model.

7. Experimental Results (Binary Classification)

This section presents the detailed results for each dataset and compares the performance of this study with that of the reference study.

7.1. German Dataset

The results for the German test set are presented in Tables 5 and 6. A comparative analysis of the different classification routes (Routes 1 to 5) applied to the German dataset revealed a clear improvement in the hybrid approaches that integrated unsupervised and SOM techniques. The individual models (Route 1) showed the weakest performance for all metrics. In contrast, Route 2 to Route 5 showed significant improvement, particularly in terms of MCC, AUC, ACC, recall, and precision. Among these, Route 5, which combines hierarchical clustering (Agglomerative or BIRCH) with SOM, stands out with the best values for Agglomerative (MCC (0.5708), ACC (0.825), precision (0.8322)) and BIRCH (AUC (0.7859), recall (0.9420), and the lowest Type II error (0.0579)), illustrating the excellent ability to detect positive classes without generating too many false negatives. Compared

with the reference study (MCC (0.490), ACC (0.792)), this model showed a significant improvement. Aggregating SOM model predictions improved robustness and accuracy. Although Route 3 (DT) exhibited the best performance in terms of Type I error (0.3548), Route 5 maintained a good balance between all metrics. These results confirm that the integration of clustering techniques with SOM enables a more relevant structure of the data and leads to a general improvement in classification performance. Consequently, Route 5 appeared to be the most robust and effective for processing this type of data. The consensus models (Routes 2, 4, and 5) outperformed the individual and cluster-based models, confirming the advantage of SOM in improving the classification performance.

Table 5. Model performance for All Routes—German Dataset (Part 1).

German Dataset	**R1: Individual Models**							**R2: SOM Cons.**
Metric	**ANN**	**DT**	**GBDT**	**k-NN**	**LR**	**RF**	**SVM**	
MCC	0.3454	0.2952	0.3445	0.2677	0.3643	0.3974	0.3894	0.4868
AUC	0.6727	0.6441	0.6517	0.6061	0.6545	0.6847	0.6617	0.7669
ACC	0.72	0.705	0.74	0.72	0.75	0.755	0.76	0.79
Recall	0.72	0.705	0.74	0.72	0.75	0.755	0.76	0.8913
Precision	0.72	0.6987	0.7246	0.6974	0.7363	0.7444	0.7496	0.82
Type I error	0.4516	0.5161	0.5806	0.6935	0.5967	0.5	0.5967	0.4354
Type II error	0.2028	0.1956	0.1159	0.0942	0.0942	0.1304	0.0797	0.1086

Table 6. Model performance for All Routes—German Dataset (Part 2).

German Dataset	**R3: k-Means Cluster Models**							**R4: k-Means SOM**	**R5: Cluster-Based SOM**	
Metric	**ANN**	**DT**	**GBDT**	**k-NN**	**LR**	**RF**	**SVM**		**Agglom.**	**Birch**
MCC	0.4094	0.3506	0.3172	0.3050	0.3488	0.4269	0.3894	0.5176	**0.5708**	0.5568
AUC	0.7802	0.7111	0.7484	0.7435	0.7837	0.7692	0.7796	0.7851	**0.7859**	0.7822
ACC	0.765	0.7	0.735	0.73	0.745	0.77	0.76	0.805	**0.825**	0.82
Recall	0.9057	0.7246	0.9057	0.8985	0.9057	0.8985	0.9202	0.9275	0.9347	**0.9420**
Precision	0.7861	0.8196	0.7575	0.7560	0.7668	0.7948	0.7743	0.8152	**0.8322**	0.8227
Type I error	0.5483	**0.3548**	0.6451	0.6451	0.6129	0.5161	0.5967	0.4677	**0.4193**	0.4516
Type II error	0.2753	0.0942	0.1014	0.0942	0.1014	0.0797	0.0724	0.0652	**0.0579**	0.0579

7.2. Australian Dataset

The results for the Australian test set are presented in Tables 7 and 8.

An analysis of the results obtained using the Australian dataset revealed significant differences in the performance between the different approaches. The individual models (Route 1) performed fairly well, with an MCC of 0.7395 (SVM) and an AUC of 0.8739 (SVM). The SOM consensus model (Route 2) significantly improved all indicators, achieving an MCC of 0.8598, an AUC of 0.9042, an ACC of 0.9347, a precision of 0.9565, and a Type I error rate of 0.0229. Compared to the baseline, this model improved the MCC (0.725), ACC (0.862), and precision (0.806) values. The models resulting from k-means segmentation (Route 3) obtained performances comparable to or slightly inferior to those of individual models. In contrast, the combination of k-means and SOM (Route 4) showed substantial improvement, particularly for MCC (0.8138) and AUC (0.9118). Finally, the models based on clustering followed by SOM consensus (Route 5) clearly dominated all routes; in particular, the variant using the Agglomerative clustering achieved the best scores for almost all metrics: MCC (0.8765), AUC (0.9107), ACC (0.9420), precision (0.9777), and Type I error rate reduced to 0.0114. In conclusion, the integration of clustering and SOM provides a particularly powerful classification framework that offers the best compromise between performance indicators, clearly outperforming traditional or simple cluster-based approaches.

Table 7. Model performance for All Routes—Australian Dataset (Part 1).

Australian Dataset	**R1: Individual Models**							**R2: SOM Cons.**
Metric	**ANN**	**DT**	**GBDT**	**k-NN**	**LR**	**RF**	**SVM**	
MCC	0.5413	0.6554	0.7331	0.7197	0.7226	0.7163	0.7395	**0.8598**
AUC	0.7643	0.8248	0.8617	0.8356	0.8640	0.8478	0.8739	0.9042
ACC	0.7898	0.8405	0.8768	0.8695	0.8695	0.8695	0.8768	**0.9347**
Recall	0.7898	0.8405	0.8768	0.8695	0.8695	0.8695	**0.8768**	0.8627
Precision	0.7870	0.8395	0.8760	0.8760	0.8709	0.8693	0.8790	**0.9565**
Type I error	0.1379	0.1149	0.0804	**0.0344**	0.1149	0.0689	0.1149	0.0229
Type II error	0.3333	0.2352	0.1960	0.2941	0.1568	0.2352	**0.1372**	**0.1372**

Table 8. Model performance for All Routes—Australian Dataset (Part 2).

Australian Dataset	**R3: k-Means Cluster Models**							**R4: k-Means SOM**	**R5: Cluster-Based SOM**	
Metric	**ANN**	**DT**	**GBDT**	**k-NN**	**LR**	**RF**	**SVM**		**Agglom.**	**Birch**
MCC	0.6701	0.6889	0.6918	0.7197	0.7226	0.7163	0.7395	0.8138	**0.8765**	0.8291
AUC	**0.9145**	0.9019	0.8843	0.8928	0.9051	0.9215	0.9229	0.9118	0.9107	0.9118
ACC	0.8478	0.8550	0.8550	0.8695	0.8695	0.8695	0.8768	0.9130	**0.9420**	0.9202
Recall	0.7647	0.8039	0.8235	0.7058	0.8431	0.7647	0.8627	0.8039	**0.8627**	0.8235
Precision	0.8125	0.8039	0.7924	0.9230	0.8113	0.8666	0.8148	0.9534	**0.9777**	0.9545
Type I error	0.1034	0.1149	0.1264	0.0344	0.1149	0.0689	0.1149	0.0229	**0.0114**	0.0229
Type II error	0.2352	0.1960	0.1764	0.2941	0.1568	0.2352	**0.1372**	0.1960	**0.1372**	0.1764

7.3. Taiwan Dataset

The results of the Taiwan test set are presented in Tables 9 and 10, where the performance analysis on the Taiwanese dataset shows a more modest improvement than in the German and Australian cases but still highlights interesting differences between the different approaches. The individual models (Route 1) obtained relatively homogeneous results, with MCCs of approximately 0.39 for the best (RF and GBDT) and a precision of approximately 0.80. However, their recall remained high (approximately 0.81), suggesting a good ability to detect positive cases, but at the cost of a high Type II error rate (often >0.64). The SOM consensus approach (Route 2) improved the MCC (0.4014) and AUC (0.7710) while maintaining good precision (0.6807). However, this method had a significantly lower recall (0.3556), indicating a reduction in the sensitivity in detecting positive cases. The models derived from k-means (Route 3) did not offer any clear advantage, with MCC and AUC comparable to the individual models and slightly lower performance in terms of recall and precision. The combined k-means and SOM model (Route 4) offered a slight gain, with an MCC of 0.4038, an AUC of 0.7606, and an ACC of 0.823. Finally, in Route 5, the consensus methods based on Agglomerative or BIRCH clustering yielded consistent results: the BIRCH version had the best overall MCC (0.4057), an ACC of 0.8231, and the lowest Type II error rate (0.6359).

Table 9. Model performance for All Routes—Taiwan Dataset (Part 1).

Taiwan dataset	**R1: Individual Models**							**R2: SOM Cons.**
Metric	**ANN**	**DT**	**GBDT**	**k-NN**	**LR**	**RF**	**SVM**	
MCC	0.3847	0.3615	0.3919	0.3463	0.3250	0.3936	0.3264	**0.4014**
AUC	0.6499	0.6440	0.6490	0.6251	0.6033	0.6524	0.6018	**0.7710**
ACC	0.8175	0.8100	0.8205	0.8110	0.8098	0.8201	0.8105	0.8225
Recall	0.8175	0.8100	0.8205	0.8110	0.8098	0.8201	0.8105	0.3556
Precision	0.7987	0.7890	**0.8030**	0.7896	0.7914	0.8024	0.7937	0.6807
Type I error	0.0520	0.0608	0.0460	0.0443	**0.0294**	0.0492	0.0270	0.0462
Type II error	0.6481	0.6511	0.6557	0.7052	0.7638	0.6458	0.7692	0.6443

Table 10. Model performance for All Routes—Taiwan Dataset (Part 2).

Taiwan Dataset	R3: k-Means Cluster Models							R4: k-Means SOM	R5: Cluster-Based SOM	
Metric	**ANN**	**DT**	**GBDT**	**k-NN**	**LR**	**RF**	**SVM**		**Agglom.**	**Birch**
MCC	0.3829	0.3766	0.3774	0.3510	0.3236	0.3832	0.3771	**0.4038**	0.4011	**0.4057**
AUC	0.7637	0.7580	0.7626	0.7304	0.7229	0.7734	0.7162	0.7606	0.7605	0.7606
ACC	0.8185	0.8188	0.8193	0.8113	0.8100	0.8183	0.8190	**0.8230**	0.8226	**0.8231**
Recall	0.3351	0.3069	0.3031	0.3061	0.2277	0.3381	0.3069	0.3587	0.3526	**0.3604**
Precision	0.6707	0.6948	0.7019	0.6452	0.7035	0.6776	0.6960	0.6816	**0.6838**	0.6789
Type I error	0.0460	0.0377	0.0360	0.0471	**0.0268**	0.0471	0.0375	0.0469	**0.0456**	0.0482
Type II error	0.6648	0.6930	0.6968	0.6938	0.7722	0.6618	0.6930	0.6412	0.6473	**0.6359**

Although the improvements are subtler than those for the German and Australian datasets, the approaches based on consensus and Route 5 once again proved to be the most robust. They optimized the balance between the key metrics while maintaining better error rates. Overall, for the Taiwan dataset, all models were closely matched; however, Route 5 with BIRCH performed slightly better.

Analysis of the three datasets allowed us to draw several conclusions. Although the results differed from those presented in the reference study, they showed a similar overall trend, leading to a conclusion closely aligned with that of the reference study.

The adoption of the consensus model clearly improves the performance of individual models. Furthermore, applying dataset clustering not only improves the performance of each individual model but also enables additional performance gains when combined with the consensus strategy. In addition, the Matthews correlation coefficient (MCC) values of the individual supervised models increased significantly when the consensus model and dataset clustering were used together.

8. Robustness Clustering Analysis (Binary Classification)

This analysis focuses on binary classification and, as such, does not extend to multiclass data. However, for multiclass classification, the results were mainly analyzed in terms of overall performance and qualitative comparison.

Clustering plays an essential role in representing and grouping data. Analyzing the quality of clustering and its ability to accurately represent data is crucial because this step can have a direct impact on how classifiers interact with data and, consequently, on their performance. This part of the study examines the clustering process in the model. The central question addressed was whether the performance of cluster-based models depends on the quality of the clustering.

Furthermore, this study examines whether there is a correlation between the cluster quality and the observed performance of the model. To assess this correlation, we compared the clusters obtained with the original class labels, thus analyzing the distribution of class instances among the clusters. The silhouette score [14] is used as a measure of cluster quality, reflecting the degree of separation between clusters. The Adjusted Rand Index (ARI) [15] was used to measure the similarity between the clustering results and the true class labels, thus providing information on the alignment of clusters with the classes.

8.1. Clustering Metrics Comparison

We calculated the metrics and performed a comparison between the obtained clusters and the original class labels after determining the number k of optimal clusters for each dataset. The results for the Australian training set are presented in Table 11.

Table 11. Australian training set clustering methods comparison.

Australian Dataset Yes (1) No (0)					
Clustering Methods	**Number of Clusters**	**Silhouette Score**	**Adjusted Rand Score**	**Nber of Obs. in Each Class**	**Nber of Obs. in Each Cluster**
k-means	2	0.9598	0.0024	0: 296 1: 256	0: 548 1: 4
Agglomerative	2	0.8968	0.0012	0: 296 1: 256	0: 550 1: 2
Birch	2	0.5059	0.0024	0: 296 1: 256	0: 548 1: 4

The following are comments on the results obtained for the German and Taiwanese training sets.

German training set:

The results of the clustering methods for the German training set are listed in Table 12.

Table 12. German training set clustering methods comparison.

German Dataset POSITIVE (1); NEGATIVE (0)					
Clustering Methods	**Number of Clusters**	**Silhouette Score**	**Adjusted Rand Score**	**Nber of Obs. in Each Class**	**Nber of Obs. in Each Cluster**
k-means	2	0.7119	0.0450	0: 238 1: 562	0: 149 1: 651
Agglomerative	2	0.7461	0.0192	0: 238 1: 562	0: 790 1: 10
Birch	2	0.7139	0.0480	0: 238 1: 562	0: 662 1: 138

For k-means , the silhouette score obtained was 0.7119, indicating a good separation between the clusters. However, the ARI was only 0.0450, indicating a very weak correspondence between the clusters and the true classes. Table 13 shows that the majority of observations in the two classes are unbalanced within the clusters.

Table 13. Correspondence matrix of German training set for k-means.

↓ Class/Cluster →	**0**	**1**	**Sum**
0	59	179	238
1	90	472	562
Sum	149	651	800

For the Agglomerative (linkage = "average"), the silhouette score was even higher, at 0.7461, showing a very clear separation between clusters. However, the ARI was only 0.0192, reflecting a very low fidelity to real classes. Table 14 shows that almost all observations were grouped together in a single cluster, reducing the usefulness of this method to reflect the real structure of the classes.

Table 14. Correspondence matrix of German training set for Agglomerative.

↓ Class/Cluster →	**0**	**1**	**Sum**
0	231	559	790
1	7	3	10
Sum	238	562	800

For BIRCH , the silhouette score reached 0.7139, again indicating good separation between clusters. However, the ARI remains low, at 0.0480, again reflecting poor correspondence with the original classes. Table 15 shows that, although the observations are distributed slightly more evenly between the clusters than in the previous case, they do not perfectly map to their actual classes.

Table 15. Correspondence matrix of German training set for BIRCH.

↓ Class/Cluster →	0	1	Sum
0	182	480	662
1	56	82	138
Sum	238	562	800

Although all three showed high silhouette scores, indicating good separation between clusters, the very low ARI values revealed a poor match with actual classes. This suggests that the structures identified by the algorithms do not faithfully reflect the original classification.

In summary, although all three methods produced well-separated clusters according to the silhouette score, they failed to capture the true structure of the classes (low ARI). This reveals the limitations of unsupervised clustering in this particular case.

Taiwan training set:

The results of the clustering methods for the Taiwanese dataset are presented in Table 16. Although each of these clustering methods showed relatively good separation between clusters, none of them convincingly matched the actual classes, as revealed by the low ARI scores. The results suggest that the cluster structure does not fully capture the distribution of classes.

Table 16. Taiwan training set clustering methods comparison.

		Taiwan Dataset Yes (1) No (0)			
Clustering Methods	**Number of Clusters**	**Silhouette Score**	**Adjusted Rand Score**	**Nber of Obs. in Each Class**	**Nber of Obs. in Each Cluster**
k-means	2	0.5539	−0.0177	0: 18677 1: 5323	0: 20701 1: 3299
Agglomerative	2	0.8968	−0.0003	0: 18677 1: 5323	0: 23999 1: 1
Birch	2	0.5059	−0.0089	0: 18677 1: 5323	0: 20568 1: 3432

For k-means:

The silhouette score obtained was 0.5539, indicating moderate separation between the clusters. This means that overall, the points are closer to their assigned cluster than to neighboring clusters. However, the ARI, which was −0.0177, indicated that the clusters generated by k-means did not align with the original classes (Table 17).

Table 17. Correspondence matrix of Taiwan training set for k-means.

↓ Class/Cluster →	0	1	Sum
0	15,997	4704	20,701
1	2680	619	3299
Sum	18,677	5323	24,000

For the Agglomerative (linkage = "average"), the silhouette score was 0.8968, indicating a very clear separation between the clusters. However, the ARI was −0.0003, indicating once again that there was no real correspondence with the actual classes. As shown in Table 18, Cluster 0 contained the majority of Class 0 observations (18,677), and only one observation of Class 1, whereas Cluster 1 contained 5323 observations of Class 0 and no observation of Class 1.

Table 18. Correspondence matrix of Taiwan training set for Agglomerative.

↓ Class/Cluster →	0	1	Sum
0	18,676	5323	23,999
1	1	0	1
Sum	18,677	5323	24,000

For BIRCH , the silhouette score was 0.5059, indicating lower separation compared to the other methods. The ARI, which is −0.0089, also shows poor alignment with the real classes. The obtained clusters contained 18,677 and 5323 observations. As shown in Table 19, Cluster 0 mainly contained Class 0 observations (15,949) and Class 1 (4619). Cluster 1 also contains a mixture of the two classes (2728 from Class 0 and 704 from Class 1), although the separation is less marked than that with k-means and Agglomerative.

Table 19. Correspondence matrix of Taiwan training set for BIRCH.

↓ Class/Cluster →	0	1	Sum
0	15,949	4619	20,568
1	2728	704	3432
Sum	18,677	5323	24,000

Although the three clustering methods applied to the three datasets provided reasonable cluster separations, all of them showed shortcomings in their correspondence with actual classes.

Figure 9 shows the conflicting clusters after the k-means clustering. The clusters are colored purple and yellow (circles), respectively. Correctly formed clusters correspond to those groups that have good alignment with the true labels (shown with blue crosses). By contrast, conflicting clusters contain data points belonging to different true labels (represented by red crosses within the same cluster), indicating separation errors or conflicts in the cluster assignment. Conflicting clusters after Agglomerative and BIRCH clustering of German and Australian datasets are shown in Figures A1 and A2 in the Appendix A.

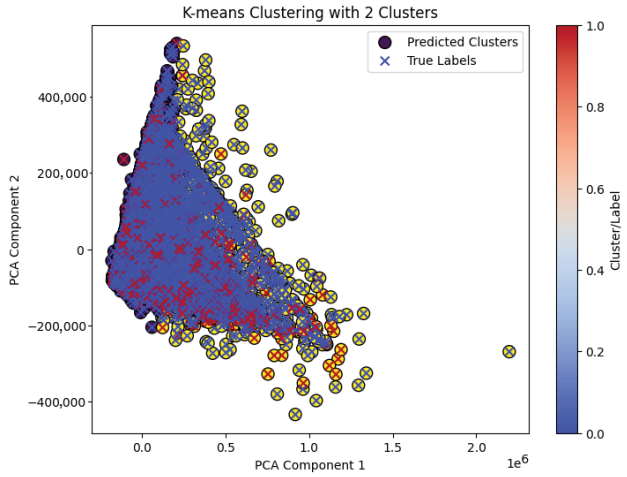


Figure 9. Conflicting clusters of Taiwan dataset after k-means.

8.2. Model Performance Regarding Clustering Metrics

The objective was to analyze the evolution of the metrics as a function of the silhouette scores and ARI for each dataset. We would like to determine whether better clustering corresponds to better model performance.

The analysis in Table 20 reveals a clear relationship between clustering quality and performance on other evaluation metrics, such as MCC, AUC, precision, and overall ACC. The clustering quality improved; noticeable improvements were observed in these performance metrics for all models across the datasets. The Australian dataset showed an exceptionally high silhouette score of 0.9598, corresponding to an ARI of 0.0024, indicating the best overall performance for the evaluation metrics in this study. The Australian dataset provided the best improvements in MCC, AUC, and ACC. This improvement demonstrates that the good quality of the clustering is ultimately beneficial for the model's performance.

Table 20. k-means performance regarding silhouette score and ARI.

					k-means				
Datasets	**Silhouette Score**	**ARI**	**MCC**	**AUC**	**ACC**	**Recall**	**Precision**	**Type I Error**	**Type II Error**
Taiwan	0.5539	−0.0177	**0.4038**	**0.7606**	0.823	0.3587	**0.6816**	0.0469	0.6412
German	0.7119	0.0450	**0.5176**	**0.7851**	0.805	0.9275	**0.8152**	0.4677	0.0724
Australian	0.9598	0.0024	**0.8138**	**0.9118**	0.9130	0.8039	**0.9534**	0.0229	0.1960

As shown in Table 21, a higher clustering quality of the Agglomerative is associated with better performance on several different evaluation metrics, including MCC, AUC, precision, and overall ACC.

Table 21. Agglomerative performance regarding silhouette score and ARI.

					Agglomerative				
Datasets	**Silhouette Score**	**ARI**	**MCC**	**AUC**	**ACC**	**Recall**	**Precision**	**Type I Error**	**Type II Error**
hlTaiwan	0.4057	−0.0089	**0.4057**	**0.7606**	0.8231	0.3604	**0.6789**	0.0482	0.6359
German	0.7139	0.0480	**0.5568**	**0.7859**	0.82	0.9420	**0.8227**	0.4516	0.0579
Australian	0.9598	0.0024	**0.8291**	**0.9118**	0.9202	0.8235	**0.9545**	0.0229	0.1764

The same trend was observed for the BIRCH clustering, as shown in Table 22, which indicates that better clustering quality is linked to better predictive performance, as shown by the results of the k-means and Agglomerative clustering approaches.

Table 22. BIRCH performance regarding silhouette score and ARI.

					Birch				
Datasets	**Silhouette Score**	**ARI**	**MCC**	**AUC**	**ACC**	**Recall**	**Precision**	**Type I Error**	**Type II Error**
Taiwan	0.8968	−0.0003	**0.4011**	**0.7605**	0.8226	0.3526	**0.6838**	0.0456	0.6473
German	0.7461	0.0192	**0.5708**	**0.7822**	0.825	0.9347	**0.8322**	0.4193	0.0652
Australian	0.9652	0.0012	**0.8765**	**0.9107**	0.9420	0.8627	**0.9777**	0.0114	0.1372

Thus, although the silhouette scores provide an indication of the quality of cluster separation, they are not sufficient to guarantee that the clustering corresponds to the structure of the real classes. The improved performance we observe arises not from accurate class replication (as measured by ARI), but from the ability of clustering to highlight local data patterns that individual classifiers learn differently.

9. Case of Multiclass Classification

9.1. Description of Credit Score Multiclass Dataset

The Credit Score Classification dataset [43] used contains 100,000 instances divided into three classes: poor (0) 28,998 (29%), standard (1) 53,174 (53%), and good (2) 17,828 (18%). It comprises 28 variables (27 explanatory + 1 target). After selecting the characteristics, 20 variables were retained, with a slight improvement in accuracy (from 0.7872 to 0.7892). Missing values and outliers are also considered. The dataset was unbalanced, with less representation of the "good" class. This dataset was used to extend the proposed approach to multiclass classifications. For the number k of clusters we chose the length of the class (3).

9.2. Experimental Results

The results obtained using the multiclass credit score dataset are presented in Tables 23 and 24. A comparative analysis of the different classification approaches (Routes 1 to 5) revealed the superiority of hybrid approaches that integrate unsupervised techniques and SOM. The individual models (Route 1) performed the worst overall on all indicators (MCC, AUC, ACC, recall, precision, and Type I and II errors), with MCC values below 0.66 for all models except RF and k-NN. Conversely, Routes 2 to 5 show a significant improvement, particularly Routes 4 and 5, which incorporate both SOM consensus models and clustering techniques. Route 2 (SOM consensus) stands out for its excellent overall performance (MCC, 0.6674; AUC, 0.9084; ACC, 0.7979), which is often close to or equal to that obtained with Routes 4 and 5. However, Routes 4 and 5 refine the classification slightly, particularly for certain sensitive classes (reduction in Type I and II errors). This improvement remains modest but significant in contexts where class accuracy and robustness against false negatives are critical. Therefore, although Route 2 is already very powerful, Route 5, particularly in combination with BIRCH, offers the most complete solution for multiclass processing of complex data. Route 5 (BIRCH and SOM) achieved the best values for MCC (0.6702), AUC (0.9067), and ACC (0.7970), while maintaining a good balance between recall (0.7958) and accuracy (0.7875). In terms of errors, this configuration showed the best results for Type I errors for Class 2 (0.0534), as well as low Type II errors, particularly for Classes 1 and 2. These results confirm that integrating clustering techniques with SOM enables more relevant structuring of data, leading to an overall improvement in classification performance, particularly in the complex context of multiclassing. Although some isolated models (e.g., GBDT, RF, or k-NN in Route 1 or 3) show good results over time, they struggle to maintain the overall stability across all classes. Conversely, the SOM-based consensus models (Routes 2, 4, and 5) delivered robust and balanced performance, confirming the value of these hybrid approaches for processing complex multiclass data.

Table 23. Model performance for All Routes—Credit Score Multiclass Dataset (Part 1).

Dataset	**R1: Individual Models**							**R2: SOM Cons.**
Metric	**ANN**	**DT**	**GBDT**	**k-NN**	**LR**	**RF**	**SVM**	
MCC	0.5138	0.5218	0.6312	0.6535	0.3780	0.6571	0.5443	**0.6674**
AUC	0.8511	0.8446	0.9001	**0.9138**	0.7913	**0.9172**	0.7259	0.9084
ACC	0.7078	0.7103	0.7790	0.7905	0.6402	**0.7937**	0.7259	0.7979
Recall	0.7078	0.7103	0.7790	0.7905	0.6402	0.7937	0.7259	0.7925
Precision	0.7082	0.7136	0.7786	0.7908	0.6382	0.7935	0.7270	0.7861
Type I error (0)	0.2846	0.2721	0.2232	0.2136	0.3298	0.2125	0.2565	**0.0993**
Type I error (1)	0.2597	0.2495	**0.2085**	**0.1898**	0.3542	**0.1894**	0.2434	**0.1898**
Type I error (2)	0.3994	0.4207	0.2568	0.2575	0.4375	0.2475	0.3891	**0.0517**

Table 24. Model performance for All Routes—Credit Score Multiclass Dataset (Part 2).

Dataset	R3: k-Means Cluster Models							R4: k-Means SOM	R5: Cluster-Based SOM	
Metric	**ANN**	**DT**	**GBDT**	**k-NN**	**LR**	**RF**	**SVM**		**Agglom.**	**Birch**
MCC	0.5074	0.5410	0.6242	0.6525	0.3772	0.6537	0.5446	**0.6713**	0.6685	**0.6702**
AUC	0.8509	0.8263	0.8976	0.9136	0.7901	0.9142	0.8594	0.9065	0.9063	0.9067
ACC	0.7017	0.7241	0.7750	0.7898	0.6398	0.7920	0.7259	0.7972	0.7970	**0.7997**
Recall	0.7017	0.7241	0.7750	0.7898	0.6398	0.7920	0.7270	**0.7998**	0.7952	0.7958
Precision	0.7071	0.7241	0.7746	0.7901	0.6377	0.7916	0.7259	0.7820	0.7830	**0.7875**
Type I error (0)	0.2631	0.2661	0.2229	0.1857	0.3302	0.2139	0.2588	0.1117	0.1067	**0.0950**
Type I error (1)	0.2627	0.2447	**0.2121**	**0.2121**	0.3546	**0.1921**	0.2425	**0.1637**	0.1757	**0.1895**
Type I error (2)	0.4328	0.3857	0.2688	0.2571	0.4381	0.2476	0.3876	0.0568	0.0545	**0.0534**
Type II error (0)	0.3561	0.2745	0.2374	**0.1857**	0.4657	**0.1918**	0.2951	**0.1312**	0.1440	**0.1678**
Type II error (1)	0.2557	0.2403	**0.1956**	0.2080	0.2309	**0.1910**	0.2352	**0.2303**	**0.2198**	**0.2053**
Type II error (2)	0.3297	0.3847	**0.2920**	**0.2571**	0.5724	**0.2857**	0.3555	**0.2387**	**0.2503**	**0.2392**

9.3. Comparison with Binary Classification

Hybrid approaches that integrate clustering techniques with SOM, particularly Route 5, offer the best performance for binary and multiclass classifications. Although the gains are more pronounced in binary classification (particularly on the German and Australian datasets), multiclass classification also benefits from a notable improvement, with better stability and a good balance between metrics.

10. Discussion and Limitations

The results obtained herein confirm the effectiveness of the proposed method and extend and improve the reference work [10]. Reference restricted the feasible search space through early classifier restrictions, which reduces the ability of the model to explore potential solutions. By contrast, our method broadens the search space, providing more flexibility for the model to identify optimal outcomes (see Table A2 in the Appendix B). Through iterative improvements, the algorithm has sufficient time to optimize the weights and converge to the best possible values. In addition, a grid search before re-fitting ensures that the model is retrained on a diverse dataset with the best hyperparameters to maximize accuracy.

We also grouped all datasets into $k = 2$ (binary class) clusters based on the elbow method and confirmed this choice using the silhouette score, a metric that quantifies how well each data point fits into its assigned cluster. It is worth noting that both the elbow method and silhouette tend to favor low k, especially in imbalanced datasets. This preference for a low k can lead to an oversimplification of the data, where important underlying patterns may be missed, particularly in more complex datasets. Consequently, the generalization ability of the model may be compromised, as the clustering process might fail to capture the full variability in the data, potentially limiting the robustness of the consensus model. By contrast, the reference study chose $k = 5$ for the German and Australian datasets. However, using a higher number of clusters can result in fragmented and less meaningful clusters, reducing the overall clustering quality and complicating the analysis. For $k = 2$, our model offers a simpler, more interpretable, and more robust structure.

Analysis of the results across different models and configurations for Routes 1, 2, 3, 4, and 5 reveals key insights into the strengths and weaknesses of the classical and cluster-based approaches. Although each dataset has its own particularities, Route 2, 4 and 5 have demonstrated the ability to deliver the best performance. This confirmed that the integration of clustering techniques with SOM represents a particularly robust and effective approach for processing complex data. SOM provides insights that go beyond noise reduction. SOM does not simply create arbitrary partitions, but groups prediction vectors according to their joint structure, thus enabling majority voting informed by classifier agreement rather than independent votes.

This confirmed that the integration of clustering techniques with SOM represents a particularly robust and effective approach for processing complex data.

This study also analyses the relationship between clustering quality (measured by silhouette score and ARI) and classification model performance, in order to determine whether better clustering correspond to better predictive performance. The results obtained from the datasets highlight the crucial role of integrating clustering techniques, especially hybrid approaches with self-organizing maps (SOMs). The integration of the SOM model with clustering techniques notably enhances the classification performance compared with individual models, thereby emphasizing the importance of consensus in improving precision, recall, ACC, and AUC. Hybrid approaches, such as those combining k-means, BIRCH, or Agglomerative clustering with SOM, have yielded promising results, confirming that the integration of clustering techniques contributes to improved data structuring and more effective detection of positive classes. Among the various configurations tested, Route 5 was the most effective, particularly when employing Agglomerative or BIRCH clustering, offering the best balance between precision, recall, and error rates. Furthermore, the aggregation of SOM model predictions provides greater stability and robustness in the results, leading to an overall performance improvement compared with clustering models alone. A high silhouette score and a poor ARI (close to zero) indicate that the clustering algorithm has found well-separated clusters that do not correspond to the target classes. Therefore, the benefit of clustering may come from discovering alternative data structures, not from alignment with true class boundaries.

The hyperparameters of hierarchical clustering methods have a significant influence on the results. Although we used default values for our tests, we recognize that adjusting these hyperparameters, such as the distance threshold, branching factor, or number of clusters, could improve the quality and robustness of the clustering, depending on the specific characteristics of the data. We suggest that further hyperparameter testing be carried out to assess the impact of different settings on overall performance.

Our experiments have demonstrated significant improvements in model performance across multiple datasets and classifiers. However, a major limitation of our study is the absence of statistical significance tests and uncertainty quantification to validate these improvements. Indeed, the absence of such analyses prevents us from rigorously confirming whether the observed differences are statistically significant or whether they may be due to random fluctuations. Although the use of performance improvements across different datasets and classifiers has been adopted as an indicator of robustness, we recognize that this approach is not a substitute for formal significance tests.

Also the choice of 80/20 ratio for dividing the data into training and test sets was adopted to prioritize model training and for simplicity, it is possible that a relatively small test set (20%) may not sufficiently reflect the diversity of the data, which could undermine the robustness of the model evaluations. Distributions such as 70/30 or 60/40 would likely offer a better compromise, allowing for both more representative training and a more reliable evaluation of the model's performance on unseen data. To overcome this limitation, we used 5-fold cross-validation, which strengthens the robustness of the model selection process. However, it remains relevant to explore different data splitting strategies and validate their impact on the overall performance of the model.

11. Implications and Contributions

This research demonstrated the general strengths of Self-Organizing Maps to model complex data, particularly in consensus configurations (Route 2) and approaches based on clustering techniques (Routes 4 and 5). These methodologies contribute to a significant im-

provement in classification performance, particularly when the data present complex internal structures that are difficult to obtain with conventional supervised learning algorithms.

By organizing data through SOM, the models benefit from improved structure recognition, which translates into better classification metrics such as MCC, AUC, and ACC. The consensus strategy increases the model stability and robustness, whereas the integration of clustering techniques enables a more meaningful segmentation of the data before classification. These findings support the hypothesis that clustering quality reflected by high silhouette scores and appropriate ARI can directly influence model effectiveness.

The use of SOM to organize data before classification proves to be especially relevant in real-world scenarios, such as credit scoring or fraud detection, where balancing multiple performance indicators is critical. In these domains, high recall is essential to effectively detect positive cases. However, minimizing Type I and Type II errors is equally important to avoid false alarms or missed detections.

In addition, this study emphasizes the importance of cost-sensitive evaluation in areas such as credit scoring, where classification errors have significant financial consequences. Although we did not include metrics such as the F1-score, which combines both precision and recall, we chose to use the MCC (Matthews Correlation Coefficient), a metric that, like the F1-score, is sensitive to false positives and false negatives. The MCC proves particularly robust in situations of class imbalance, as it considers all values in the confusion matrix (including true positives, false positives, true negatives, and false negatives). However, specific cost-sensitive metrics such as expected loss, misclassification cost, and profit-based criteria are essential for a more accurate and cost-sensitive evaluation of models. Although these aspects are outside the scope of this replication and extension study, we emphasize that they are of great importance for the practical implementation of models in credit scoring systems.

This research also demonstrated the importance of cluster structure in model performance, from a practical point of view. Datasets with better-defined clusters tend to produce better classification outcomes when used in hybrid models. This implies that the quality of the clustering has a direct impact on the model's ability to classify data correctly; therefore, clustering should not be treated only as a preprocessing step but as a strategic component of the modeling pipeline.

Compared to other paradigms such as reinforcement learning, rule-based learning, neuro-symbolic AI, and neuromorphic engineering, our approach stands out for its ability to efficiently process static tabular data and capture complex non-linear relationships between borrower characteristics. Although each paradigm has its own advantages and limitations, the hybrid approach we propose is suited to credit scoring data and the needs of financial institutions. Furthermore, our methodology offers a significant economic advantage over alternative paradigms. It delivers improved performance with low development, implementation and execution costs. Our method uses algorithms (k-means clustering, hierarchical clustering, SOM, standard classifiers) that are well understood, stable and widely available in open-source libraries such as Scikit-learn. It can be run on standard hardware (CPU) without the need for specialised infrastructure. It requires expertise in 'classical' data science, a more common and less expensive resource than expertise in RL or neuromorphic engineering.

Once trained, the model is very fast to execute. The prediction for a new applicant consists of assigning them to a cluster, calculating the classifiers' prediction on the enriched data, and then aggregating the results via SOM consensus. This computationally inexpensive process can easily be scaled to process thousands of credit applications on standard infrastructure. This could facilitate its adoption by institutions without excessive investment. Even a marginal improvement in scoring accuracy (reduction in false positives and false negatives) translates into direct and substantial savings in terms of

reduced payment defaults and financial losses. The development and execution costs are low compared to these potential benefits. The proposed framework can be extended to other structured datasets for credit scoring. Nonetheless, the performance outcomes may vary due to differences in dataset characteristics such as feature heterogeneity, class imbalance, or sample size.

12. Conclusions and Future Directions

Recent literature has reached a consensus on the fact that hybrid methods are consistently superior to standalone methods in assessing credit risk. this work extends this paradigm by examining how the use of various clustering techniques affects ultimate classification effectiveness. The benefits of applying clustering and consensus techniques to classification were verified. Integrating a clustering method and a classifier is critical because it determines how the performance of the model will be better or worse. The effectiveness of clustering affects the performance of the classifier in terms of making correct predictions. A good clustering process can support the classifier, whereas poor clustering can seriously harm the classifier when making predictions. Thus, careful consideration should be given to the clustering phase because it ensures that the integrated model achieves the best results. This implies that attention should be given in practice to (i) the quality of cluster separations in preprocessing phases, (ii) the choice of the clustering algorithm in such a way that it matches the structure of the data, and (iii) using consensus models in complex and unbalanced contexts as robust solutions.

While this study provides a thorough understanding of the various clustering algorithms, it remains limited to the theoretical aspects of the knowledge without fully consider the practical experience or skills needed to apply them effectively in real-world contexts. The impact of these algorithms in complex situations remains to be explored.

Future research will include statistical tests significance (e.g., McNemar and bootstrap confidence intervals), uncertainty quantification to determine more reliably whether the observed performance improvements are truly significant. These analyses will validate the conclusions of this study and provide a more rigorous assessment of the impact of different approaches on model performance. Also treating cases where there is a high separation between clusters (high silhouette score) and strong alignment with real classes (high Adjusted Rand Index) could lead to a better performance. Another study integrated other clustering algorithms, such as DBSCAN combined with SOM models, to obtain better clustering results. In addition, it would be interesting to use other complex data types, such as time series, to improve the existing knowledge on the behavior of clustering and the SOM model in different environments.

Author Contributions: Conceptualization, R.G.A.; methodology, R.G.A.; software, R.G.A.; validation, R.G.A.; formal analysis, R.G.A., T.S., I.M. and M.A.; investigation, R.G.A.; resources, R.G.A.; data curation, R.G.A.; writing—original draft preparation, R.G.A.; writing—review and editing, T.S., I.M. and M.A.; visualization, R.G.A. and M.A.; supervision, T.S., I.M. and M.A.; project administration, R.G.A., T.S., I.M. and M.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data supporting the results of this study can be found in the publicly available datasets referenced in the references section.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ACC	Accuracy
ANN	Artificial Neural Network
ARI	Adjusted Rand Index
AUC	Area Under the Curve
BIRCH	Balanced Iterative Reducing and Clustering using Hierarchies
DT	Decision Tree
GBDT	Gradient Boosting Decision Tree
IoT	Internet of Things
IQR	Interquartile Range
k-NN	k-Nearest Neighbors
LR	Logistic Regression
MCC	Matthews Correlation Coefficient
MLP	Multi-Layer Perceptron
MV	Majority Voting
RF	Random Forest
ROC	Receiver Operating Characteristic
SOM	Self-Organizing Map
SVM	Support Vector Machine

Appendix A

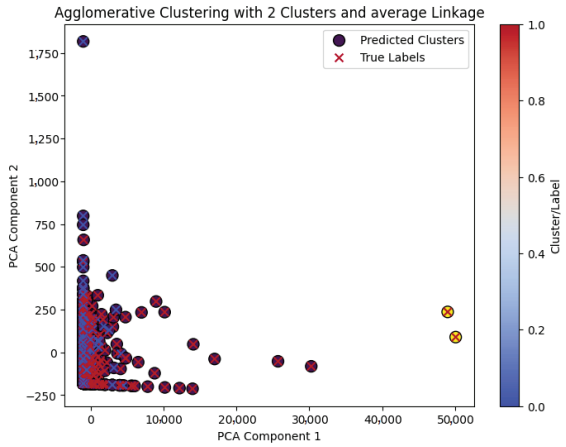


Figure A1. Conflicting clusters of the Australian dataset after Agglomerative clustering.

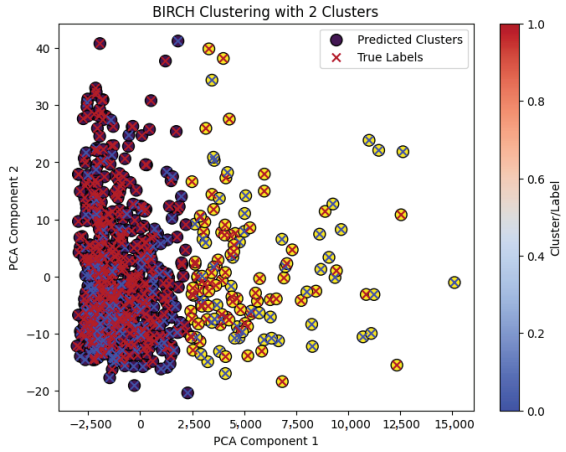


Figure A2. Conflicting clusters of the German dataset after BIRCH clustering.

Appendix B

Table A1. Metrics used in this study.

Abbreviation	Description
MCC	Balance measure that considers true and false positives and negatives.
AUC	Measures the model's ability to distinguish between classes.
ACC	Measures the proportion of correct predictions made by the model.
Recall	Measures the proportion of actual positives that were correctly identified.
Precision	Indicates the proportion of positive cases that were correct.
Type I error	False positive rate: some negative cases are incorrectly predicted as positive.
Type II error	False negative rate: the model misses some positive cases.

Table A2. Hyperparameter tuning.

Model	Baseline Paper	Research	Best Parameters Found in GridSearchCV
ANN (MLP)	ann_params = { 'hidden_layer_sizes': [(i,) for i in range (10, 51)], # hidden_nodes from 10 to 50 'alpha': [0.001, 0.01, 0.1, 1] }	mlp = MLPClassifier(max_iter=2300) param_grid_mlp = { 'hidden_layer_sizes': [(50,), (100,), (50, 50)], 'activation': ['relu', 'tanh'], 'solver': ['adam'], 'alpha': [0.0001, 0.001, 0.01], 'learning_rate_init': [0.001, 0.01, 0.1] }	{'activation': 'relu', 'alpha': 0.0001, 'hidden_layer_sizes': (50,), 'learning_rate_init': 0.1, 'solver': 'adam'}
Decision Tree	dt_params = { 'max_depth': range (5, 16), 'min_samples_split': range (5, 21) }	dt = DecisionTreeClassifier() param_grid_dt = { 'max_depth': [None, 10, 20, 30], 'min_samples_split': [2, 5, 10], 'min_samples_leaf': [1, 2, 4], 'criterion': ['gini', 'entropy'] }	{'criterion': 'gini', 'max_depth': 10, 'min_samples_leaf': 1, 'min_samples_split': 2}
GBDT	gbdt_params = { 'n_estimators': range (100, 501, 100), 'max_depth': range (5, 26), 'learning_rate': [0.001, 0.01, 0.1, 1] }	gb = GradientBoostingClassifier() param_grid_gb = { 'n_estimators': [50, 100, 200, 300], 'learning_rate': [0.01, 0.1, 0.5, 1.0], 'max_depth': [3, 5, 7, None] }	{'learning_rate': 0.01, 'max_depth': 3, 'n_estimators': 300}
k-NN	knn_params = { 'n_neighbors': range (3, 11) }	param_grid_knn = { 'n_neighbors': [3, 5, 7, 10], 'weights': ['uniform', 'distance'], 'p': [1, 2] }	{'n_neighbors': 10, 'p': 1, 'weights': 'uniform'}
Logistic Regression	Not mentioned	lr = LogisticRegression(solver = 'liblinear') param_grid_lr = { 'C': [0.01, 0.1, 1, 10], 'penalty': ['l1', 'l2'], 'solver': ['liblinear']}	{'C': 1, 'penalty': 'l2', 'solver': 'liblinear'}
Random Forest	rf_params = { 'n_estimators': range (100, 501, 100), 'min_samples_leaf': range (3, 11), 'max_depth': range (5, 26) }	rf = RandomForestClassifier(random_state=42) param_grid_rf = { 'n_estimators': [50, 100, 200, 300], 'max_depth': [None, 10, 20, 30], 'min_samples_split': [2, 5, 10], 'min_samples_leaf': [1, 2, 4], 'bootstrap': [True, False] }	{'bootstrap': False, 'max_depth': 20, 'min_samples_leaf': 1, 'min_samples_split': 5, 'n_estimators': 300}
SVM	svm_params = { 'C': [2**i for i in range (−10, 11)], 'gamma': [2**i for i in range (−10, 11)] }	svm = SVC (probability = True) param_grid_svm = { 'C': [0.1, 1, 10, 100], 'kernel': ['linear', 'poly', 'rbf', 'sigmoid'], 'degree': [2, 3, 4], 'gamma': ['scale', 'auto'] {'C': 1, 'degree': 2, 'gamma': 'scale', 'kernel': 'rbf'}	
k-Means Clustering	kmeans_params = { 'n_clusters': [8] # for Chinese P2P credit dataset 'n_clusters': [5] # For German and Australian datasets}	'n_clusters': [2] # For all datasets	-

Appendix C. Illustrative Example of Route 5

This appendix provides an illustrative step-by-step process of Route 5 using a simple set of data. This example is to show the key mechanics of this approach and how it then feeds into a supervised classification, and how that result is combined with the Self Organising Map (SOM). The purpose is to provide a clear understanding of this approach

without reproducing all components of our computational process. For empirical validation and detailed results, refer to Sections 6–9.

Appendix C.1. Credit dataset and Train-Test Split

Consider a small sample of 8 credit applicants, each described by three normalized features: F1: Income Level, F2: Credit History Score, F3: Debt Ratio, and target: Credit Risk → Good (1) or Bad (0). The dataset is split into a training set and a test set as follows:

Table A3. Credit dataset with features and true labels.

Applicant	F1 (Income Level)	F2 (Credit History)	F3 (Debt Ratio)	True Label (Target)
A1	0.8	0.9	0.1	1
A2	0.7	0.8	0.2	1
A3	0.3	0.4	0.8	0
A4	0.2	0.3	0.9	0
A5	0.6	0.5	0.6	0
A6	0.9	0.7	0.3	1

Test set (2 applicants):

Table A4. Test dataset with features and true labels.

Applicant	F1 (Income Level)	F2 (Credit History)	F3 (Debt Ratio)	True Label
A7	0.5	0.6	0.5	0
A8	0.8	0.8	0.2	1

Note: The dataset shows class overlaps with challenging cases (e.g., A5 has mixed characteristics).

Appendix C.2. Hierarchical Clustering and Feature Enrichment (Training Phase)

We apply Agglomerative clustering on the training set to group similar applicants without using the labels. The algorithm identifies two clusters:

- Cluster 0: {A1, A2, A6} → Applicants with higher income and better credit history.
- Cluster 1: {A3, A4, A5} → Applicants with lower income and higher debt.

Extension of training dataset with cluster feature:

Table A5. Extended training dataset with clusters and true labels.

Applicant	F1 (Income Level)	F2 (Credit History Score)	F3 (Debt Ratio)	Cluster	True Label
A1	0.8	0.9	0.1	0	1
A2	0.7	0.8	0.2	0	1
A3	0.3	0.4	0.8	1	0
A4	0.2	0.3	0.9	1	0
A5	0.6	0.5	0.6	1 *	0
A6	0.9	0.7	0.3	0	1

* The result of the clustering put A5 in the cluster 1, rather than cluster 0.

Appendix C.3. Supervised Classification on Extended Data (Training Phase)

We train individual classifiers on the extended training dataset:

- Classifier CL1: (e.g., Decision Tree).

- Classifier CL2: (e.g., Logistic Regression).

Individual predictions:

Table A6. Individual classifier predictions on the extended training dataset.

Applicant	CL1 (Decision Tree)	CL2 (Logistic Regression)	True Label
A1	1	1	1
A2	1	1	1
A3	0	0	0
A4	0	0	0
A5	0 *	1 *	0
A6	1	1	1

* The result of CL1 predicted A5 as 0 and CL2 as 1.

Training classifier performance:

- CL1 Training accuracy: 5/6 = 0.833.
- CL2 Training accuracy: 5/6 = 0.833.

Appendix C.4. Test Set Processing and SOM Consensus (Testing Phase)

Step 1: Assign test applicants to clusters using the clustering model trained on the training data:

- A7 → Cluster 1 (characteristics similar to the lower-income group).
- A8 → Cluster 0 (characteristics similar to the higher-income group).

Step 2: Individual classifier predictions on the test set:

Table A7. Individual classifier predictions on the test dataset.

Applicant	CL1 (Decision Tree)	CL2 (Logistic Regression)	True Label
A7	0	1	0
A8	1	1	1

Step 3: SOM Consensus method on test predictions:

Table A8. SOM Consensus on the test dataset.

Applicant	CL1	CL2	SOM unit	Majority vote	Final Label	True Label
A7	0	1	U1	1 *	1	0
A8	1	1	U2	1	1	1

* The Majority vote consensus predicted A7 as 1.

Appendix C.5. Performance Evaluation on Test Set

Confusion matrix (test set):

Table A9. Confusion matrix.

	Predicted Good (1)	Predicted Bad (0)
Actual Good (1)	TP = 1 (A8)	FN = 0
Actual Bad (0)	FP = 1 (A7)	TN = 0

Detailed test performance metrics:

- Accuracy: $(TP + TN)/\text{Total} = (1 + 0)/2 = 0.50$.
- Precision: $TP/(TP + FP) = 1/(1 + 1) = 0.50$.
- Recall: $TP/(TP + FN) = 1/(1 + 0) = 1.00$.
- Type I Error (False Positive): 1.
- Type II Error (False Negative): 0.
- MCC: $\frac{(TP \times TN - FP \times FN)}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} = \frac{0}{\sqrt{0}} =$ none (due to small test set size).

Appendix C.6. Discussion and Link to Our Results

The example above shows how we applied our methodology in practice. The addition of a cluster feature to the dataset extended the original dataset with an additional dimension, allowing for a more thorough analysis of both the original data and the clustering results. The incorporation of cluster memberships during the training of each classifier enabled each model to leverage both the original features of the data as well as those derived from the clustering process. This methodology helped the model integrate the complexity of the data. The consensus from the SOM was critical in resolving differences between classifiers, as seen in A5, and illustrates our method's ability to resolve ambiguity. The results reflect expected outcomes, similar to those obtained empirically in Sections 6–8, with Route 5 showing the highest performance level, though it has not yet reached perfection.

References

1. Kaminskyi, A.; Nehrey, M.; Zomchak, L. Machine learning methods application for consumer banking. In Proceedings of the 9th International Conference on Monitoring, Modeling & Management of Emergent Economy (M3E2 2021), Odessa, Ukraine, 26–28 May 2021 ; EDP Sciences: Paris, France, 2021; Volume 107, p. 12001. [CrossRef]
2. Ayari, H.; Guetari, R.; Kraïem, N. Machine learning powered financial credit scoring: A systematic literature review. *Artif. Intell. Rev.* **2025**, *59*, 13. [CrossRef]
3. Ileberi, E.; Sun, Y.; Wang, Z. Performance Evaluation of Machine Learning Methods for Credit Card Fraud Detection Using SMOTE and AdaBoost. *IEEE Access* **2021**, *9*, 165286–165294. [CrossRef]
4. Chang, V.; Sivakulasingam, S.; Wang, H.; Wong, S.T.; Ganatra, M.A.; Luo, J. Credit Risk Prediction Using Machine Learning and Deep Learning: A Study on Credit Card Customers. *Risks* **2024**, *12*, 174. [CrossRef]
5. Noriega, J.P.; Rivera, L.A.; Herrera, J.A. Machine Learning for Credit Risk Prediction: A Systematic Literature Review. *Data* **2023**, *8*, 169. [CrossRef]
6. Dastile, X.; Celik, T.; Potsane, M. Statistical and machine learning models in credit scoring: A systematic literature survey. *Appl. Soft Comput.* **2020**, *91*, 106263. [CrossRef]
7. Sachan, S.; Yang, J.B.; Xu, D.L.; Benavides, D.E.; Li, Y. An explainable AI decision-support-system to automate loan underwriting. *Expert Syst. Appl.* **2020**, *144*, 113100. [CrossRef]
8. Hand, D.J.; Henley, W.E. Statistical Classification Methods in Consumer Credit Scoring: A Review. *J. R. Stat. Society. Ser. A Stat. Soc.* **1997**, *160*, 523–541. [CrossRef]
9. Abellán, J.; Castellano, J.G. A comparative study on base classifiers in ensemble methods for credit scoring. *Expert Syst. Appl.* **2017**, *73*, 1–10. [CrossRef]
10. Bao, W.; Lianju, N.; Yue, K. Integration of unsupervised and supervised machine learning algorithms for credit risk assessment. *Expert Syst. Appl.* **2019**, *128*, 301–315. [CrossRef]
11. Kohonen, T. *Self-Organizing Maps*; Springer: Berlin/Heidelberg, Germany, 2001.
12. Murtagh, F. A Survey of Recent Advances in Hierarchical Clustering Algorithms. *Comput. J.* **1983**, *26*, 354–359. [CrossRef]
13. Zhang, T.; Ramakrishnan, R.; Livny, M. BIRCH: A New Data Clustering Algorithm and Its Applications. *Data Min. Knowl. Discov.* **1997**, *1*, 141–182. [CrossRef]
14. Rousseeuw, P.J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* **1987**, *20*, 53–65. [CrossRef]
15. Dunn, J.C. Well-Separated Clusters and Optimal Fuzzy Partitions. *J. Cybern.* **1974**, *4*, 95–104. [CrossRef]
16. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
17. Freund, Y.; Schapire, R.E. Experiments with a new boosting algorithm. In Proceedings of the Proceedings of the Thirteenth International Conference on International Conference on Machine Learning, San Francisco, CA, USA, 3–6 July 1996; pp. 148–156.

18. Kuncheva, L.; Whitaker, C. Measures of Diversity in Classifier Ensembles and Their Relationship with the Ensemble Accuracy. *Mach. Learn.* **2003**, *51*, 181–207. [CrossRef]
19. Muniappan, M.; Paruvachi Subramanian, N.D. A Majority Voting Mechanism-Based Ensemble Learning Approach for Financial Distress Prediction in Indian Automobile Industry. *J. Risk Financ. Manag.* **2025**, *18*, 197. [CrossRef]
20. Kohonen, T. The self-organizing map. *Proc. IEEE* **1990**, *78*, 1464–1480. [CrossRef]
21. D'Urso, P.; De Giovanni, L.; Massari, R. Smoothed self-organizing map for robust clustering. *Inf. Sci.* **2020**, *512*, 381–401. [CrossRef]
22. Drobics, M.; Winiwater, W.; Bodenhofer, U. Interpretation of self-organizing maps with fuzzy rules. In Proceedings of the 12th IEEE Internationals Conference on Tools with Artificial Intelligence, ICTAI 2000, Vancouver, BC, Canada, 13–15 November 2000; pp. 304–311. [CrossRef]
23. Ryzhikov, I.; Huovinen, M.; Hiltunen, Y. Increasing interpretability and prediction rate by combining self-organizing maps with modeling algorithms. In Proceedings of the First SIMS EUROSIM Conference on Modelling and Simulation, SIMS EUROSIM 2021, and 62nd International Conference of Scandinavian Simulation Society, SIMS 2021, Virtual Conference, Finland, 21–23 September 2022; pp. 92–98.
24. Hsieh, N.C. Hybrid mining approach in the design of credit scoring models. *Expert Syst. Appl.* **2005**, *28*, 655–665. [CrossRef]
25. Arutjothi, G.; Senthamarai, C. Assessment of Probability Defaults Using K-Means Based Multinomial Logistic Regression. *Int. J. Comput. Theory Eng.* **2022**, *14*, 84–88. [CrossRef]
26. Yu, T.; Huang, W.; Tang, X.; Zheng, D. A hybrid unsupervised machine learning model with spectral clustering and semi-supervised support vector machine for credit risk assessment. *PLoS ONE* **2025**, *20*, e0316557. [CrossRef]
27. Wang, Y.; Jia, Y.; Fan, S.; Xiao, J. Deep reinforcement learning based on balanced stratified prioritized experience replay for customer credit scoring in peer-to-peer lending. *Artif. Intell. Rev.* **2024**, *57*, 93. [CrossRef]
28. Du, H.; Tang, Y.; Tao, J. Construction of Credit Risk Assessment System Based on Deep Reinforcement Learning and Machine Learning–Taking P2p Lending Platform as An Example. *ITM Web Conf.* **2025**, *78*, 01017. [CrossRef]
29. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction*; MIT Press: Cambridge, Cambridge, MA, USA, 1998; Volume 1.
30. Almutairi, M.; Stahl, F.; Bramer, M. ReG-Rules: An Explainable Rule-Based Ensemble Learner for Classification. *IEEE Access* **2021**, *9*, 52015–52035. [CrossRef]
31. Garcez, A.d.; Lamb, L.C. Neurosymbolic AI: The 3rd wave. *Artif. Intell. Rev.* **2023**, *56*, 12387–12406. [CrossRef]
32. Besold, T.R.; Garcez, A.d.; Bader, S.; Bowman, H.; Domingos, P.; Hitzler, P.; Kühnberger, K.U.; Lamb, L.C.; Lima, P.M.V.; de Penning, L.; et al. Chapter1. Neural-Symbolic Learning and Reasoning: A Survey and Interpretation. In *Neuro-Symbolic Artificial Intelligence: The State of the Art*; Besold, T.R., Garcez, A.d., Hitzler, P., Eds.; IOS Press: 2021; Volume 342, pp. 1–51. [CrossRef]
33. Vanarse, A.; Osseiran, A.; Rassau, A. Neuromorphic engineering—A paradigm shift for future IM technologies. *IEEE Instrum. Meas. Mag.* **2019**, *22*, 4–9. [CrossRef]
34. Schuman, C.D.; Kulkarni, S.R.; Parsa, M.; Mitchell, J.P.; Date, P.; Kay, B. Opportunities for neuromorphic computing algorithms and applications. *Nat. Comput. Sci.* **2022**, *2*, 10–19. [CrossRef]
35. Al-Slehat, Z.A.F.; Almanaseer, S.R.; Al Sharif, B.M.M.; Al-Haraisa, Y.E.; Aloshaibat, S.D.; Almahasneh, M.A. Creditworthiness Criteria According to the 5Cs Model and Credit Decision: The Moderating Role of Intellectual Capital. *Int. Rev. Manag. Mark.* **2024**, *14*, 274–287. [CrossRef]
36. Karsh, S.M.A.; Abumwais, R.M.; Karsh, S.M.A. Evaluation Strategies of Credit Risk used by Commercial Banks Listed in Palestine Stock Exchange (PSE). *Glob. J. Manag. Bus. Res.* **2017**, *17*, 27–38.
37. Halkidi, M.; Batistakis, Y.; Vazirgiannis, M. On Clustering Validation Techniques. *J. Intell. Inf. Syst.* **2001**, *17*, 107–145. [CrossRef]
38. South German Credit. Available online: https://archive.ics.uci.edu/dataset/573/south+german+credit+update (accessed on 6/19/2020).
39. Statlog (Australian Credit Approval). Available online: https://archive.ics.uci.edu/dataset/143/statlog+australian+credit+approval
40. Default of Credit Card Clients. Available online: https://archive.ics.uci.edu/dataset/350/default+of+credit+card+customers (accessed on 1/25/2016).
41. Muntasir, F.; Anower, M.S.; Nahiduzzaman, M. Majority Voting Ensemble Approach for Predicting Diabetes Mellitus in Female Patients from Unbalanced Dataset. In Proceedings of the 2023 International Conference on Electrical, Computer and Communication Engineering (ECCE), 23–25 February 2023; pp. 1–6.
42. Dietterich, T.G. Ensemble Methods in Machine Learning. In *International Workshop on Multiple Classifier Systems* ; Springer: Berlin/Heidelberg, Germany, 2000; pp. 1–15.

43. Wiewer, A. Credit_Score_Classification. Available online: https://www.openml.org/search?type=data&status=any&id=46441 (accessed on 25 November 2024).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

applied sciences

MDPI

Article

Extracting Behavioral Rules from Health Survey Data with Interpretable Models

Piotr Lasek

Faculty of Exact and Technical Sciences, University of Rzeszów, Al. Rejtana 16C, 35-959 Rzeszów, Poland; plasek@ur.edu.pl; Tel.: +48-17-851-8516

Abstract

This study investigates the use of interpretable machine learning techniques to identify behavioral and demographic patterns associated with diabetes, based on structured population survey data from the Canadian Community Health Survey (CCHS). A decision tree classifier was applied to a dataset comprising 16,824 respondents and 38 preprocessed features covering lifestyle, well-being, and sociodemographic factors. The model was optimized through grid search with five-fold stratified cross-validation, achieving a test accuracy of 61.3% (mean 62.6% $\pm$ 0.6% across a 10 $\times$ 5 repeated stratified cross-validation). Feature importance analysis revealed that age, alcohol consumption patterns, daily energy expenditure, and physical activity were the most influential factors associated with diabetes status, with the top three features exhibiting stable importance across all cross-validation folds. The model produced a set of 32 human-readable decision rules; a sensitivity analysis confirmed that these rules are stable across encoding choices and cross-validation folds. Several model variants were evaluated: a class-weighted decision tree, a logistic regression baseline, an age-only decision tree, and an age and sex logistic regression. The class-weighted model improved minority-class recall (from 0.25 to 0.53) at the cost of overall accuracy. A one-hot encoding sensitivity analysis showed that replacing ordinal label encoding of nominal variables with one-hot encoding produces virtually identical results (accuracy: 61.4% vs. 61.3%), confirming that the main rules are not artifacts of the encoding choice. Although the classification accuracy is moderate and not significantly better than a majority-class baseline (McNemar's test, $p = 0.455$), the extracted rules confirmed several known associations and revealed interactions between social and lifestyle variables. These rules are intended as hypothesis-generating population-level descriptors rather than validated clinical decision tools, and no causal inference is claimed. This approach demonstrates the value of rule-based models for exploratory public health research.

Keywords: medical data exploration; diabetes; health informatics; decision trees; machine learning in healthcare; interpretable machine learning

1. Introduction

Diabetes mellitus is one of the most prevalent chronic diseases worldwide and represents a major public health challenge. According to the International Diabetes Federation, approximately 537 million adults were living with diabetes in 2021, a figure projected to rise to 783 million by 2045 [1]. A pooled analysis of 751 population-based studies found that the global age-standardized prevalence of diabetes has nearly doubled since 1980, rising from 4.3% to 9.0% in men and from 5.0% to 7.9% in women [2]. The financial impact

is also significant: global healthcare spending on diabetes was estimated at USD 966 billion in 2021 [1].

Canada is no exception to this global trend. The IDF estimates that approximately 3.7 million Canadians were living with diabetes in 2021, representing a prevalence of approximately 11% among adults aged 20–79 [1]. National surveillance data indicate that at least 3.4 million Canadians aged 20 years or older had a physician-diagnosed diabetes in a recent surveillance period, a figure expected to rise substantially as the population ages [3]. In Canada, direct healthcare costs related to diabetes exceed CAD 3.8 billion annually, with indirect costs from lost productivity adding substantially to the overall burden. The Canadian Community Health Survey (CCHS), provides one of the most comprehensive sources of population-level data on health behaviors, chronic conditions, and sociodemographic characteristics [4]. Conducted regularly, the CCHS surveys about 130,000 respondents per cycle and collects self-reported information on many lifestyle factors, including physical activity, smoking, alcohol consumption, diet, emotional well-being, and social support. This makes it a valuable source of data for chronic disease research. Unlike clinical registries, a full CCHS cycle is broadly representative of the Canadian population when its survey weights are applied, and it includes individuals who may not regularly access healthcare. The present study analyzes an unweighted analytic subset of this data for exploratory pattern discovery, making it well suited for studying how behavioral factors are associated with chronic disease.

The analysis of health survey data using machine learning methods has gained increased attention in recent years [5,6]. However, many studies in this domain focus mainly on predictive accuracy rather than interpretability. They often use black box models such as deep neural networks, random forests, or gradient boosting machines [6,7]. While these models can achieve high accuracy, they offer limited insight into how decisions are made, which is a significant drawback in healthcare applications where transparency, trust, and clinical usefulness are essential [8,9].

In contrast, interpretable models produce predictions that domain experts and stakeholders can understand directly. Decision trees are especially useful because they create rule-based structures that clearly show the conditions leading to different predicted outcomes [10]. This transparency is particularly important in public health, where policymakers and clinicians need to understand not only what a model predicts, but also why it makes that prediction. Rudin [8] strongly supports this view, arguing that in high-stakes settings, interpretable models should be preferred over post-hoc explanations of black box models.

A key gap in the existing literature is that most machine learning studies on diabetes rely on clinical or laboratory data, such as blood glucose levels, HbA1c, body mass index (BMI), and blood pressure, as input features [11–13]. Fewer studies have focused on using behavioral and psychosocial survey data as the main inputs for diabetes prediction. Survey-based features, including self-reported physical activity, smoking, alcohol use, and perceived health status, differ from clinical biomarkers in important ways. They are less expensive to collect, available at the population level, and directly relevant to prevention-focused interventions. However, they are also noisier and less directly connected to disease biology, which creates a different modeling challenge.

This paper addresses this gap by applying an interpretable decision tree classifier to behavioral and sociodemographic data from the CCHS. Instead of focusing mainly on maximizing predictive accuracy, the main goal is to identify meaningful, human-readable rules that show how combinations of lifestyle and demographic factors are related to diabetes diagnosis. This study builds on previous work using unsupervised methods for

https://doi.org/10.3390/app16126146

health data analysis [14]. It contributes to the literature by focusing on supervised learning, model interpretability, and quantitative evaluation using structured survey data.

The specific objectives of this study are (1) to construct a decision tree classifier on CCHS behavioral data and evaluate its performance using standard classification metrics; (2) to extract human-readable decision rules from the trained model; (3) to identify the key demographic and behavioral factors associated with diabetes status through feature importance analysis; and (4) to discuss the practical implications of the extracted rules for exploratory public health research, while acknowledging the observational nature of the study and the absence of causal claims.

The remainder of this paper is organized as follows. Section 2 reviews related work on interpretable machine learning and its application to healthcare and diabetes data. Section 3 describes the dataset used in the study, including its structure, preprocessing steps, and key characteristics. Section 4 outlines the methodology, including the modeling approach, hyperparameter optimization, and evaluation protocol. Section 5 presents the quantitative and qualitative results of the analysis. Section 6 discusses the findings in the context of the existing literature and addresses limitations. Finally, Section 7 summarizes the contributions and proposes directions for future work.

2. Related Work

Data-driven methods are increasingly used to study chronic diseases, especially diabetes. This growth has been supported by the availability of large population health surveys and advances in machine learning methods. This section reviews related work in three areas: machine learning for diabetes prediction, interpretable machine learning methods, and the use of national health surveys for chronic disease modeling.

2.1. Machine Learning for Diabetes Prediction

Numerous studies have addressed the problem of predicting diabetes using machine learning. A large body of work has focused on standard clinical benchmarks such as the Pima Indians Diabetes Dataset, which contains laboratory measurements including glucose concentration, BMI, and blood pressure. On such datasets, studies have reported classification accuracies ranging from 65% to 78% using methods such as decision trees, support vector machines, naive Bayes, and ensemble methods [5,11]. More recently, Zou et al. [12] compared multiple classifiers on clinical diabetes data and found that random forests and gradient boosting achieved the highest accuracy, but at the cost of interpretability.

Beyond clinical datasets, several studies have used population health surveys for diabetes-related analysis. Maniruzzaman et al. [13] applied multiple classifiers to demographic and behavioral features and demonstrated that ensemble methods can achieve reasonable accuracy even without laboratory measurements. Studies using survey-based features, including physical activity, BMI category, and smoking status, have shown that such data can predict diabetes with moderate accuracy [12,13]. These studies highlight the feasibility of using self-reported data for chronic disease screening, albeit with lower accuracy than clinical data-based models.

In the Canadian context, Lasek and Mei [14] conducted a detailed study on clustering and visualizing high-dimensional data related to diabetes using the Canadian Community Health Survey. Their work emphasized the importance of effective data representation and dimensionality reduction and identified meaningful patient groups using characteristics such as self-perceived health, income, physical activity, and living conditions. The present study extends this line of work by applying supervised learning to the same data source.

https://doi.org/10.3390/app16126146

2.2. Interpretable Machine Learning

In the context of supervised learning, decision trees and rule-based classifiers have been recognized for their interpretability, especially in medical applications where transparent decision-making is essential [8,15]. These models provide direct insight into the relative importance of variables and the structure of decision rules. In contrast to black box models such as neural networks, decision trees enable domain experts to validate and understand model behavior, which is critical in sensitive domains such as healthcare.

Beyond decision trees, several other interpretable or explainability-oriented methods have been developed. LIME (Local Interpretable Model-agnostic Explanations) provides local explanations for individual predictions by fitting simple surrogate models in the neighborhood of each instance [16]. SHAP (SHapley Additive exPlanations) assigns feature importance values based on cooperative game theory and provides both local and global explanations [17]. Generalized additive models (GAMs) and their extensions, such as Explainable Boosting Machines (EBMs), offer a middle ground between interpretability and predictive power by modeling non-linear feature effects in an additive structure [18]. Rule-based methods such as Bayesian Rule Lists provide ordered decision rules with probabilistic guarantees [19].

While these methods offer valuable alternatives, decision trees remain one of the most widely used interpretable models due to their simplicity, computational efficiency, and direct production of decision rules without requiring post-hoc explanation [10]. For the present study, a decision tree classifier was selected precisely because the goal is to extract explicit behavioral rules, not merely to explain predictions after the fact.

2.3. Data Preprocessing and Visualization

Data preprocessing plays a important role in improving both model performance and interpretability. In medical datasets, attributes often include a mix of categorical, ordinal, and continuous variables, sometimes with missing values or categories that are not applicable. Techniques such as discretization of numeric values into ordinal categories and encoding of categorical variables are standard approaches to prepare data for interpretable models [20–22].

Visualization is another key component of data analysis. Interactive data visualization techniques can support exploratory analysis, improve communication of findings, and enhance the transparency of results [23,24]. These methods are particularly useful in public health contexts, where policymakers and stakeholders need clear and concise summaries of patterns emerging from complex datasets.

Although deep learning and ensemble methods such as random forests and gradient boosting have demonstrated high predictive accuracy in diabetes diagnosis and risk estimation tasks [6,7], these models often lack interpretability, making them less suitable for use cases where decisions must be explained. Therefore, interpretable models such as decision trees remain valuable tools, especially when combined with thoughtful preprocessing and visualization strategies.

This work aims to extract meaningful behavioral patterns from structured health survey data by applying an interpretable supervised learning approach. Instead of grouping individuals through unsupervised methods, we focus on generating explicit decision rules that reveal how specific combinations of features relate to a health outcome. In doing so, we seek to identify and explain which behavioral and demographic characteristics are most indicative of risk, providing insights that are both data-driven and easy to interpret.

https://doi.org/10.3390/app16126146

3. Materials

3.1. Data Source

The dataset used in this study is based on a subset of responses from the Canadian Community Health Survey (CCHS), a cross-sectional survey conducted by Statistics Canada that collects information on the health status, healthcare utilization, and health determinants of the Canadian population [4]. The CCHS covers approximately 130,000 respondents per cycle and includes both numeric and categorical attributes spanning demographics, lifestyle, chronic conditions, and psychosocial well-being.

For the purposes of this experiment, a labeled version of the dataset was used in which each record represents an individual respondent and includes a binary decision attribute (`CCC_101`) indicating whether the individual has been diagnosed with diabetes. The raw dataset contained 16,836 records and 117 columns. After filtering to retain only records with a definitive diabetes diagnosis (`YES` or `NO`), the working dataset comprised 16,824 respondents.

3.2. Feature Selection and Filtering

The original metadata file associated with the survey describes 1206 variables. Of these, 117 were included in the dataset used for this study. The selection was performed by excluding administrative and operational variables (interview identifiers, sample weights, collection codes), geopolitical precision codes, derived aggregate indices, and items from clinical diagnostic modules unrelated to behavioral risk factors; only variables pertaining to health behaviors, demographics, socioeconomic status, psychosocial well-being, and general health perceptions were retained. The 117 selected variables covered a range of domains:

- Demographics: age (`DHHGAGE`), sex (`DHH_SEX`), marital status (`DHHGMS`), household size, province of residence, education level (`EDUDH04`), household income (`INCGHH`).
- Lifestyle and behavior: type of smoker (`SMK_202`), type of drinker (`ALCDTTM`), physical activity index (`PACDPAI`), daily energy expenditure (`PACDEE`), monthly frequency of physical activity (`PACDFM`).
- Psychosocial characteristics: self-perceived health (`GEN_01`), perceived mental health (`GENDMHI`), satisfaction with life (`GENGSWL`), health compared to one year ago (`GEN_02`).
- Other: pain and discomfort (`HUPDPAD`), health region (`GEODPMF`), immigration status (`SDCFIMM`), cultural background (`SDCGCGT`), and lifestyle segmentation cluster (`Segment`).

To prevent data leakage, the variable `CCC_121` (heart disease diagnosis), which belongs to the same chronic conditions module as the target variable, was removed prior to modeling.

Columns with more than 10% of values marked as `NOT_APPLICABLE` were excluded from the analysis, resulting in the removal of 74 columns. An additional 3 columns were removed due to having more than 10% of values recorded as `NOT_STATED`. After these filtering steps, 38 features remained for model training.

Table 1 presents a sample of the key features retained in the analysis along with their human-readable descriptions, as obtained from the survey metadata file.

Table 1 lists a representative subset of the retained features; the complete list of all 38 features used for model training—together with each variable's type, encoding, and an assessment of possible post-diagnosis relevance—is provided in Appendix A.

The `Segment` variable is a derived psychosocial segmentation variable produced by the CCHS that assigns respondents to lifestyle clusters based on their reported emotional well-being, social support, stress levels, and coping behaviors. It captures an integrative psychosocial profile that is not directly represented by any single survey question.

https://doi.org/10.3390/app16126146

It should be noted that several features in the dataset, particularly self-perceived health (GEN_01) and perceived mental health (GENDMHI), may partially reflect post-diagnosis status: respondents already living with diabetes may rate their health lower as a consequence of the disease rather than as an independent antecedent. This potential post-diagnosis confounding cannot be resolved with cross-sectional data, and all associations reported in this study should be interpreted as correlational rather than causal (see Section 6, Limitations).

Table 1. Sample of key features retained in the analysis with descriptions from the CCHS metadata. The Encoding column indicates the encoding used for model training: OrdLE = ordinal label encoding (integer codes preserve natural category order); OHE = one-hot encoding in the sensitivity analysis (Section 5.3); Cont. = treated as continuous (numeric or already integer-coded).

Variable Code	Type	Encoding	Description
DHHGAGE	Ordinal	OrdLE	Age group (e.g., 45 TO 49 YEARS, 80 YEARS OR MORE)
DHH_SEX	Nominal	OHE	Sex (MALE/FEMALE)
ALCDTTM	Nominal	OHE	Type of drinker in past 12 months (REGULAR/OCCASIONAL/NO DRINK)
PACDPAI	Ordinal	OrdLE	Physical activity index (ACTIVE/MODERATE ACTIVE/INACTIVE)
PACDEE	Continuous	Cont.	Daily energy expenditure (kcal/kg/day)
PACDFM	Continuous	Cont.	Monthly frequency of physical activity (>15 min)
SMK_202	Nominal	OHE	Type of smoker (DAILY/OCCASIONALLY/NOT AT ALL)
GEN_01	Ordinal	OrdLE	Self-perceived health (EXCELLENT to POOR)
GENDMHI	Ordinal	OrdLE	Perceived mental health (EXCELLENT to POOR)
EDUDH04	Ordinal	OrdLE	Highest level of education (household, 4 levels)
INCGHH	Ordinal	OrdLE	Total household income (grouped)
DHHGMS	Nominal	OHE	Marital status (unordered categories)
GEODPMF	Nominal	OHE	Health region (geographic, unordered)
Segment	Nominal	OHE	Derived psychosocial lifestyle cluster (e.g., DEPRESSED, GLOOMY, HAPPY)

3.3. Dataset Summary

Table 2 provides a summary of the dataset used for training the decision tree classifier.

Table 2. Summary of the dataset used for training the decision tree classifier.

Statistic	Value
Total number of records (raw)	16,836
Records after filtering (YES/NO only)	16,824
Original number of columns	117
Columns removed (leakage)	1
Columns removed (NOT_APPLICABLE > 10%)	74
Columns removed (NOT_STATED > 10%)	3
Features retained for modeling	38
Target attribute	CCC_101 (Diabetes)
Class distribution: YES (diabetic)	10,218 (60.7%)
Class distribution: NO (non-diabetic)	6606 (39.3%)

The dataset exhibits a moderate class imbalance, with diabetic respondents constituting approximately 61% of the filtered records. To examine whether class imbalance affects model performance, a class-weighted variant of the decision tree (class_weight = 'balanced')

https://doi.org/10.3390/app16126146

was evaluated alongside the standard unweighted model and a logistic regression baseline; results from all configurations are compared in Section 5.

3.4. Descriptive Statistics

Table 3 presents the distribution of the two most important features—age group and drinking type—stratified by diabetes status. These features were identified by the decision tree as the strongest predictors (see Section 5).

Table 3. Distribution of key features by diabetes status (NO = non-diabetic, YES = diabetic). Row percentages are within-class. Chi-square tests of independence are reported per feature.

Feature/Category	NO (n = 6606)	%	YES (n = 10,218)	%
DHHGAGE (Age group)	$\chi^2 = 573.0$, $p < 0.001$			
≤44 years	364	5.5	760	7.4
45–64 years	1860	28.2	3794	37.1
65–79 years	2662	40.3	4303	42.1
≥80 years	1720	26.0	1361	13.3
ALCDTTM (Type of drinker)	$\chi^2 = 137.3$, $p < 0.001$			
Regular drinker	3264	49.4	4139	40.5
Occasional drinker	1117	16.9	2175	21.3
Non-drinker	2089	31.6	3708	36.3
Not stated	136	2.1	196	1.9

The age distribution reveals a striking contrast: respondents aged ≥ 80 years make up 26.0% of the non-diabetic class but only 13.3% of the diabetic class. This likely reflects survivor bias and changing disease patterns in the oldest age groups captured by self-reported data. In contrast, the 45–64 year range is substantially more prevalent among diabetic respondents (37.1% vs. 28.2%), consistent with the well-documented middle-age peak in type 2 diabetes onset. For drinking type, regular drinkers are more prevalent in the non-diabetic class (49.4% vs. 40.5%), reflecting the apparent protective association with moderate alcohol consumption that is also captured by the decision tree (discussed in Section 6).

4. Methodology

4.1. Preprocessing Pipeline

The data processing pipeline consisted of the following steps:

1. Data loading: The dataset was loaded from a comma-separated values (`CSV`) file containing 16,836 records and 117 columns.
2. Target filtering: Records with values other than `YES` or `NO` for the target attribute `CCC_101` were removed, yielding 16,824 usable records. Twelve records marked as `DON'T KNOW` were excluded.
3. Leakage removal: The variable `CCC_121` (heart disease diagnosis) was removed to prevent data leakage. Because `CCC_121` belongs to the same Chronic Conditions module (`CCC_*`) as the target, retaining it would allow the model to exploit co-diagnosis patterns rather than learning from behavioral and demographic features—the focus of this study.
4. Column filtering: Columns with more than 10% of values marked as `NOT_APPLICABLE` or `NOT_STATED` were excluded. The rationale for the 10% threshold is that higher proportions of non-informative responses reduce the effective sample size for that variable and introduce systematic missingness that cannot be reliably imputed in a

https://doi.org/10.3390/app16126146

survey context [25]. This step removed 77 columns (74 due to `NOT_APPLICABLE`, 3 due to `NOT_STATED`), leaving 38 features.

5. Feature encoding: The 38 retained features were encoded as follows. Ordinal variables (e.g., age group, education level, self-perceived health, physical activity index) were encoded with ordinal label encoding (OrdLE), where integer codes follow the natural category order. This encoding preserves the ordering information and is appropriate for tree-based splits on ordinal scales. Nominal variables—those with no natural ordering, including type of drinker (`ALCDTTM`), type of smoker (`SMK_202`), marital status, health region (`GEODPMF`), and the psychosocial segment—were also encoded with label encoding in the primary analysis, assigning arbitrary integer codes. Label encoding of nominal variables imposes a spurious ordering that may constrain split candidates for decision trees; however, this does not change which feature is selected at each split, and the dominant splits in this dataset (age, alcohol consumption, energy expenditure) involve features where the ordering either is natural (age) or is empirically irrelevant (the tree finds the correct split threshold regardless of alphabetical assignment). To verify this claim, a one-hot encoding (OHE) sensitivity analysis was conducted in which all nominal variables were expanded into binary indicator columns; the results are reported in Section 5.3. The primary label-encoding variant was retained for the main analysis because it preserves a single Gini importance score per original survey variable, which is important for the feature importance interpretation (Section 5).
6. Target encoding: The binary target variable was encoded as 0 (`NO`) and 1 (`YES`).

4.2. Train/Test Split

The dataset was divided into a training set (80%) and a test set (20%) using stratified random sampling with a fixed random seed (`random_state = 42`) to ensure reproducibility and preserve class proportions across both splits. This resulted in 13,459 training samples and 3365 test samples.

4.3. Model Selection and Hyperparameter Optimization

A decision tree classifier was selected for this task due to its ability to produce interpretable models in the form of decision rules [10]. The algorithm identifies conditional relationships between feature values and the target outcome, allowing the extraction of clear, human-readable paths from input data to predictions.

To determine the optimal hyperparameters, a grid search procedure with five-fold stratified cross-validation was conducted. The grid search evaluated combinations of three key parameters:

- Maximum depth of the tree: `max_depth` $\in \{5, 10, 15, 20, \text{None}\}$
- Minimum samples required to split a node: `min_samples_split` $\in \{2, 5, 10\}$
- Minimum samples required at a leaf node: `min_samples_leaf` $\in \{1, 2, 4\}$

This produced a total of 45 parameter combinations. Accuracy was used as the scoring metric, and the stratified cross-validation ensured that class distributions were preserved across all folds.

A key finding of the grid search was that `max_depth` is the dominant hyperparameter: within each depth level, variations in `min_samples_split` and `min_samples_leaf` produced negligible differences in accuracy (less than 0.02 percentage points). For this reason, Table 4 reports the best cross-validation result for each depth level rather than listing all 45 combinations, and includes training accuracy and the number of leaf nodes to illustrate the bias–variance trade-off.

Several observations emerge from this analysis. First, the optimal depth is 5, yielding a compact tree with 32 leaves that generalizes best to unseen data. Second, the gap

https://doi.org/10.3390/app16126146

between training accuracy and cross-validation accuracy grows sharply with depth (from 1.3 percentage points at depth 5 to 30.7 percentage points for the unconstrained tree) providing evidence of overfitting. Third, only the depth-5 tree exceeds the majority-class baseline of 60.7%, and it does so by a margin of 0.6 percentage points on the test set (61.3% vs. 60.7%). This margin reflects the difficulty of predicting a clinical diagnosis from self-reported behavioral data alone; however, as discussed in Section 5, the model's value lies not in marginal accuracy gains but in the interpretable risk patterns it reveals.

Table 4. Grid search results grouped by `max_depth`. For each depth, the best-performing combination of `min_samples_split` and `min_samples_leaf` is reported. The majority-class baseline (always predicting `YES`) achieves 60.7% accuracy.

Max_Depth	Leaves	Train Acc.	CV Acc. ± Std.	Test Acc.	Δ vs. Baseline
Baseline (majority)	1	0.607	—	0.607	—
5	32	0.645	0.631 ± 0.007	0.613	+0.6%
10	532	0.733	0.597 ± 0.006	0.605	−0.3%
15	1398	0.837	0.560 ± 0.009	0.550	−5.8%
20	1981	0.863	0.549 ± 0.007	0.553	−5.5%
None (27)	1807	0.854	0.547 ± 0.004	0.541	−6.6%

The optimal model was configured with `max_depth` = 5, `min_samples_split` = 2, and `min_samples_leaf` = 1.

4.4. Model Evaluation

Given the moderate class imbalance (61%/39%), overall accuracy is an insufficient evaluation metric (a majority-class classifier that always predicts `YES` would achieve 60.7% accuracy). To provide a more complete picture of model performance, we report:

- Per-class precision, recall, and F1-score, which reveal how the model performs on each class independently. In a screening context, recall for the diabetic class (sensitivity) is important, as it reflects the proportion of true cases that the model detects.
- Macro-averaged metrics, which weight both classes equally regardless of their prevalence, exposing weaknesses in minority-class prediction.
- Weighted-averaged metrics, which account for class prevalence and reflect overall expected performance.
- A confusion matrix, which provides the full breakdown of true positives, false positives, true negatives, and false negatives.

All metrics were computed on the held-out test set, which was not used during hyperparameter optimization or model training. The majority-class baseline (60.7% accuracy) serves as the lower bound against which the decision tree's performance is compared.

4.5. Software Environment

The analysis was implemented in Python 3.11.9 using the scikit-learn library [20] for classification and model evaluation, pandas for data manipulation, and matplotlib [26] for visualization. The data processing and modeling workflow was organized into a reproducible pipeline.

5. Results

5.1. Classification Performance

Table 5 presents the classification performance of the optimal decision tree model on the held-out test set (3365 samples).

The confusion matrix for the test set is shown in Table 6.

https://doi.org/10.3390/app16126146

The model achieves an overall accuracy of 61.3%, with notably asymmetric per-class performance. For the diabetic class (YES), the model achieves a recall of 0.85, meaning it correctly identifies 85% of diabetic individuals. However, precision for this class is 0.64, indicating that 36% of predicted diabetic cases are false positives. For the non-diabetic class (NO), precision is low (0.52) and recall is very low (0.25), reflecting the model's tendency to over-predict the majority class.

Table 5. Classification performance of the decision tree model on the test set.

Class	Precision	Recall	F1-Score	Support
NO (non-diabetic)	0.52	0.25	0.34	1321
YES (diabetic)	0.64	0.85	0.73	2044
Macro average	0.58	0.55	0.53	3365
Weighted average	0.59	0.61	0.57	3365
Overall accuracy		0.613		

Table 6. Confusion matrix for the decision tree classifier on the test set.

	Predicted NO	Predicted YES
Actual NO	333	988
Actual YES	313	1731

5.2. Model Comparison and Statistical Testing

To contextualize the decision tree's performance, we compared four models on the same held-out test set: the majority-class baseline (always predicting YES), the standard depth-5 decision tree (DT-orig), a class-weighted variant of the same tree (class_weight = 'balanced', DT-balanced), and a logistic regression classifier (LR) as a second interpretable baseline.

Several observations emerge. First, the age-only DT (using only DHHGAGE) achieves 61.8% accuracy—slightly above the full 38-feature model (61.3%). The age + sex LR degenerates to predicting the majority class for all samples, achieving exactly the majority baseline (60.7%). These results indicate that adding age as the sole predictor captures most of the accuracy signal, and that the behavioral variables in the full model do not provide additional predictive accuracy beyond age alone. The primary contribution of the full model therefore lies in the structured behavioral rules it produces, not in improved accuracy over a demographic-only baseline. Second, applying class weights (DT-balanced) substantially improves non-diabetic recall (from 0.25 to 0.53) and yields a higher macro F1 (0.562 vs. 0.533), but reduces overall accuracy to 57.2% and diabetic recall to 0.60. Third, logistic regression achieves nearly identical overall accuracy (61.2%) but with even greater class asymmetry (YES-recall 0.91, NO-recall 0.16). Fourth, the OHE sensitivity model achieves 61.4% accuracy, differing from the label-encoding model by only 0.09 percentage points, with virtually identical per-class performance; the implications for rule stability are discussed in Section 5.3.

To assess whether the observed accuracy differences are statistically significant, we applied McNemar's test [27] to each pairwise model comparison on the test set. The primary DT (label encoding) does not significantly outperform the majority baseline ($\chi^2 = 0.56$, $p = 0.455$), confirming that the 0.6 percentage-point accuracy improvement represents a negligible gain. The DT-balanced model differs significantly from DT-orig ($\chi^2 = 21.23$, $p < 0.001$), reflecting the meaningful redistribution of predictions between classes. Logistic regression does not differ significantly from DT-orig ($\chi^2 = 0.03$, $p = 0.870$). These results

https://doi.org/10.3390/app16126146

confirm that the decision tree's primary value lies not in superior predictive accuracy but in the interpretable rule structure it produces.

5.3. One-Hot Encoding Sensitivity

To verify that the main results are not artifacts of the label-encoding of nominal variables, a sensitivity analysis was conducted in which all nominal variables (`ALCDTTM`, `SMK_202`, `DHHGMS`, `Segment`, `GEODPMF`, `DHH_SEX`, and three sociodemographic indicators) were replaced by one-hot indicator columns, expanding the feature matrix from 38 to 162 columns. A depth-5 decision tree was trained and evaluated on the same train/test split.

The results are shown in Table 7 (row "DT depth-5, one-hot encoding"). Test accuracy was 61.4%, differing from the label-encoding model by 0.09 percentage points. The per-class performance (NO-Prec 0.518, NO-Rec 0.254, YES-Prec 0.637, YES-Rec 0.847) is virtually identical to the label-encoding variant. Crucially, the root split was `DHHGAGE` in both models, and the depth-1 splits both involved `ALCDTTM` (as `ALCDTTM_REGULAR DRINKER` binary indicator in the OHE model). These findings confirm that the main extracted rules—and the ranking of features associated with diabetes status—are robust to the choice of encoding for nominal variables.

Table 7. Model comparison on the held-out test set ($n = 3365$). NO-Rec and YES-Rec denote per-class recall; Macro F1 weights both classes equally. Simple demographic baselines (rows 2–3) use only age or age+sex to isolate the contribution of the behavioral features.

Model	Acc.	Mac.F1	NO-P	NO-R	YES-P	YES-R
Majority baseline (all YES)	0.607	0.378	0.000	0.000	0.607	1.000
Age-only DT (depth-5)	0.618	0.536	0.530	0.251	0.639	0.856
Age+sex LR	0.607	0.378	0.000	0.000	0.607	1.000
DT depth-5, label enc. (primary)	0.613	0.533	0.515	0.252	0.637	0.847
DT depth-5, OHE (sensitivity)	0.614	0.534	0.518	0.254	0.637	0.847
DT depth-5 (class-weighted)	0.572	0.562	0.461	0.534	0.664	0.596
Logistic Regression (38 features)	0.612	0.491	0.519	0.158	0.625	0.905

5.4. Feature Importance

Age was by far the strongest factor associated with diabetes status, accounting for 51.7% of the model's decision-making power, followed by alcohol consumption patterns (15.5%) and daily energy expenditure (10.0%). Together, the top three features account for over 77% of total feature importance. Physical activity measures (monthly frequency and leisure participation), education level, and self-perceived health also contributed meaningfully to the model. The top ten features are summarized in Table 8.

To assess the stability of both importance rankings and specific split conditions across different data splits, we evaluated the decision tree within each fold of the five-fold stratified cross-validation. The top three features—`DHHGAGE`, `ALCDTTM`, and `PACDEE`—appeared in the top three positions across all five folds (5/5), with mean importances of 0.537 ± 0.015, 0.167 ± 0.019, and 0.074 ± 0.021 respectively (mean $\pm$ standard deviation across folds).

Beyond feature importance rankings, the stability of the specific split conditions was also examined. Table 9 reports the split conditions (feature, direction, and category label) that appeared in the upper levels (depth 0–2) of the decision tree across all five folds. The root split (`DHHGAGE` $\leq$ `75 TO 79 YEARS`) was identical in all five folds. The depth-1 split on alcohol consumption (`ALCDTTM` $\leq$ `OCCASIONAL DRINKER`) also appeared in all five folds. Four of the six most stable conditions appeared in all 5/5 folds, confirming that the key behavioral splits—not only the feature rankings—are robust to sampling variation.

To further assess performance stability, a 10-repeat $\times$ 5-fold repeated stratified cross-validation was conducted (50 evaluation folds total). Mean accuracy across the 50 folds

https://doi.org/10.3390/app16126146

was 62.6% ± 0.6% (range: 61.4–64.4%), confirming that the depth-5 model's performance is stable and that the single-run estimate of 61.3% test accuracy is representative of the model's generalization ability.

Table 8. Feature importance scores from the decision tree model (Gini importance). Only features with non-zero importance are shown.

Rank	Feature (Variable Code)	Description	Importance
1	DHHGAGE	Age group	0.517
2	ALCDTTM	Type of drinker	0.155
3	PACDEE	Daily energy exp.	0.100
4	GEODPMF	Health region	0.049
5	PACDFM	Monthly phys. act. freq.	0.030
6	EDUDH04	Education level	0.021
7	Segment	Emotional segment	0.018
8	DHH_SEX	Sex	0.016
9	GEN_02	Health vs. 1 yr ago	0.015
10	PACFLEI	Leisure phys. act.	0.013
11	GEN_01	Self-perceived health	0.013
12	SDC_8	Current student	0.012
13	GEN_02A2	Health change (alt.)	0.008
14	DHHG611	Children 6–11 in HH	0.008
15	GENDMHI	Mental health	0.007
16	GEN_02B	Health change (scale)	0.007
17	DHHGMS	Marital status	0.007
18	SMK_202	Type of smoker	0.005

Table 9. Stability of split conditions across 5 CV folds. Only conditions appearing in ≥4 folds are shown. Conditions are from the root and depth-1/depth-2 nodes of a depth-5 DT trained on each fold.

Split Condition	Folds Present
DHHGAGE ≤ 75 TO 79 YEARS	5/5
DHHGAGE > 80 YEARS OR MORE	5/5
ALCDTTM ≤ OCCASIONAL DRINKER	5/5
ALCDTTM > REGULAR DRINKER	5/5
DHHGAGE ≤ 70 TO 74 YEARS	5/5
DHHGAGE > 75 TO 79 YEARS	5/5

5.5. Decision Tree Structure and Rules

The trained decision tree has a maximum depth of five and contains 32 leaf nodes. Figure 1 shows a simplified version of the tree that highlights the most significant decision paths. The root split is based on age: respondents aged 80 years or more are separated from younger individuals at the first decision node.

The key decision pathways extracted from the model can be summarized as follows:

1. Age as root predictor: Respondents aged 80 years or more are separated at the root node, indicating that advanced age is the single strongest predictor. Among younger individuals, alcohol consumption patterns form the next decision point.
2. Alcohol and physical activity: Among respondents under 80 who are not regular drinkers, physical activity level and smoking status determine the classification. Inactive individuals who smoke are more likely to be classified as diabetic.
3. Regular drinkers: Among regular drinkers under 80, physical inactivity combined with residence in Quebec and daily smoking leads to diabetic classification, while active individuals are classified as non-diabetic regardless of other factors.

https://doi.org/10.3390/app16126146

4. Elderly subgroups: For respondents aged 80 or older, the model further differentiates based on alcohol consumption, physical activity frequency, perceived health status, and sex.

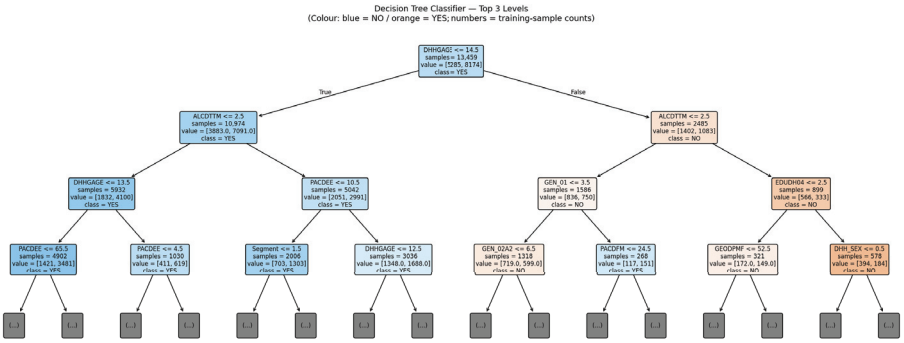


Figure 1. The decision tree diagram showing the top three levels. Node labels show the split condition, the Gini impurity, the number of training samples reaching that node, and the sample counts per class [`NO`, `YES`]. Blue shading indicates a majority of non-diabetic samples (`NO`); orange shading indicates a majority of diabetic samples (`YES`). The full depth-5 tree contains 32 leaf nodes; only the three most informative levels are displayed for readability.

Table 10 presents a summary of the behavioral and demographic characteristics associated with each diabetes outcome.

Table 10. Summary of behavioral and demographic features associated with each diabetes outcome in the decision tree analysis. All associations are observational; no causal interpretation is implied.

Factor	Associated with Diabetes (`YES`)	Associated with No Diabetes (`NO`)
Age	80 years or older	Under 80 years
Physical Activity	Inactive or minimally active	Active or moderately active
Smoking	Daily or occasional smoker	Not at all
Drinking Habits	Not a regular drinker	Regular drinker
Self-Perceived Health	Poor or fair	Good to excellent
Energy Expenditure	Low daily expenditure	Higher daily expenditure
Sex	Male (in elderly subgroups)	Female (in certain subgroups)
Education	Lower education level	Post-secondary graduate
Region	Specific health regions	Other regions

Figure 2 illustrates the top-8 extracted IF–THEN rules ranked by their training sample coverage (support), providing a compact model-derived summary of the dominant demographic and behavioral patterns associated with each predicted class in the dataset.

5.6. Formal IF–THEN Rules

Each of the 32 leaf nodes in the decision tree corresponds to a formal IF–THEN rule, defined by the sequence of conditions along the path from the root to that leaf. Table 11 presents a representative selection of eight rules chosen to cover both predicted classes and a range of support levels. For each rule, support denotes the number of training samples reaching the leaf, and precision denotes the fraction of those samples matching the predicted class. Conditions are expressed using the original CCHS variable codes and categories; `PACDEE` refers to daily energy expenditure in kcal/kg/day, and `Segment` refers to the derived psychosocial lifestyle cluster.

The two highest-support rules (support = 2217 and 1837) together account for 4054 of the 13,459 training samples (30.1%) and both predict the diabetic class. They capture a similar population: non-elderly, non-regular drinkers with low daily energy expenditure,

https://doi.org/10.3390/app16126146

differentiated only by health region. This confirms that the dominant behavioral pattern associated with diabetes in this dataset—moderate or no drinking combined with low physical energy expenditure in the non-elderly population—is geographically widespread. Among the non-diabetic rules, Rule 8 (female, elderly, educated, regular drinker with stable health) achieves the highest precision (0.860), representing a compact, highly coherent profile of low-risk respondents.

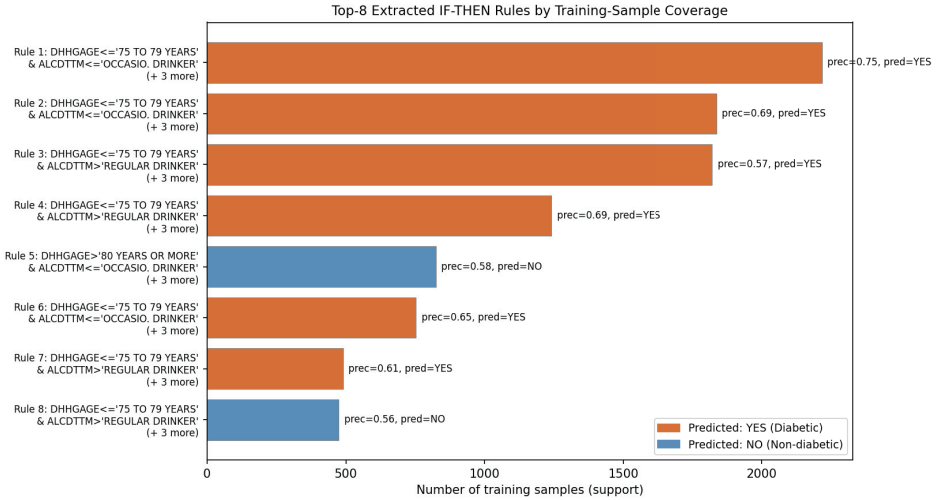


Figure 2. Top-8 extracted IF–THEN rules ranked by training sample coverage (support). Orange bars indicate rules predicting the diabetic class (`YES`); blue bars indicate rules predicting the non-diabetic class (`NO`). Precision (fraction of samples correctly classified at each leaf) is annotated on each bar. This visualization is directly derived from the trained depth-5 decision tree and provides a compact summary of the dominant behavioral and demographic patterns identified in the dataset.

Table 11. Selected decision rules from the depth-5 decision tree. Support = number of training samples (out of 13,459) reaching the leaf; Precision = fraction correctly classified. Rules are ordered by support within each predicted class. The full set of 32 rules is available upon request.

IF Conditions	Pred.	Support	Precision
Predicted class: YES (Diabetic)			
DHHGAGE $<$ 75 yr; ALCDTTM $\neq$ Regular; DHHGAGE $<$ 70 yr; PACDEE $\leq$ 3.0; health region $\leq$ Prairie	YES	2217	0.752
DHHGAGE $<$ 75 yr; ALCDTTM $\neq$ Regular; DHHGAGE $<$ 70 yr; PACDEE $\leq$ 3.0; health region $>$ Prairie	YES	1837	0.695
DHHGAGE $<$ 75 yr; ALCDTTM $=$ Regular; PACDEE $\geq$ 1.1; DHHGAGE $<$ 70 yr; eastern/coastal regions	YES	1822	0.565
DHHGAGE $<$ 75 yr; ALCDTTM $=$ Regular; PACDEE $\leq$ 1.0; Segment $\geq$ Gloomy; 65–70 yr	YES	1241	0.691
Predicted class: NO (Non-diabetic)			
DHHGAGE $\geq$ 80 yr; ALCDTTM $\neq$ Regular; GEN_01 not very good; GEN_02A2 $\geq$ 8; no children 6–11	NO	824	0.576
DHHGAGE 70–79 yr; ALCDTTM $=$ Regular; PACDEE $\geq$ 1.1; monthly PA freq. $\geq$ 32	NO	474	0.557
DHHGAGE $\geq$ 80 yr; ALCDTTM $=$ Regular; EDUDH04 $=$ Post-sec. grad.; Sex $=$ Male; leisure PA $=$ Yes	NO	252	0.671
DHHGAGE $\geq$ 80 yr; ALCDTTM $=$ Regular; EDUDH04 $=$ Post-sec. grad.; Sex $=$ Female; health not worsening	NO	100	0.860

6. Discussion

6.1. Interpretation of Results

The decision tree classifier achieved an overall test accuracy of 61.3%, which is moderate but consistent with the inherent challenge of predicting a clinical diagnosis from self-reported behavioral data alone. For comparison, studies using clinical biomarkers

https://doi.org/10.3390/app16126146

(blood glucose, BMI, HbA1c) on the Pima Indians dataset report accuracies in the range of 65–78% [5,11]. The lower accuracy in the present study is expected, as self-reported survey features are inherently noisier, subject to recall bias, and less directly related to the physiological mechanisms underlying diabetes.

Importantly, the McNemar test confirms that the depth-5 decision tree does not achieve a statistically significant accuracy improvement over the majority-class baseline ($p = 0.455$). This finding should be interpreted honestly: the model's value lies not in its marginal accuracy gain but in the interpretable, stable rule structure it produces. The extracted rules reveal which specific combinations of age, drinking behavior, physical activity, and psychosocial characteristics are associated with diabetes in this population-level dataset—a type of insight that a majority-class classifier cannot provide.

The asymmetric per-class performance (85% recall for diabetic, 25% for non-diabetic) reflects the well-known tendency of cost-insensitive classifiers to favor the majority class. The class-weighted variant (DT-balanced) improves non-diabetic recall to 53% and macro F1 to 0.562 (vs. 0.533 for the standard model), but reduces diabetic recall from 85% to 60% and overall accuracy to 57.2%. In a population-level risk stratification context—as distinct from individual clinical screening—the standard model's higher diabetic sensitivity may be preferable, as it is more conservative about labeling respondents as non-diabetic.

6.2. Consistency with Established Epidemiological Knowledge

The behavioral patterns identified by the decision tree are largely consistent with established epidemiological knowledge. Advanced age is the strongest factor associated with diabetes status (51.7% of total feature importance), which aligns with the well-documented increase in type 2 diabetes prevalence with aging [28]. While age dominates the global importance ranking, the result that age alone (without any behavioral features) achieves 61.8% accuracy underscores that demographic variables are the primary signal in this dataset. The behavioral patterns extracted in the sub-trees below the age split—particularly those involving alcohol consumption, energy expenditure, and psychosocial segment among non-elderly respondents—represent the interpretable contribution of this study. These patterns would not be reported by a model using age alone, even though they do not add overall predictive accuracy.

Physical inactivity and sedentary behavior are consistently associated with type 2 diabetes in the epidemiological literature [29], and their prominence in the model's feature importance is consistent with this population-level association in survey data. Smoking behavior also contributes to the model, consistent with a systematic review and meta-analysis that found active smokers have a higher risk of developing type 2 diabetes [30]. Education level and daily energy expenditure also appear among the top features, reflecting the well-documented socioeconomic gradient in chronic disease prevalence [29]. Self-perceived health status, which appears in the tree's deeper branches, is associated with objective health outcomes and mortality in the broader literature [31]. The presence of psychosocial and self-perception variables in the model highlights the multidimensional nature of diabetes-associated patterns, extending beyond purely physiological factors. All of these associations are cross-sectional and observational, and no causal interpretation should be drawn from this analysis.

6.3. The Alcohol–Diabetes Association

An apparent protective association with regular alcohol consumption was observed in the decision tree, where regular drinkers were less likely to be classified as diabetic. This finding is consistent with meta-analyses reporting a U-shaped relationship between alcohol consumption and type 2 diabetes risk, with moderate drinkers showing lower risk than abstainers [32]. However, this association must be interpreted with caution. The "sick quitter" hypothesis suggests that some non-drinkers may have stopped drinking due to

https://doi.org/10.3390/app16126146

pre-existing health conditions, including diabetes [33]. Additionally, regular moderate drinking may be a proxy for higher socioeconomic status, social engagement, and overall healthier lifestyle patterns rather than reflecting a direct protective effect of alcohol [34]. The CCHS data cannot distinguish between these explanations, and this finding should not be interpreted as evidence that alcohol consumption prevents diabetes.

6.4. Limitations

Several limitations should be acknowledged. First, this study is entirely observational and cross-sectional. The features identified as associated with diabetes status cannot be interpreted as causes of diabetes, and the study cannot be used to support causal public health interventions. All extracted rules should be read as population-level associations that may reflect shared confounders, reverse causation, or other non-causal mechanisms. Longitudinal study designs, instrumental variable analyses, or causal inference methods would be required to establish directionality.

Second, some psychosocial features—particularly self-perceived health (`GEN_01`) and perceived mental health (`GENDMHI`)—may partially reflect post-diagnosis status rather than independent antecedents. Respondents already living with diabetes may rate their health lower as a consequence of the disease; this reverse-causation pathway cannot be separated from a prospective association using cross-sectional data.

Third, the dataset relies entirely on self-reported data, which is subject to recall bias and social desirability effects. Studies have shown that self-reported diabetes prevalence may underestimate true clinical prevalence by 20–30% [35], meaning that some respondents classified as `NO` in the CCHS may have undiagnosed diabetes. The accuracy of self-reported behavioral variables—including physical activity duration, alcohol consumption frequency, and perceived health—is also known to vary substantially from objectively measured equivalents. These limitations mean that the model should not be used for individual-level clinical diagnosis; its appropriate use is for exploratory population-level pattern identification.

Fourth, the analysis is based on a single survey cycle, which limits the temporal generalizability of the findings. Patterns identified in one cohort may not hold across different time periods or populations.

Fifth, while the class imbalance is moderate (61%/39%), the model's tendency to favor the majority class (`YES`) is evident in the asymmetric per-class performance. The class-weighted DT variant evaluated in this study improves non-diabetic recall (from 0.25 to 0.53) and macro F1 (0.533 $\rightarrow$ 0.562), but at the cost of reduced diabetic recall and overall accuracy. The choice between these models depends on the intended application: the class-weighted model is preferable when balanced detection of both classes is required, while the standard model better preserves high sensitivity for the diabetic class.

Sixth, the maximum tree depth of five constrains the model's expressiveness. While this shallow structure enhances interpretability and reduces overfitting, it may prevent the model from capturing more complex feature interactions.

Finally, the appearance of regional features (health region, province) in the decision tree should be interpreted with caution. The `GEODPMF` health-region variable may proxy for population-level confounders—such as differences in healthcare access, urbanization, dietary norms, ethnic composition, or regional survey response patterns—rather than reflecting genuine geographic variation in diabetes risk. Rules involving regional splits should be treated as population-level descriptors of the training data rather than individual-level predictors, and should not be applied to individuals in other geographic contexts without re-training the model on locally representative data.

https://doi.org/10.3390/app16126146

6.5. Implications for Public Health

Despite these limitations, the approach demonstrates several practical advantages for public health research and exploratory analysis. The extracted decision rules are expressed in plain language and can be directly communicated to public health practitioners. Unlike complex ensemble models that require specialized software for interpretation, a decision tree with five levels can be printed on a single page and used as a visual summary of population-level risk patterns. This transparency is particularly valuable in research and policy contexts where understanding the structure of risk—rather than achieving maximum predictive accuracy—is the primary goal.

The identified behavioral and demographic patterns associated with diabetes—particularly the interaction of age, physical inactivity, and psychosocial profile—suggest targets for further investigation. Public health programs could use similar rule-based analyses to stratify populations by risk level, prioritize research and outreach efforts, or design targeted educational materials for specific demographic subgroups. It should be emphasized, however, that the model was trained on self-reported survey data without external validation on an independent cohort, and the McNemar test showed that its accuracy advantage over a trivial baseline is not statistically significant. Therefore, the extracted rules should be treated as hypothesis-generating descriptors of population-level associations, not as validated clinical decision rules suitable for individual risk assessment.

Furthermore, the use of survey-based features rather than clinical biomarkers means that the input data for such analyses could be collected through questionnaires or telephone interviews, without requiring laboratory tests. This makes the approach potentially scalable for population-level deployment, complementing rather than replacing clinical diagnostic pathways. National health surveys such as the CCHS are conducted on a regular cycle, which means that models trained on survey data can be periodically updated to reflect evolving population characteristics and health trends.

7. Conclusions and Future Work

This study applied an interpretable decision tree classifier to behavioral and sociodemographic data from the Canadian Community Health Survey to extract human-readable rules associated with diabetes diagnosis. The main contributions are as follows:

1. Interpretable rule extraction: The decision tree model produced a concise set of 32 IF-THEN rules linking age, physical activity, alcohol consumption, and self-perceived health to diabetes status, all expressed in a transparent, human-readable format using original survey category labels.
2. Quantitative evaluation: The model was rigorously evaluated using standard classification metrics, achieving 61.3% accuracy ($62.6 \pm 0.6\%$ over a 10×5 repeated stratified CV). McNemar's test confirms that this accuracy is not statistically significantly better than a majority-class baseline ($p = 0.455$); the value of the model lies in the interpretable rule structure, not in accuracy alone.
3. Feature stability: Age, alcohol consumption patterns, and daily energy expenditure were identified as the dominant factors associated with diabetes status (top-3 across all five CV folds). Rule condition stability analysis showed that six split conditions appeared identically in all 5/5 CV folds, confirming that the extracted rules themselves—not only the feature rankings—are robust to sampling variation.
4. Comprehensive model comparison: Seven model configurations were evaluated: majority baseline, age-only DT, age + sex LR, standard depth-5 DT (label encoding and OHE variant), class-weighted DT, and full logistic regression. The age-only DT achieves comparable accuracy (61.8%) to the full model, confirming that behavioral features do not add predictive accuracy over demographic-only baselines. The OHE

https://doi.org/10.3390/app16126146

sensitivity analysis (61.4% accuracy, 0.09 pp difference) confirms that the main rules are not artifacts of encoding choices.

The moderate classification accuracy reflects the fundamental difficulty of predicting a clinical diagnosis from self-reported behavioral data alone. The extracted rules should be treated as hypothesis-generating, observational descriptors of population-level associations—not as validated clinical decision tools and not as evidence of causal relationships. The interpretability and cross-validation stability of the identified patterns, and their consistency with established epidemiological knowledge, support the value of this approach for exploratory public health research.

Several directions for future work are envisioned. First, applying the methodology to external datasets, such as the Behavioral Risk Factor Surveillance System (BRFSS) in the United States or the Health Survey for England, would help assess the robustness and generalizability of the identified patterns across different populations.

Second, more advanced interpretable models could be explored. Explainable Boosting Machines (EBMs) [18] offer a middle ground between decision trees and ensemble methods, producing interpretable additive models with potentially higher accuracy. Bayesian Rule Lists [19] provide ordered decision rules with probabilistic uncertainty estimates. Post hoc explanation methods such as SHAP [17] could be applied to ensemble models to compare the feature importance patterns with those obtained from the decision tree.

Third, the decision rules extracted from the model could serve as the basis for developing practical screening tools. A web-based risk calculator that implements the top-level decision paths could provide a simple, accessible tool for preliminary risk stratification, subject to external validation on independent cohorts. Such tools could be integrated into clinical decision support systems or public health dashboards to assist in population-level monitoring and targeted intervention planning.

Fourth, future experiments could integrate additional data sources, such as clinical or longitudinal health records, to complement self-reported information and improve both prediction accuracy and the clinical validity of the extracted rules. Combining survey-based behavioral features with clinical biomarkers (e.g., fasting glucose, HbA1c, blood pressure) could yield models that are both interpretable and clinically actionable, bridging the gap between population-level surveillance and individual-level diagnosis.

Finally, the methodology developed in this study is not limited to diabetes and could be applied to other chronic conditions captured in the CCHS, such as heart disease, asthma, or mental health disorders. Comparative analysis across multiple conditions using the same survey-based approach could reveal shared and condition-specific behavioral risk factors, contributing to a more holistic understanding of chronic disease determinants at the population level.

Funding: This research received no external funding.

Institutional Review Board Statement: This study is based solely on publicly available, anonymized microdata from the Canadian Community Health Survey (CCHS), administered by Statistics Canada. No primary data were collected from human participants by the authors. Therefore, ethics board approval was not required.

Informed Consent Statement: Informed consent was not applicable, as this study used publicly available, de-identified survey data collected and anonymized by Statistics Canada under their approved ethical and privacy framework.

Data Availability Statement: Data derived from the Canadian Community Health Survey (CCHS).

Acknowledgments: The author thanks Zhen Mei (Manifold Data Mining Inc., Toronto, ON, Canada) for valuable discussions and contributions during earlier stages of this work.

Conflicts of Interest: The author declares no conflicts of interest.

https://doi.org/10.3390/app16126146

Appendix A. Complete List of Retained Features

For completeness, Table A1 lists all 38 features retained after preprocessing (Section 3) and used for model training. For each variable, the table reports the description from the CCHS metadata, the variable type, the encoding used in the primary (label-encoding) analysis, and an assessment of whether the variable may partly reflect post-diagnosis status (i.e., its value may be a consequence of living with diabetes rather than an independent antecedent). In the one-hot encoding sensitivity analysis (Section 5.3), all nominal variables are expanded into binary indicator columns. Variables flagged "Possible" (self-perceived and mental health, satisfaction, pain, and physical-activity measures) should be interpreted with appropriate caution, as cross-sectional data cannot establish their temporal ordering relative to diagnosis.

Table A1. Complete list of the 38 features retained for model training, with type, encoding, and possible post-diagnosis relevance. `OrdLE` = ordinal label encoding; `OHE` = one-hot encoding (sensitivity analysis); `Binary` = binary 0/1 label; `Cont.` = treated as continuous.

Code	Description	Type	Encoding	Post-dx. Relevance
Demographics and household				
DHHGAGE	Age group	Ordinal	OrdLE	No
DHH_SEX	Sex	Binary	Binary	No
DHHGMS	Marital status	Nominal	OHE	No
DHHGLE5	Persons ≤ 5 yr in household	Binary	Binary	No
DHHG611	Persons 6–11 yr in household	Binary	Binary	No
DHHGL12	Persons < 12 yr in household	Binary	Binary	No
DHHGLVG	Living arrangement	Nominal	OHE	No
DHHGHSZ	Household size	Ordinal	OrdLE	No
DHH_OWN	Dwelling owned/rented	Nominal	OHE	No
Geography and socioeconomic status				
GEOGPRV	Province of residence	Nominal	OHE	No
GEODPMF	Health region	Nominal	OHE	No
EDUDH04	Highest education (household)	Ordinal	OrdLE	No
INCGHH	Total household income	Ordinal	OrdLE	No
SDCGCB12	Country of birth (Canada/other)	Nominal	OHE	No
SDCGLHM	Language(s) spoken at home	Nominal	OHE	No
SDCFIMM	Immigrant status	Binary	OHE	No
SDCGCGT	Cultural/racial origin	Nominal	OHE	No
SDC_8	Current student	Nominal	OHE	No
Psychosocial and perceived health				
Segment	Psychosocial lifestyle cluster	Nominal	OHE	Possible
GEN_01	Self-perceived health	Ordinal	OrdLE	Possible
GEN_02	Health vs. 1 year ago	Ordinal	OrdLE	Possible
GEN_02A2	Satisfaction with life	Ordinal	OrdLE	Possible
GEN_02B	Self-perceived mental health	Ordinal	OrdLE	Possible
GEN_07	Perceived life stress	Ordinal	OrdLE	Possible
GEN_10	Sense of community belonging	Ordinal	OrdLE	Possible
GENDHDI	Perceived health (derived)	Ordinal	OrdLE	Possible
GENDMHI	Perceived mental health (derived)	Ordinal	OrdLE	Possible
GENGSWL	Satisfaction with life (derived)	Ordinal	OrdLE	Possible
PMH_04	Felt contributing to society	Ordinal	OrdLE	Possible
PMHDCLA	Positive mental-health class	Nominal	OHE	Possible
HUPDPAD	Pain/discomfort	Ordinal	OrdLE	Possible
Health behaviors				
PACDEE	Daily energy expenditure	Continuous	Cont.	Possible
PACDFM	Monthly freq. physical activity	Continuous	Cont.	Possible
PACDPAI	Physical activity index	Ordinal	OrdLE	Possible
PACFLEI	Leisure physical activity (part.)	Nominal	OHE	Possible
SMK_202	Type of smoker	Nominal	OHE	No
ETS_35	Smoking allowed in house	Nominal	OHE	No
ALCDTTM	Type of drinker	Nominal	OHE	Possible

https://doi.org/10.3390/app16126146

References

1. International Diabetes Federation. *IDF Diabetes Atlas*, 10th ed.; International Diabetes Federation: Brussels, Belgium, 2021.
2. NCD Risk Factor Collaboration. Worldwide trends in diabetes since 1980: A pooled analysis of 751 population-based studies with 4.4 million participants. *Lancet* **2016**, *387*, 1513–1530. [PubMed]
3. Pelletier, L.; Dai, S.; Roberts, K.C.; Bienek, A.; Onysko, J.; Pelletier, C. Diabetes prevalence and care practices in Canada. *Chronic Dis. Inj. Can.* **2012**, *33*, 54–56. [CrossRef]
4. Statistics Canada. *Canadian Community Health Survey (CCHS)—Annual Component: User Guide*; Technical Report Catalogue No. 82-620-M; Statistics Canada: Ottawa, ON, Canada, 2022.
5. Kavakiotis, I.; Tsave, O.; Salifoglou, A.; Maglaveras, N.; Vlahavas, I.; Chouvarda, I. Machine learning and data mining methods in diabetes research. *Comput. Struct. Biotechnol. J.* **2017**, *15*, 104–116. [CrossRef] [PubMed]
6. Miotto, R.; Li, F.; Kidd, B.A.; Dudley, J.T. Deep patient: An unsupervised representation to predict the future of patients from the electronic health records. *Sci. Rep.* **2016**, *6*, 26094. [CrossRef] [PubMed]
7. Chen, T.; Guestrin, C. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*; ACM: New York, NY, USA, 2016; pp. 785–794.
8. Rudin, C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nat. Mach. Intell.* **2019**, *1*, 206–215. [CrossRef] [PubMed]
9. Lipton, Z.C. The mythos of model interpretability. *Queue* **2018**, *16*, 31–57. [CrossRef]
10. Breiman, L.; Friedman, J.H.; Olshen, R.A.; Stone, C.J. *Classification and Regression Trees*; Wadsworth International Group: Belmont, CA, USA, 1984.
11. Sisodia, D.; Sisodia, D.S. Prediction of diabetes using classification algorithms. *Procedia Comput. Sci.* **2018**, *132*, 1578–1585. [CrossRef]
12. Zou, Q.; Qu, K.; Luo, Y.; Yin, D.; Ju, Y.; Tang, H. Predicting diabetes mellitus with machine learning techniques. *Front. Genet.* **2018**, *9*, 515. [CrossRef] [PubMed]
13. Maniruzzaman, M.; Kumar, N.; Menhazul Abedin, M.; Islam, M.S.; Suri, H.S.; El-Baz, A.S.; Suri, J.S. Comparative approaches for classification of diabetes mellitus data: Machine learning paradigm. *Comput. Methods Programs Biomed.* **2017**, *152*, 23–34. [CrossRef] [PubMed]
14. Lasek, P.; Mei, Z. Clustering and visualization of a high-dimensional diabetes dataset. *Procedia Comput. Sci.* **2019**, *159*, 2179–2188. [CrossRef]
15. Guidotti, R.; Monreale, A.; Ruggieri, S.; Turini, F.; Giannotti, F.; Pedreschi, D. A survey of methods for explaining black box models. *ACM Comput. Surv.* **2018**, *51*, 1–42. [CrossRef]
16. Ribeiro, M.T.; Singh, S.; Guestrin, C. "Why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*; ACM: New York, NY, USA, 2016; pp. 1135–1144.
17. Lundberg, S.M.; Lee, S.I. A unified approach to interpreting model predictions. *Adv. Neural Inf. Process. Syst.* **2017**, *30* , 4765–4774.
18. Lou, Y.; Caruana, R.; Gehrke, J. Intelligible models for classification and regression. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*; ACM: New York, NY, USA, 2012; pp. 150–158.
19. Letham, B.; Rudin, C.; McCormick, T.H.; Madigan, D. Interpretable classifiers using rules and Bayesian analysis: Building a better stroke prediction model. *Ann. Appl. Stat.* **2015**, *9*, 1350–1371. [CrossRef]
20. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
21. Domingos, P. A few useful things to know about machine learning. *Commun. ACM* **2012**, *55*, 78–87. [CrossRef]
22. Kotsiantis, S.; Kanellopoulos, D. Discretization techniques: A recent survey. *GESTS Int. Trans. Comput. Sci. Eng.* **2006**, *32*, 47–58.
23. Shneiderman, B. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings 1996 IEEE Symposium on Visual Languages*; IEEE: Piscataway, NJ, USA, 1996; pp. 336–343.
24. Godfrey, P.; Gryz, J.; Lasek, P. Interactive visualization of large data sets. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 2142–2157. [CrossRef]
25. Little, R.J.A.; Rubin, D.B. *Statistical Analysis with Missing Data*, 3rd ed.; John Wiley & Sons: Hoboken, NJ, USA, 2019.
26. Hunter, J.D. Matplotlib: A 2D graphics environment. *Comput. Sci. Eng.* **2007**, *9*, 90–95. [CrossRef]
27. McNemar, Q. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika* **1947**, *12*, 153–157. [CrossRef] [PubMed]
28. Whiting, D.R.; Guariguata, L.; Weil, C.; Shaw, J. IDF diabetes atlas: Global estimates of the prevalence of diabetes for 2011 and 2030. *Diabetes Res. Clin. Pract.* **2011**, *94*, 311–321. [CrossRef] [PubMed]
29. Marmot, M. Social determinants of health inequalities. *Lancet* **2005**, *365*, 1099–1104. [CrossRef] [PubMed]
30. Willi, C.; Bodenmann, P.; Ghali, W.A.; Faris, P.D.; Cornuz, J. Active smoking and the risk of type 2 diabetes: A systematic review and meta-analysis. *JAMA* **2007**, *298*, 2654–2664. [CrossRef] [PubMed]

https://doi.org/10.3390/app16126146

31. Idler, E.L.; Benyamini, Y. Self-rated health and mortality: A review of twenty-seven community studies. *J. Health Soc. Behav.* **1997**, *38*, 21–37. [CrossRef] [PubMed]
32. Koppes, L.L.; Dekker, J.M.; Hendriks, H.F.; Bouter, L.M.; Heine, R.J. Moderate alcohol consumption lowers the risk of type 2 diabetes: A meta-analysis of prospective observational studies. *Diabetes Care* **2005**, *28*, 719–725. [CrossRef] [PubMed]
33. Stockwell, T.; Zhao, J.; Panwar, S.; Roemer, A.; Naimi, T.; Chikritzhs, T. Do "moderate" drinkers have reduced mortality risk? A systematic review and meta-analysis of alcohol consumption and all-cause mortality. *J. Stud. Alcohol Drugs* **2016**, *77*, 185–198. [CrossRef] [PubMed]
34. Naimi, T.S.; Stockwell, T.; Zhao, J.; Xuan, Z.; Dangardt, F.; Konttinen, H.; Bassett, M.T.; Chikritzhs, T. Confounding and effect modification in studies of alcohol consumption and mortality. *Open Heart* **2017**, *4*, e000526.
35. Gorber, S.C.; Schofield-Hurwitz, S.; Hardt, J.; Levasseur, G.; Tremblay, M. The accuracy of self-reported smoking: A systematic review of the relationship between self-reported and cotinine-assessed smoking status. *Nicotine Tob. Res.* **2009**, *11*, 12–24. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

https://doi.org/10.3390/app16126146

applied sciences

MDPI

Article

Using Hybrid LSTM Neural Networks to Detect Anomalies in the Fiber Tube Manufacturing Process

Zbigniew Gomolka [1,*], Ewa Zeslawska [1] and Lukasz Olbrot [2]

[1] College of Natural Sciences, University of Rzeszow, Rejtana St. 16C, 35-959 Rzeszow, Poland; ezeslawska@ur.edu.pl
[2] FIBRAIN Sp. z o.o., Zaczernie 190F, 36-062 Zaczernie, Poland; l.olbrot@fibrain.pl
* Correspondence: zgomolka@ur.edu.pl; Tel.: +48-17-851-8755

Abstract: The production process of tubes for fiber optic cables is a complex process, where proper execution is crucial to the quality of the final product. This process has a complex state vector whose structure and coordinates dynamically change during the tube extrusion process. Small fluctuations in process parameters, such as temperature, extrusion pressure, production speed, and optical fiber tension, affect the optical attenuation of the final product. Such defects necessitate the withdrawal of the product. Due to the high number of process coordinates and the technological inability to automatically label those segments of the production process that cause anomalies in the final product, the authors used data clustering methods to create a training set that enabled the use of neural tools for anomaly detection. The system proposed in the main part of the paper includes a hybrid Long short-term memory (LSTM) network model, which is fed with data streams recorded on the tube extrusion production line. The input module, which performs preprocessing of input data, conducts multiresolution analysis of recorded process parameters, and recommends the process state's belonging to a set of classes describing individual production anomalies to appropriate LSTM network modules. The learning process of the three–channel network allowed effective recognition of five classes of the monitored tube production process. The fit level of the proposed network model reached R2 values of $\geq$0.85.

Keywords: optical tubes extrusion process; data clustering; LSTM networks; anomalies detection

1. Introduction

Fiber optic cables in modern civilization are the primary medium used for efficient and reliable information transmission. The continuous expansion of computer network infrastructure drives a growing demand for such products. Solutions are expected to include both long segments of optical medium as well as multi–core and multi–mode solutions, which allow for parallelization and thus acceleration of information transmission. The basic raw material is usually optical fiber, which has specific parameters describing its attenuation, rigidity, and physical dimensions. It is obtained by process of drawing the fiber in special towers from prepared glass billets of high chemical purity, which ensures appropriately low optical attenuation. Typically, at this stage, the fiber is coated with a dedicated polymer that ensures light confinement within it. The polymer layers can be combined with a color that ensures proper separation of fibers already at the stage of digital network construction. The next stage of optical cable manufacturing in industrial

conditions is the extrusion process of the tube, which involves continuous heating and extrusion of plastic or metal materials to form hollow tubes that serve as a hydrophobic coating surrounding bundles of optical fibers. The subsequent step is the twisting of tube assemblies, and the final product of this process is an optical cable.

As can be seen, the above process is complex, with many different state coordinates, and each stage of production can potentially be a source of defects in the final product. In this context, predicting process anomalies using data from production equipment and taking preventive action is crucial. This is a key leitmotif that motivates many research teams around the world [1–3] to seek solutions and tools for monitoring this process and preventing defects that degrade the quality of the final product. Typically, the anomaly detection process involves collecting and analyzing data from sensors that monitor temperature, pressure, vibrations, chemical parameters, diameter, ovality, and other indicators specific to a given production line's state. In recent years, statistical algorithms (e.g., linear regression), machine learning algorithms (e.g., decision trees, Support Vector Machines (SVM), neural networks [2,4]), and deep learning algorithms (e.g., Recurrent Neural Networks (RNN), Autoencoders) have been used for this purpose. The primary task of these algorithms is to monitor the given process in real–time, maintain predictive maintenance [5], and ensure quality control. Currently, companies in the fiber optic cable manufacturing sector are developing various solutions to minimize the percentage of products manufactured out of specification. Among the challenges they face are the scalability and adaptability of algorithms, their ability to integrate with production systems, and their effectiveness in minimizing waste and downtime [6–12].

By reviewing the latest literature, State-of-the-Art Review on Anomaly Detection in Various Domains [13], several leading themes can be distinguished. The detection of anomalies has emerged as a critical focus across diverse domains, particularly within the Internet of Things (IoT), biomedical signal processing, industrial processes, and satellite telemetry data. This review synthesizes recent advances in methods and approaches, highlighting their unique characteristics, applications, and associated challenges. Anomaly detection in IoT systems often employs dual-view mechanisms, combining time reconstructors, frequency reconstructors, and dual-view adversarial learning [14,15]. These methods aim to analyze data streams comprehensively, addressing both temporal and spectral dimensions to enhance detection accuracy. Biomedical data, such as signals related to Parkinson's disease, are processed using unsupervised deep networks. These methods focus on uncovering latent patterns within data without labeled training samples, providing significant insights into disease recognition while addressing specific anomaly types. The use of singular value decomposition (SVD) in deep neural network (DNN) models aids in encrypting hidden features of signals. Techniques incorporating Fourier transforms allow for robust analysis of anomalies while maintaining data integrity [16]. Recent developments include the implementation of self-adversarial variational autoencoders (VAEs) integrated with contrastive learning to enhance anomaly detection capabilities [16]. These architectures excel in capturing nuanced differences between normal and anomalous states. Multi-node knowledge graphs fused with expert knowledge and Bi-directional Long Short-Term Memory (Bi-LSTM) networks have been proposed [17] as powerful tools for local feature extraction in anomaly detection. This approach leverages both structural and temporal relationships within the data. The multi-scale sparse attention module effectively extracts global features across scales, preserving pattern information critical for detecting process disturbances [18]. This approach is particularly adept at handling data from industrial monitoring systems [19]. For multidimensional processes, local feature masking can amplify state differences to visually highlight anomalies [20]. Similarly, mechanisms for identifying highly divergent

patterns based on data density help streamline the detection process [21]. A knowledge distillation approach [22], transferring insights from large models to smaller, more efficient models, has proven effective in speeding up anomaly detection. Stochastic approaches utilizing partial Markov chains are employed to mask irrelevant process features, such as in image data [23]. These methods improve reconstruction quality and focus anomaly detection efforts on significant features. An imbalance in training datasets is a persistent issue, especially in industrial processes where generating anomalies artificially is infeasible [24]. Few-shot anomaly detection using positive-unlabeled learning addresses this by preparing well-structured datasets, albeit with challenges in real-world implementation [25]. Preprocessing transformations, such as Kalman filtering, are used to simplify input data for anomaly detection in diverse systems, including ECG signals and cooling devices [26]. Additionally, federated learning enhances defect detection, achieving high accuracy, precision, recall, and F1 scores in certain industrial settings [24]. Frequency-domain techniques generate augmented pseudo-anomalous images, allowing for the discovery of key anomaly characteristics [27]. Analyzing short video sequences of industrial processes has achieved notable success, such as the MSTE environment yielding 90.7% accuracy [28]. However, video-based methods face limitations due to measurement constraints and environmental variability. In satellite telemetry anomaly detection, deep learning algorithms contend with significant noise and data gaps [29]. To address data scarcity in anomaly states, mechanisms such as feature injection and heterogeneous spatiotemporal graphs have been developed [30]. These techniques enable artificial supplementation of training datasets, improving detection accuracy in processes with high anomaly deficits. In unsupervised scenarios, attributed networks leveraging graph-based models enhance the differentiation between normal states and anomalies by reducing interference [31].

In the study [6], anomaly detection in smart factories was explored using machine learning techniques, particularly addressing challenges related to diverse sensors and their transferability across different production lines. A combination of machine learning algorithms, including neural networks and transfer learning, was employed to facilitate knowledge transfer between production lines. The authors achieved high accuracy in anomaly detection using datasets from various sensors, and the knowledge transfer between sensors enhanced system effectiveness in new, unfamiliar production environments. The study [10] proposed the creation of a digital twin for a production line, enabling improved monitoring and real-time anomaly prediction. This approach integrated digital twin modeling with machine learning algorithms, such as Long Short-Term Memory (LSTM) networks and autoencoders. The system demonstrated high accuracy in anomaly prediction, while the digital twin allowed for dynamic production parameter adjustments, reducing defects by 40% compared to standard methods. In [5], the authors introduced a system for predicting production costs in the context of Industry 4.0. This system integrated data from multiple sources, including industrial sensors and ERP systems, to forecast changes in production costs. Advanced ML models, such as LSTM networks and decision trees, were applied to analyze time-series data from various sources. The model achieved a 92% accuracy rate in production cost predictions, significantly reducing financial losses from unplanned downtimes and production defects. The study [7] focused on anomaly detection in optical fiber networks using machine learning algorithms. The primary application involved monitoring the integrity of optical cables and detecting damages in communication networks. Support Vector Machines (SVM) and neural networks, including LSTM models, were used to process sensor data and detect cable irregularities. The LSTM model excelled in real-time anomaly detection, achieving 94% accuracy, and effectively identified both minor and major damages. Finally, Ref. [12] investigated automatic anomaly detection in

manufacturing machines based on sensor data, aiming to minimize machine failures and downtimes by predicting potential issues. Statistical algorithms such as linear regression and k-Nearest Neighbors (k-NN) were combined with advanced machine learning methods, including LSTM networks. These models successfully identified anomalies in 87% of cases, leading to a 25% reduction in production process downtimes.

Table 1 presents a summary of the technologies employed and the performance of the models in the context of research on anomaly detection and production process monitoring.

Table 1. Summary of key elements, applied models, and results for selected works on anomaly detection in production processes.

Authors (Year)	Research Description	Applied Models	Key Results
Abdallah et al. (2023) [6]	Anomaly detection in smart factories with sensor-to-sensor transfer	LSTM, Transfer learning, Neural networks	High anomaly detection accuracy, effective knowledge transfer between sensors
Kakavandi et al. (2023) [10]	Digital twin for real-time production line monitoring	Digital twin, LSTM, Autoencoders	40% defect reduction, dynamic adjustment of production parameters
Soleimani et al. (2023) [5]	Production cost prediction in Industry 4.0	LSTM, Decision trees	92% prediction accuracy for costs, reduction of defect-related losses
Abdelli et al. (2022) [7]	Anomaly detection in fiber optic monitoring	LSTM, SVM	94% anomaly detection accuracy, effective detection of optical cable damages
Pittino et al. (2020) [12]	Automatic anomaly detection in production machines	LSTM, Linear regression, k–NN	87% anomaly detection accuracy, 25% downtime reduction

Regarding above, the industry practice shows, that often heuristic strategies are needed to overcome these industrial challenges, which frequently hinder effective detection [32,33]. Emerging trends such as federated learning, knowledge graphs, and advanced neural architectures offer promising directions for overcoming these challenges. By integrating domain expertise with sophisticated computational techniques, anomaly detection continues to evolve, bridging gaps across diverse applications. Although the recent reviews have attempted to unify diverse methods to facilitate objective comparisons, but unfortunately significant challenges persist due to process-specific measurement differences and the unique nature of anomalies across domains, which often hinder generalization [34–40].

In this study, the authors refer to the concept of anomalies as a set of various disturbances that may accompany the production process. These deviations from the correct process, which has its proper signature representing a normal and unthreatened production state, can become, in the absence of corrective actions, a potential source of defects in the final product. This may result in either a reduction in the product's final quality or, in a worst-case scenario, render it unusable for further application.

This paper presents the results obtained from monitoring the tube extrusion process based on actual measurements recorded on the production line at the industrial plant FIBRAIN Sp. z o.o. This company specializes in the production and delivery of fiber optic technology solutions and is one of the leading manufacturers of fiber optic cables and telecommunication components in Central and Eastern Europe. The designed anomaly detection system allows for the reduction of disposal and repair costs for tube sections produced out of specification on the tube extrusion production line. The system described in the main section of the paper incorporates a hybrid Long Short-Term Memory (LSTM) network model, designed to process data streams collected from a tube extrusion production line. The input module, responsible for preprocessing, performs multiresolution analysis of the recorded process parameters. It subsequently assigns the current process state to one of the predefined classes, representing specific production anomalies, and

directs it to the appropriate LSTM network modules. The three-channel network's training process enabled the effective identification of five distinct classes within the monitored tube production process.

2. Materials and Methods

Given the current and continuously growing scale of production, the number of occurrences of various defects disqualifying a given fiber optic cable is a source of significant financial losses for the manufacturer in the form of wasted resources, the cost of disposing of defective cables, or their potential repair. The entire research work encompassed three different production lines, as shown in Figure 1: the RLR line for tube extrusion (a), the RLV line for coating extrusion on the cable core (b), and the RLM line for twisting cable cores from tubes (c).



(**a**) (**b**) (**c**)

Figure 1. The production lines used by FIBRAIN Sp. z o.o.: RLR (**a**), RLV (**b**) oraz RLM (**c**).

These are stages where both the number and type of sensors used make it impossible to implement a uniform system that, in a hybrid form, would simultaneously record and detect anomalies from different processes. Therefore, at the current stage of work, the considerations presented by the research team focus on the production stage involving a single RLM production line.

In this paper, a hybrid LSTM network model is presented, which pertains to measurements from the RLM line. The other production lines are subjects of ongoing research and will be discussed in future works focusing on the engineering applications of AI tools. The RLM line is responsible for the production of micro–module cables with fiberglass rods (FRP) (MDC–FM, AERO–FM), aerial cables (ADSS, AERO–AS), road cables reinforced with fiberglass or metal rods in the cable sheath (AERO–DDF, Burry, VC–Tx), easy access cables (EAC), duct cables (BDC), micro cables (MK–LX), central tube cables (EXO), and steel tape armored cables (SST).

2.1. Factory RLM Line Data Acquisition

As part of the conducted work, the line was equipped with devices that allow for monitoring and recording the tension of individual modules or tight tubes, as well as the temperature of the process water located directly in the cooling baths. Based on the recommendations of tube manufacturing technology experts, 232 measurement points were designated, where the production process was recorded at a frequency of $1[s]$. This set included both the established process recipe parameters and process variables. The complete data set contained 180 randomly recorded production sessions lasting from 30 min

to 150 min, corresponding to different working hours of the line and the teams of operators handling it. An example fragment of measurement data from a single session is presented in Table 2. This includes a small range of approximately a one–hour measurement session in which 20 km of optical cable was produced.

Table 2. Sample of process data recorded for RLM line.

Time Stamps 29 November 2023	BAZ1 _iTens	BAZ2 _iLoad	BAZ2 _iMetLo	BAZ2 _iSpeed	EXT1 _iLoad	EXT1 _iSpeed	⋯	SPE2 _iLoad	SPE2 _iSpeed
22:17:14	0.274658	6.427	11	53.9844	33.5266	30.957	⋯	15.4602	55.0488
22:17:15	0.219727	6.38428	21	53.9844	33.783	30.957	⋯	17.218	55.0879
22:17:16	0.183106	6.49414	30	53.9941	33.5205	30.9961	⋯	16.5955	54.7461
22:17:17	0.146484	6.46973	39	54.043	33.5571	30.957	⋯	15.5945	54.6094
22:17:18	0.164795	6.51245	48	53.9941	33.7036	30.9473	⋯	16.9556	54.4238
22:17:19	0.201416	6.5918	59	54.0234	33.7341	30.957	⋯	16.1804	54.209
22:17:20	0.201416	6.5918	68	54.0234	33.5022	30.9668	⋯	16.6687	54.209
22:17:21	0.274658	6.37817	77	53.9551	33.5449	30.9668	⋯	17.4622	54.1406
22:17:22	0.274658	6.50024	86	53.9844	33.8196	30.918	⋯	16.2292	54.3359
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
22:25:29	0.128174	6.79321	4644	54.9609	34.4177	31.6602	⋯	3.62549	−0.2929
22:25:30	0.366211	6.86035	4654	54.9707	34.1492	31.6113	⋯	3.62549	−0.2929
22:25:31	0.1	7.00073	4663	54.9902	34.2896	31.6113	⋯	3.62549	−0.2929
22:25:32	0.1	6.98242	4673	54.9707	34.1919	31.6016	⋯	3.62549	−0.2929
22:25:33	0.146484	7.04346	4682	54.9707	34.3018	31.5723	⋯	3.62549	−0.2929
22:25:34	0.146484	6.92749	4691	54.9805	34.0881	31.6211	⋯	3.62549	−0.2929
22:25:35	0.146484	6.88477	4701	55.0293	34.3445	31.6309	⋯	3.62549	−0.2929
22:25:36	0.128174	6.88477	4711	54.9805	34.0271	31.6504	⋯	3.62549	−0.2929
22:25:37	0.1	6.89697	4721	55.0098	34.1125	31.5723	⋯	3.62549	−0.2929
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
22:53:52	0.347901	7.88574	23544	79.9707	38.7207	46.8945	⋯	3.62549	−0.2929
22:53:53	0.347901	7.72095	23558	80.0195	38.4277	46.8945	⋯	3.62549	−0.2929
22:53:54	0.347901	7.72095	23571	80.0195	38.4277	46.9531	⋯	3.62549	−0.2929
22:53:55	0.219727	7.73315	23585	80.0586	38.269	46.9336	⋯	3.62549	−0.2929
22:53:56	0.219727	7.69043	23599	79.9805	38.855	46.9922	⋯	3.62549	−0.2929
22:53:57	0.366211	7.67212	23612	79.9902	38.3606	46.9629	⋯	3.62549	−0.2929
22:54:03	0.146484	7.59277	23696	80	38.8367	47.002	⋯	3.62549	−0.2929
22:54:04	0.347901	7.45239	23710	80.0391	38.6475	46.9336	⋯	3.62549	−0.2929
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
23:16:44	0.109863	6.75659	40728	53.9746	33.252	31.3477	⋯	3.62549	−0.2929
23:16:45	0.238037	6.71997	40737	53.9941	33.4595	31.3086	⋯	3.62549	−0.2929
23:16:46	0.384522	6.79932	40746	54.0137	33.3801	31.2598	⋯	3.62549	−0.2929
23:16:47	0.1	6.93359	40755	54.0039	33.5266	31.2598	⋯	3.62549	−0.2929
23:16:48	0.1	6.92749	40766	53.9941	33.1848	31.2012	⋯	−0.00610	−0.1464

During the conducted work related to recording measurement data, the team observed random gaps in the process data. These gaps were caused either by improper operator handling of the production line or by malfunctions in individual sensors. Naturally, there are fundamentally different strategies to counteract such issues, ranging from simple methods of discarding damaged data sequences to advanced strategies for repairing them using sophisticated statistical methods, machine learning, or interpolation and extrapolation techniques. Each approach has a specific level of accuracy and computational complexity, which can be crucial when selecting the appropriate method for various industrial applications.

For the experiments involving the RLM line, the method of extrapolating missing data ("gaps") was adopted, when the size of the gaps did not exceeded 10 s. In cases where the negative effect was more pronounced, the affected data packets were discarded. Considering the strict technological conditions, the occurrence of missing data or abrupt changes in the recorded values was extremely rare.

These data were subsequently analyzed using a sliding window with sizes of [64, 128, 256, 512, 1024]. Considering the dynamics of the manufacturing process, it was determined

that the most optimal window size, from a technological standpoint and accounting for the possible reaction time of the line operator in the hypothetical event of an anomaly presence, would be approximately 4 min, corresponding to a 256-element window. Other time intervals might also highlight interesting properties of the process; however, for the purposes of this study and its limited scope, they were excluded from further consideration.

2.2. Preprocessing of Input Data for Training Set

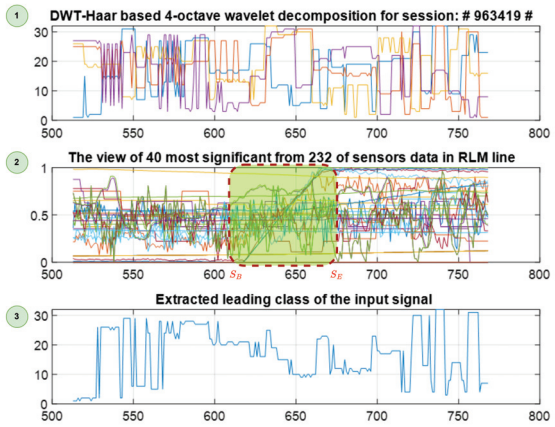
A fundamental problem faced by research teams working on anomaly detection is the inability to precisely construct a training sequence, due to the fact that the vast majority of defects caused by anomalies are detected only after the completion of the production process during the final product quality checks. Usually, determining the precise temporal coincidence of a specific moment in the production process is either impossible or associated with a large spatial localization error. To address this, an initial assessment of the input data was performed by analyzing the significance of features in the recorded process state vector using the Discrete Wavelet Transform (DWT) with a Haar wavelet basis function, applying a 4–octave wavelet decomposition level. This resulted in sets of 32 features for successive windows, which illustrate different process states. For the purposes of further discussion, the following mathematical notation have been assumed. Let $x_i(t)$ represent the discrete input signal registered from the i-th sensor channel of the RLM production line, where: $i = 1, 2, \ldots, S_N$, $t = 0, 1, 2, \ldots, W_s$, $S_N = 232$—denotes the number of mounted sensors in RLM line, $W_s = 256$—denotes the window size, (i.e., 256 time stamps in the moving window of length 256). The matrix representation of the complete signal data from all sensors of consecutive channels within the moving window can be represented as a matrix:

$$X(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \cdots \\ x_{S_N}(t) \end{bmatrix} \tag{1}$$

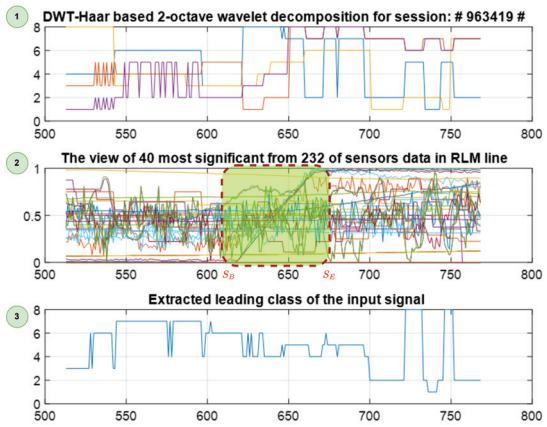
where $X(t) \in R^{S_N \times W_S}$. The moving window mechanism we assume respectively: t_w denote the moving window size (here $t_w = W_S$). For every sensors channels i, the windowed segment can be expressed to as: $x_i(t) = \{x_i(k) \mid k = t,\ t+1,\ \ldots,\ t+t_w-1\}$. The initial research assumptions were that a pyramidal wavelet decomposition would be performed on the process coordinates so recorded, which, due to the absence of constraints on the periodicity and continuity of the signal characteristic of Fourier frequency decomposition, allows the extraction of selected components of the transformed signal into complementary components. For this purpose, we used the algorithm proposed by [41] which uses signal decomposition $x_i(t)$ into approximations and details at various levels j: $j : DWT\{x_i(t)\} = \{A_j, D_j\}_{j=1}^{J}$, where A_j is the approximation coefficients at level j, D_j is the detail coefficients at level j and J is the maximum decomposition level, depending on the length of $x_i(t)$. Taking into account the technological requirements determined by the size of the analyzed window, it was assumed for the purposes of the preliminary analysis that the $J \in \{2, 3, 4, 5, 6\}$. The multi-sensor processing with DWT is applied to each feed individually:

$$DWT\{X(t)\} = \begin{bmatrix} DWT\{x_1(t)\} \\ DWT\{x_2(t)\} \\ \cdots \\ DWT\{x_{S_N}(t)\} \end{bmatrix} \tag{2}$$

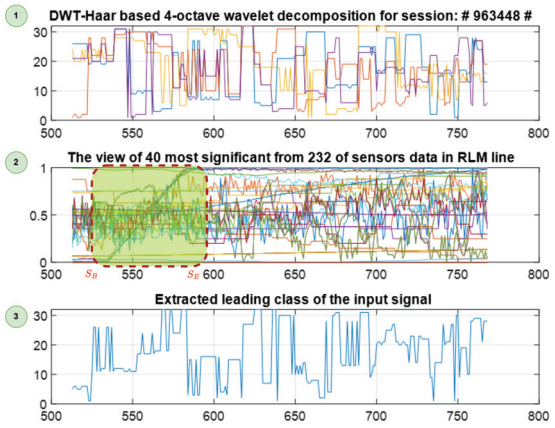
Presented above mathematical framework ensures clarity in processing, accommodating the 232 sensor channels, 256—sample moving window, and 1—second sampling interval. Below (Figure 2) are sample fragments of the signal recorded from the RLM production line for wavelet decomposition using the Haar mother wavelet with 2 and 4 octave sizes, respectively.



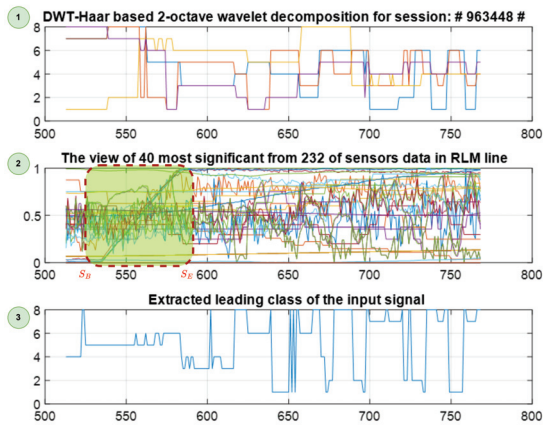
(**a**)



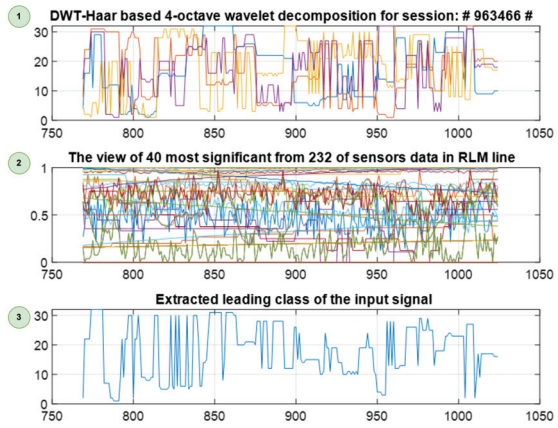
(**b**)



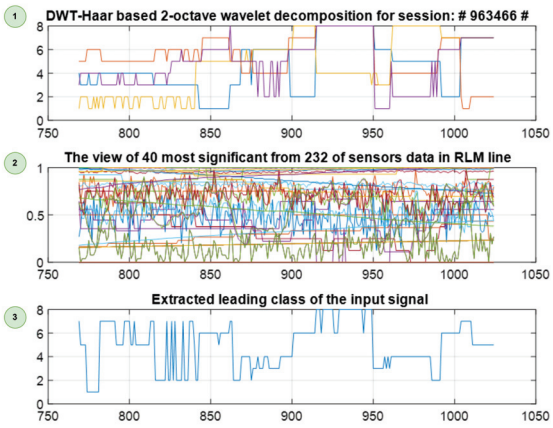
(**c**)



(**d**)



(**e**)



(**f**)

Figure 2. *Cont*.

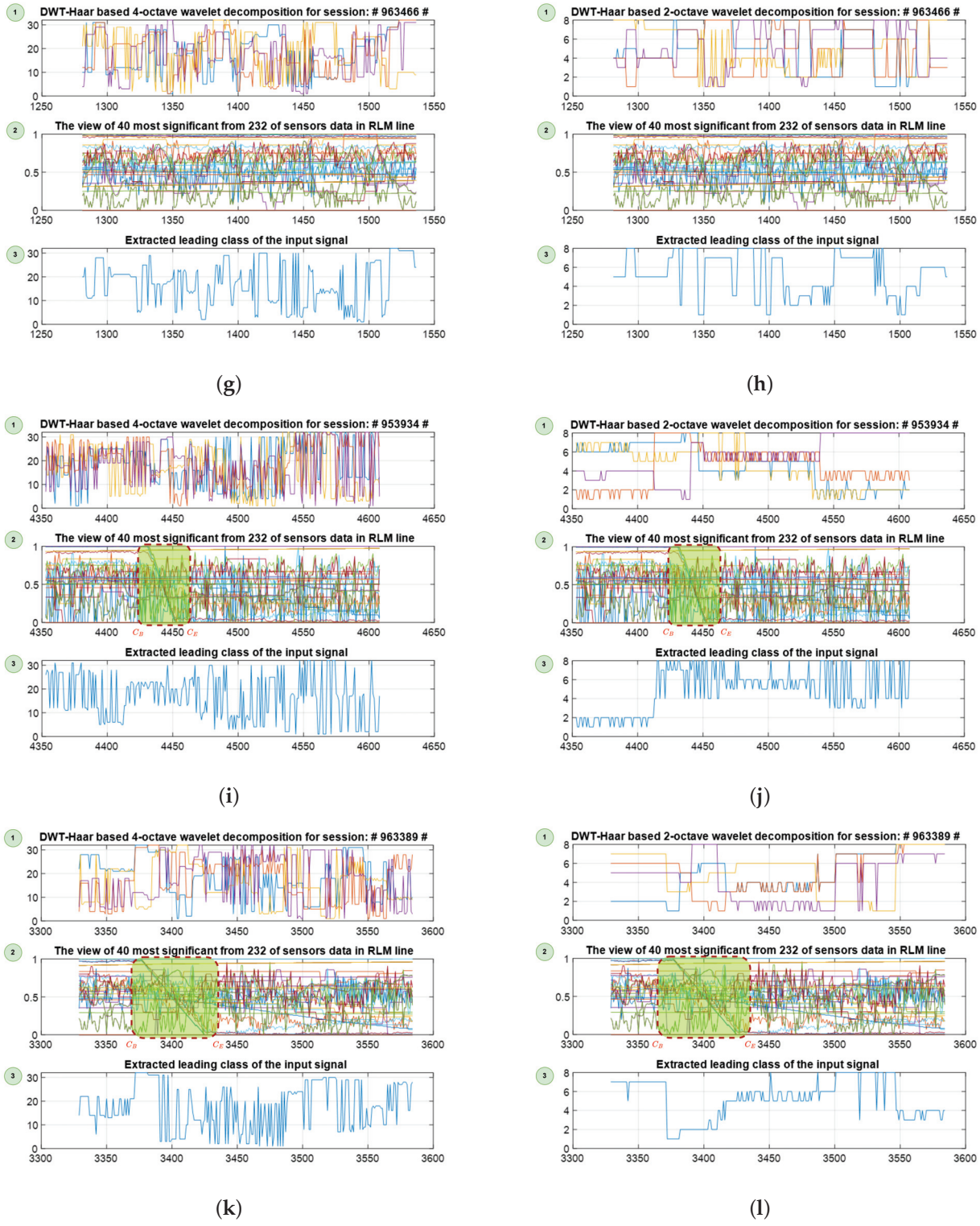


Figure 2. Preliminary analysis of the manufacturing process aimed at using the Haar-based DWT transform clustering algorithm.

For all waveforms in Figure 2, the designations (1), (2) and (3) stand for: the waveform of the components $\{A_j, D_j\}$ of the DWT transform, the waveforms of the leading 40 components of the production process, and the membership of the RLM line signal to the different classes extracted in the clustering process. The left column (Figure 2a,c,e,g,i,k) corresponds to the decomposition of the signal recorded during the measurement session from the RLM production line using a 4–octave Haar–based DWT, while the right column (Figure 2b,d,f,h,j,l) corresponds to the decomposition of the same signal using a 2–octave Haar–based DWT. The waveforms (Figure 2a–d) show example sections of RLM line acceleration, in which the time interval $\langle S_B, \ldots, S_E \rangle$ is marked with a rectangular area. The symbols S_B and S_E, denote the start of the fiber twisting process and the end of the twisting process of a given tube section, respectively. Graphs (Figure 2e–h) contain waveforms for the ongoing basic production process also for two different randomly selected sessions. The recorded process data from the RLM line $X \in R^{S_N \times W_S}$, were normalized $\{a = 0, b = 1\}$ for each signal channel data independently:

$$x_{i,j,\text{norm}} = a + (b - a) \cdot \frac{x_{i,j} - min(X_{i,:})}{max(X_{i,:}) - min(X_{i,:})} \tag{3}$$

where: X_i is the i-th row of X. The plots (Figure 2i–l) show sample stages of the finalization of the fiber twisting process for a given tube section, marked as $\langle C_B, \ldots, C_E \rangle$, where the symbols C_B and C_E denote the start and end of the production completion process for the given tube section, respectively. All plots marked with the number (3) represent an attempt to visualize the potential membership of the production process state to a priori selected number of classes.

Subsequently, a univariate feature selection algorithm for classification using the Chi–square test was applied. This allowed for the visualization and assessment of the importance of individual leading features in the recorded data stream, represented as a cluster structure corresponding to 32 different process states, as shown in Figure 3. The Table 3 contains the coordinates and names of the individual leading components of the process.

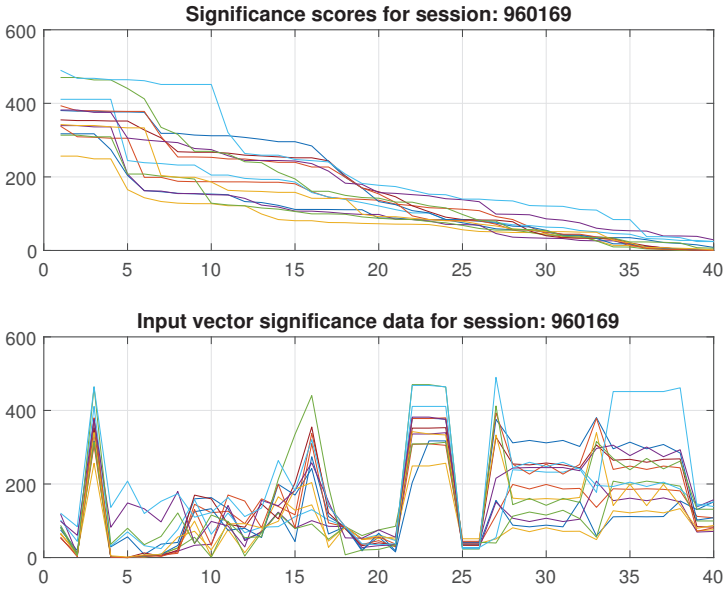


Figure 3. Input data significance distribution.

Table 3. Corresponding process coordinates names.

Column Indexes	Name
{[4]}	{'BAZ1_iSpeed'}
{[30]}	{'KAP1_iTens17_20'}
{[31]}	{'KAP1_iTens21_24'}
{[32]}	{'KAP1_iTens5_8'}
{[57]}	{'POF1_iDancMode_N'}
{[58]}	{'POF1_iFltCodeTrav'}
{[59]}	{'POF1_iLoad'}
{[60]}	{'POF1_iReelDiam'}
{[61]}	{'POF1_iSpeed'}
{[63]}	{'QSD2_ValDel'}
{[49]}	{'MES_MeterCnt'}
{[56]}	{'POF1_iDancerPos'}
{[50]}	{'MES_MeterCnt2'}
{[52]}	{'MES_OKaltRe'}
{[40]}	{'MES_DWarmY'}

In the course of observing the impact of individual process characteristics on cluster distribution, it was found that it is possible to use an automatic clustering mechanism to construct a training set for Deep Neural Networks (DNN). At the same time, it was noted that the DWT showed high sensitivity to momentary impacts of some parameters, such as the state of the cable length counter when changing the winding drum. Such a phenomenon caused strong interference, which negatively affected the segmentation process of individual process states. In addition, for analytical purposes, the authors attempted to graphically interpret the distribution of individual clusters and the distance between them, using the Hierarchical Clustering with Binary Clustering Tree method. Assuming that the given is a dataset with n observations, where each observation is represented as a d-dimensional vector $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{id})$, the steps for agglomerative hierarchical clustering we obtained by iterative merging: by serching for the pair of clusters (C_i, C_j) with the smallest distance $d(C_i, C_j)$. Then merging clusters C_i and C_j into a new cluster C_{ij} and eventually update the distance matrix to reflect the distance between consecutive new cluster C_{ij} and all other. As the linkage criteria we assumed the average distance method to measure the distance between clusters:

$$d(C_i, C_j) = \frac{1}{|C_i||C_j|} \sum_{\mathbf{x}_a \in C_i} \sum_{\mathbf{x}_b \in C_j} d(\mathbf{x}_a, \mathbf{x}_b) \tag{4}$$

By the repetition of merging loop we obtained a Binary Clustering Tree of the monitored process clusters. This preliminary preprocessing of RLM line data we performed for all sessions and the resulting dendrogram and 3D view of the most significant components are shown on the Figure 4 respectively.

As we can observe a binary clustering tree is as a kind of dendrogram, where each node represents a cluster, and each leaf node represents an individual observation. The height of the nodes in the tree corresponds to the distance (or dissimilarity) at which consecutive clusters are merged. A spatial visualization of the location of the leading features of the clusters is provided in Figure 4b. As can be seen, there is a classic "energy leakage effect" causing the extracted classes of process states to interpenetrate and individual clusters to differ to a very small degree. Therefore, a multi–resolution K—means Clustering mechanism was adopted in later part of this work to cluster the input data at different resolutions of the space of feature channels $Ch_j \in \{8, 16, 32\}$, for $j = 1, 2, 3$. So, having an RLM line dataset with $n = 256$ observations for each stepping window, where each observation is represented as a d-dimensional vector we have:

$$\mathbf{x}_i^{Ch_j} = \left(x_{i1}^{Ch_j}, x_{i2}^{Ch_j}, \ldots, x_{id}^{Ch_j}\right) \tag{5}$$

Then we performed K—means clustering in each of the channels to partition the observations into channel independent clusters:

$$K^{Ch_j} \in \left\{C_1^{Ch_j}, C_2^{Ch_j}, \ldots, C_K^{Ch_j}\right\} \tag{6}$$

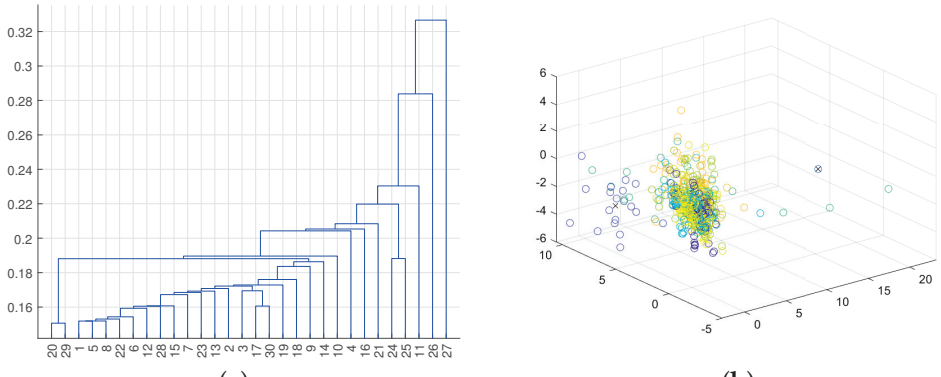
such that sum of squared distances between observations and their corresponding cluster centroids is minimized. Given $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_k$, as initial cluster centroids we assign each observation $\mathbf{x}_i^{Ch_j}$ to the cluster:

$$C_k^{Ch_j} = \left\{\mathbf{x}_i^{Ch_j} \mid \arg\min_k \| \mathbf{x}_i^{Ch_j} - \boldsymbol{\mu}_k \|^2\right\}, k = 1, \ldots, Ch_j \tag{7}$$

where $\| \mathbf{x}_i^{Ch_j} - \boldsymbol{\mu}_k \|$ is the Euclidean distance between observation $\mathbf{x}_i^{Ch_j}$ and centroid $\boldsymbol{\mu}_k$. The update of the centroid for each cluster in a given channel Ch_j, is calculated as a mean of the observations assigned to that cluster:

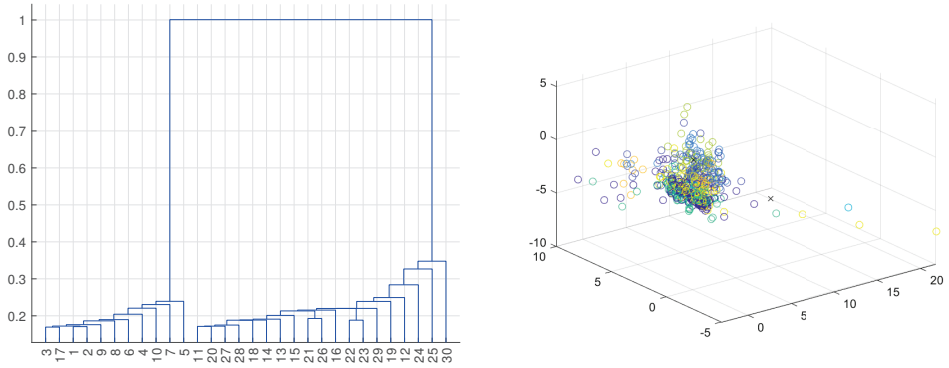
$$\boldsymbol{\mu}_k = \frac{1}{|C_k^{Ch_j}|} \sum_{\mathbf{x}_i^{Ch_j} \in C_k^{Ch_j}} \mathbf{x}_i \tag{8}$$

where $|C_k^{Ch_j}|$ is the number of observations in cluster $C_k^{Ch_j}$. This way of preprocessing the input data was intended to maximize the disparity of the process states in order to facilitate the task of building a training set for the LSTM network. One of the key difficulties that arises in anomaly detection tasks is that technological limitations do not allow precise labeling of training set classes. Therefore, an analysis of individual measurement sessions was carried out in cooperation with RLM line technologists. As a result of it, it was indicated after which sessions the quality procedure checks showed any defects in the final product. In this way, it was possible to estimate the index of potential defect for each measurement session: $p = 1$ for normal production, $p = 0$ for the interval of the production process in which any defects were detected during the post manufacturing quality control in a given measurement session. A dendrogram and a spatial visualization of the most relevant features were constructed again for such a completed input data set, see Figure 5.



(**a**) (**b**)

Figure 4. Production RLM line data feautures. The visualisation of the hierarchical, binary cluster tree (**a**), 3D view of clusters first three principal components (**b**).



(**a**) (**b**)

Figure 5. Production RLM line data feautures with bad sessions indicator *p* component. The hierarchical, binary cluster tree (**a**) and 3D view of clusters first three principal components (**b**) respectively.

By incorporating this additional indicator for the rough spatial localization of defects, a sufficiently effective mechanism for distinguishing anomalies responsible for defects in the final product was achieved. The described data preprocessing stage was used to construct the training dataset.

2.3. Hybrid LSTM Network Model for Anomaly Detection

To process the data stream recorded during the manufacturing of fiber optic cables, which are subjected to *K*—means clustering in three channels simultaneously, a dedicated model of a hybrid LSTM network has been proposed. Its task is to analyze the states of the production line in each of the channels, see Figure 6. The integrating module of the network is an aggregating layer along with a set of auxiliary layers, whose task is to detect anomalies based on the recommendations of the three predictors operating in each of the channels.

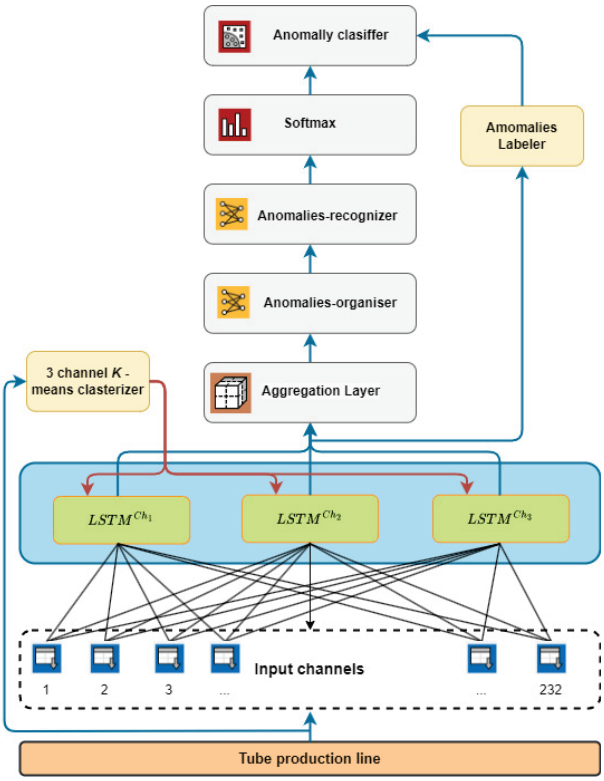


Figure 6. The diagram of the hybrid LSTM network designed for the classification of production process states.

Considering the three–channel architecture of the LSTM network, the following single cell model was used, see Figure 7.

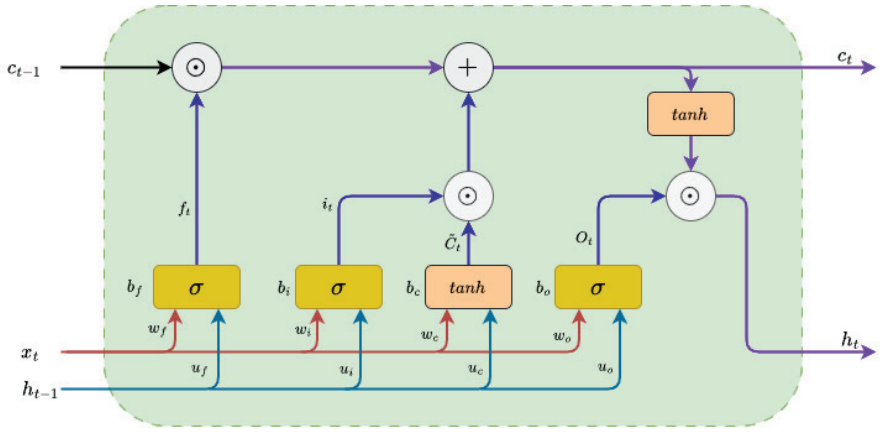


Figure 7. LSTM cell from the given channel layer.

Assuming that the input data of a single cell at time step t is the input vector x_t, the mathematical model for a fixed time step enabling the update of the memory state c_t is as follows:

$$f_t^{Ch} = \sigma\left(u_f h_{t-1} + w_f x_t + b_f\right) \tag{9}$$

$$i_t^{Ch} = \sigma(u_i h_{t-1} + w_i x_t + b_i) \tag{10}$$

$$\tilde{c}_t^{Ch} = tanh(u_c h_{t-1} + w_c x_t + b_c) \tag{11}$$

$$o_t^{Ch} = \sigma(u_o h_{t-1} + w_o x_t + b_o) \tag{12}$$

$$h_t^{Ch} = tanh(o_t \odot c_t) \tag{13}$$

$$c_t^{Ch} = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{14}$$

and the generalized matrix is written in the form:

$$W^{Ch} = \left[w_f, w_i, w_c, w_o\right]^T \tag{15}$$

where: σ—is the sigmoid activation function, h_{t-1}—previous hidden state and $tanh$ is the hyperbolic tangent activation function and W^{Ch} denotes the given channel weights set. Figure 8 represents the structure of a single LSTM layer operating in each of the channels.

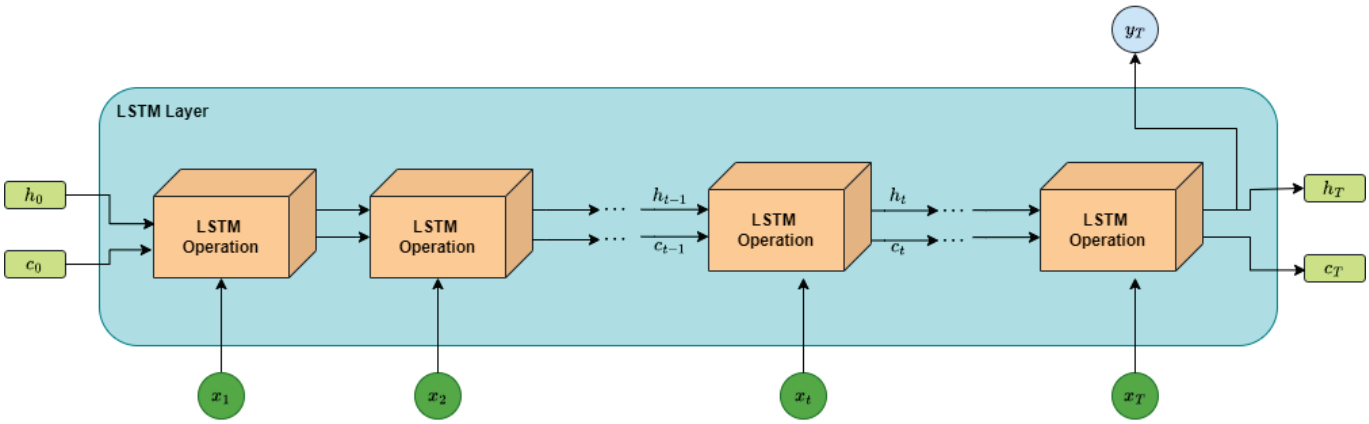


Figure 8. LSTM Layer Diagram.

Assuming the clustering mechanism described by Equation (7), we obtain the relationship for the expected values of the signature at time t of the industrial process on the RLM line:

$$y^{Ch_j}(t) = C^k_{Ch_j}(X(t)) \tag{16}$$

As in the supervised training model the goal of the LSTM model is to learn a function $f : R^{S_N \times W_S} \rightarrow \left\{1, 2, \ldots, C_k^{Ch_j}\right\}$ such that the output $\hat{y}^{Ch_j}(t)$ predicted by the model is as close as possible to the true label $y^{Ch_j}(t)$. This we achieved by minimizing the loss function $L\left(W^{Ch}\right)$:

$$L\left(W^{Ch}\right) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{C_k^{Ch_j}} y^{Ch_j} log\left(\hat{y}^{Ch_j}\left(W^{Ch}\right)\right) \tag{17}$$

where W^{Ch} represents the weight set collection sought and N denotes the number of samples in the learning set.

Individual channels of $LSTM^{Ch_{(8,16,32)}}$ have the following numerical details, see Tables 4 and 5.

Table 4. Hybrid LSTM networks parameters Channels 1–3.

Name	Type	Activations	Learnables	
Seq 1	Sequence input	232	–	
Seq 1	LSTM	512	Input Weights Recurrent Weights Bias	2048 × 232 2048 × 512 2048 × 1
Seq 3	Full Connected	8/16/32	Weights Bias	8/16/32 × 512 8/16/32 × 32
Seq 4	Softmax	8/16/32	–	
Seq 5	Classification Output	8/16/32	–	

Table 5. Hybrid LSTM networks parameters Aggregation Pipeline.

Name	Type	Activations	Learnables	
Channel 16	Feature input	1	–	
Channel 32	Feature input	1	–	
Channel 8	Feature input	1	–	
Aggregation	Concatenation	3	–	
Anomalies-organiser	Full Connected	50	Weights Bias	50 × 3 50 × 1
Anomalies-recognizer	Full Connected	5	Weights Bias	5 × 50 5 × 1
Softmax	Softmax	5	–	
Anommaly	Classification Output	5	–	

For the chosen LSTM network model, a training set was constructed with parameters summarized in Table 6. They represent the time interval from which individual measurement sessions originate. In total, recorded coordinates of the production process and extracted classes corresponding to individual diagnostic states and the actual production state were used.

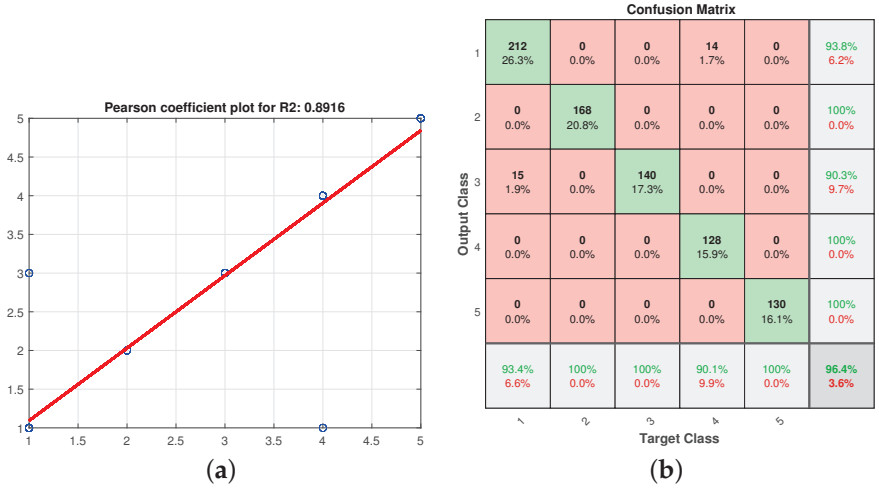
Table 6. Training data set information.

Name	Data	
Production line	RLM	
Session measurement dates	Start time 1 June 2021	End time 1 December 2023
Total measurement time	688,896 [s], ∼191 [h]	
Number of measurement points	232	
Size of the moving window	256 [s]	
Window shift step	5	
Number of samples in the learning set	2691	
Training set size	70%	
Test set size	30%	

3. Results and Discussion

The training process of the hybrid LSTM network was conducted independently in each of the submodules $LSTM^{Ch_{(8,16,32)}}$. Subsequently, for the frozen weight sets $W^{Ch_{(8,16,32)}}$, a convergent learning process of the aggregating module was carried out using the Anomalies labeler block (see Figure 6). The course of this process and the achieved degree of model fit for the entire LSTM network are shown in Figure 9. The proportion of the training set split into training and testing sets was set at 70% and 30%, respectively. The obtained model fit coefficient $R^2 > 0.85$ demonstrates the high efficiency of the model on the tested data set.

```
RLM production line
Root Mean Squared Error:  0.475
F-statistic vs.  constant model:  6.62 × 10^3, p-value = 0
R =
1.0000 0.9442
0.9442 1.0000
Obtained Pearson correlation coefficients:  0.9442
Obtained R2 determination coefficient:  0.892
```



(**a**) (**b**)

Figure 9. Pearson coefficient (**a**) and Confusion matrix (**b**).

To evaluate the effectiveness of anomaly detection, a detailed experimental study was conducted using identical data sets, with models: Random Forest, SVM, and a simple RNN as benchmarks. The performance of each of these models was evaluated using standard metrics, including Precision, Recall, F-1 Score, and R^2, which allow an objective assessment of the quality of anomaly detection and prediction [42]. Detailed results are shown in Table 7.

Table 7. Summary of results for each forecasting model, taking into account Precision, Recall, F-1 Score and R^2 indicators (the higher the values, the better the performance).

Metric	LSTM	Random Forest	SVM	RNN
Precision	**>0.94**	~0.85	~0.88	~0.87
Recall	**>0.96**	~0.83	~0.85	~0.86
F–1 Score	**>0.91**	~0.84	~0.86	~0.86
R^2	**>0.86**	~0.85	~0.85	~0.85

The obtained results show that LSTM significantly outperforms other models, particularly in the context of detecting complex temporal and sequential dependencies. The Precision and Recall metrics for LSTM exceed 0.94 and 0.96, respectively, while the F1-score is >0.91, indicating a balance between the model's accuracy and sensitivity. To thoroughly evaluate the effectiveness of the proposed model, a detailed performance analysis for individual classes was conducted using Precision, Recall, and F1-score metrics. The results, presented in Table 8, reveal that Proper achieved the highest Precision (0.9414) and Recall (0.9695), indicating an excellent fit of the model to this class. In contrast, the lowest Precision (0.6647) and Recall (0.7513) values were observed for Anomaly 4, highlighting some challenges for the model in detecting anomalies in this class.

Table 8. Performance indicator results for each class.

Name	Precision	Recall	F1-Score
Proper	0.9414	0.9695	0.9553
Anomaly 1	0.9180	0.9438	0.9307
Anomaly 2	0.8554	0.7513	0.7997
Anomaly 3	0.9185	0.9185	0.9185
Anomaly 4	0.6647	0.8248	0.7368

The experimental results confirm that the proposed hybrid LSTM model outperforms other approaches in terms of anomaly detection effectiveness in production processes. The application of advanced deep learning methods represents a significant step toward improving anomaly detection quality in data with complex temporal structures, which is critically important in the context of industrial process monitoring.

The resulting network model was implemented in the FIBRAIN industrial plant in a technological supervision system with the task of providing real–time support for RLM line operators during the production process. Figures 10 and 11 show examples of the monitoring view during: normal process operation and the occurrence of an anomaly threatening a defect in the final product, respectively. The bar graph represents the probability values of the current process signature belonging to one of the classes: `Proper, Anomaly 1-4`. For the operator's convenience, the values on this graph correspond to the output signals of the penultimate Softmax layer of the aggregating module. The system operator's task

was to control and tune the line's working parameters so that the process state signature recognized by the LSTM network corresponded to its proper operation. The number of distinguishable anomalies for the RLM line was set at 4, based on heuristic assumptions and consultations with experts supervising the production technology on this line.

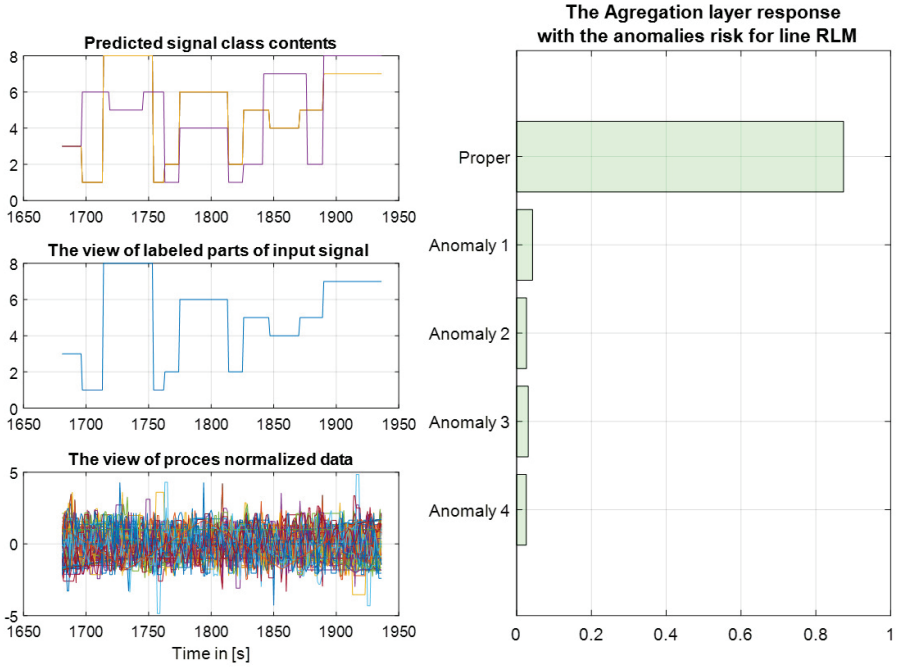


Figure 10. Signalization of proper operation in the analysis of the technological line's performance.

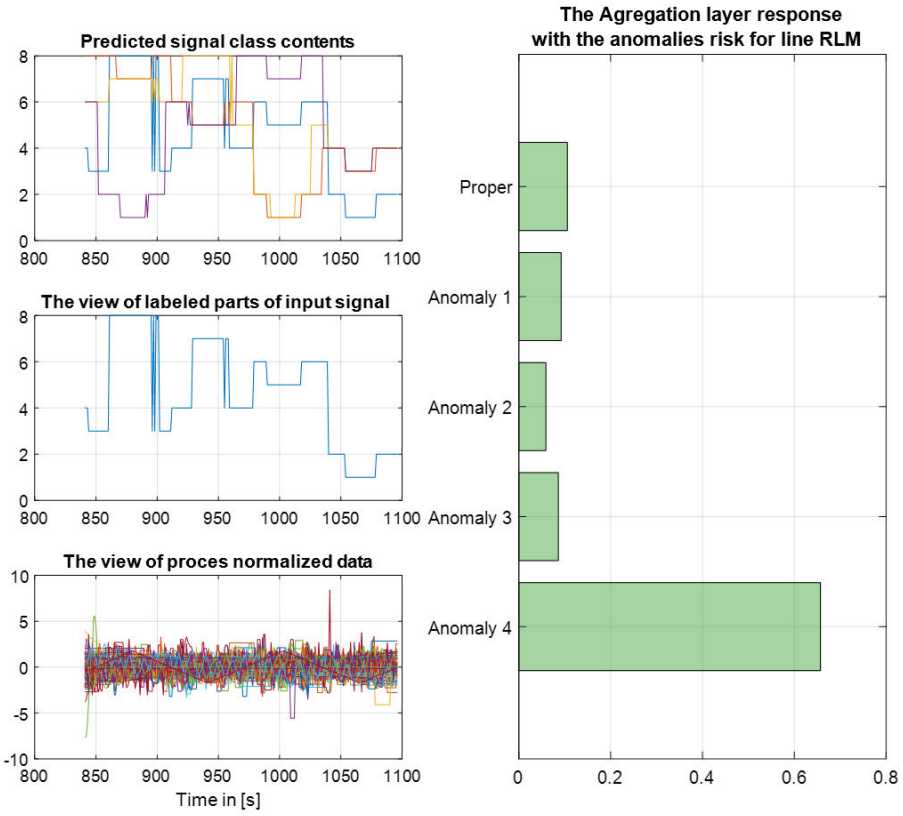


Figure 11. Signalization of proper operation in the analysis of the technological line's performance.

During the period from 1 January 2024, to 28 June 2024, the constructed LSTM network model was test–integrated into the production process monitoring system. Based on the observations made, the following coincidences between detected anomalies and corrective actions taken were identified:

- Anomaly 1: Excessive fluctuations in pressure and temperature alter the geometry and texture of the produced item. Significant pressure variations cause diameter changes within the range of 0.2 [mm] to 0.4 [mm], leading to the product being classified as non–compliant. In the cable coating extrusion process, large pressure changes result in discontinuities in the coating material, causing the final product to be divided into short segments, which are often unacceptable to customers.
- Anomaly 2: Excessive deviations in the pressure and temperature of the hydrophobic gel. Pressure changes lead to variations in the external and internal diameters of the semi–finished product. Changes in the external diameter result in weakened strength at the constriction points of the semi–finished product.
- Anomaly 3: Excessive production speed and the associated tension force of the production line and winding device. According to experts, this is a crucial anomaly that causes excess fiber in the tube, negatively affecting the transmission properties and strength of the finished fiber optic cable. Moreover, its frequent occurrence indicates wear and tear of the drive and consumable parts of the line.
- Anomaly 4: Temperature fluctuations in the cooling bath water affect the surface condition of semi–finished and finished products, as well as the dynamics of secondary shrinkage, which negatively impacts semi–finished and finished cables many days after their production.

During the analyzed period of the test implementation, a reduction in the amount of defective product by 66.3% was achieved, which should be considered a significant reduction in unnecessary production costs per unit length of the final product. A notable drawback of the developed approach is the relatively high sensitivity of the monitoring module to sterile measurement conditions that must be maintained when using such a large number of sensors. Its susceptibility to transient spikes or temporary loss of measurement values may pose a significant challenge in adapting the proposed method to similar production processes.

4. Conclusions

Combining statistical methods, machine learning, and sensor-based monitoring provides a robust approach to detecting and addressing anomalies. The proposed approach effectively integrates the multi-resolution K-means clustering method with recursive LSTM network architectures. The analysis conducted in this study on the available data from the monitored RLM production line identified key parameters determining product quality during technological processes: pressure and temperature of the material and gel, and constant production speed, which is crucial for maintaining the correct dimensions of the extruded product. The achieved results indicate the validity of using deep learning with LSTM networks for the analysis and classification of technological data.

This work makes a significant contribution to the field of anomaly detection by applying an innovative hybrid architecture that combines clustering techniques with LSTM networks, enabling the automatic creation of training datasets even in the absence of manually labeled data. The multi-channel temporal data analysis system from production lines provided a new perspective on real-time monitoring of industrial processes, which can also be adapted to other industries. However, certain limitations of the developed model should be noted. The high sensitivity of the system to input data disturbances, particularly

when using multiple sensors, necessitates precise measurement conditions. Additionally, the scalability of the model presents a challenge when deploying it across production lines with differing characteristics. Despite these challenges, the proposed method significantly reduces the number of defects and allows for better management of the production process.

The ultimate application of the described system is full integration with production lines in real industrial facilities, enabling further monitoring and optimization of the manufacturing processes for optical fiber tubes. The system facilitates a significant reduction in waste and improvement in the quality of final products, making an essential contribution both economically and environmentally. The application of anomaly detection methods using LSTM in other manufacturing processes, especially within the Industry 4.0 framework, can enhance quality control across a wide range of technologies. Ultimately, the findings of this study demonstrate that further advancements in optical cable production technology and advanced machine learning techniques will continue to improve the precision and efficiency of anomaly detection methods, thus supporting better quality control and the advancement of industrial process automation.

Author Contributions: Conceptualization, Z.G.; methodology, Z.G.; software, Z.G. and E.Z.; validation, Z.G. and E.Z.; formal analysis, Z.G.; investigation, Z.G., E.Z. and L.O.; resources, L.O.; data curation, Z.G.; writing—original draft preparation, Z.G. and E.Z.; visualization, Z.G. and E.Z.; supervision, Z.G.; project administration, Z.G.; funding acquisition, L.O., Z.G. and E.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This project received funding from the National Centre for Research and Development under grant agreement No. POIR.01.01.01-00-1425/20.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Acknowledgments: We gratefully acknowledge the support of the NCBR, as part of the competition: 6/1.1.1/2020 6/1.1.1/2020 SS Duze/MSP/JN 4, project number POIR.01.01.01-00-1425/20-00 and Fibrain Sp. z.o.o, which made this research possible.

Conflicts of Interest: Lukasz Olbrot was employed by the company FIBRAIN Sp. z o.o. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DNN	Deep Neural Network
DWT	Discrete Wavelet Transform
LSTM	Long short-term memory
RNN	Recurrent neural network
RLR	line for tube extrusion
RLV	line for coating extrusion on the cable core
RLM	line for twisting cable cores from tubes
SVM	Support Vector Machines

References

1. Gordan, M.; Sabbagh-Yazdi, S.R.; Ismail, Z.; Ghaedi, K.; Carroll, P.; McCrum, D.; Samali, B. State-of-the-art review on advancements of data mining in structural health monitoring. *Measurement* **2022**, *193*, 110939. [CrossRef]

2. Kotsiopoulos, T.; Sarigiannidis, P.; Ioannidis, D.; Tzovaras, D. Machine Learning and Deep Learning in smart manufacturing: The Smart Grid paradigm. *Comput. Sci. Rev.* **2021**, *40*, 100341. [CrossRef]
3. Ribeiro, R.; Pilastri, A.; Moura, C.; Morgado, J.; Cortez, P. A data-driven intelligent decision support system that combines predictive and prescriptive analytics for the design of new textile fabrics. *Neural Comput. Appl.* **2023**, *35*, 17375–17395. [CrossRef]
4. Bock, F.E.; Aydin, R.C.; Cyron, C.J.; Huber, N.; Kalidindi, S.R.; Klusemann, B. A review of the application of machine learning and data mining approaches in continuum materials mechanics. *Front. Mater.* **2019**, *6*, 452701. [CrossRef]
5. Soleimani, M.; Naderian, H.; Afshinfar, A.H.; Savari, Z.; Tizhari, M.; Agha Seyed Hosseini, S.R. A Method for Predicting Production Costs Based on Data Fusion from Multiple Sources for Industry 4.0: Trends and Applications of Machine Learning Methods. *Comput. Intell. Neurosci.* **2023**, *2023*, 6271241. [CrossRef] [PubMed]
6. Abdallah, M.; Joung, B.G.; Lee, W.J.; Mousoulis, C.; Raghunathan, N.; Shakouri, A.; Sutherland, J.W.; Bagchi, S. Anomaly Detection and Inter-Sensor Transfer Learning on Smart Manufacturing Datasets. *Sensors* **2023**, *23*, 486. [CrossRef] [PubMed]
7. Abdelli, K.; Cho, J.Y.; Azendorf, F.; Griesser, H.; Tropschug, C.; Pachnicke, S. Machine Learning-based Anomaly Detection in Optical Fiber Monitoring. *J. Opt. Commun. Netw.* **2022**, *14*, 365–375. [CrossRef]
8. Abdula, S.P.; Llagas, M.J.; Fernandez, A.M.; Arboleda, E. Machine Learning Applications for Fault Tracing and Localization in Optical Fiber Communication Networks: A Review. *Preprints* **2024**. [CrossRef]
9. Glass, S.W.; Fifield, L.S.; Spencer, M.P. Transition to Online Cable Insulation Condition Monitoring. In Proceedings of the 2021 48th Annual Review of Progress in Quantitative Nondestructive Evaluation, QNDE 2021, Virtual, 28–30 July 2021.
10. Kakavandi, F.; Gomes, C.; de Reus, R.; Badstue, J.; Jensen, J.L.; Larsen, P.G.; Iosifidis, A. Towards Developing a Digital Twin for a Manufacturing Pilot Line: An Industrial Case Study. In *Digital Twin Driven Intelligent Systems and Emerging Metaverse*; Springer: Singapore, 2023; pp. 39–64.
11. Kane, A.P.; Kore, A.S.; Khandale, A.N.; Nigade, S.S.; Joshi, P.P. *Predictive Maintenance Using Machine Learning*; SPD Technology: London, UK, 2022.
12. Pittino, F.; Puggl, M.; Moldaschl, T.; Hirschl, C. Automatic Anomaly Detection on In-Production Manufacturing Machines Using Statistical Learning Methods. *Sensors* **2020**, *20*, 2344. [CrossRef]
13. Fan, J.; Wang, Z.; Wu, H.; Sun, D.; Wu, J.; Lu, X. An Adversarial Time–Frequency Reconstruction Network for Unsupervised Anomaly Detection. *Neural Netw.* **2023**, *168*, 44–56. [CrossRef] [PubMed]
14. Zhu, W.; Li, W.; Dorsey, E.R.; Luo, J. Unsupervised anomaly detection by densely contrastive learning for time series data. *Neural Netw.* **2023**, *168*, 450–458. [CrossRef] [PubMed]
15. Zhang, X.; Shi, S.; Sun, H.; Chen, D.; Wang, G.; Wu, K. ACVAE: A novel self-adversarial variational auto-encoder combined with contrast learning for time series anomaly detection. *Neural Netw.* **2024**, *171*, 383–395. [CrossRef] [PubMed]
16. Yao, Y.; Ma, J.; Feng, S.; Ye, Y. SVD-AE: An asymmetric autoencoder with SVD regularization for multivariate time series anomaly detection. *Neural Netw.* **2024**, *170*, 535–547. [CrossRef] [PubMed]
17. Li, Q.; Wang, Y.; Dong, J.; Zhang, C.; Peng, K. Multi-node knowledge graph assisted distributed fault detection for large-scale industrial processes based on graph attention network and bidirectional LSTMs. *Neural Netw.* **2024**, *173*, 106210. [CrossRef] [PubMed]
18. Fan, J.; Ge, Y.; Zhang, X.; Wang, Z.; Wu, H.; Wu, J. Learning the feature distribution similarities for online time series anomaly detection. *Neural Netw.* **2024**, *180*, 106638. [CrossRef] [PubMed]
19. Lyu, S.; Mo, D.; Wong, W.K. REB: Reducing biases in representation for industrial anomaly detection. *Knowl.-Based Syst.* **2024**, *290*, 111563. [CrossRef]
20. Kang, B.; Zhong, Y.; Sun, Z.; Deng, L.; Wang, M.; Zhang, J. MSTAD: A masked subspace-like transformer for multi-class anomaly detection. *Knowl.-Based Syst.* **2024**, *283*, 111186. [CrossRef]
21. Iqbal Basheer, M.Y.; Mohd Ali, A.; Abdul Hamid, N.H.; Mohd Ariffin, M.A.; Osman, R.; Nordin, S.; Gu, X. Autonomous anomaly detection for streaming data. *Knowl.-Based Syst.* **2024**, *284*, 111235. [CrossRef]
22. Hong, J.; Kang, S. Score distillation for anomaly detection. *Knowl.-Based Syst.* **2024**, *295*, 111842. [CrossRef]
23. Wang, X.; Li, W.; He, X. MTDiff: Visual anomaly detection with multi-scale diffusion models. *Knowl.-Based Syst.* **2024**, *302*, 112364. [CrossRef]
24. Guo, W.; Jiang, P. Weakly Supervised anomaly detection with privacy preservation under a Bi-Level Federated learning framework. *Expert Syst. Appl.* **2024**, *254*, 124450. [CrossRef]
25. An, S.; Kim, J.; Kim, S.; Chikontwe, P.; Jung, J.; Jeon, H.; Park, S.H. Few-shot anomaly detection using positive unlabeled learning with cycle consistency and co-occurrence features. *Expert Syst. Appl.* **2024**, *256*, 124890. [CrossRef]
26. Shen, L.; Wei, Y.; Wang, Y.; Li, H. AFMF: Time series anomaly detection framework with modified forecasting. *Knowl.-Based Syst.* **2024**, *296*, 111912. [CrossRef]

27. Bai, Y.; Zhang, J.; Chen, Z.; Dong, Y.; Cao, Y.; Tian, G. Dual-path Frequency Discriminators for few-shot anomaly detection. *Knowl.-Based Syst.* **2024**, *302*, 112397. [CrossRef]
28. Liu, Y.; Ju, B.; Yang, D.; Peng, L.; Li, D.; Sun, P.; Li, C.; Yang, H.; Liu, J.; Song, L. Memory-enhanced spatial-temporal encoding framework for industrial anomaly detection system. *Expert Syst. Appl.* **2024**, *250*, 123718. [CrossRef]
29. Di, Y.; Wang, F.; Zhao, Z.; Zhai, Z.; Chen, X. An interpretable graph neural network for real-world satellite power system anomaly detection based on graph filtering. *Expert Syst. Appl.* **2024**, *254*, 124348. [CrossRef]
30. Chen, A.; Wu, J.; Zhang, H. FIAD: Graph anomaly detection framework based feature injection. *Expert Syst. Appl.* **2025**, *259*, 125216. [CrossRef]
31. Lei, T.; Ou, M.; Gong, C.; Li, J.; Yang, K. An unsupervised deep global–local views model for anomaly detection in attributed networks. *Knowl.-Based Syst.* **2024**, *300*, 112185. [CrossRef]
32. Mejri, N.; Lopez-Fuentes, L.; Roy, K.; Chernakov, P.; Ghorbel, E.; Aouada, D. Unsupervised anomaly detection in time-series: An extensive evaluation and analysis of state-of-the-art methods. *Expert Syst. Appl.* **2024**, *256*, 124922. [CrossRef]
33. Chi, J.; Mao, Z. Deep domain-adversarial anomaly detection with robust one-class transfer learning. *Knowl.-Based Syst.* **2024**, *300*, 112225. [CrossRef]
34. Zhu, T.; Liu, L.; Sun, Y.; Lu, Z.; Zhang, Y.; Xu, C.; Chen, J. Semi-supervised noise-resilient anomaly detection with feature autoencoder. *Knowl.-Based Syst.* **2024**, *304*, 112445. [CrossRef]
35. Han, H.; Fan, H.; Huang, X.; Han, C. Self-supervised multi-transformation learning for time series anomaly detection. *Expert Syst. Appl.* **2024**, *253*, 124339. [CrossRef]
36. Yu, L.-R.; Lu, Q.-H.; Xue, Y. DTAAD: Dual Tcn-attention networks for anomaly detection in multivariate time series data. *Knowl.-Based Syst.* **2024**, *295*, 111849. [CrossRef]
37. Wei, S.; Wei, X.; Ma, Z.; Dong, S.; Zhang, S.; Gong, Y. Few-shot online anomaly detection and segmentation. *Knowl.-Based Syst.* **2024**, *300*, 112168. [CrossRef]
38. Amini, A.; Kalantari, R. Gold price prediction by a CNN-Bi-LSTM model along with automatic parameter tuning. *PLoS ONE* **2024**, *19*, e0298426. [CrossRef] [PubMed]
39. Huang, W.; Lin, Y.; Liu, M.; Min, H. Velocity-aware spatial-temporal attention LSTM model for inverse dynamic model learning of manipulators. *Front. Neurorobotics* **2024**, *18*, 1353879. [CrossRef] [PubMed]
40. Liu, L.; Feng, J.; Li, J.; Chen, W.; Mao, Z.; Tan, X. Multi-layer CNN-LSTM network with self-attention mechanism for robust estimation of nonlinear uncertain systems. *Front. Neurosci.* **2024**, *18*, 1379495. [CrossRef] [PubMed]
41. Misiti, M.; Misiti, Y.; Oppenheim, G.; Poggi, J.M. Clustering Signals Using Wavelets. In *Computational and Ambient Intelligence*; Sandoval, F., Prieto, A., Cabestany, J., Graña, M., Eds.; Springer: Berlin/Heidelberg, Germany, 2007; pp. 514–521.
42. Campos, G.O.; Zimek, A.; Sander, J.; Campello, R.J.; Micenková, B.; Schubert, E.; Assent, I.; Houle, M.E. On the evaluation of unsupervised outlier detection: Measures, datasets, and an empirical study. *Data Min. Knowl. Discov.* **2016**, *30*, 891–927. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

applied sciences

MDPI

Article

Explainable AI (XAI) Techniques for Convolutional Neural Network-Based Classification of Drilled Holes in Melamine Faced Chipboard

Alexander Sieradzki [1], Jakub Bednarek [2], Albina Jegorowa [3] and Jarosław Kurek [1,*]

[1] Department of Artificial Intelligence, Institute of Information Technology, Warsaw University of Life Sciences, Nowoursynowska 159, 02-776 Warsaw, Poland
[2] Faculty of Medicine, Medical University of Lodz, Kościuszki 4, 90-419 Łódź, Poland
[3] Department of Mechanical Processing of Wood, Institute of Wood Sciences and Furniture, Warsaw University of Life Sciences, Nowoursynowska 159, 02-776 Warsaw, Poland
* Correspondence: jaroslaw_kurek@sggw.edu.pl

Abstract: The furniture manufacturing sector faces significant challenges in machining composite materials, where quality issues such as delamination can lead to substandard products. This study aims to improve the classification of drilled holes in melamine-faced chipboard using Explainable AI (XAI) techniques to better understand and interpret Convolutional Neural Network (CNN) models' decisions. We evaluated three CNN architectures (VGG16, VGG19, and ResNet101) pretrained on the ImageNet dataset and fine-tuned on our dataset of drilled holes. The data consisted of 8526 images, divided into three categories (Green, Yellow, Red) based on the drill's condition. We used 5-fold cross-validation for model evaluation and applied LIME and Grad-CAM as XAI techniques to interpret the model decisions. The VGG19 model achieved the highest accuracy of 67.03% and the lowest critical error rate among the evaluated models. LIME and Grad-CAM provided complementary insights into the decision-making process of the model, emphasizing the significance of certain features and regions in the images that influenced the classifications. The integration of XAI techniques with CNN models significantly enhances the interpretability and reliability of automated systems for tool condition monitoring in the wood industry. The VGG19 model, combined with LIME and Grad-CAM, offers a robust solution for classifying drilled holes, ensuring better quality control in manufacturing processes.

Keywords: explainable AI (XAI); LIME; Grad-CAM; convolutional neural networks; CNN; drilled holes classification; melamine-faced chipboard; tool condition monitoring

1. Introduction

The furniture manufacturing sector faces numerous challenges, particularly in the machining of composite materials where quality issues, such as delamination, can lead to substandard products. This is particularly true during the drilling operation, in which multiple factors might lead to undesirable results.

Laminated panels are widely utilized in the manufacture of furniture [1–7]. Research indicates that delamination in wood-based material cutting is primarily caused by tool wear. Studies show a clear correlation between tool wear and defects on the laminate surface of wood panels [4–6,8]. Monitoring tool condition is crucial, as identifying the optimal time for tool replacement is essential to prevent delamination and ensure product quality. Manual inspection of tools without prior condition assessment can lead to unnecessary downtime, while delaying tool replacement can damage materials. Thus, automating this monitoring process is increasingly important and a topic of extensive research.

The furniture manufacturing sector faces numerous challenges, particularly in the machining of composite materials where quality issues, such as delamination, can lead to

substandard products. This is particularly true during the drilling operation, in which multiple factors might lead to undesirable results. In recent years, the application of machine learning and data augmentation techniques has emerged as a promising approach to address such challenges, especially when working with limited datasets. Capturing sufficient and balanced data for intelligent fault diagnosis remains a significant issue across various domains. For instance, recent research by Liu et al. (2024) highlights how counterfactual-augmented few-shot contrastive learning can effectively identify faults even with limited and imbalanced samples by focusing on discriminative representations [9]. Similarly, the use of interpretable data-augmented adversarial variational autoencoders (AVAE) has been shown to improve diagnostic accuracy in imbalanced datasets, a common scenario in fault diagnosis tasks like those encountered in the wood processing industry [10]. These advancements underscore the importance of integrating such approaches into automated systems for tool condition monitoring in the wood industry, where consistent quality control is critical.

Explainable Artificial Intelligence (XAI) has gained significant attention in recent years due to its potential to address the black-box problem associated with advanced deep learning models, especially in critical applications such as healthcare, finance, and industrial automation. XAI enables the interpretation of AI model outputs, fostering trust and transparency in AI systems, which are crucial for their broader adoption.

Kim et al. [11] explored the application of Convolutional Neural Networks (CNNs) for partial discharge (PD) classification in cast-resin transformers. The study highlights the challenge posed by the lack of transparency in CNNs, which leads to difficulties in understanding and trusting model predictions. The authors propose the use of Grad-CAM, an XAI technique, to provide visual explanations for model decisions, thereby enhancing the reliability of PD classification systems. The research demonstrated a classification accuracy of approximately 97%, and the incorporation of XAI helped identify the criteria used by the model, ultimately leading to improved trust and robustness in the system.

Apicella et al. [12] focused on leveraging XAI not only for interpretability, but also to enhance the performance of machine learning models. The authors assessed well-known XAI techniques, such as Integrated Gradients, in the context of classification tasks using datasets like Fashion-MNIST, CIFAR10, and STL10. The study revealed that XAI methods could be used strategically to identify critical features that directly contribute to improving model accuracy. This approach suggests that XAI can serve as a dual-purpose tool—both for interpretability and performance enhancement.

In another study, Miller et al. [13] applied XAI to light direction reconstruction models used in augmented reality. The study emphasized the caution needed when interpreting XAI outputs due to potential unreliability. Nevertheless, by carefully leveraging the meta-information provided by XAI methods, the authors were able to enhance the architecture and training dataset of a regression model, leading to improved performance. This work underscores the role of XAI in guiding the optimization of model design and training.

Selvaraju et al. [14] introduced Grad-CAM, a widely used XAI technique that produces visual explanations for CNN-based models by highlighting relevant regions in input images. Grad-CAM is applicable to various CNN architectures, and has demonstrated its utility across multiple tasks, including image classification, captioning, and visual question answering (VQA). By combining Grad-CAM with fine-grained visualizations, the study achieved superior model interpretability and generalization, addressing key challenges such as adversarial robustness and dataset bias.

Barredo Arrieta et al. [15] presented a comprehensive review of the XAI field, providing a taxonomy of existing methods and highlighting challenges in achieving responsible AI. The review emphasizes the significance of explainability in machine learning models, particularly in deep learning, where the complexity of models often obscures their decision-making processes. The authors propose a framework for responsible AI that integrates fairness, transparency, and accountability, paving the way for the large-scale implementation of AI in real-world applications.

These studies collectively underscore the critical role of XAI in enhancing the transparency, reliability, and performance of AI models. The integration of XAI not only aids in the interpretability of complex models, but also provides avenues for improving model performance through more informed architectural and training choices.

The foundational work in tool condition monitoring dates back nearly 40 years to [16–18], focusing on identifying reliable signals and features for detecting tool wear during production without downtime. This area of research has evolved, applying various approaches [19–27], typically involving multiple sensors to detect changes in physical parameters from the machining zone.

Conventional methods usually require an array of sensors, such as acoustic emission, vibration testing, noise measurement, or evaluating cutting torque and feed force [28–33]. While these methods can be accurate, they are often complex and time-consuming, requiring careful sensor placement and calibration. Initial setup errors can lead to unstable and inaccurate results, especially if regular folding of sensors is necessary. These drawbacks make sensor-based approaches challenging, particularly in dynamic production environments where conditions frequently change.

Given the multitude of factors influencing drilling quality, machine learning (ML) algorithms emerge as a logical solution. In recent years, ML methods have gained traction in the wood industry for various applications, from monitoring machining processes [34–38] to wood species recognition. Tailored solutions or adaptable ML algorithms are available for specific problems.

In this research, the study relies on photographs of drilled holes as the primary dataset, working from the premise that the condition of the drill edges can reveal the wear on the tool, thereby indicating the appropriate timing for tool replacement. This theory has yielded encouraging outcomes through the application of various techniques and algorithms. Techniques based on deep learning and transfer learning have demonstrated efficiency and a strong alignment with this specific challenge [39–43]. Additional investigations propose that the implementation of classifier ensembles and data augmentation techniques could improve the overall performance [44,45], with various classifiers being evaluated and showing considerable potential for real-world usage.

Explainable AI (XAI) techniques, such as Class Activation Mapping (CAM), have been applied to identify points of concern in image-based Deep Neural Networks (DNNs) used for failure analysis in industrial applications [46–50]. The study highlights the importance of XAI tools in providing insights into the decision process of DNNs and emphasizes the need for further analysis of their correctness. Local Interpretable Model-agnostic Explanations (LIME) and Gradient-weighted Class Activation Mapping (GRAD-CAM) have shown their utility in various industrial applications. CNN and GRAD-CAM techniques have been utilized in detecting defects and pinpointing their locations on industrial products. However, issues related to accurately identifying defective regions and the time required for processing have been noted [51–54]. Additionally, the application of LIME in the automated detection of fractures during material testing has been explored, offering insights into the capability for critical feature detection and the automation of crack detection.

Challenges in drill state recognition have been addressed by transfer learning using CNN, offering the advantage of training classification models with a small portion of data [19]. Data augmentation techniques combined with transfer learning have improved the accuracy of drill wear recognition, even with a small original dataset [20].

Despite the absence of direct information on the specific use of LIME and GRAD-CAM for drilled hole classification in melamine-faced chipboard, the insights from XAI techniques and their application in industrial contexts suggest their potential applicability. LIME and GRAD-CAM are valuable XAI techniques that provide insights into the decision-making process of CNN-based classification models for drilled holes in melamine-faced chipboard. These methods provide benefits regarding the accuracy of explanations and the visual representation of specific features. However, there are drawbacks that require further investigation in upcoming studies.

This paper explores the use of LIME and GRAD-CAM for enhancing the interpretability of CNN-based models in classifying drilled holes in melamine-faced chipboard. By applying these XAI techniques, we aim to better understand the model's decision-making process, ensure its reliability, and improve the overall accuracy of the classification.

2. Materials and Methods

2.1. Data Collection

In the experiment, a Busselato automated Computerized Numerical Control (CNC) workstation (model Jet 100) was utilized. This system guaranteed repeatability in the experiment. The drilling process involved using a 12 mm diameter drill from FABA (model WP-01, as shown in Figure 1). The drill operated at a rotational speed of 4500 RPM and advanced at a feed rate of 1.35 m/min, following the recommendations provided by the manufacturer of the drill. The material subjected to the test was a standard 18 mm thick laminated board, commonly used in the furniture industry, produced by the Polish company KRONOPOL (model U 511).

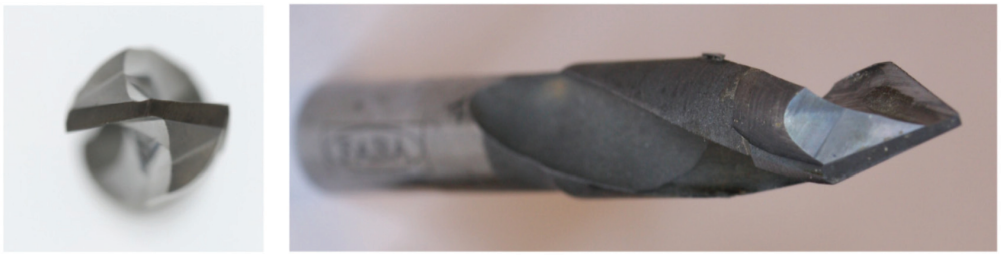


Figure 1. The provided description illustrates the FABA WP-01 drill from two different perspectives used in the experiments. From the left, a top view of the drill bit is depicted, showcasing its cutting edges and the general shape. Meanwhile, the image on the right offers a side perspective, emphasizing the length and contour of the drill bit.

The density profile for the chipboard was analyzed with a GreCon DAX device by Fagus-GreCon Greten GmbH & Co. KG (Alfeld, Germany), as illustrated in Figure 2. Additional characteristics of the material include a bending modulus of rupture at 15.4 MPa, a bending modulus of elasticity at 2950 MPa, and a Brinell surface hardness (HB) of 2.1. The evaluations of these material properties were conducted in accordance with standards [55,56], utilizing an Instron 3382 testing apparatus from Instron in Norwood, MA, USA, and a Brinell CV 3000LDB hardness testing equipment from the Bowers Group located in Camberley, UK, respectively.

For an in-depth learning and testing exercise, the condition of drill bits was accurately observed and evaluated using a conventional workshop microscope equipped with a digital camera (Model TM—505; manufactured by Mitutoyo, based in Kawasaki, Japan). In this context, the degree of wear and tear observed on the external corner of the drill bits was considered a direct measure of their condition. This wear (denoted as W) was quantitatively measured for each of the drill bits' two cutting edges, utilizing the formula provided below:

$$W = W'' - W' \tag{1}$$

where W'' is the initial width of a brand-new cutting edge near the outer corner (mm), and W' is the current width of a brand-new cutting edge near the outer corner (mm) (Figure 3).

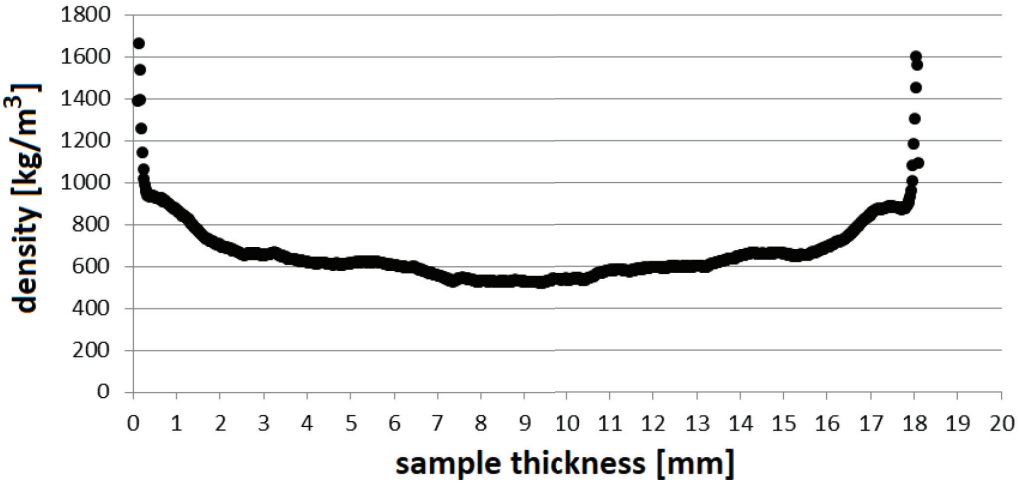


Figure 2. Density profile of the chipboard utilized in the conducted experiments.

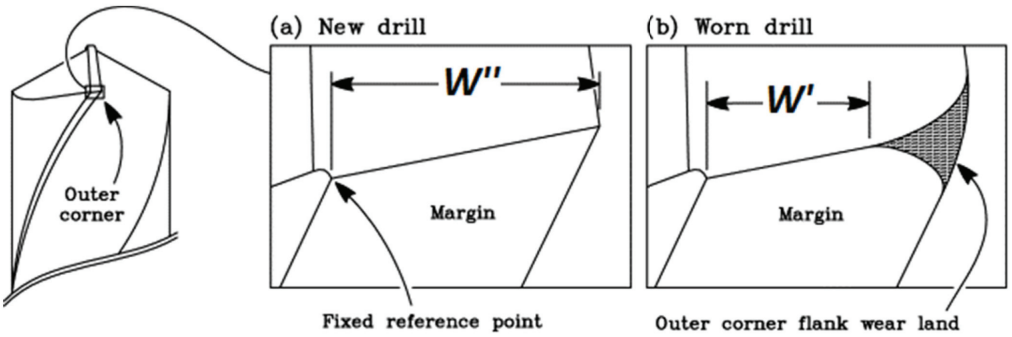


Figure 3. Approach to measuring tool degradation: (**a**) appearance of a freshly manufactured tool, (**b**) appearance of a tool after wear [57].

According to Equation (3), the state of the drill is divided into three distinct classifications:

- "Green" tool wear class $W < 0.2$ mm.
- "Yellow" tool wear class $0.2 \leq W < 0.35$ mm.
- "Red" tool wear class $W \geq 0.35$ mm.

Figure 1 illustrates the FABA WP-01 drill used in the experimental setup. The left part of the figure provides a top-down view of the drill bit, clearly showing the sharp cutting edges designed for efficient drilling into materials. This view emphasizes the precision and engineering behind the drill's design, crucial for achieving clean and accurate holes in melamine-faced chipboard.

The right part of the figure offers a side view of the drill bit, showcasing its overall length and the angled cutting surfaces. This perspective highlights the drill bit's structure, which is designed to maintain stability and effectiveness during high-speed rotations and feed rates. Understanding both the top and side views of the drill bit helps in comprehensively analyzing its performance and wear characteristics during the drilling process.

The figure above (Figure 4) illustrates examples of drilled holes categorized into three different classes, based on the condition of the drill and the quality of the holes. The first row shows holes classified as "Green", indicating that they were made with a new or minimally worn drill bit, resulting in clean and precise holes. The second row shows holes classified as "Yellow", which were made with a moderately worn drill bit, resulting in slightly less clean holes with some rough edges. The third row shows holes classified as "Red", indicating that they were made with a significantly worn drill bit, resulting in rough and irregular holes with noticeable defects. These examples are used for research analysis

to train and evaluate the performance of the Convolutional Neural Network (CNN) models in classifying the condition of the drilled holes.

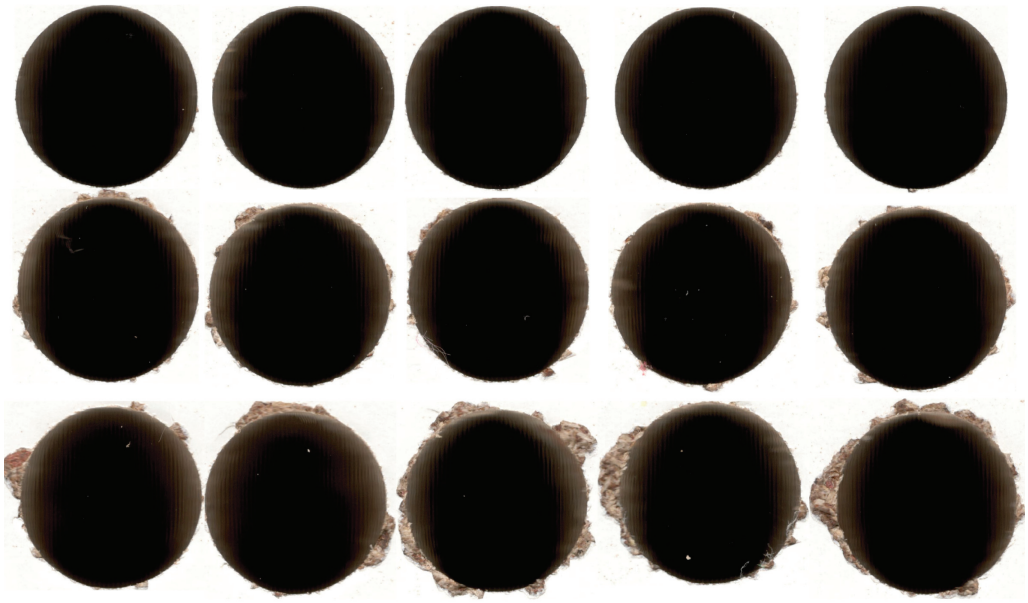


Figure 4. For illustration purposes, examples of drilling holes are categorized into three classes. The "Green" class examples are listed in the first row, followed by the "Yellow" class examples in the second row, and examples of the "Red" class are displayed in the third row. Refer to for additional details.

The evaluation of the drill's condition was routinely performed using a Mitutoyo microscope (model TM-500), enabling the classification of each hole into one of three categories:

- Good—represents a new, unworn drill that is suitable for continued use;
- Worn—applies to a used drill that either needs manual inspection to determine its state or is still in a condition acceptable for ongoing production activities;
- Requiring replacement—describes a used drill that is in a condition deemed unserviceable and must be replaced without delay.

Table 1 provides a summary of data collection and drill wear measurements. The table includes three columns: Drill Number, Green/Yellow/Red, and Total Images. The Drill Number column lists the five different drills used in the experiments, numbered from 1 to 5. The Green/Yellow/Red column shows the number of images categorized into three classes (Green, Yellow, Red) for each drill, indicating the condition of the drill bit. Specifically, Green represents a new or minimally worn drill bit, Yellow indicates a moderately worn drill bit, and Red signifies a significantly worn drill bit. The Total Images column provides the total number of images collected for each drill.

Table 1. Summary of image data collection and drill wear classification.

Drill Number	Green/Yellow/Red	Total Images
1	840/420/406	1666
2	840/700/280	1820
3	700/560/420	1680
4	840/560/280	1680
5	560/560/560	1680
Total	3780/2800/1946	8526

Drill 1 has 840 images classified as Green, 420 as Yellow, and 406 as Red, summing up to a total of 1666 images. Drill 2 has 840 Green, 700 Yellow, and 280 Red images, with a total of 1820 images. Drill 3 has 700 Green, 560 Yellow, and 420 Red images, totaling 1680 images. Drill 4 has 840 Green, 560 Yellow, and 280 Red images, also totaling 1680 images. Drill 5 has an equal distribution of 560 images in each category, amounting to 1680 images. The total count for all drills combined is 3780 Green, 2800 Yellow, and 1946 Red images, leading to an overall total of 8526 images.

2.2. Cost Analysis of Conventional Methods for Sensor Replacement and Calibration

In conventional machining processes, sensor-based monitoring systems play a critical role in detecting tool wear, ensuring production quality, and maintaining operational efficiency. However, these systems come with significant costs, both in terms of initial setup and ongoing maintenance. The costs associated with replacing or calibrating sensors can become substantial over time, especially in dynamic production environments.

One of the primary cost factors involves the periodic replacement of sensors. Sensors such as those used for acoustic emission, vibration analysis, and cutting torque measurement are subject to wear and degradation due to exposure to harsh machining conditions. Frequent replacement of these sensors is necessary to maintain accurate readings, which incurs direct material costs and the additional cost of halting production during the replacement process.

Another considerable expense arises from the need for regular calibration. Precision monitoring requires sensors to be regularly recalibrated to ensure accuracy, especially in environments where conditions change frequently. Calibration typically requires skilled technicians and specialized equipment, contributing to both labor and operational costs. Additionally, calibration processes may necessitate machine downtime, further increasing the overall expense.

A comparative study of conventional and AI-based monitoring systems highlights the cost-effectiveness of automated solutions. While AI-based approaches, such as those using Convolutional Neural Networks (CNNs) and Explainable AI (XAI) techniques, require initial investments in software and model training, they significantly reduce the ongoing costs associated with sensor management. AI-driven models can monitor tool wear and quality without the need for extensive sensor networks, thus minimizing both direct and indirect expenses.

In summary, conventional sensor-based methods for tool condition monitoring are associated with recurring costs related to sensor replacement, calibration, and production downtime. By integrating AI-based solutions, manufacturers can mitigate these costs while improving the reliability and efficiency of their monitoring systems.

2.3. CNN Models Architecture

In our research, we assessed the capabilities of three distinct Convolutional Neural Network (CNN) models—VGG16, VGG19, and ResNet101—for identifying and classifying drilled holes in melamine-faced chipboards, as referenced in various works [40,45,58–60]. These models were selected because of their proven proficiency in a variety of image analysis tasks. Herein, we detail each model's architecture and the specific setups utilized in our study.

Introduced by Simonyan and Zisserman [61], the VGG16 architecture is favored for its straightforward design and depth, featuring 16 weight layers. This configuration comprises 13 convolutional layers paired with three fully connected layers at the end. It applies 3×3 convolutional filters through its layers, enabling it to discern intricate patterns in the visuals. Max-pooling layers are strategically placed to decrease the spatial dimensionality and computational demands.

Expanding upon VGG16, the VGG19 model [61] incorporates an additional three convolutional layers, totaling 19 weight layers. This enhancement is aimed at augmenting the model's ability to learn more elaborate features, albeit at the expense of increased computational requirements and extended training periods.

The ResNet101 model, brought forth by He et al. [62], belongs to the Residual Networks lineage. It introduces an approach of using residual learning to facilitate the training of profoundly deep networks. With its 101 layers, ResNet101 surpasses both VGG16 and VGG19 in depth. Its residual blocks are designed to learn identity functions, aiding in overcoming the issue of vanishing gradients and permitting the cultivation of deeper networks.

Our investigation into the efficacy of various CNN architectures for the categorization of drilled holes in melamine-faced chipboards involved an analysis of three renowned pretrained models: VGG16, VGG19, and ResNet101. We took into account each architecture's depth, scale, parameter count, and optimal input image dimensions to gauge their effectiveness and efficiency for our specific task.

These models were initially trained on the extensive ImageNet dataset, which encompasses a wide assortment of image categories. We then fine-tuned them on our targeted dataset of drilled holes in melamine-faced chipboards. The images were resized to dimensions of 224 × 224 pixels to conform to each model's input requirements. To enhance the training dataset's variety and fortify the models' resilience, data augmentation techniques like rotation, flipping, and scaling were applied.

Table 2 presents a synopsis of the salient features of the pretrained CNN models explored in this study [63–77].

Table 2. Comparison of pretrained CNN models.

Pretrained CNN Model	Depth	Size	Parameters (Millions)	Image Input Size
Resnet101	101	167 MB	44.6	224-by-224
VGG16	16	515 MB	138	224-by-224
VGG19	19	535 MB	144	224-by-224

The comparison of these pretrained CNN models is based on several criteria, including depth, model size, number of parameters, and input image size. Each of these factors plays a crucial role in determining the model's performance, computational requirements, and ability to generalize to the classification task at hand.

- Depth—ResNet101, with its 101 layers, is the deepest model among the three. The depth of a model often correlates with its capacity to learn complex features. However, deeper models also require more computational resources and can be more challenging to train effectively.
- Size—VGG16 and VGG19 are significantly larger in terms of model size compared to ResNet101. Larger models can store more information, but they also require more memory and storage, which can be a limitation in resource-constrained environments.
- Parameters—VGG16 and VGG19 have a much higher number of parameters (138 million and 144 million, respectively) compared to ResNet101 (44.6 million). While having more parameters can increase a model's learning capacity, it also makes the model more prone to overfitting and increases the computational load.
- Image input size—all models were adapted to accept input images of 224 × 224 pixels. This standardization ensures a fair comparison and aligns with the common practice in image classification tasks.

The following sections provide detailed evaluation results and analysis of each model's performance in classifying drilled holes in melamine-faced chipboard, highlighting their strengths and areas for improvement.

2.4. Techniques for Explainable AI (XAI)

In this section, we explore different methods applied to enhance the interpretability and explainability of AI models, especially focusing on Convolutional Neural Networks (CNNs). These methods are crucial for deciphering how these models arrive at their decisions and offer clarity on which aspects of the input data play a pivotal role in the model's predictive outcomes. Our primary attention is directed towards two key techniques in Explainable AI (XAI): Local Interpretable Model-Agnostic Explanations (LIME) and Gradient-weighted Class Activation Mapping (Grad-CAM).

2.4.1. Local Interpretable Model-Agnostic Explanations (LIME)

Interpreting the predictions made by deep neural networks can be quite challenging due to their inherent complexity. Local Interpretable Model-Agnostic Explanations (LIME) is a technique developed to shed light on the reasons behind the classification decisions of deep neural networks [53,78–83].

Deep neural networks are often likened to "black boxes" due to the complexity and opacity of their decision-making processes. To mitigate this, LIME approximates the behavior of these intricate networks with a model that is simpler and more interpretable, such as a regression tree. This approach makes it easier to understand which features in the input data are significant in influencing the decisions made by the network.

LIME is particularly beneficial in the realm of image classification. It enables the creation of feature importance maps using the following key steps:

1. Segmenting features—the input image is divided into segments based on similar pixel characteristics such as color, texture, or intensity.
2. Creating segment masks—a binary mask is generated for each segment, indicating whether the segment is included (1) or excluded (0) in the synthetic image.
3. Generating synthetic images—synthetic images are produced by modifying the original image according to the segment masks, where excluded segments are replaced with average pixel values to obscure their information.
4. Identifying synthetic images—every synthetic image undergoes analysis through the neural network to ascertain how the presence or lack of specific segments influences the result of the classification.
5. Fitting an interpretable model—to establish a simpler, understandable model like a regression tree, the link between the presence of segments and their classification scores is leveraged.
6. Determining feature importance—the simpler model helps identify the significance of each segment, which is then mapped back to the original image to highlight influential regions.
7. Mapping feature importance—the significant values identified from segments are applied back to the initial image. This process generates a map that highlights the areas of the image that were crucial for the model's decision-making.

By applying LIME in our image classification tasks, we gain a clearer understanding of how our Convolutional Neural Network (CNN) models classify drilled holes in melamine-faced chipboard. This insight is crucial for validating the models' performance and ensuring they focus on the relevant features of the images.

In our study, with the VGG19 model, LIME was utilized to explain both correct and incorrect classifications. Figures 5 and 6 display the LIME-generated explanations for the VGG19 model's predictions on images of drilled holes. These visualizations are invaluable for identifying the critical regions in the images that the model considered, thus allowing us to assess its reliability and identify areas for potential improvement.

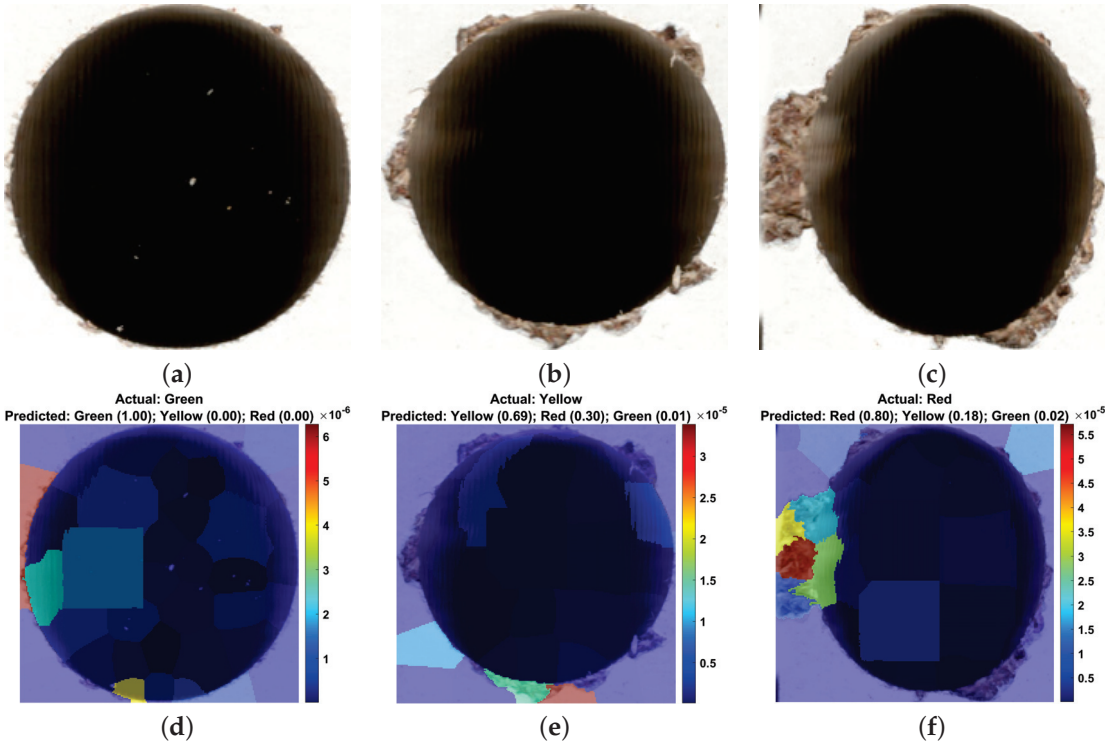


Figure 5. LIME-generated explanations for accurate predictions made by the VGG19 model on images featuring drilled holes are presented. The first row (**a**–**c**) displays the original images, while the second row (**d**–**f**) presents the LIME explanations corresponding to these images: (**a**) true class: Green, predicted class: Green; (**b**) true class: Yellow, predicted class: Yellow; (**c**) true class: Red, Predicted class: Red; (**d**) LIME explanation for image (**a**); (**e**) LIME explanation for image (**b**); (**f**) LIME explanation for image (**c**).

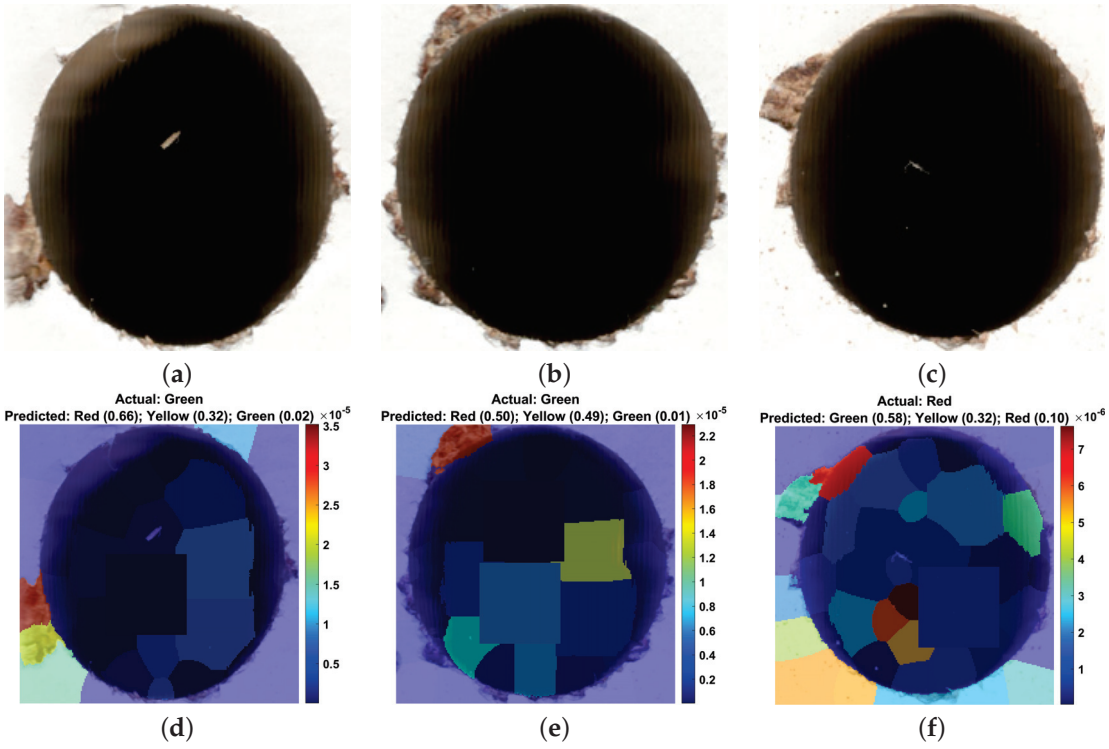


Figure 6. Explanations generated by LIME for misclassified predictions by the VGG19 model on images of drilled holes. The first row (**a**–**c**) presents the original images, while the second row

(**d**–**f**) displays the LIME-generated explanations corresponding to each image: (**a**) real category: Green, predicted category: Red; (**b**) real category: Green, predicted category: Red; (**c**) real category: Red, predicted category: Green; (**d**) explanation by LIME for image (**a**); (**e**) explanation by LIME for image (**b**); (**f**) explanation by LIME for image (**c**).

2.4.2. Gradient-Weighted Class Activation Mapping (Grad-CAM)

Gradient-weighted Class Activation Mapping (Grad-CAM) serves as a technique in deep learning that facilitates the visualization and interpretation of the reasoning behind decisions made by Convolutional Neural Networks (CNNs) [84–92]. By leveraging the gradients of the target class heading into the last convolutional layer, Grad-CAM produces a localization map. This map accentuates the critical regions within an input image that play a pivotal role in the model's predictive process.

The Grad-CAM technique involves several key steps:

- Forward pass—the input image is processed through the CNN to extract feature maps from the last convolutional layer.
- Gradient calculation—the gradients of the classification score with respect to the feature maps are computed. These gradients indicate the sensitivity of the output value of the target class to changes in the feature map values.
- Weight averaging—the gradients are averaged to obtain weights, which signify the importance of each feature map channel for the target class.
- Weighted combination—a weighted combination of the feature maps is performed using these weights, resulting in an initial coarse localization map.
- ReLU activation—the ReLU activation function is applied to the weighted combination to retain only positive influences, highlighting the regions in the image that positively impact the target class score.
- Upsampling—the coarse localization map is upsampled to match the dimensions of the input image, producing the final Grad-CAM heatmap.

This heatmap can be overlaid on the original image to visualize which parts of the image the network focused on during its prediction.

In our research, we utilized Grad-CAM technology with a customized VGG16 network, originally developed for classifying drilled holes in melamine-coated chipboard. Our objective was to identify the specific areas the CNN model focuses on when making predictions. This approach helped us delve deeper into the model's rationale and improved the clarity of our automated tool condition monitoring system.

3. Results and Discussion of CNN Models Classification

In this section, we disclose the findings from our study on Convolutional Neural Network (CNN) models applied to classify drilled holes in melamine-faced chipboard. We assessed the efficacy of each model by examining their accuracy, significant error frequencies, and confusion matrices over various folds. Detailed examinations of the outcomes for the VGG16, VGG19, and ResNet101 models are elaborated in the subsequent sections.

The evaluation metrics used in this study include:

- Accuracy—refers to the ratio of correctly identified instances to the overall number of instances.
- Critical errors—instances where the model misclassifies a "Green" hole as "Red" or vice versa, which are particularly significant due to their potential impact in practical applications.
- Confusion matrix—a table that provides a summary of the model's performance by displaying the count of accurate and inaccurate forecasts for every category.

Every model underwent training and assessment through a 5-fold cross-validation method, in which the dataset was segmented into five parts. Each part served as a test set on one occasion, with the rest of the parts merged to create the training set. Such a strategy guarantees a thorough appraisal of the model's effectiveness.

The subsequent subsections present comprehensive findings for each model, encompassing their precision, significant error frequencies, and confusion matrices. These findings are then examined in detail to pinpoint the advantages and disadvantages of each model and to identify the most appropriate model for the classification task.

3.1. Numerical Experiments

Figure 7 presents a flowchart illustrating the workflow for the numerical experiments conducted in this study. The process began with the acquisition of data, which included collecting images of drilled holes categorized into three classes: Green, Yellow, and Red. The next step was data preparation, where the dataset was organized and augmented to enhance model training.

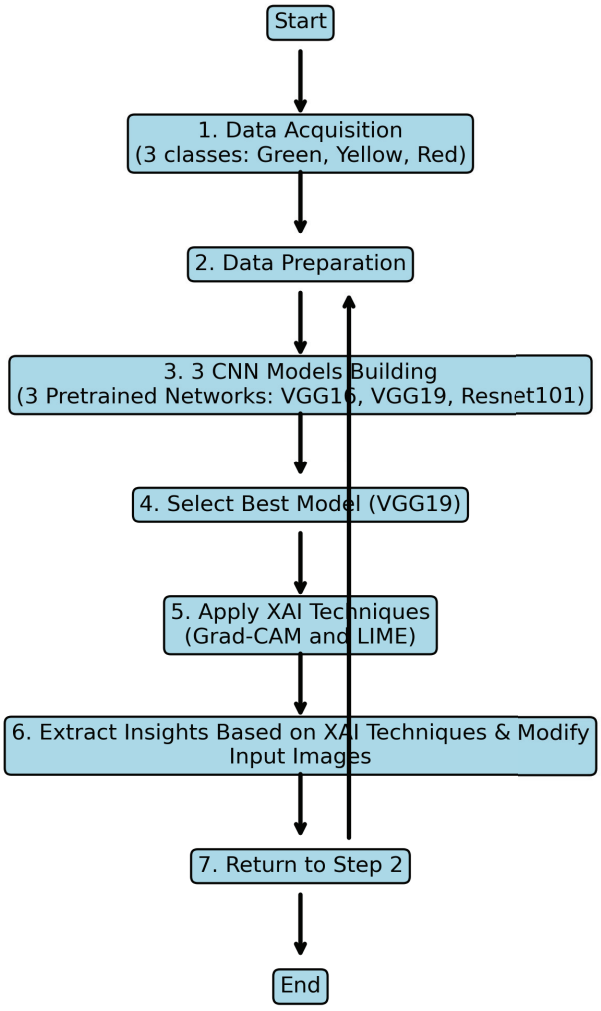


Figure 7. Flowchart illustrating the workflow for the numerical experiments conducted in this study.

Three CNN models—VGG16, VGG19, and ResNet101—were then constructed and fine-tuned for the classification task. After model evaluation, the best-performing model, which in this study was VGG19, was selected for further analysis. Explainable AI (XAI) techniques, specifically Grad-CAM and LIME, were applied to this model to gain insights into its decision-making process.

The insights gathered from these XAI techniques were then used to adjust and improve the input images or training process, after which the entire cycle was repeated if necessary. This iterative process ensures continuous model refinement and improved classification performance.

In this section, we detail the numerical tests carried out to assess how well CNN models can classify drilled holes in melamine-faced chipboard. We applied a 5-fold cross-validation method (involving five different drills), dividing the dataset into five parts. Each part was alternately used as a test set, with the other four parts serving for training.

We arranged the image data into five separate folders, each folder corresponding to distinct data collection dates. The folders included images classified into three categories: "Green", "Yellow", and "Red", indicating the state of the drilled holes.

For cross-validation, we generated five folds. In every fold, a single subset was selected as the test set, and the rest were amalgamated to create the training set. This strategy guaranteed that each subset was utilized as a test set once, offering a thorough assessment of the model's effectiveness.

For each fold, we performed the following steps:

1. Data preparation—the images in the training set were loaded and labeled accordingly.
2. Model configuration—we used the pretrained CNN network, modifying its final layers to fit our classification task. The fully connected layer was adjusted to have three outputs corresponding to the three classes, and the final classification layer was added.
3. Training—the network was trained using stochastic gradient descent with momentum (SGDM) for 10 epochs. Training parameters such as the mini-batch size (32) and initial learning rate (1×10^{-4}) were optimized to achieve the best performance.
4. Evaluation—the trained model was evaluated on the test set for each fold. We recorded the accuracy and generated confusion matrices to analyze the classification performance.

We have applied a 5-fold cross-validation approach, which involved splitting the dataset into training and test sets in such a way that each fold had data from one drill reserved for testing, while the remaining four drills' data were used for training. This division was suggested by industry experts to ensure the robustness and generalization of the models across different tools.

Table 3 provides a summary of the data distribution in each of the 5 folds. Each fold includes a test set from one drill and a training set from the remaining four drills. The table presents the number of images in both the training and test sets for each fold.

Table 3. Summary of data distribution in each fold of the 5-fold cross-validation.

Fold	Test Drill Number	Training Set Size	Test Set Size	Training/Test Percentage
1	1	6860	1666	80.46%/19.54%
2	2	6706	1820	78.65%/21.35%
3	3	6846	1680	80.30%/19.70%
4	4	6846	1680	80.30%/19.70%
5	5	6846	1680	80.30%/19.70%

3.2. Framework and Implementation Details

This section outlines the framework and implementation details for the use of Convolutional Neural Networks (CNNs) including VGG16, VGG19, and ResNet101 models, as well as Explainable AI (XAI) techniques such as Grad-CAM and LIME. The entire workflow was implemented in Python 3.10, utilizing well-established libraries and frameworks commonly used for deep learning and model interpretability.

3.2.1. Framework Overview

The models and XAI techniques were developed and executed using Python 3.10. The primary libraries used include:

- TensorFlow and Keras. These were used for building, training, and fine-tuning the CNN models (VGG16, VGG19, and ResNet101). TensorFlow and Keras offer high-level abstractions for defining and training deep neural networks.

- OpenCV and Pillow. These libraries were used for image preprocessing, including resizing, normalization, etc.
- Matplotlib and Seaborn. Visualization of the training process, evaluation metrics, and the output of the XAI techniques was performed using these libraries.
- scikit-learn. For data splitting, evaluation metrics, and auxiliary tasks such as generating cross-validation folds.

3.2.2. Implementation of CNN Models

The VGG16, VGG19, and ResNet101 models were implemented using the Keras API, which provides pre-trained models accessible from the keras.applications module. The models were fine-tuned to the task of classifying drilled holes into three classes: Green, Yellow, and Red.

The training process involved applying data augmentation techniques such as random rotations, flips, and scaling to increase dataset variability. Stochastic Gradient Descent (SGD) with momentum was used as the optimizer, while categorical cross-entropy was applied as the loss function.

3.2.3. Explainable AI (XAI) Techniques

To interpret the decisions of the CNN models, two XAI techniques were integrated: Gradient-weighted Class Activation Mapping (Grad-CAM) and Local Interpretable Model-agnostic Explanations (LIME).

Grad-CAM was implemented using TensorFlow and Keras. This technique generates class-discriminative localization maps by computing gradients of the target class concerning the feature maps in the final convolutional layer. The resulting heatmaps highlight the regions of the input image that contribute the most to the model's prediction. The implementation followed standard practices, using the tf.GradientTape() for computing gradients and overlaying the heatmaps on the original images for visualization.

LIME was implemented using the lLIME package in Python. This technique approximates the model's behavior by generating perturbed versions of the input image and fitting an interpretable model (such as a linear model) to the local predictions. The LIME explanations provide a feature importance map, indicating which parts of the image were critical for the classification decision. The ImageExplainer class was used for generating explanations specific to image data.

3.2.4. Model Evaluation and Interpretation

The models were evaluated using 5-fold cross-validation, with accuracy, precision, recall, and F1-score as the primary metrics. Grad-CAM and LIME were applied both to correctly classified and misclassified samples, providing insights into model behavior and highlighting areas for potential improvement.

By combining the power of CNN architectures with explainability methods like Grad-CAM and LIME, the study not only achieved reasonable classification accuracy, but also offered transparency into the decision-making process, which is crucial for industrial applications.

3.3. Results for VGG16 Model

The evaluation results for the VGG16 model are presented in Table 4. The model's accuracy varied across the folds, with the highest accuracy of 71.00% observed in the first fold and the lowest accuracy of 56.13% observed in the fourth fold. The overall accuracy across all folds was 66.60%.

Table 4. Evaluation results of VGG16 model.

Fold / Drill #	Accuracy	Critical Errors		
		Green as Red	Red as Green	Total
1	71.00%	1	3	4
2	70.82%	8	11	19
3	63.98%	1	58	59
4	56.13%	61	0	61
5	70.77%	1	15	16
Total	66.60%	72	87	159

The evaluation metrics for the VGG16 model, as presented in Table 5, provide a detailed analysis of the model's performance across three classes: "Green", "Yellow", and "Red". The table highlights key indicators such as precision, sensitivity, specificity, accuracy, and F1 score.

Table 5. Detailed evaluation metrics for VGG16 model.

Class	Precision (%)	Sensitivity (%)	Specificity (%)	Accuracy (%)	F1 Score (%)
Green	79.58	84.05	82.83	83.37	81.76
Yellow	51.97	52.86	76.11	68.47	52.41
Red	60.62	52.52	89.91	81.37	56.28

For the "Green" class, the model demonstrates strong performance with a precision of 79.58%, a sensitivity of 84.05%, and an F1 score of 81.76%. The high values in these metrics suggest that the model is effective in correctly identifying and classifying instances as "Green", with a balanced trade-off between precision and recall.

In contrast, the "Yellow" class shows a significant drop in performance. The precision and sensitivity are 51.97% and 52.86%, respectively, with a corresponding F1 score of 52.41%. These results indicate that the model struggles to correctly classify instances in the "Yellow" category, leading to a higher number of misclassifications. The specificity for the "Yellow" class is higher at 76.11%, reflecting a better ability to distinguish "Yellow" instances from non-"Yellow" instances, but overall accuracy remains lower at 68.47%.

The "Red" class also faces challenges, with a precision of 60.62% and a sensitivity of 52.52%. Although the specificity is high at 89.91%, the F1 score of 56.28% reflects the model's difficulty in accurately classifying instances as "Red", with a tendency to misclassify these instances as either "Green" or "Yellow".

Overall, the table indicates that the model performs well for the "Green" class, but struggles to maintain the same level of accuracy and precision for the "Yellow" and "Red" classes. These findings highlight areas for improvement, especially in refining the classification of intermediate and highly worn drill holes.

The VGG16 model's confusion matrix, depicted in Figure 8, offers an in-depth examination of its classification efficacy over various folds. This matrix displays the count of instances that were accurately and inaccurately categorized for each class.

In the confusion matrix, the true positive rates for the classes "Green", "Yellow", and "Red" are as follows:

- The "Green" class had 3177 correctly classified instances, with 531 instances misclassified as "Yellow" and 72 instances as "Red".
- The "Yellow" class had 1480 correctly classified instances, with 728 instances misclassified as "Green" and 592 instances as "Red".
- The "Red" class had 1022 correctly classified instances, with 837 instances misclassified as "Yellow" and 87 instances as "Green".

These results indicate that the VGG16 model achieved moderate accuracy, with varying degrees of misclassification among the classes. The model showed a higher tendency to misclassify instances of the "Yellow" class, particularly as "Green" and "Red".

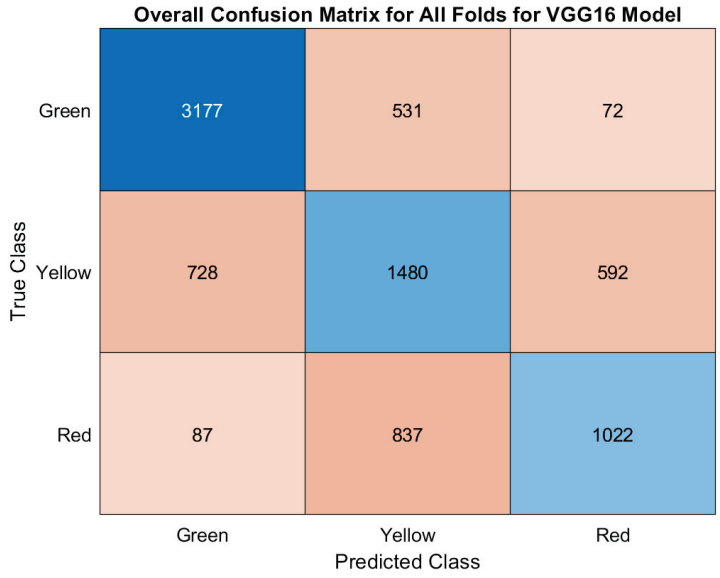


Figure 8. Overall confusion matrix for all folds for VGG16 model.

Critical errors refer to the misclassification of instances where the model incorrectly identifies a "Green" drill hole as "Red" or vice versa. Such errors are particularly concerning due to their potential implications in real-world applications.

- Fold 1—the model demonstrated a relatively high accuracy of 71.00%. Critical errors included one instance where a "Green" hole was classified as "Red" and three instances where "Red" holes were classified as "Green". The total number of critical errors in this fold was four.
- Fold 2—the accuracy for this fold was 70.82%. Critical errors increased significantly, with eight instances of "Green" holes misclassified as "Red", and 11 instances of "Red" holes misclassified as "Green". The total number of critical errors in this fold was 19.
- Fold 3—this fold showed a lower accuracy of 63.98%. There was one instance where a "Green" hole was misclassified as "Red", and a substantial number of 58 instances where "Red" holes were misclassified as "Green". The total number of critical errors was 59.
- Fold 4—the accuracy dropped to 56.13% in this fold. A significant number of 61 instances where "Green" holes were classified as "Red" were observed. No instances of "Red" holes misclassified as "Green" were recorded. The total number of critical errors was 61.
- Fold 5—the accuracy for this fold was 70.77%. Critical errors included one instance where a "Green" hole was classified as "Red", and 15 instances where "Red" holes were classified as "Green". The total number of critical errors in this fold was 16.

The analysis of critical errors reveals several insights into the performance of the VGG16 model:

1. Consistency in Green to Red misclassification—across all folds, there is a consistent occurrence of "Green" holes being misclassified as "Red". This misclassification is particularly problematic, as it could lead to unnecessary corrective actions or rework.
2. Variation in Red to Green misclassification—the misclassification of "Red" holes as "Green" varies significantly across folds. For instance, Fold 4 had no such misclassifications, while Fold 3 had a substantial number (58 instances).
3. Impact on overall accuracy—the high number of critical errors, especially in certain folds, significantly impacts the overall accuracy of the model. Folds with higher critical errors generally correspond to lower accuracy rates.
4. Need for model improvement—the analysis suggests that while the VGG16 model shows promise, there is a need for further refinement to reduce the critical error rate.

This could involve tuning the model parameters, using more advanced architectures, or incorporating additional data preprocessing techniques.

By addressing these critical errors, the reliability and applicability of the VGG16 model for classifying drilled holes in melamine-faced chipboard can be enhanced, leading to more consistent and accurate results in practical applications.

3.4. Results for VGG19 Model

The evaluation results for the VGG19 model are presented in Table 6. The model's accuracy varied across the folds, with the highest accuracy of 71.60% observed in the fifth fold and the lowest accuracy of 52.38% observed in the fourth fold. The overall accuracy across all folds was 67.03%.

Table 6. Evaluation results of VGG19 model.

Fold / Drill #	Accuracy	Critical Errors		
		Green as Red	Red as Green	Total
1	69.50%	2	2	4
2	70.76%	19	5	24
3	70.59%	1	10	11
4	52.38%	57	0	57
5	71.60%	0	11	11
Total	67.03%	79	28	107

The evaluation metrics for the VGG19 model, as shown in Table 7, provide an in-depth analysis of the model's performance across the "Green", "Yellow", and "Red" classes. The table includes metrics such as precision, sensitivity, specificity, accuracy, and F1 score, which give a comprehensive understanding of the model's classification capabilities.

Table 7. Detailed evaluation metrics for VGG19 model.

Class	Precision (%)	Sensitivity (%)	Specificity (%)	Accuracy (%)	F1 Score (%)
Green	84.86	78.89	88.79	84.40	81.77
Yellow	51.51	58.32	73.16	68.29	54.71
Red	59.72	56.53	88.72	81.37	58.08

For the "Green" class, the model exhibits strong performance, with a precision of 84.86%, sensitivity of 78.89%, and an F1 score of 81.77%. The high values for specificity (88.79%) and accuracy (84.40%) suggest that the model effectively identifies and classifies instances belonging to the "Green" category, with minimal misclassifications.

The "Yellow" class shows moderate performance, with a precision of 51.51% and sensitivity of 58.32%. The F1 score of 54.71% indicates that the model struggles with distinguishing "Yellow" instances, often confusing them with the other classes. The specificity of 73.16% and accuracy of 68.29% further highlight the challenges the model faces in correctly classifying "Yellow" instances, resulting in a higher rate of misclassification.

For the "Red" class, the model achieves a precision of 59.72% and a sensitivity of 56.53%. Although the specificity is high at 88.72%, indicating a good ability to correctly exclude non-"Red" instances, the F1 score of 58.08% reflects the model's difficulty in accurately classifying "Red" instances. The accuracy for the "Red" class stands at 81.37%, suggesting some degree of reliability, though the model still struggles with edge cases.

In summary, while the VGG19 model shows robust performance for the "Green" class, it has noticeable difficulties in accurately classifying the "Yellow" and "Red" classes. These results point to the need for further fine-tuning and model improvements, particularly for intermediate and heavily worn drill conditions.

The confusion matrix of the VGG19 model, depicted in Figure 9, offers a comprehensive analysis of its classification accuracy across all folds. It details the count of both accurately and inaccurately classified instances by class.

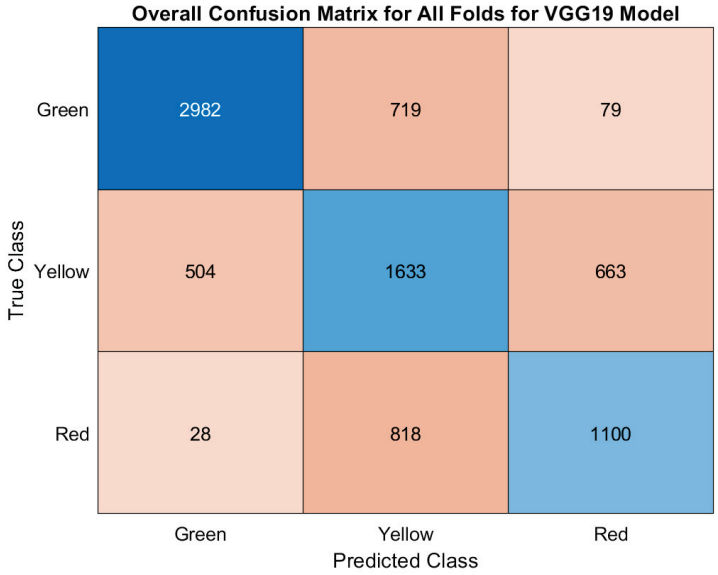


Figure 9. Overall confusion matrix for all folds for VGG19 model.

In the confusion matrix, the true positive rates for the classes "Green", "Yellow", and "Red" are as follows:

- The "Green" class had 2982 correctly classified instances, with 719 instances misclassified as "Yellow" and 79 instances as "Red".
- The "Yellow" class had 1633 correctly classified instances, with 504 instances misclassified as "Green" and 663 instances as "Red".
- The "Red" class had 1100 correctly classified instances, with 818 instances misclassified as "Yellow" and 28 instances as "Green".

These results indicate that the VGG19 model achieved moderate accuracy, with varying degrees of misclassification among the classes. The model showed a higher tendency to misclassify instances of the "Yellow" class, particularly as "Green" and "Red".

Critical errors refer to the misclassification of instances where the model incorrectly identifies a "Green" drill hole as "Red", or vice versa. Such errors are particularly concerning due to their potential implications in real-world applications.

- Fold 1—the model demonstrated a relatively high accuracy of 69.50%. Critical errors included two instances where a "Green" hole was classified as "Red", and two instances where "Red" holes were classified as "Green". The total number of critical errors in this fold was four.
- Fold 2—the accuracy for this fold was 70.76%. Critical errors increased significantly, with 19 instances of "Green" holes misclassified as "Red", and five instances of "Red" holes misclassified as "Green". The total number of critical errors in this fold was 24.
- Fold 3—this fold showed an accuracy of 70.59%. There was one instance where a "Green" hole was misclassified as "Red", and 10 instances where "Red" holes were misclassified as "Green". The total number of critical errors was 11.
- Fold 4—the accuracy dropped to 52.38% in this fold. A significant number of 57 instances where "Green" holes were classified as "Red" were observed. No instances of "Red" holes misclassified as "Green" were recorded. The total number of critical errors was 57.

- Fold 5—the accuracy for this fold was 71.60%. Critical errors included no instances where a "Green" hole was classified as "Red", and 11 instances where "Red" holes were classified as "Green". The total number of critical errors in this fold was 11.

The analysis of critical errors reveals several insights into the performance of the VGG19 model:

1. Consistency in Green to Red misclassification—across all folds, there is a consistent occurrence of "Green" holes being misclassified as "Red". This misclassification is particularly problematic, as it could lead to unnecessary corrective actions or rework.
2. Variation in Red to Green misclassification—the misclassification of "Red" holes as "Green" varies significantly across folds. For instance, Fold 4 had no such misclassifications, while Fold 2 had a higher number (19 instances).
3. Impact on overall accuracy—the high number of critical errors, especially in certain folds, significantly impacts the overall accuracy of the model. Folds with higher critical errors generally correspond to lower accuracy rates.
4. Need for model improvement—the analysis suggests that while the VGG19 model shows promise, there is a need for further refinement to reduce the critical error rate. This could involve tuning the model parameters, using more advanced architectures, or incorporating additional data preprocessing techniques.

By addressing these critical errors, the reliability and applicability of the VGG19 model for classifying drilled holes in melamine-faced chipboard can be enhanced, leading to more consistent and accurate results in practical applications.

3.5. Results for Resnet101 Model

The evaluation results for the Resnet101 model are presented in Table 8. The model's accuracy varied across the folds, with the highest accuracy of 66.42% observed in the fifth fold and the lowest accuracy of 56.04% observed in the second fold. The overall accuracy across all folds was 61.19%.

Table 8. Evaluation results of Resnet101 model.

Fold / Drill #	Accuracy	Critical Errors		
		Green as Red	Red as Green	Total
1	66.26%	26	13	39
2	56.04%	158	7	165
3	65.59%	5	39	44
4	56.13%	64	2	66
5	66.42%	3	16	19
Total	61.19%	256	77	333

The evaluation metrics for the ResNet101 model, as presented in Table 9, provide a comprehensive analysis of the model's performance across the "Green", "Yellow", and "Red" classes. These metrics include precision, sensitivity (recall), specificity, accuracy, and F1 score, offering a detailed view of how well the model performs for each class.

Table 9. Detailed evaluation metrics for ResNet101 model.

Class	Precision (%)	Sensitivity (%)	Specificity (%)	Accuracy (%)	F1 Score (%)
Green	81.45	80.48	85.40	83.22	80.96
Yellow	47.52	36.96	80.04	65.89	41.58
Red	46.23	62.08	78.65	74.87	52.99

For the "Green" class, the model achieves a precision of 81.45% and a sensitivity of 80.48%. The F1 score of 80.96% indicates a balanced performance in identifying "Green" instances, with relatively few misclassifications. The model's specificity of 85.40% and

overall accuracy of 83.22% suggest that it is generally reliable in distinguishing the "Green" class from others.

The "Yellow" class poses a significant challenge for the model, as reflected by the lower precision of 47.52% and sensitivity of 36.96%. The F1 score of 41.58% highlights the model's struggle to accurately classify "Yellow" instances, often leading to confusion with the "Green" and "Red" classes. Although the specificity for the "Yellow" class is reasonably high at 80.04%, the overall accuracy is lower at 65.89%, indicating that a considerable number of "Yellow" instances are misclassified.

For the "Red" class, the model shows moderate performance with a precision of 46.23% and a sensitivity of 62.08%. The F1 score of 52.99% reflects the model's tendency to misclassify "Red" instances, although it performs slightly better than for the "Yellow" class. The specificity of 78.65% and accuracy of 74.87% suggest that the model can somewhat differentiate "Red" instances, but still faces challenges in correctly classifying all of them.

In summary, while the ResNet101 model demonstrates reasonably strong performance for the "Green" class, its effectiveness decreases significantly for the "Yellow" and "Red" classes. The lower precision and F1 scores for these classes indicate a need for further refinement and optimization of the model, particularly in addressing the classification challenges associated with intermediate and heavily worn drill conditions.

The confusion matrix for the Resnet101 model, as shown in Figure 10, provides a detailed breakdown of the classification performance across all folds. The matrix indicates the number of correctly and incorrectly classified instances for each class.

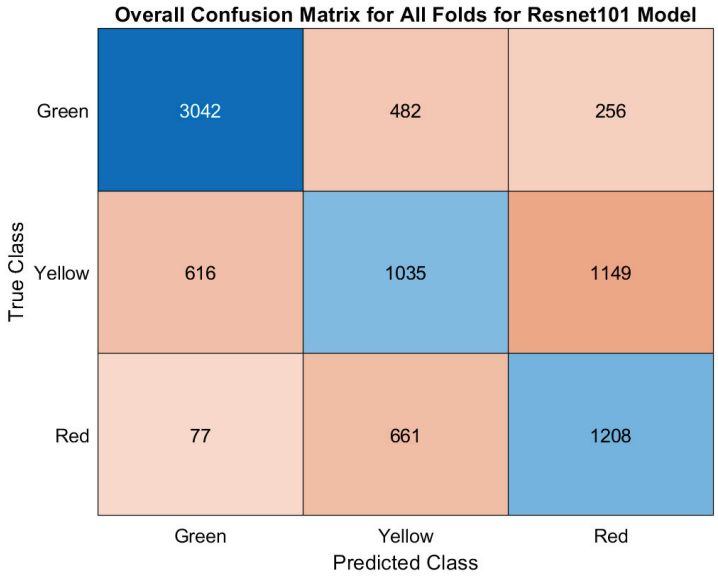


Figure 10. Overall confusion matrix for all folds for Resnet101 model.

In the confusion matrix, the true positive rates for the classes "Green", "Yellow", and "Red" are as follows:

- The "Green" class had 2780 correctly classified instances, with 846 instances misclassified as "Yellow" and 256 instances as "Red".
- The "Yellow" class had 1394 correctly classified instances, with 623 instances misclassified as "Green" and 938 instances as "Red".
- The "Red" class had 962 correctly classified instances, with 906 instances misclassified as "Yellow" and 77 instances as "Green".

These results indicate that the Resnet101 model achieved moderate accuracy, with varying degrees of misclassification among the classes. The model showed a higher tendency to misclassify instances of the "Yellow" class, particularly as "Green" and "Red".

Critical errors refer to the misclassification of instances where the model incorrectly identifies a "Green" drill hole as "Red" or vice versa. Such errors are particularly concerning due to their potential implications in real-world applications.

- Fold 1—the model demonstrated a relatively high accuracy of 66.26%. Critical errors included 26 instances where a "Green" hole was classified as "Red", and 13 instances where "Red" holes were classified as "Green". The total number of critical errors in this fold was 39.
- Fold 2—the accuracy for this fold was 56.04%. Critical errors increased significantly, with 158 instances of "Green" holes misclassified as "Red", and seven instances of "Red" holes misclassified as "Green". The total number of critical errors in this fold was 165.
- Fold 3—this fold showed an accuracy of 65.59%. There were five instances where a "Green" hole was misclassified as "Red", and 39 instances where "Red" holes were misclassified as "Green". The total number of critical errors was 44.
- Fold 4—the accuracy dropped to 56.13% in this fold. A significant number of 64 instances where "Green" holes were classified as "Red" were observed. Only two instances of "Red" holes misclassified as "Green" were recorded. The total number of critical errors was 66.
- Fold 5—the accuracy for this fold was 66.42%. Critical errors included three instances where a "Green" hole was classified as "Red", and 16 instances where "Red" holes were classified as "Green". The total number of critical errors in this fold was 19.

The analysis of critical errors reveals several insights into the performance of the Resnet101 model:

1. Consistency in Green to Red misclassification—across all folds, there is a consistent occurrence of "Green" holes being misclassified as "Red". This misclassification is particularly problematic as it could lead to unnecessary corrective actions or rework.
2. Variation in Red to Green misclassification—the misclassification of "Red" holes as "Green" varies significantly across folds. For instance, Fold 4 had only two such misclassifications, while Fold 3 had a higher number (39 instances).
3. Impact on overall accuracy—the high number of critical errors, especially in certain folds, significantly impacts the overall accuracy of the model. Folds with higher critical errors generally correspond to lower accuracy rates.
4. Need for model improvement—the analysis suggests that while the Resnet101 model shows promise, there is a need for further refinement to reduce the critical error rate. This could involve tuning the model parameters, using more advanced architectures, or incorporating additional data preprocessing techniques.

By addressing these critical errors, the reliability and applicability of the Resnet101 model for classifying drilled holes in melamine-faced chipboard can be enhanced, leading to more consistent and accurate results in practical applications.

3.5.1. Comparison of VGG16, VGG19, and Resnet101 Models

The evaluation of the VGG16, VGG19, and Resnet101 models focused on assessing their accuracy and significant error rates. A summary of the evaluation metrics is provided in Table 10.

Table 10. Comparison of VGG16, VGG19, and Resnet101 models.

Model	Average Accuracy	Total Critical Errors	Consistency of Performance
VGG16	66.60%	159	Moderate
VGG19	67.03%	107	High
Resnet101	61.19%	333	Low

The VGG16 architecture reached a mean accuracy level of 66.60%, accompanied by 159 critical errors. Conversely, the VGG19 architecture demonstrated a modest improvement, attaining a mean accuracy of 67.03% while registering 107 critical errors, marking it as a more accurate and reliable choice than the VGG16. Meanwhile, the Resnet101 model recorded the least impressive performance, with an average accuracy rate of 61.19% and the greatest amount of critical errors, totaling 333.

3.5.2. Selection of the Best CNN Model

Based on the comparison, the VGG19 model stands out as the best among the three models evaluated. It not only achieved the highest average accuracy, but also had the lowest number of critical errors. The consistency of its performance across different folds further supports its robustness and reliability.

In practical applications, the lower critical error rate of the VGG19 model is particularly advantageous, as it reduces the risk of misclassifications that could lead to costly errors in the manufacturing processes. Therefore, the VGG19 model is selected as the best model for the classification of drilled holes in melamine-faced chipboard.

In conclusion, the VGG19 model demonstrated superior performance in terms of accuracy and critical error rate compared to the VGG16 and Resnet101 models. Its consistent performance across different folds and lower propensity for critical errors make it the most reliable choice for the classification task. Future work will focus on further refining the VGG19 model to enhance its accuracy and reduce misclassification rates even further.

4. Results and Discussion of Explainable AI (XAI) for VGG19 Model

In this section, we discuss the application of two Explainable AI (XAI) techniques, namely Local Interpretable Model-agnostic Explanations (LIME) and Gradient-weighted Class Activation Mapping (Grad-CAM), to the VGG19 model. The aim is to interpret the model's decisions and identify the most influential features contributing to the classification of drilled holes in melamine-faced chipboard.

LIME Results

Figure 5 shows LIME-generated explanations for correctly classified VGG19 model's predictions on images of drilled holes. The top row (a–c) shows the original images, and the bottom row (d–f) shows the corresponding LIME explanations:

- (a) Actual class: Green, predicted class: Green. The LIME analysis in (d) underscores the sections of the picture that had the most significant impact on the model's identification of the "Green" class.
- (b) Actual class: Yellow, predicted class: Yellow. The LIME explanation in (e) indicates the areas that influenced the model's decision to classify the image as "Yellow".
- (c) Actual class: Red, predicted class: Red. The LIME explanation in (f) shows the key features that led the model to correctly identify the image as belonging to the "Red" class.

Figure 6 shows LIME-generated explanations for misclassified VGG19 model's predictions on images of drilled holes. The top row (a–c) shows the original images, and the bottom row (d–f) shows the corresponding LIME explanations:

- (a) Actual class: Green, predicted class: Red. The LIME explanation in (d) highlights the regions of the image that contributed most to the model's prediction of the "Red" class. Based on the explanation image, it is evident that the actual condition should be Red instead of Green, and the VGG19 model predicted correctly (66% for the red class). However, the problem is that sometimes even with a good drill bit, damages (artifacts in the image) can occur due to local poor quality of the material or other issues not resulting from drill bit wear.
- (b) Actual class: Green, predicted class: Red. The LIME explanation in (e) indicates the areas that influenced the model's decision to classify the image as "Red". Similarly,

in this case, the state according to the image should be Red/Yellow. The model decided on the Red class (50% for red, 49% for yellow) on the boundary.

- (c) Actual class: Red, predicted class: Green. The LIME explanation in (f) shows the key features that led the model to identify the image as belonging to the "Green" class. In this case, the problem lies with the model; it should have assigned the Red/Yellow class. However, it shows 58% for the green class and 32% for the yellow class, and only 10% for the Red class. The issue might be the greater number of artifacts at the bottom left of the drilled hole.

Grad-CAM Results

Figure 11 shows Grad-CAM-generated explanations for correctly classified VGG19 model's predictions on images of drilled holes. The top row (a–c) shows the original images, and the bottom row (d–f) shows the corresponding Grad-CAM explanations:

- (a) Actual class: Green, predicted class: Green. The Grad-CAM explanation in (d) highlights the regions of the image that contributed most to the model's prediction of the "Green" class.
- (b) Actual class: Yellow, predicted class: Yellow. The Grad-CAM explanation in (e) indicates the areas that influenced the model's decision to classify the image as "Yellow".
- (c) Actual class: Red, predicted class: Red. The Grad-CAM explanation in (f) shows the key features that led the model to correctly identify the image as belonging to the "Red" class.

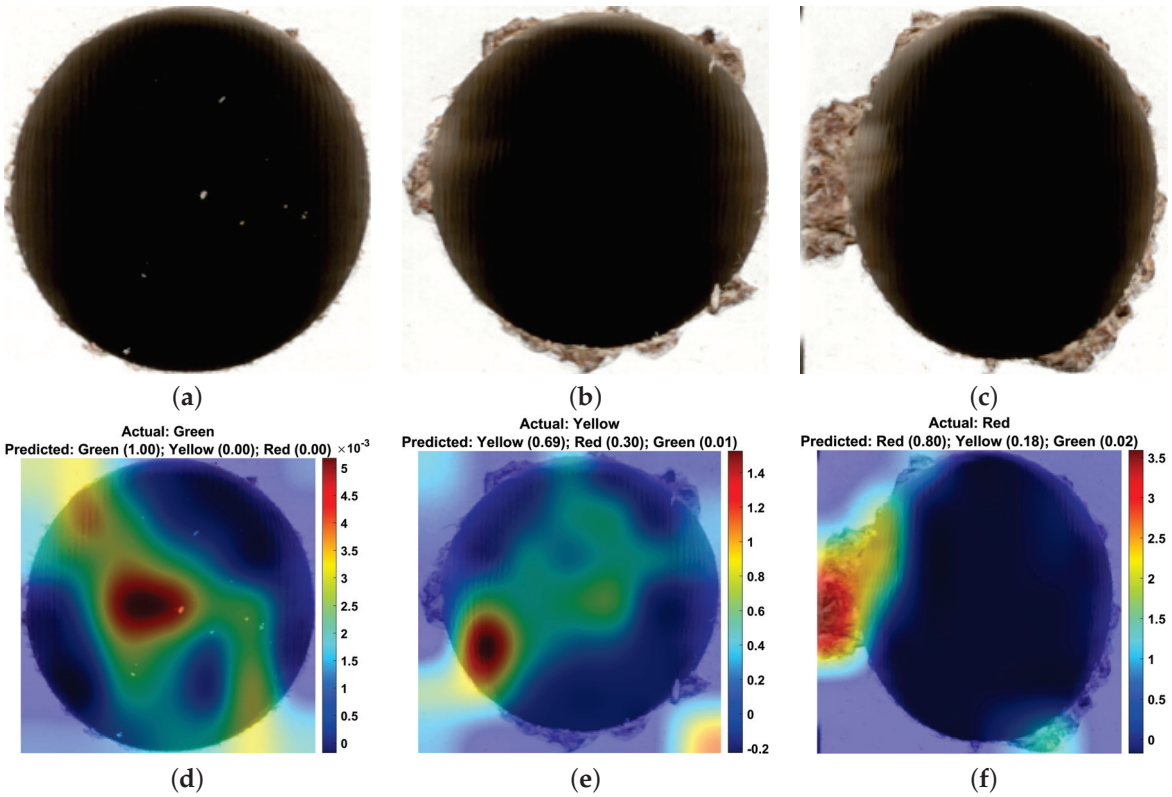


Figure 11. Explanations generated by Grad-CAM for the VGG19 model's correct classifications of images showcasing drilled holes are presented. The initial row, denoted as (**a**–**c**), displays the original images, while the subsequent row, labeled as (**d**–**f**), presents the Grad-CAM interpretations corresponding to these images: (**a**) true class: Green, model's prediction: Green; (**b**) true class: Yellow, model's prediction: Yellow; (**c**) true class: Red, model's prediction: Red; (**d**) presents the Grad-CAM explanation for image (**a**), (**e**) for image (**b**), and (**f**) for image (**c**).

Figure 12 shows Grad-CAM-generated explanations for misclassified VGG19 model's predictions on images of drilled holes. The top row (a–c) shows the original images, and the bottom row (d–f) shows the corresponding Grad-CAM explanations:

- (a) Actual class: Green, predicted class: Red. The Grad-CAM explanation in (d) highlights the regions of the image that contributed most to the model's prediction of the "Red" class. Here, it is evident that based on the image alone, the class should be Red, and the model performed very well with its decision (66% for the Red class, 32% for the Yellow class). The problem seems to be related to issues other than a worn drill bit, such as local poor quality of the material or other reasons not necessarily connected to a blunt drill bit.
- (b) Actual class: Green, predicted class: Red. The Grad-CAM explanation in (e) indicates the areas that influenced the model's decision to classify the image as "Red". Similarly, in this case, the model made a very good decision, indicating the class should be Red (50%) or Yellow (49%).
- (c) Actual class: Red, predicted class: Green. The Grad-CAM explanation in (f) shows the key features that led the model to identify the image as belonging to the "Green" class. In this case, the model did not perform well. The class should be Red, or possibly Yellow (based on the drilled hole image alone), but the model assigned the class as Green (58%) or Yellow (32%), with only 10% for the Red class. The issue might be the greater number of artifacts at the bottom left of the drilled hole.

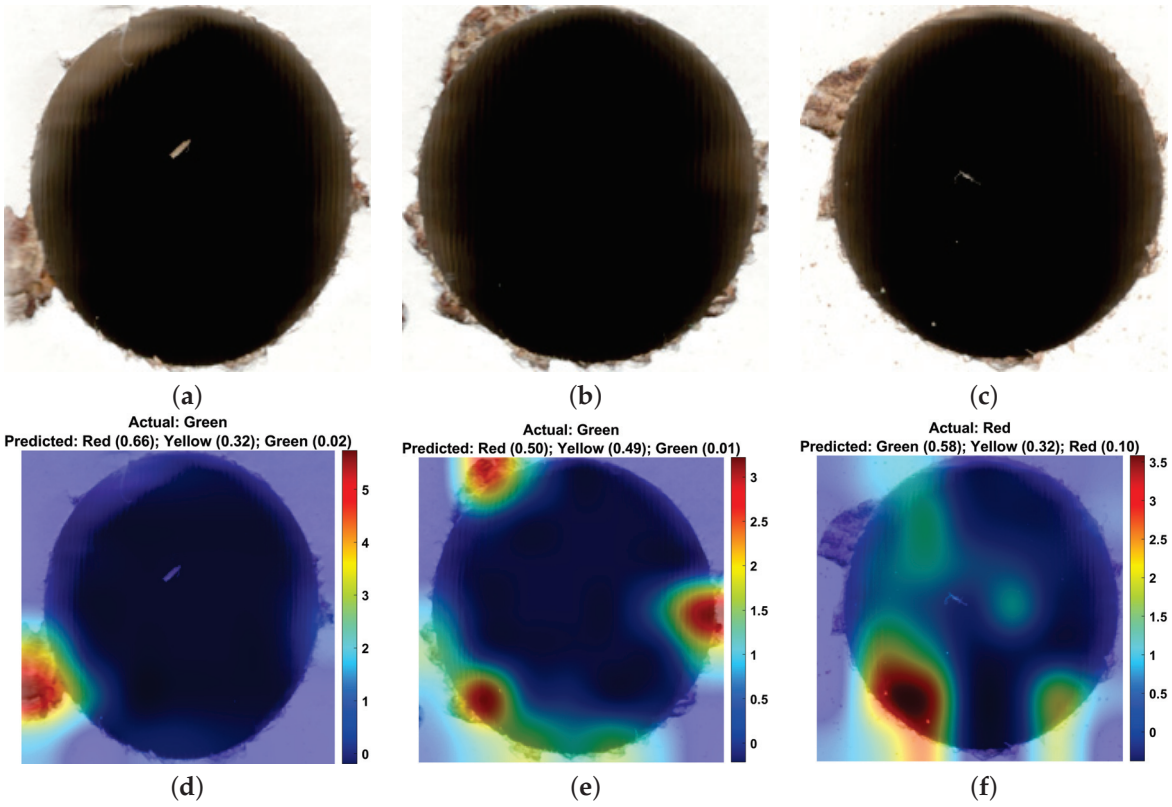


Figure 12. Explanations provided by Grad-CAM for incorrectly categorized images by the VGG19 model, focusing on drilled holes. The initial row (**a**–**c**) presents the genuine pictures, while the subsequent row (**d**–**f**) illustrates the Grad-CAM justifications corresponding to each: (**a**) true class: Green, identified as: Red; (**b**) true class: Green, identified as: Red; (**c**) true class: Red, identified as: Green; (**d**) is the Grad-CAM rationale for (**a**), (**e**) is for (**b**), and (f) for (**c**).

4.1. Comparison of LIME and Grad-CAM Explanations for VGG19

The explanations generated by Local Interpretable Model-agnostic Explanations (LIME) and Gradient-weighted Class Activation Mapping (Grad-CAM) provide insights into the VGG19 model's decision-making process. This section compares the two techniques based on their ability to explain the classifications of drilled holes in melamine-faced chipboard.

LIME produces explanations by perturbing the input data and observing the changes in the model's predictions. It generates local approximations of the model's behavior, highlighting the features most influential in a particular decision. Grad-CAM, on the other hand, uses the gradients of the target class flowing into the final convolutional layer to create a localization map highlighting the important regions in the input image.

4.1.1. Correctly Classified Instances

For correctly classified instances, both LIME and Grad-CAM highlight relevant regions that contribute to the VGG19 model's predictions. Figures 5 and 11 illustrate the explanations for the same set of correctly classified images. LIME focuses on specific features and areas in the images, showing which parts of the drilled holes most influenced the classification as "Green", "Yellow", or "Red". Grad-CAM highlights broader regions within the images that are important for the model's decisions. It provides a more holistic view of the areas contributing to the classification.

Both techniques show a high degree of agreement in identifying the critical regions for classification. However, LIME's explanations are more granular, focusing on smaller image patches, while Grad-CAM provides a more general overview of the important areas.

4.1.2. Misclassified Instances

Figures 6 and 12 show the explanations for misclassified instances by LIME and Grad-CAM, respectively. In cases of misclassification, LIME identifies specific features that misled the model. For example, it shows which regions of a "Green" hole led to its misclassification as "Red". Grad-CAM indicates larger areas that influenced the wrong classification, providing a broader context for understanding the misclassification.

In nearly all cases, both LIME and Grad-CAM indicate similar regions (features) of explainability in the image. However, there is an exception in Figures 6c and 12c—the same case where the model struggled more with the decision. This discrepancy in the explainability regions highlighted by the two algorithms indicates that the model's decision-making process was less certain.

4.1.3. Discrepancies in Explanations

The discrepancies between LIME and Grad-CAM in the case shown in Figures 6c and 12c can be attributed to several factors:

- Local vs. global interpretations—LIME provides local interpretations by focusing on small perturbed regions, while Grad-CAM offers a global view based on gradients. The difference in their approach can lead to highlighting different aspects of the image.
- Sensitivity to perturbations—LIME's reliance on perturbations can sometimes make it sensitive to noise and local variations, which might not significantly affect the gradient-based approach of Grad-CAM.
- Model uncertainty—the model's lower confidence in its prediction for this particular case likely caused the divergence. When the model is uncertain, the explanations from different methods can vary more significantly, reflecting the underlying ambiguity in the model's decision process.
- Feature importance—different algorithms might prioritize different features. LIME might highlight specific small regions that it considers crucial, while Grad-CAM might emphasize broader regions that overall influence the decision.

4.1.4. Strengths and Weaknesses

- LIME:
  - Strengths—provides detailed, feature-level explanations. Useful for understanding specific reasons behind individual predictions.
  - Weaknesses—can be computationally intensive due to the need for multiple perturbations. May not capture the global context as effectively.
- Grad-CAM:
  - Strengths—offers a high-level view of important regions, useful for understanding the model's overall focus. Computationally efficient compared to LIME.
  - Weaknesses—less detailed than LIME, may not highlight specific features as precisely.

4.1.5. Discussion of Yellow Class Misclassifications

The misclassification of instances belonging to the "Yellow" class deserves special attention, as it represents an intermediate state in the condition of drilled holes. The "Yellow" category is particularly challenging to classify, due to its transitional nature between the "Green" (good) and "Red" (requiring replacement) classes. This ambiguity introduces uncertainties in both model predictions and human assessments.

One of the primary challenges in accurately identifying the "Yellow" class lies in its inherent variability. Unlike the "Green" and "Red" classes, which have more clearly defined characteristics (new or significantly worn drill bits, respectively), the "Yellow" class represents a range of wear levels that are subject to interpretation. Factors such as minor surface defects, slight roughness around the edges of the drilled hole, or localized material inconsistencies can cause confusion between adjacent classes.

The CNN models tested, including VGG16, VGG19, and ResNet101, frequently struggled with distinguishing "Yellow" from either "Green" or "Red". As seen in the confusion matrices, a substantial proportion of "Yellow" instances were misclassified as either "Green" or "Red". This tendency was particularly noticeable in cases where the visual characteristics of "Yellow" overlapped with those of the other two classes. These misclassifications are not unexpected, considering the visual continuity in the degradation of drill bit performance over time.

From an industrial perspective, the misclassification of "Yellow" instances as "Green" or "Red" can have different implications. If a "Yellow" hole is misclassified as "Green", there is a risk of continuing to use a drill bit that is approaching its wear limit, potentially leading to quality issues in subsequent operations. On the other hand, misclassifying a "Yellow" hole as "Red" could lead to premature replacement of a drill bit, resulting in unnecessary downtime and increased operational costs.

To reduce the frequency of these misclassifications, several strategies could be explored:

- Improved data labeling. One approach involves refining the labeling process to better capture the nuances within the "Yellow" class. Introducing subcategories or gradations within the "Yellow" class might help the model learn the subtle differences between moderate wear levels.
- Enhanced feature extraction. Incorporating additional feature extraction techniques, such as edge detection or texture analysis, could improve the model's ability to distinguish between "Yellow" and other classes. These features could provide more detailed information about the quality of the drilled holes, leading to more accurate classifications.
- Artifact removal based on LIME/Grad-CAM explanations. Analysis using LIME and Grad-CAM can identify specific artifacts in the images that lead to misclassifications. By focusing on these artifacts and either removing them from the dataset or adjusting the image preprocessing pipeline to mitigate their effects, the model's performance can be improved, reducing the likelihood of incorrect class assignments.
- Ensemble methods. Using ensemble models that combine the predictions of multiple CNN architectures may help to mitigate misclassifications. By aggregating the outputs

of different models, it is possible to achieve a more balanced and reliable classification, especially for ambiguous cases.
- Human-in-the-loop systems. Implementing a hybrid system where "Yellow" classifications are verified by human experts could provide a safety net for cases where the model's confidence is low. This approach allows for continuous improvement through iterative feedback and model retraining.

Further research should focus on understanding the visual and contextual factors that contribute to "Yellow" misclassifications. Analyzing the decision boundaries in more detail using Explainable AI (XAI) techniques, such as LIME and Grad-CAM, can provide insights into the features the model struggles with. Additionally, investigating how the model's uncertainty varies across the three classes could guide the development of more robust classification systems.

In conclusion, while the "Yellow" class poses significant challenges for both machine learning models and human operators, targeted strategies can help to minimize misclassification rates and improve overall system performance. Addressing these issues is critical for developing reliable tool condition monitoring systems that can effectively support decision-making in industrial environments.

4.1.6. Conclusions

LIME and Grad-CAM both contribute significantly to understanding how the VGG19 model makes decisions. The selection of either technique hinges on the analysis's particular requirements. If in-depth explanations at the feature level are necessary, LIME is the ideal choice. On the other hand, Grad-CAM offers a wider perspective on the areas that impact the model's predictions. Utilizing both techniques together can provide a well-rounded analysis of the model's actions, thereby improving the AI system's interpretability and reliability in identifying drilled holes in melamine-faced chipboard.

5. Conclusions

This study explored the application of Explainable AI (XAI) techniques, specifically LIME and Grad-CAM, to enhance the interpretability of Convolutional Neural Network (CNN) models used for the classification of drilled holes in melamine-faced chipboard. The primary focus was on understanding the decision-making process of these models to ensure reliability and improve classification accuracy.

The research involved evaluating three pre-trained CNN architectures: VGG16, VGG19, and ResNet101. The performance of these models was assessed based on accuracy and critical error rates using a 5-fold cross-validation approach. The VGG19 model emerged as the best performer, achieving the highest average accuracy of 67.03% and the lowest critical error rate. This model was therefore selected for further analysis using XAI techniques.

Both LIME and Grad-CAM were applied to the VGG19 model to generate explanations for its predictions. These techniques provided valuable insights into the features and regions of the images that influenced the model's classifications. LIME produced detailed, feature-level explanations, highlighting specific areas that contributed to the model's decisions. Grad-CAM, on the other hand, offered a more general overview, indicating broader regions that were important for the model's predictions.

The comparison between LIME and Grad-CAM revealed that both methods generally agreed on the influential regions within the images, though LIME provided more granular details. In cases of misclassification, both techniques identified regions that misled the model, with some discrepancies attributed to the different approaches of the two methods. LIME's perturbation-based approach and Grad-CAM's gradient-based technique highlighted complementary aspects of the model's decision-making process.

The study underscores the importance of using XAI techniques to interpret and validate the predictions of deep learning models in industrial applications. By understanding the factors influencing the model's decisions, it is possible to improve the reliability and robustness of automated systems for tool condition monitoring.

Future research will focus on refining the VGG19 model further to enhance its accuracy and reduce critical errors. Additionally, exploring other XAI techniques and combining them with LIME and Grad-CAM could provide a more comprehensive understanding of model behavior. Implementing these improvements will contribute to more effective and trustworthy AI systems in the wood industry and beyond.

In conclusion, the integration of XAI techniques with CNN models for the classification of drilled holes in melamine-faced chipboard has demonstrated significant potential. The insights gained from this study pave the way for more transparent and reliable AI applications in manufacturing and other fields where understanding model decisions is crucial.

Author Contributions: Conceptualization, J.K. and A.S.; methodology, J.K. and A.S.; software, J.K. and A.S.; validation, J.B. and A.J.; formal analysis, J.K.; investigation, J.K. and A.S.; resources, J.K.; data curation, A.J.; writing—original draft preparation, J.K.; writing—review and editing, J.K., A.S., A.J. and J.B.; visualization, J.B.; supervision, J.K.; project administration, J.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available upon request from the corresponding author. The data are not publicly available due to privacy and confidentiality.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Ozlusoylu, I.; Istek, A. Effects of Surface Lamination Process Parameters on Medium Density Fiberboard (MDF) Properties. *BioResources* **2023**, *18*, 767–777. [CrossRef]
2. Kun, W.; Shen, Q.; Wang, C.; Liu, C. Influence of pneumatic pressure on delamination factor of drilling medium density fiberboard. *Wood Res.* **2015**, *60*, 429–440.
3. Park, B.D.; Kang, E.C.; Lee, S.M.; Park, J.Y. Formaldehyde emission of wood-based composite panels with different surface lamination materials using desiccator method. *J. Korean Wood Sci. Technol.* **2016**, *44*, 600–606. [CrossRef]
4. Szwajka, K.; Trzepieciński, T. Effect of tool material on tool wear and delamination during machining of particleboard. *J. Wood Sci.* **2016**, *62*, 305–315. [CrossRef]
5. Szwajka, K.; Trzepieciński, T. An examination of the tool life and surface quality during drilling melamine-faced chipboard. *Wood Res.* **2017**, *62*, 307–318.
6. Śmietańska, K.; Podziewski, P.; Bator, M.; Górski, J. Automated monitoring of delamination factor during up (conventional) and down (climb) milling of melamine-faced MDF using image processing methods. *Eur. J. Wood Wood Prod.* **2020**, *78*, 613–615. [CrossRef]
7. Huang, X.; Tao, Z.; You, H.; Li, C.; Liu, J. Research on cutting method of wood panels based on improved particle swarm algorithm. *J. For. Eng.* **2024**, *9*, 125–131. [CrossRef]
8. Zaida, H.; Bouchelaghem, A.M.; Chehaidia, S.E. Experimental study of tool wear evolution during turning operation based on DWT and RMS. *Defect Diffus. Forum* **2021**, *406*, 392–405. [CrossRef]
9. Liu, Y.; Jiang, H.; Yao, R.; Zeng, T. Counterfactual-augmented few-shot contrastive learning for machinery intelligent fault diagnosis with limited samples. *Mech. Syst. Signal Process.* **2024**, *216*, 111507. [CrossRef]
10. Liu, Y.; Jiang, H.; Yao, R.; Zhu, H. Interpretable data-augmented adversarial variational autoencoder with sequential attention for imbalanced fault diagnosis. *J. Manuf. Syst.* **2023**, *71*, 342–359. [CrossRef]
11. Kim, H.S.; Jung, J.; Hwang, R.; Park, S.C.; Lee, S.J.; Kim, G.T.; Lee, B.W. Classification of PRPD Pattern in Cast-Resin Transformers Using CNN and Implementation of Explainable AI (XAI) with Grad-CAM. *IEEE Access* **2024**, *12*, 53623–53632. [CrossRef]
12. Apicella, A.; Di Lorenzo, L.; Isgrò, F.; Pollastro, A.; Prevete, R. Strategies to Exploit XAI to Improve Classification Systems. *Commun. Comput. Inf. Sci.* **2023**, *1901*, 147–159. [CrossRef]
13. Miller, M.; Ronczka, S.; Nischwitz, A.; Westermann, R. Light Direction Reconstruction Analysis and Improvement using XAI and CG. *J. WSCG* **2022**, *2022*, 189–198. [CrossRef]

14. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *Int. J. Comput. Vis.* **2020**, *128*, 336–359. [CrossRef]
15. Barredo Arrieta, A.; Díaz-Rodríguez, N.; Del Ser, J.; Bennetot, A.; Tabik, S.; Barbado, A.; Garcia, S.; Gil-Lopez, S.; Molina, D.; Benjamins, R.; et al. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Inf. Fusion* **2020**, *58*, 82–115. [CrossRef]
16. Lemaster, R.L.; Tee, L.B.; Dornfeld, D.A. Monitoring tool wear during wood machining with acoustic emission. *Wear* **1985**, *101*, 273–282. [CrossRef]
17. Dutta, S.; Pal, S.; Mukhopadhyay, S.; Sen, R. Application of digital image processing in tool condition monitoring: A review. *CIRP J. Manuf. Sci. Technol.* **2013**, *6*, 212–232. [CrossRef]
18. Zhu, K. Tool Condition Monitoring with Sparse Decomposition. In *Smart Machining Systems: Modelling, Monitoring and Informatics*; Springer Series in Advanced Manufacturing; Springer: Cham, Switzerland, 2022; pp. 235–266. [CrossRef]
19. Lemaster, R.L.; Lu, L.; Jackson, S. The use of process monitoring techniques on a CNC wood router. Part 1. sensor selection. *For. Prod. J.* **2000**, *50*, 31.
20. Lemaster, R.L.; Lu, L.; Jackson, S. The use of process monitoring techniques on a CNC wood router. Part 2. Use of a vibration accelerometer to monitor tool wear and workpiece quality. *For. Prod. J.* **2000**, *50*, 59.
21. Zhu, N.; Tanaka, C.; Ohtani, T.; Usuki, H. Automatic detection of a damaged cutting tool during machining I: Method to detect damaged bandsaw teeth during sawing. *J. Wood Sci.* **2000**, *46*, 437–443. [CrossRef]
22. Zhu, N.; Tanaka, C.; Ohtani, T.; Takimoto, Y. Automatic detection of a damaged router bit during cutting. *Holz Als Roh-Und Werkst.* **2004**, *62*, 126–130. [CrossRef]
23. Suetsugu, Y.; Ando, K.; Hattori, N.; Kitayama, S. A tool wear sensor for circular saws using wavelet transform signal processing. *For. Prod. J.* **2005**, *55*, 79.
24. Szwajka, K.; Górski, J. Evaluation tool condition of milling wood on the basis of vibration signal. *J. Physics: Conf. Ser.* **2006**, *28*, 1205. [CrossRef]
25. Wilkowski, J.; Górski, J. Vibro-acoustic signals as a source of information about tool wear during laminated chipboard milling. *Wood Res.* **2011**, *56*, 57–66.
26. Górski, J.; Szymanowski, K.; Podziewski, P.; Śmietańska, K.; Czarniak, P.; Cyrankowski, M. Use of cutting force and vibro-acoustic signals in tool wear monitoring based on multiple regression technique for compreg milling. *Bioresources* **2019**, *14*, 3379–3388. [CrossRef]
27. Kurek, J.; Antoniuk, I.; Świderski, B.; Jegorowa, A.; Bukowski, M. Application of siamese networks to the recognition of the drill wear state based on images of drilled holes. *Sensors* **2020**, *20*, 6978. [CrossRef] [PubMed]
28. Osowski, S.; Kurek, J.; Kruk, M.; Górski, J.; Hoser, P.; Wieczorek, G.; Jegorowa, A.; Wilkowski, J.; Śmietańska, K.; Kossakowska, J. Developing automatic recognition system of drill wear in standard laminated chipboard drilling process. *Bull. Pol. Acad. Sci. Tech. Sci.* **2016**, *64*, 633–640.
29. Xie, Z.; Lu, Y.; Chen, X. A multi-sensor integrated smart tool holder for cutting process monitoring. *Int. J. Adv. Manuf. Technol.* **2020**, *110*, 853–864. [CrossRef]
30. Kuo, R. Multi-sensor integration for on-line tool wear estimation through artificial neural networks and fuzzy neural network. *Eng. Appl. Artif. Intell.* **2000**, *13*, 249–261. [CrossRef]
31. Bhuiyan, M.; Choudhury, I.; Dahari, M.; Nukman, Y. An Investigation into Turning of ASSAB-705 Steel Using Multiple Sensors. *Mater. Manuf. Process.* **2016**, *31*, 896–904. [CrossRef]
32. Jemielniak, K.; Urbański, T.; Kossakowska, J.; Bombiński, S. Tool condition monitoring based on numerous signal features. *Int. J. Adv. Manuf. Technol.* **2012**, *59*, 73–81. [CrossRef]
33. Panda, S.; Singh, A.K.; Chakraborty, D.; Pal, S. Drill wear monitoring using back propagation neural network. *J. Mater. Process. Technol.* **2006**, *172*, 283–290. [CrossRef]
34. Nasir, V.; Cool, J.; Sassani, F. Intelligent machining monitoring using sound signal processed with the wavelet method and a self-organizing neural network. *IEEE Robot. Autom. Lett.* **2019**, *4*, 3449–3456. [CrossRef]
35. Nasir, V.; Sassani, F. A review on deep learning in machining and tool monitoring: Methods, opportunities, and challenges. *Int. J. Adv. Manuf. Technol.* **2021**, *115*, 2683–2709. [CrossRef]
36. Nautiyal, A.; Mishra, A. Drill Bit Selection and Drilling Parameter Optimization using Machine Learning. *IOP Conf. Ser. Earth Environ. Sci.* **2023**, *1261*, 012027. [CrossRef]
37. Feng, Z.; Gani, H.; Damayanti, A.D.; Gani, H. An explainable ensemble machine learning model to elucidate the influential drilling parameters based on rate of penetration prediction. *Geoenergy Sci. Eng.* **2023**, *231*, 212231. [CrossRef]
38. Mendez, M.; Ahmed, R.; Karami, H.; Nasser, M.; Hussein, I.A.; Garcia, S.; Gonzalez, A. Applications of Machine Learning Methods to Predict Hole Cleaning in Horizontal and Highly Deviated Wells. *SPE Drill. Complet.* **2023**, *38*, 606–617. [CrossRef]
39. Kurek, J.; Swiderski, B.; Jegorowa, A.; Kruk, M.; Osowski, S. Deep learning in assessment of drill condition on the basis of images of drilled holes. In Proceedings of the Eighth International Conference on Graphic and Image Processing (ICGIP 2016) SPIE, Tokyo, Japan, 29–31 October 2016; Volume 10225, pp. 375–381.
40. Kurek, J.; Wieczorek, G.; Kruk, B.S.M.; Jegorowa, A.; Osowski, S. Transfer learning in recognition of drill wear using convolutional neural network. In Proceedings of the 2017 18th International Conference on Computational Problems of Electrical Engineering (CPEE), Kutna Hora, Czech Republic, 11–13 September 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 1–4.

41. Zhu, J.; Su, Z.; Han, Z.; Lan, Z.; Wang, Q.; Ho, M.M.P. An ensemble approach for enhancing generalization and extendibility of deep learning facilitated by transfer learning: Principle and application in curing monitoring. *Smart Mater. Struct.* **2023**, *32*, 115022. [CrossRef]
42. Bhuiyan, M.R.; Uddin, J. Deep Transfer Learning Models for Industrial Fault Diagnosis Using Vibration and Acoustic Sensors Data: A Review. *Vibration* **2023**, *6*, 218–238. [CrossRef]
43. Chai, Z.; Wang, J.; Zhao, C.; Ding, J.; Sun, Y. Deep transfer learning methods for typical supervised tasks in industrial monitoring: State-of-the-art, challenges, and perspectives. *Sci. Sin. Informationis* **2023**, *53*, 821–840. [CrossRef]
44. Kurek, J.; Antoniuk, I.; Górski, J.; Jegorowa, A.; Świderski, B.; Kruk, M.; Wieczorek, G.; Pach, J.; Orłowski, A.; Aleksiejuk-Gawron, J. Classifiers ensemble of transfer learning for improved drill wear classification using convolutional neural network. *Mach. Graph. Vis.* **2019**, *28*, 13–23. [CrossRef]
45. Kurek, J.; Antoniuk, I.; Górski, J.; Jegorowa, A.; Świderski, B.; Kruk, M.; Wieczorek, G.; Pach, J.; Orłowski, A.; Aleksiejuk-Gawron, J. Data augmentation techniques for transfer learning improvement in drill wear classification using convolutional neural network. *Mach. Graph. Vis.* **2019**, *28*, 3–12. [CrossRef]
46. Arrighi, L.; Barbon Junior, S.; Pellegrino, F.A.; Simonato, M.; Zullich, M. Explainable Automated Anomaly Recognition in Failure Analysis: Is Deep Learning Doing it Correctly? In *World Conference on Explainable Artificial Intelligence*; Communications in Computer and Information Science; Springer: Cham, Switzerland, 2023; Volume 1902, pp. 420–432. [CrossRef]
47. Famiglini, L.; Campagner, A.; Barandas, M.; La Maida, G.A.; Gallazzi, E.; Cabitza, F. Evidence-based XAI: An empirical approach to design more effective and explainable decision support systems. *Comput. Biol. Med.* **2024**, *170*, 108042. [CrossRef] [PubMed]
48. Gkartzonika, I.; Gkalelis, N.; Mezaris, V. Learning Visual Explanations for DCNN-Based Image Classifiers Using an Attention Mechanism. In *European Conference on Computer Vision*; Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); Springer: Cham, Switzerland, 2023; Volume 13808, pp. 396–411. [CrossRef]
49. Makridis, G.; Theodoropoulos, S.; Dardanis, D.; Makridis, I.; Separdani, M.M.; Fatouros, G.; Kyriazis, D.; Koulouris, P. XAI enhancing cyber defence against adversarial attacks in industrial applications. In Proceedings of the 2022 IEEE 5th International Conference on Image Processing Applications and Systems (IPAS), Genova, Italy, 5–7 December 2022. [CrossRef]
50. Li, S.; Li, T.; Sun, C.; Yan, R.; Chen, X. Multilayer Grad-CAM: An effective tool towards explainable deep neural networks for intelligent fault diagnosis. *J. Manuf. Syst.* **2023**, *69*, 20–30. [CrossRef]
51. Shimizu, T.; Nagata, F.; Arima, K.; Miki, K.; Kato, H.; Otsuka, A.; Watanabe, K.; Habib, M.K. Enhancing defective region visualization in industrial products using Grad-CAM and random masking data augmentation. *Artif. Life Robot.* **2024**, *29*, 62–69. [CrossRef]
52. Arima, K.; Nagata, F.; Shimizu, T.; Otsuka, A.; Kato, H.; Watanabe, K.; Habib, M.K. Improvements of detection accuracy and its confidence of defective areas by YOLOv2 using a data set augmentation method. *Artif. Life Robot.* **2023**, *28*, 625–631. [CrossRef]
53. Zafar, M.R.; Khan, N. Deterministic Local Interpretable Model-Agnostic Explanations for Stable Explainability. *Mach. Learn. Knowl. Extr.* **2021**, *3*, 525–541. [CrossRef]
54. Harishyam, B.; Jenarthanan, M.; Rishivanth, R.; Rajesh, R.; Sai Girish, N. Visual inspection of mechanical components using visual imaging and machine learning. *Mater. Today Proc.* **2023**, *72*, 2557–2563. [CrossRef]
55. *EN 310:1994*; Wood-Based Panels: Determination of Modulus of Elasticity in Bending and of Bending Strength. European Standard: Brussels, Belgium, 1994.
56. *EN 1534*; Wood Flooring and Parquet: Determination of Resistance to Indentation—Test Method. European Standard: Brussels, Belgium, 2002.
57. Czarniak, P.; Szymanowski, K.; Panjan, P.; Górski, J. Initial Study of the Effect of Some PVD Coatings ("TiN/AlTiN" and "TiAlN/a-C:N") on the Wear Resistance of Wood Drilling Tools. *Forests* **2022**, *13*, 286. [CrossRef]
58. Tuunainen, T.; Isohanni, O.; Jose, M.R. A comparative study on the application of Convolutional Neural Networks for wooden panel defect detection. In Proceedings of the 2024 IEEE 22nd World Symposium on Applied Machine Intelligence and Informatics (SAMI), Stará Lesná, Slovakia, 25–27 January 2024; pp. 321–326. [CrossRef]
59. Kurek, J.; Szymanowski, K.; Chmielewski, L.J.; Orłowski, A. Advancing Chipboard Milling Process Monitoring through Spectrogram-Based Time Series Analysis with Convolutional Neural Network using Pretrained Networks. *Mach. Graph. Vis.* **2023**, *32*, 89–108. [CrossRef]
60. Karathanasopoulos, N.; Hadjidoukas, P. Deep learning based automated fracture identification in material characterization experiments. *Adv. Eng. Inform.* **2024**, *60*, 102402. [CrossRef]
61. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition.*arXiv* **2015**, arXiv:1409.1556.
62. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. *arXiv* **2015**, arXiv:1512.03385.
63. ImageNet. Available online: http://www.image-net.org (accessed on 10 August 2024)
64. Iandola, F.N.; Han, S.; Moskewicz, M.W.; Ashraf, K.; Dally, W.J.; Keutzer, K. SqueezeNet: AlexNet-level accuracy with 50× fewer parameters and <0.5 MB model size. *arXiv* **2016**, arXiv:1602.07360.
65. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.

66. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826.
67. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the CVPR, Honolulu, HI, USA, 21–26 July 2017; Volume 1, p. 3.
68. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 4510–4520.
69. Chollet, F. Xception: Deep Learning with Depthwise Separable Convolutions. *arXiv* **2017**, arXiv:1610.02357.
70. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A.A. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. In Proceedings of the AAAI, San Francisco, CA, USA, 4–9 February 2017; Volume 4, p. 12.
71. Zhang, X.; Zhou, X.; Lin, M.; Sun, J. ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices. *arXiv* **2017**, arXiv:1707.01083v2.
72. Zoph, B.; Vasudevan, V.; Shlens, J.; Le, Q.V. Learning Transferable Architectures for Scalable Image Recognition. *arXiv* **2017**, *2*, arXiv:1707.07012.
73. Redmon, J. Darknet: Open Source Neural Networks in C. Available online: https://pjreddie.com/darknet (accessed on 10 August 2024).
74. Tan, M.; Le, Q.V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *arXiv* **2019**, arXiv:1905.1194.
75. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
76. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis. (IJCV)* **2015**, *115*, 211–252. [CrossRef]
77. Zhou, B.; Khosla, A.; Lapedriza, A.; Torralba, A.; Oliva, A. Places: An image database for deep scene understanding. *arXiv* **2016**, arXiv:1610.02055.
78. Tan, L.; Huang, C.; Yao, X. A Concept-Based Local Interpretable Model-Agnostic Explanation Approach for Deep Neural Networks in Image Classification. In *International Conference on Intelligent Information Processing*; IFIP Advances in Information and Communication Technology; Springer: Cham, Switzerland, 2024; Volume 704, pp. 119–133. [CrossRef]
79. Meng, H.; Wagner, C.; Triguero, I. SEGAL time series classification—Stable explanations using a generative model and an adaptive weighting method for LIME. *Neural Netw.* **2024**, *176*, 106345. [CrossRef] [PubMed]
80. Visani, G.; Bagli, E.; Chesani, F. OptiLIME: Optimized lime explanations for diagnostic computer algorithms. *arXiv* **2020**, *2699*, arXiv:2006.05714.
81. Messalas, A.; Aridas, C.; Kanellopoulos, Y. Evaluating MASHAP as a faster alternative to LIME for model-agnostic machine learning interpretability. In Proceedings of the 2020 IEEE International Conference on Big Data (Big Data), Atlanta, GA, USA, 10–13 December 2020; pp. 5777–5779. [CrossRef]
82. Dhurandhar, A.; Ramamurthy, K.N.; Ahuja, K.; Arya, V. Locally Invariant Explanations: Towards Stable and Unidirectional Explanations through Local Invariant Learning. In Proceedings of the Advances in Neural Information Processing Systems 36 (NeurIPS 2023), New Orleans, LA, USA, 10–16 December 2023; Volume 36.
83. Barr Kumarakulasinghe, N.; Blomberg, T.; Liu, J.; Saraiva Leao, A.; Papapetrou, P. Evaluating local interpretable model-agnostic explanations on clinical machine learning classification models. In Proceedings of the 2020 IEEE 33rd International Symposium on Computer-Based Medical Systems (CBMS), Rochester, MN, USA, 28–30 July 2020; pp. 7–12. [CrossRef]
84. Chakraborty, T.; Trehan, U.; Mallat, K.; Dugelay, J.L. Generalizing Adversarial Explanations with Grad-CAM. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, New Orleans, LO, USA, 18–24 June 2022; pp. 186–192. [CrossRef]
85. Liu, Y.; Tang, L.; Liao, C.; Zhang, C.; Guo, Y.; Xia, Y.; Zhang, Y.; Yao, S. Optimized Dropkey-Based Grad-CAM: Toward Accurate Image Feature Localization. *Sensors* **2023**, *23*, 8351. [CrossRef] [PubMed]
86. Desai, S.; Ramaswamy, H.G. Ablation-CAM: Visual explanations for deep convolutional network via gradient-free localization. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Snowmass Village, CO, USA, 1–5 March 2020; pp. 972–980. [CrossRef]
87. Li, Z.; Xu, M.; Yang, X.; Han, Y.; Wang, J. A Multi-Label Detection Deep Learning Model with Attention-Guided Image Enhancement for Retinal Images. *Micromachines* **2023**, *14*, 705. [CrossRef] [PubMed]
88. Chen, L.; Chen, J.; Hajimirsadeghi, H.; Mori, G. Adapting grad-CAM for embedding networks. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Virtual, 5–9 January 2021; pp. 2783–2792. [CrossRef]
89. Song, W.; Dai, S.; Wang, J.; Huang, D.; Liotta, A.; Di Fatta, G. Bi-gradient verification for grad-CAM towards accurate visual explanation for remote sensing images. In Proceedings of the 2019 International Conference on Data Mining Workshops (ICDMW), Beijing, China, 8–11 November 2019; pp. 473–479. [CrossRef]
90. Brahmaiah, O.V.; Raju, M.S.N.; Jahnavi, V.; Varshini, M. Dense Net-Based Acute Lymphoblastic Leukemia Classification and Interpretation through Gradient-Weighted Class Activation Mapping. In Proceedings of the 2024 Third International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS), Krishnankoil, India, 14–16 March 2024. [CrossRef]

91. Xiao, M.; Zhang, L.; Shi, W.; Liu, J.; He, W.; Jiang, Z. A visualization method based on the Grad-CAM for medical image segmentation model. In Proceedings of the 2021 International Conference on Electronic Information Engineering and Computer Science (EIECS), Changchun, China, 23–26 September 2021; pp. 242–247. [CrossRef]
92. Qiu, Z.; Rivaz, H.; Xiao, Y. Is Visual Explanation with Grad-CAM More Reliable for Deeper Neural Networks? A Case Study with Automatic Pneumothorax Diagnosis. In *International Workshop on Machine Learning in Medical Imaging*; Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); Springer: Cham, Switzerland, 2024; Volume 14349, pp. 224–233. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

applied sciences

MDPI

Article

Deep Learning-Based Layout Analysis Method for Complex Layout Image Elements

Yunfei Zhong *, Yumei Pu, Xiaoxuan Li, Wenxuan Zhou, Hongjian He, Yuyang Chen, Lang Zhong and Danfei Liu

School of Packaging Engineering, Hunan University of Technology, Zhuzhou 412007, China; 17830278323@163.com (Y.P.); m22077300009@stu.hut.edu.cn (X.L.); m23085600004@stu.hut.edu.cn (W.Z.); m24077300002@stu.hut.edu.cn (H.H.); m230805z1005@stu.hut.edu.cn (Y.C.); m24080500004@stu.hut.edu.cn (L.Z.); liudanfei@hut.edu.cn (D.L.)

* Correspondence: yfzhong@hut.edu.cn; Tel.: +86-0731-2218-2088

Featured Application: A specific application of this work is an automated design element segmentation tool for converting complex graphic designs, such as movie posters, into editable layered formats. By leveraging the improved DeepLabv3+ model, the tool can accurately identify and segment text, images, logos, and other layout elements within a raster image (e.g., JPEG/PNG). This enables designers to automatically generate structured, layered files (e.g., PSD) for efficient editing, adaptation to different formats (e.g., social media, print), or trend analysis. The model's reduced computational demands and enhanced accuracy make it suitable for integration into design software, streamlining workflows, and reducing manual effort in reverse-engineering layouts.

Abstract: The layout analysis of elements is indispensable in graphic design, as effective layout design not only facilitates the delivery of visual information but also enhances the overall esthetic appeal to the audience. The combination of deep learning and graphic design has gradually turned into a popular research direction in graphic design in recent years. However, in the era of rapid development of artificial intelligence, the analysis of layout still requires manual participation. To address this problem, this paper proposes a method for analyzing the layout of complex layout image elements based on the improved DeepLabv3++ model. The method reduces the number of model parameters and training time by replacing the backbone network. To improve the effect of multi-scale semantic feature extraction, the null rate of ASPP is fine-tuned, and the model is trained by self-constructed movie poster dataset. The experimental results show that the improved DeepLabv3+ model achieves a better segmentation effect on the self-constructed poster dataset, with MIoU reaching 75.60%. Compared with the classical models such as FCN, PSPNet, and DeepLabv3, the improved model in this paper effectively reduces the number of model parameters and training time while also ensuring the accuracy of the model.

Keywords: DeepLabv3+; image segmentation; poster layout analysis; deep learning

1. Introduction

At this stage, artificial intelligence has developed into the intersection of many disciplines such as cognitive science, psychology, art, and computer science [1,2]. With the development of deep learning and neural networks, more and more researchers combine them with design, such as intelligent image color matching and intelligent image quality evaluation [3,4]. And the complex layout images obtained from text, graphics, images, and

other layout elements through the process of artistic design, pre-press processing, typesetting and so on are widely used in packaging, posters, book binding, and so on [5,6]. At present, the study of complex layout images is only for simple document images, and there are few reports on the design of complex layout images. Therefore, there is an urgent need for more scientific, intelligent, and highly accurate methods and techniques for analyzing the layout of complex layout images.

Compared with such complex images as posters, document images, newspapers, and other complex images have a single background and neatly arranged text, graphics, tables and other foregrounds. Currently, most domestic and international research on complex layout image analysis focuses on document images, newspapers, and similar structured materials, while relatively few studies have addressed layout analysis in traditional poster designs, brochures, and other visual media. Traditional layout analysis can be roughly categorized into three approaches: top-down approach [7]; bottom-up approach [8]; hybrid approach [9]. Amidst the rise of neural networks, a number of researchers have combined layout analysis with computers, deep learning, neural networks, and other techniques to explore complex layout image analysis [10]. Wu et al. [11] proposed a method for document image layout analysis with explicit edge embedding networks, which exploits the model to overcome the data scarcity by using an integrated document approach compared to the traditional methods. Guo et al. [12] proposed a design space to describe the design elements in advertising posters and introduced a design sequence to rationalize the design decisions of human designers in creating posters. There is still much room for research in the field of combining visual communication design images with deep learning techniques.

The central question of this research is how to perform automatic layout design through artificial intelligence. The first step towards enabling automatic layout design with AI is to parse and recognize the constituent components of the layout. As suggested in previous work [13], this study adopts deep learning techniques to accurately locate and classify layout elements in complex visual designs such as movie posters. The presence of different foregrounds and complex graphical backgrounds in natural images like movie posters and the text in the images with various, colors, sizes, orientations, and textures make it even more challenging to perform an image layout analysis for complex layout images like movie posters.

Aiming at the problem of difficulty in analyzing the layout of visual communication images, this paper takes movie posters as the research object and carries out the construction of movie poster dataset. Based on the improved DeepLabv3+ network model, the segmentation and recognition of text, theme and graphic regions of movie posters were performed. Finally, according to the layout segmentation results, the positional relationship of the theme area, text area and graphic area is analyzed, and the layout analysis of the movie poster is performed using the GoogLeNet image classification network model.

In summary, this study proposes a two-stage approach for analyzing complex movie poster layouts. An improved DeepLabv3+ model with a lightweight backbone and optimized ASPP is used for accurate element segmentation, and a GoogLeNet-based classifier is employed to determine layout types. A high-resolution annotated dataset of 2300 posters is also constructed to support the experiments.

The rest of this paper is organized as follows: Section 2 introduces the layout types of movie posters. Section 3 presents the proposed segmentation method. Section 4 describes dataset construction and training details. Section 5 provides experimental results and analysis. Section 6 concludes the paper.

2. Film Poster Composition Layout Method

At present, there is no uniform view in the academic world on the compositional layout of film posters, so there are various methods of classification. This study combines the existing rules of poster layout design composition and typography, and summarizes film poster images into the following eight commonly used types: centered layout, split layout, symmetrical layout, diagonal layout, wraparound layout, full-screen layout, axial layout, inclined layout, and other compositional layouts.

Centered layout will be the main elements in a centrally aligned manner, can quickly attract the eye to occupy the visual focus, as shown in Figure 1a. The split layout arranges the poster in an asymmetrical structure with the figure above the text below and the figure below the text above, ensuring a balanced and stable picture and forming a sharp contrast, as shown in Figure 1b. The symmetrical layout arranges the main elements in a symmetrical manner relative to the central axis, visually giving a feeling of rigor and rationality, as shown in Figure 1c. The diagonal layout distributes the main elements of the poster in opposite corners, adding an unstable variation that can bring a visual impact, as shown in Figure 1d. The surrounding layout is generally a graphic surrounded by text, its composition form is fuller, more information, as shown in Figure 1e. The full-screen layout is mainly graphics to support the entire layout, supplemented by text, visually give people an intuitive and strong feelings, as shown in Figure 1f. The axis layout is dominated by a hidden axis, arranging the main elements in a hidden axis to break the limitations of a centered layout and make the layout appear less dull, as shown in Figure 1g. The tilt layout mainly arranges the main elements of the poster in a tilted manner, giving a sense of visual motion and instability, as shown in Figure 1h. In practical examples, it is difficult for some posters to generalize and define the characteristics of a layout, and it may be difficult to fit into a certain layout classification. In order to simplify the analysis, this study categorizes posters that are disorganized, free-layout posters, or posters that are laid out in other layout styles within the Other Layout category. The first eight ways encompass most of the current ways of laying out a film poster.

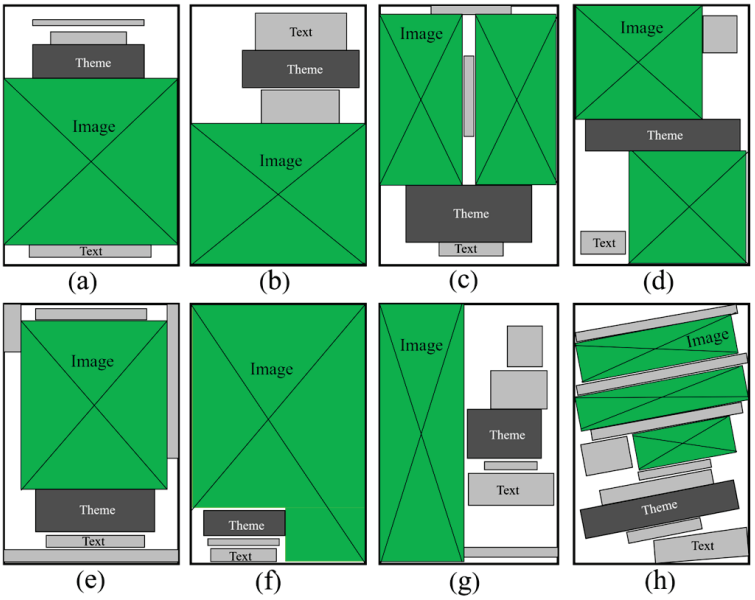


Figure 1. Schematic diagram of the poster layout. (**a**) Centered layout; (**b**) split layout; (**c**) symmetrical layout; (**d**) diagonal layout; (**e**) wraparound layout; (**f**) full-screen layout; (**g**) axial layout; (**h**) inclined layout.

3. Film Poster Layout Segmentation Method Based on Improved DeepLabv3+

3.1. Relevant Model Theory

3.1.1. DeepLabv3+ Base Model

DeepLabv3+ (Rethinking Atrous Convolution for Semantic Image Segmentation) is a semantic segmentation based on Atrous Spatial Pyramid Pooling (ASPP) module proposed by Google team in 2019 network [14], and the model is widely used in various image semantic segmentation tasks. The model introduces a decoder structure based on DeepLabv3 [15], which further fuses the underlying features with the higher-level features to improve the segmentation boundary accuracy. The overall architecture of the DeepLabv3+ model is shown in Figure 2, and the main body of its encoder is the DCNN with cavity convolution, which uses Xception [16]. The model introduces the cavity convolution, which increases the sensory field without loss of information, so that each convolution output contains a larger range of information.

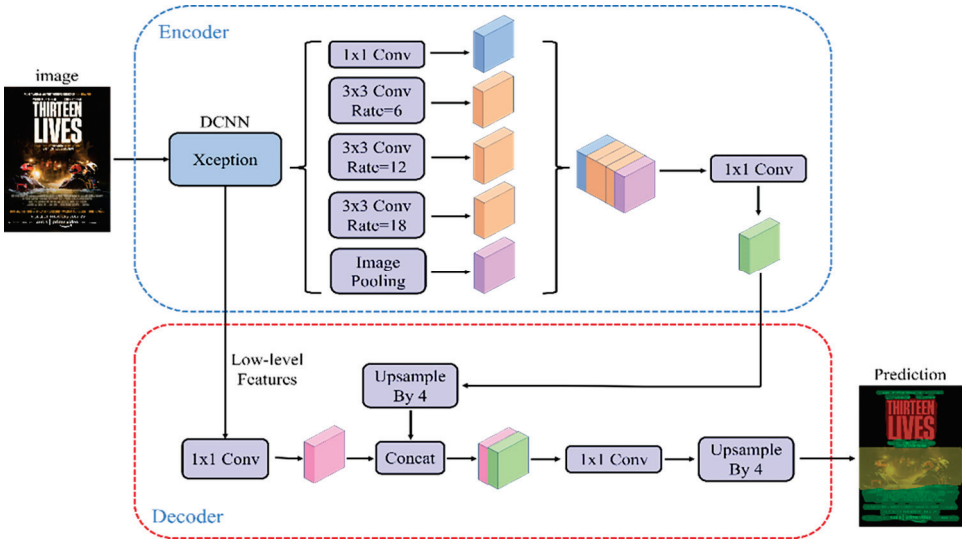


Figure 2. DeepLabv3+ basic network architecture (adapted from [14,15]).

The algorithm effectively expands the network sensory field and captures a wide range of contextual information to improve the segmentation accuracy of the image through the cavity convolution in the ASPP module. The equivalent convolution kernel K of the cavity convolution is formulated as follows:

$$K = k + (k-1)(r-1) \tag{1}$$

where k is the original convolutional kernel size, and r is the null rate.

As the backbone feature extraction network extracts both low-order and high-order features, the high-order features capture information at multiple scales. The parallel features are then fused to form a semantic-rich feature map, and the number of channels is adjusted using a 1×1 convolution. This is followed by fusion with low-order features through the upsampling operation in the decoder, and finally, a 3×3 convolution and another upsampling are applied to restore the original image size and produce the final segmentation result.

3.1.2. Mobilenetv3 Network

Mobilenetv3, proposed by Howard et al. [17], is a lightweight convolutional neural network, which has been accumulated from two generations of networks, Mobilenetv1 and Mobilenetv2, retaining the deep separable convolution of Mobilenetv1 and the inverted residual linear bottleneck block of Mobilenetv2, while improving the model structure to reduce the amount of parameters and training time. Mobilenetv3 builds on v2 by adding the Squeeze and Excitation (SE) block attention mechanism to the Bottleneck block, replacing the ReLU6 function with hard-swish and the sigmoid function with hard sigmoid, and redesigning the structure of the time-consuming layer of the network. The overall structure of the time-consuming layer of the network is shown in Figure 3, in which the structure of the small and large versions is basically the same, and the only difference lies in the number of basic units (bneck) and the internal parameters, which are mainly reflected in the difference in the number of channels.

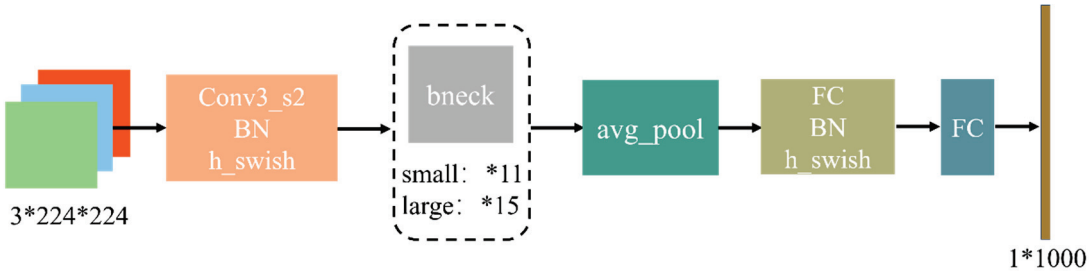


Figure 3. Mobilenetv3 network structure diagram (adapted from [17]).

3.1.3. GoogLeNet Network

In order to avoid the problems of slow network training convergence, long training time, and susceptibility to gradient vanishing and gradient explosion, the Google team proposed the GoogLeNet network in 2014, which won the first place in the classification task in the ImageNet competition that year [18]. This network introduces the Inception structure in order to fuse feature information at different scales. It uses a 1×1 convolutional kernel for both dimensionality reduction and feature mapping. A global average pooling strategy is adopted instead of a fully connected layer to reduce the number of parameters. In addition, two auxiliary classifiers are added to assist training, which not only achieve model fusion and enhance gradient flow, but also add a back-propagated gradient signal and provide additional regularization to support network training. The model consists of 22 modular structures including a convolutional layer, 9 modular structures, a global average pooling layer, and an output layer. The network is mainly used in the fields of image classification and object recognition.

3.2. Improved DeepLabv3+ Network Models

In order to effectively extract the semantic information of movie posters at different scales, the DeepLabv3+ structure is used in this paper. The algorithmic model proposed in this paper is obtained by improving the DeepLabv3+ algorithmic model, which contains encoder and decoder modules. The encoder module contains the main body of the backbone network (DCNN) with null convolution as well as the ASPP (Null Spatial Convolutional Pooling Pyramid, Atrous Spatial Pyramid Pooling) module, in which the main body of DCNN adopts the improved Xception network with cavity convolution, which mainly adopts deep separable convolution, making its computational volume smaller. On this basis, the shallow features generated by the DCNN are utilized and fed into the decoder–encoder module, and the high semantic high-level features are fed into the decoder for up-sampling,

followed by the fusion of the features using the results obtained by 1 × 1 convolution, the extraction of the features using 3 × 3 convolution, and the bilinear interpolation of the image to obtain segmentation predictions consistent with the image size.

Due to the more complex structure of the backbone network Xception in the original DeepLabv3+ model, this is more difficult for extracting smaller categories and textures in complex layout images, and when the network is extracting details and texture information, problems such as blurring and confusion may occur, resulting in the inability to continuously extract a particular category. In addition, the network has more parameters, is more computationally intensive, and takes longer to reason. In this paper, the lightweight Mobilenetv3-small network structure will be used as the feature extraction network, which has a reduced number of parameters relative to the Xception network, as well as a reduced number of computations and time required, thus increasing the speed of training and learning.

The combination of null rates for the ASPP module null convolution in the original DeepLabv3+ is 6, 12, and 18, and as the backbone network carries out feature extraction, the resolution of the feature maps decreases gradually, and the combination of 6, 12, and 18 cannot extract the features of the multi-resolution feature maps more efficiently because it does not set a smaller null rate, resulting in a lack of the ability to segment the small targets. In order to effectively extract the semantic information of complex layout images at different scales, the DeepLabv3+ structure is improved in this paper. In the improved DeepLabv3+ model, the original backbone network Xception is replaced by Mobilenetv3-small network, and at the same time two null convolution layers are added in the ASPP module. The null convolution null rate in the module is changed from a combination of 6, 12, and 18 to a combination of 2, 4, 8, 12, and 16, in order to be able to enhance the ability of the network model for the segmentation of different size categories in complex layout images. It is processed by two modules, the backbone network and ASPP, and transmitted to the decoder module, whose main structure remains unchanged. After the complex layout image processed by decoder module will output the predicted layout element segmentation, then the prediction result will be passed to the GoogLeNet classification model to classify the layout, and finally obtain the result prediction of the layout mode of the movie poster. The structure of the improved DeepLabv3+ model is shown in Figure 4.

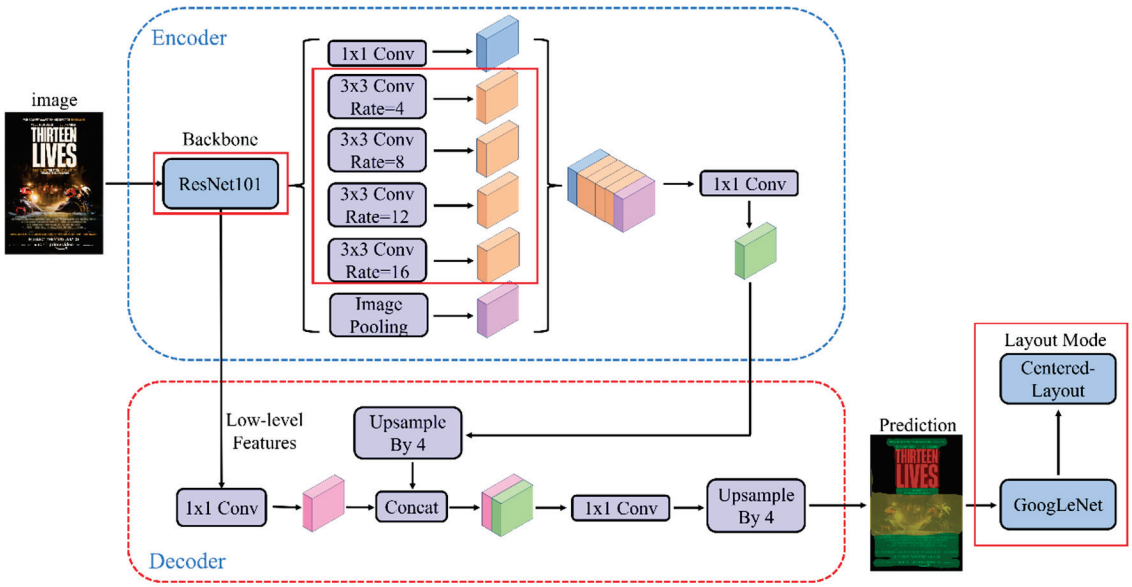


Figure 4. Improved DeepLabv3+ model architecture.

4. Dataset and Model Training

4.1. Dataset Production

In this study, the movie poster images were obtained from the IMP Awards (http://www.impawards.com/, accessed on 15 April 2025) and the Douban website (https://movie.douban.com/, accessed on 15 April 2025). The dataset contains 2300 high-resolution movie poster images, primarily in JPEG (.jpg) and PNG (.png) formats, with a minimum resolution of 505 × 755 pixels and a maximum resolution of 6889 × 9778 pixels. After completing the image sampling use the open source annotation software labelme (https://github.com/wkentaro/labelme, accessed on 15 April 2025) for manual annotation, where red is the subject class, green is the text class, and yellow is the graphic class. Figure 5 illustrates the annotation workflow using two example posters obtained from the public domain for demonstration purposes only. Figure 5b shows a sample annotation map generated with Labelme. For each annotated poster, a corresponding JSON file is automatically created, storing the polygon coordinates of the labeled regions along with essential image metadata. In total, 2300 JSON files were generated based on the annotated dataset used in this study. These files were then converted into a semantic segmentation dataset following the Pascal VOC 2012 format [19], using the official Labelme conversion script. The resulting segmentation mask, containing the three layout classes (theme, text, and graphic), is shown in Figure 5c.

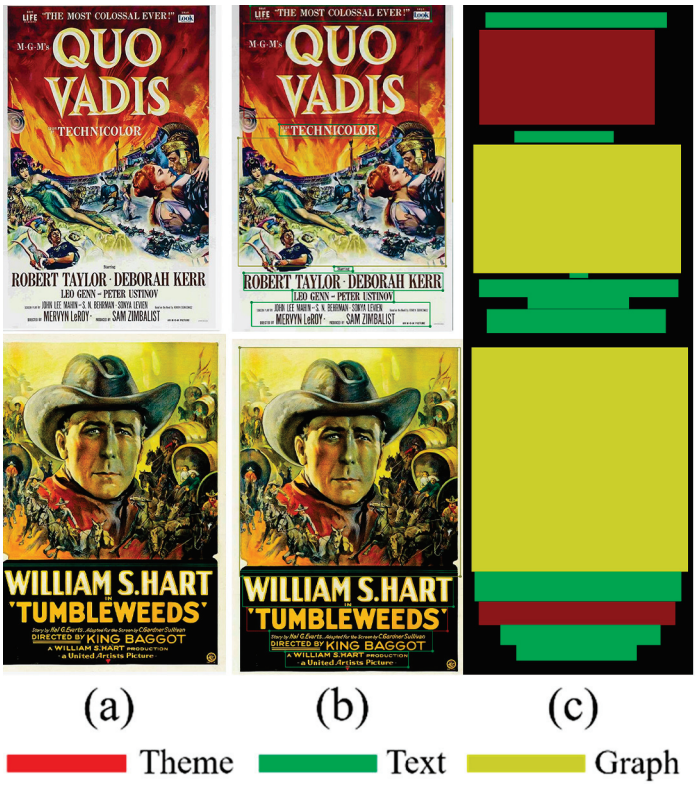


Figure 5. Demonstration of the annotation process. (**a**) Sample poster(Quo Vadis, 1951; Tumbleweeds, 1925); (**b**) split label diagram; (**c**) label split effect diagram. The images are public domain samples used only for illustration and not for training. (https://picryl.com/, accessed on 15 April 2025).

The poster dataset is divided into training, validation, and test sets in the ratio of 7:2:1, with 1610 poster images in the training set, 460 poster images in the validation set, and 230 poster images in the test set. The training set is used to train the model parameters, the

validation set is used for model hyperparameter tuning, and the test set is used to evaluate the model generalization ability.

The manually segmented 2300 labeled diagrams were divided into nine categories according to nine poster layout styles, where the distribution of category samples is shown in Figure 6 and their categories are schematically illustrated in Figure 1. Each category was divided into a training set and a validation set in the ratio of 9:1, where the training set had a total of 2080 poster images and the validation set 220 poster images. Table 1 shows the division of the poster segmentation dataset and the layout classification dataset training set, validation set, and test set.

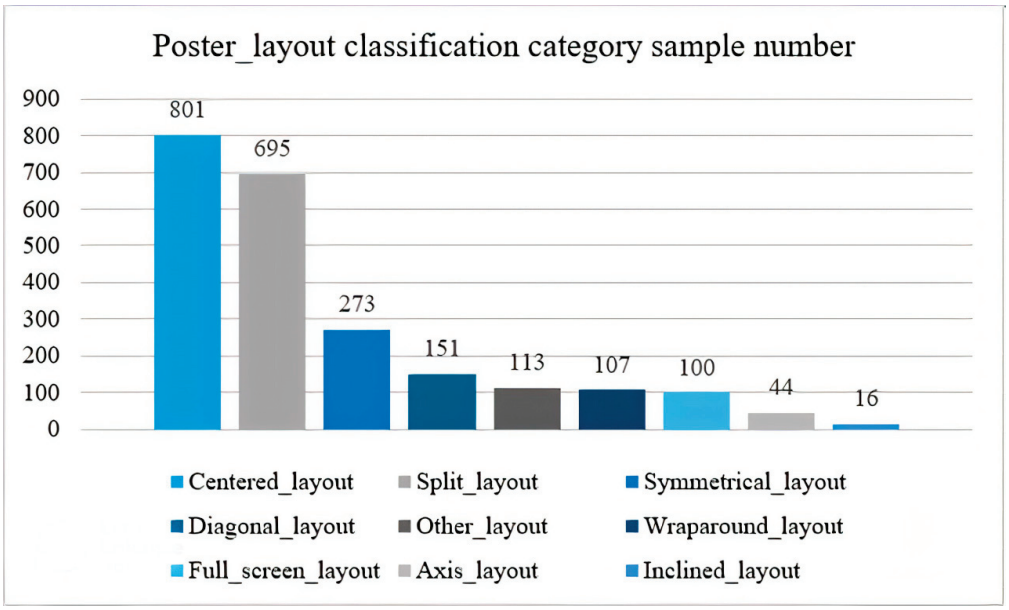


Figure 6. Poster layout categorization dataset category sample distribution.

Table 1. Poster segmentation dataset and classification dataset division.

Dataset	Train	Val	Test
Poster Segmentation	1610	460	230
Poster_layout classification	2080	220	—

4.2. Model Training

In this paper, the model is built using the Pytorch deep learning framework (version 2.2.2 + CUDA 12.1), and trained on a computer with a system environment of Win10, an Intel(R) Core (TM) i7-10700F CPU @ 2.90GHz processor (Intel, Santa Clara, CA, USA) and an NVIDIA Geforce RTX 3060 Ti 12G graphics card (NVIDIA, Santa Clara, CA, USA).

The dataset was imported into the improved DeepLabv3+ model and the model was trained. The categories for semantic segmentation are background, subject, text, and graphics, and the output segmented images are of the same size as the input. Using the migration learning approach, the model parameters that have been trained by the large-scale Pascal VOC 2012 dataset are migrated to the model in this paper, and then the model training is performed to train 200 Epochs, respectively, with the batch size set to 4. To minimize the loss function during training, the Stochastic Gradient Descent (SGD)

algorithm is employed, where model parameters are iteratively updated based on the gradient of the loss. This parameter update rule is defined as follows:

$$\theta_{t+1} = \theta_t - \alpha \nabla L(\theta_t) \tag{2}$$

where θ_t denotes the parameter values of the model at the tth iteration, α denotes the learning rate, and $\nabla L(\theta_t)$ denotes the gradient of the loss function $L(\theta)$ with respect to the model parameter θ. The momentum value was set to 0.9, initial learning rate was set to 0.007, and weight decay index was 0.0005. The poster layout classification dataset was then put into the GoogLeNet classification network model for training, with its training batch size set to 2, initial learning rate set to 0.001, and 200 epochs trained.

5. Experimental Results and Analysis

5.1. Loss Functions and Evaluation Indicators

The study presented in this paper belongs to the multi-category image segmentation problem, which are subject, text, and graphics. Therefore, the categorical cross-entropy function is chosen as the loss function of the image segmentation model, which is calculated as follows:

$$Loss = \frac{1}{N}\sum_{i=1}^{N} -[o_i ln(p_i) + (1 - o_i) \cdot ln(1 - p_i)] \tag{3}$$

where N is the number of samples, o_i is the true label value of the sample, and p_i is the probability of the predicted value of the sample. The loss value is the difference between the actual output probability and the expected output probability. The smaller the value of the cross entropy is, the closer the two probability distributions are.

This loss function is particularly suitable for multi-class pixel-wise classification tasks such as semantic segmentation, and enables stable training with differentiable loss signals.

Firstly, 2070 movie poster datasets and corresponding labeled graphs are imported into four image semantic segmentation models for training, of which 1610 images are in the training set and 460 images are in the validation set, and the changes in the loss function in the training and validation sets are observed during the training of the networks. As shown in Figure 7, it shows the change in loss function during the training process of the three models, where the horizontal coordinate represents the number of training rounds, and the vertical coordinate represents the loss function value. The first 50 rounds of all three models were trained by freezing, and the last 150 rounds were trained by unfreezing. A comparison of the loss change curves of the three models shows that the gap between the training loss and the validation loss of the improved model in this paper is small, and the training loss and the validation loss as a whole tends to decrease during the whole training process, and the loss gradually tends to flatten out after 150 rounds of iteration, which indicates that the model has reached the optimal state. It can be seen that the improved model in this study is normal throughout the training process, there is no overfitting or underfitting phenomenon, and the model is in good condition.

After the network training is completed, 230 test set images are fed into the trained model. Despite the fact that movie posters have complex graphical backgrounds and fonts of various colors and sizes, they still perform well on the improved segmentation model in this paper.

In this paper, Mean Intersection over Union (MIoU) is used as an evaluation index, which is calculated as follows:

$$MIoU = \frac{1}{n}\sum_{i} \frac{n_{ii}}{\sum_j n_{ij} + \sum_j n_{ij} - n_{ii}} \tag{4}$$

where n_{ij} denotes the number of pixels in which category i is predicted to be category j, n_{ii} denotes the number of pixels that are predicted accurately, and n is the number of target categories (containing background, theme, text, and graphics).

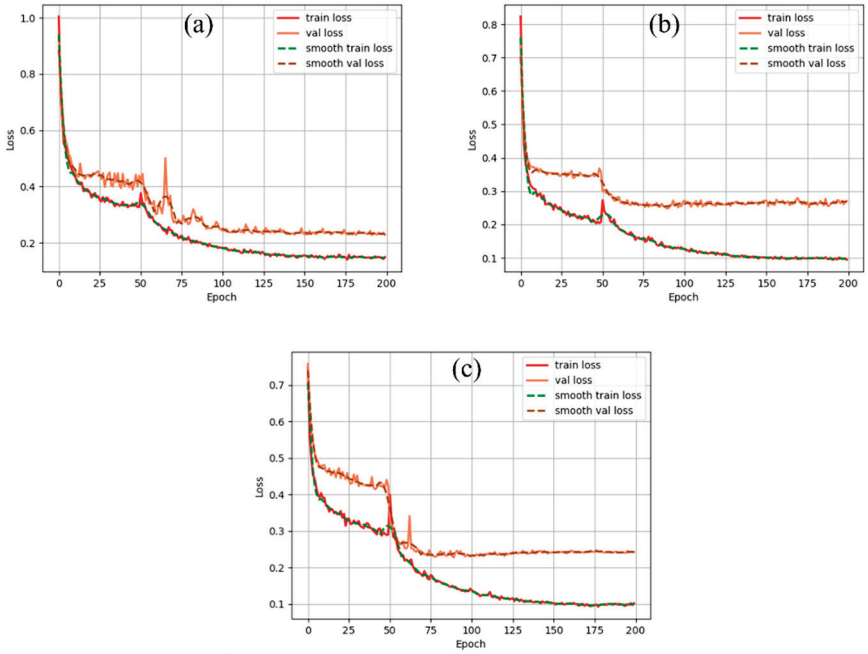


Figure 7. Loss function variation curves of the training set and validator on three different models. (**a**) The improved DeepLabv3+ model in this paper; (**b**) the original DeepLabv3+ model; (**c**) the PSPNet+ResNet50 model.

5.2. Segmentation Comparison Experiment Analysis

5.2.1. Impact of Different Backbone Feature Extraction Networks on Model Performance

We aimed to reduce the number of parameters of the model, shorten the training time of the model, and improve the segmentation accuracy of the model for small target categories. In this paper, Xception, ResNet101, Mobilenetv2, and Mobilentv3-small were selected as the backbone feature extraction networks for testing the improved model, and Table 2 shows the performance of the model when different backbone feature extraction networks are used.

Table 2. Comparison of model results using different backbone feature extraction networks. Bold-best performance.

Backbone	MIoU/%	Time/h
Xception	71.09	25.10
ResNet101	72.89	26.20
Mobilenetv2	70.05	16.20
Mobilenetv3-small	**75.60**	**15.09**

From Table 2, it can be seen that the Xception and ResNet101 networks have a higher number of parameters, more computations, and longer inference time, which took almost 10 h more compared to the Mobilenet series of networks. The Mobilenetv3 network is based on Mobilenetv1 and Mobilenetv2 with the addition of an attention mechanism and an improved time-consuming layer; thus, it can be seen that Mobilenetv3-small has a greater improvement in segmentation accuracy compared to the previous three network structures, and the training time has also been reduced to some extent. Therefore, Mobilenetv3-small network is chosen as the backbone feature extraction network for the model in this study.

5.2.2. Comparative Experimental Analysis of Different Models

In order to further validate the effectiveness and advantages of the proposed model in semantic segmentation tasks, we selected four widely used models, namely, FCN [20], PSPNet [21], DeepLabv3 [15], and the original DeepLabv3+, for comparative testing. All models were trained on the same poster dataset constructed in this study, under identical parameter settings to ensure fairness. The comparative results are presented in Table 3.

These baseline models represent three distinct architectural philosophies: FCN as an early encoder–decoder framework, PSPNet focusing on global-context modeling through pyramid pooling, and the DeepLab series employing Atrous Spatial Pyramid Pooling (ASPP) combined with a dedicated decoder for multi-scale refinement.

Other segmentation frameworks such as UNet [22], BiSeNet [23], and SegFormer [24] were not included in our experiments for specific reasons. UNet, though effective in biomedical imaging, tends to underperform on text-rich natural scenes without extensive enhancements. BiSeNet prioritizes real-time inference speed at the expense of segmentation accuracy, which is suboptimal for our boundary-sensitive poster segmentation task. SegFormer and similar transformer-based models generally require large-scale datasets and considerable computational resources to achieve stable convergence, making them less feasible under our experimental setup and computational constraints.

Table 3. Comparison experimental results of different models.

Model	Backbone	MIoU/%	Time/h
FCN	ResNet101	73.00	15.54
PSPNet	ResNet101	73.60	16.27
DeepLabv3	ResNet101	74.80	17.54
DeepLabv3+	Mobilenetv2	69.50	16.01
Textual model	Mobilenetv3-small	75.60	15.09

Therefore, our selection of baseline models aimed to strike a balance between representativeness, architectural diversity, and practical feasibility in the context of complex layout image segmentation.

From the results in Table 3, it can be seen that the improved model in this study reduces the training time by 45 min compared to the FCN model, and the segmentation accuracy is improved by 2.6%. Compared to the PSPNet model, the improved model in this paper improves 2.0% in MIoU, and the training time is reduced by 78 min; compared to the DeepLabv3 model, the segmentation accuracy of this paper's model has a small effect of improvement. Compared with the DeepLabv3 model, the model in this paper has a small effect of improving the segmentation accuracy, which is only 0.8%, but the training time is saved by 165 min, which is a larger time reduction in comparison; compared with the original DeepLabv3+ model, the model in this paper has a large improvement in the segmentation accuracy, which is 6.1% compared to the improvement, and the training time is also relatively reduced by 52 min. In summary, the performance of the improved model in this study on the self-built dataset is the most outstanding compared to the classical model.

As can be seen from Figure 8, although the DeepLabv3 model has a higher MIoU performance compared with other classical models during training, the generalization ability of its model is not high as can be seen from the test result graphs. The FCN model is not fine enough for some detail segmentation, and the learning effect of the layout style model with fewer data samples is not as good as that of the improved model in this study. Compared with PSPNet and the basic DeepLabv3+ model, the difference in segmentation

accuracy between the improved models in this paper is not large, and from the visualization results, the segmentation fineness of the improved models in this study is better than that of PSPNet and DeepLabv3+.

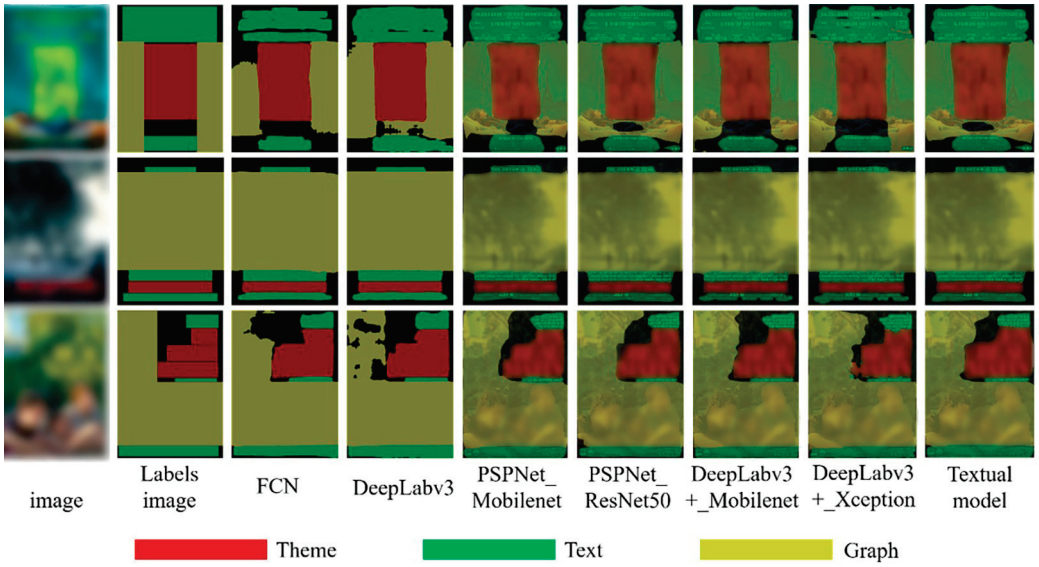


Figure 8. Comparison of page segmentation test results. The original poster images were blurred post-experiment to comply with copyright regulations, preserving layout structure for visualization without impacting model training or evaluation.

5.3. Analysis of Layout Results

The proposed approach involves two phases. In the first phase, the improved DeepLabv3+ model is used to segment the elements of movie posters, including text, subject, and graphic regions. In the second phase, the spatial relationships among these segmented elements are analyzed, and the layout type of each poster is determined using the GoogLeNet classification model. The main layout styles correspond to the common types summarized in Section 2.

The GoogLeNet model was first trained with the movie poster classification dataset, and in order to select the classification model with better performance, comparative experiments were performed using AlexNet [25], Mobilenet [26], and VGGNet [27], as shown in Figure 9, which shows the change in the loss during the training process of the four models, as well as the curve of the change in the classification accuracy rate under the same conditions.

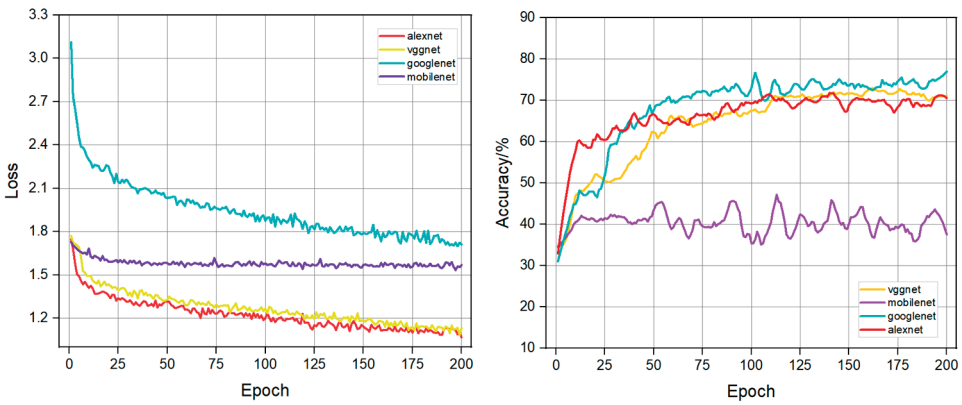


Figure 9. Loss and accuracy variation curves for four classification models.

From Figure 9, it can be seen that the GoogLeNet model has the best performance in the whole training process, so in this paper, the GoogLeNet model is chosen to analyze the layout method classification of movie posters. The network is constructed based on the Inception network, which fuses feature information from different scales to achieve better recognition results. Although the loss of the GoogLeNet model did not drop to a very low level during training, the loss was trending down throughout. As can be seen from the classification accuracy, iterating the same number of training times, GoogLeNet performs best in terms of accuracy.

6. Conclusions

Based on the analysis and generalization of the characteristics of complex layout images, this paper proposes a complex image layout analysis method based on improved DeepLabv3+ model. By improving the backbone network and the ASPP module in the DeepLabv3+ model, the network is easier to train and the image pixel recognition accuracy is higher, which enhances the model's ability to segment smaller categories of targets; secondly, based on the summarization of the poster image layout categories in this paper, we combine the GoogLeNet network model with the layout classification and analysis of the complex image. Experimental results show that the improved method proposed in this study achieves segmentation accuracy comparable to that of classical models. However, it significantly enhances the model's ability to segment smaller target categories and reduces training time while maintaining overall accuracy. Despite these advancements, the model still struggles to accurately segment text with irregular orientations—such as tilted or curved fonts—due to the complex design styles often present in poster layouts. This remains a key challenge to be addressed in future work.

In particular, future research may consider integrating recent consensus-based segmentation strategies, such as kinetic modeling frameworks [28,29], which have shown strong potential in improving robustness and handling layout uncertainty. These model-driven approaches can complement current deep learning methods and further enhance the system's generalization ability across diverse design structures.

Moreover, our experiments confirm that arbitrarily oriented text—such as slanted, rotated, and curved styles—often disrupts the alignment between text strokes and convolutional filters, leading to blurred boundaries and fragmented predictions. To address this issue, recent advances such as the Hi-SAM framework proposed by Ye et al. [30] have demonstrated promising capabilities by adapting the Segment Anything Model (SAM) for the hierarchical segmentation of complex text layouts with minimal supervision.

Additionally, the DASM module proposed by Ding et al. [31], which integrates dual spatial–channel attention into the decoder stage, has proven effective in enhancing text boundary localization under challenging lighting and distortion conditions. Building on these insights, future work will explore combining kinetic segmentation models with orientation-aware decoding, dual-attention mechanisms, and prompt-guided fine-tuning to improve robustness against spatial transformations and enhance adaptability in complex design scenarios.

Author Contributions: Conceptualization, Y.Z.; methodology, Y.Z. and Y.P.; software, Y.P.; validation, X.L., W.Z. and Y.C.; resources, H.H. and L.Z.; writing—original draft preparation, Y.P.; writing—review and editing, Y.Z. and D.L.; supervision, Y.Z.; project administration, Y.Z.; funding acquisition, Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Natural Science Foundation of Hunan Province, Grant No. 2021JJ30218.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets and source code used in this study are currently not publicly available due to copyright restrictions and ongoing research work. However, related materials can be provided upon reasonable request to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Li, H. Visual communication design of digital media in digital advertising. *J. Contemp. Educ. Res.* **2021**, *5*, 36–39. [CrossRef]
2. Jin, X.; Zhou, B.; Zou, D.; Li, X.; Sun, H.; Wu, L. Image aesthetic quality assessment: A survey. *Sci. Technol. Rev.* **2018**, *36*, 36–45. Available online: http://www.kjdb.org/CN/10.3981/j.issn.1000-7857.2018.09.005 (accessed on 1 March 2025).
3. Deng, Y.; Loy, C.C.; Tang, X. Image aesthetic assessment: An experimental survey. *IEEE Signal Process. Mag.* **2017**, *34*, 80–106. [CrossRef]
4. She, D.; Lai, Y.-K.; Yi, G.; Xu, K. Hierarchical layout-aware graph convolutional network for unified aesthetics assessment. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, Online, 19–25 June 2021; pp. 8475–8484. [CrossRef]
5. Riyanto, B. Analysis of Design Elements on Secret Magic Control Agency Movie Poster. *TAMA J. Vis. Arts* **2023**, *1*, 29–37. [CrossRef]
6. Chen, S.; Liu, D.; Pu, Y.; Zhong, Y. Advances in deep learning-based image recognition of product packaging. *Image Vis. Comput.* **2022**, *128*, 104571. [CrossRef]
7. George, N.; Sharad, C.S. Hierarchical image representation with application to optically scanned documents. In Proceedings of the 7th International Conference on Pattern Recognition (ICPR), Montreal, QC, Canada, 30 July–2 August 1984; pp. 347–349. Available online: http://digitalcommons.unl.edu/cseconfwork (accessed on 15 March 2025).
8. Mao, S.; Rosenfeld, A.; Kanungo, T.; Smith, E.H.B.; Hu, J.; Kantor, P.B. Document structure analysis algorithms: A literature survey. *Doc. Recognit. Retr. X* **2003**, *5010*, 197–207. [CrossRef]
9. Ha, J.; Haralick, R.M.; Phillips, I.T. Document page decomposition by the bounding-box project. In Proceedings of the IEEE 3rd International Conference on Document Analysis and Recognition, Montreal, QC, Canada, 14–18 August 1995; Volume 2, pp. 1119–1122. [CrossRef]
10. Pu, Y.; Liu, D.; Chen, S.; Zhong, Y. Research Progress on the Aesthetic Quality Assessment of Complex Layout Images Based on Deep Learning. *Appl. Sci.* **2023**, *13*, 9763. [CrossRef]
11. Wu, X.; Zheng, Y.; Ma, T.; Ye, H.; He, L. Document image layout analysis via explicit edge embedding network. *Inf. Sci.* **2021**, *577*, 436–448. [CrossRef]
12. Guo, S.; Jin, Z.; Sun, F.; Li, J.; Li, Z.; Shi, Y.; Cao, N. Vinci: An intelligent graphic design system for generating advertising posters. In Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, New Orleans, LA, USA, 8–13 May 2021; pp. 1–17. [CrossRef]
13. Huo, H.; Wang, F. A Study of Artificial Intelligence-Based Poster Layout Design in Visual Communication. *Sci. Program.* **2022**, *2022*, 1191073. [CrossRef]
14. Chen, L.C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking atrous convolution for semantic image segmentation. *arXiv* **2017**, arXiv:1706.05587. [CrossRef]
15. Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 834–848. [CrossRef] [PubMed]
16. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1251–1258. [CrossRef]
17. Howard, A.; Sandler, M.; Chen, B.; Wang, W.; Chen, L.-C.; Tan, M.; Chu, G.; Vasudevan, V.; Zhu, Y.; Pang, R.; et al. Searching for mobilenetv3. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1314–1324. [CrossRef]
18. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A.; Liu, W.; et al. Going deeper with convolutions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1–9. [CrossRef]
19. Everingham, M.; Eslami, S.M.A.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The pascal visual object classes challenge: A retrospective. *Int. J. Comput. Vis.* **2015**, *111*, 98–136. [CrossRef]

20. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440. [CrossRef]
21. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890. [CrossRef]
22. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In Proceedings of the MICCAI 2015, Munich, Germany, 5–9 October 2015; pp. 234–241. Available online: https://arxiv.org/abs/1505.04597 (accessed on 20 April 2025).
23. Yu, C.; Wang, J.; Peng, C.; Gao, C.; Yu, G.; Sang, N. BiSeNet: Bilateral Segmentation Network for Real-time Semantic Segmentation. In Proceedings of the ECCV 2018, Munich, Germany, 8–14 September 2018; pp. 334–349. Available online: https://arxiv.org/abs/1808.00897 (accessed on 20 April 2025).
24. Xie, E.; Wang, W.; Yu, Z.; Anandkumar, A.; Alvarez, J.M.; Luo, P. SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers. In Proceedings of the Advances in Neural Information Processing Systems, Online, 6–14 December 2021; Available online: https://arxiv.org/abs/2105.15203 (accessed on 20 April 2025).
25. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
26. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861. [CrossRef]
27. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556. [CrossRef]
28. Cabini, R.F.; Tettamanti, H.; Zanella, M. Understanding the Impact of Evaluation Metrics in Kinetic Models for Consensus-Based Segmentation. *Entropy* **2025**, *27*, 149. [CrossRef] [PubMed]
29. Cabini, R.F.; Pichiecchio, A.; Lascialfari, A.; Figini, S.; Zanella, M. A Kinetic Approach to Consensus-Based Segmentation of Biomedical Images. *Kinet. Relat. Models* **2025**, *18*, 286–311. [CrossRef]
30. Ye, M.; Zhang, J.; Liu, J.; Liu, C.; Yin, B.; Liu, C.; Tao, D. Hi-SAM: Marrying Segment Anything Model for Hierarchical Text Segmentation. *arXiv* **2024**, arXiv:2401.17904. [CrossRef] [PubMed]
31. Ding, L.; Liu, Y.; Zhao, Q.; Liu, Y. Text Font Correction and Alignment Method for Scene Text Recognition. *Sensors* **2024**, *24*, 7917. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

applied sciences

MDPI

Article

Adaptive Multi-Scale Feature Learning Module for Pediatric Pneumonia Recognition in Chest X-Rays

Petra Radočaj [1], Goran Martinović [2] and Dorijan Radočaj [3,*]

[1] Layer d.o.o., Vukovarska cesta 31, 31000 Osijek, Croatia; pradocaj@layer.agency
[2] Faculty of Electrical Engineering, Computer Science and Information Technology, Josip Juraj Strossmayer University of Osijek, Kneza Trpimira 2B, 31000 Osijek, Croatia; goran.martinovic@ferit.hr
[3] Faculty of Agrobiotechnical Sciences Osijek, Josip Juraj Strossmayer University of Osijek, Vladimira Preloga 1, 31000 Osijek, Croatia
* Correspondence: dradocaj@fazos.hr; Tel.: +385-31-554-965

Abstract: Pneumonia remains a major global health concern, particularly among pediatric populations in low-resource settings where radiological expertise is limited. This study investigates the enhancement of deep convolutional neural networks (CNNs) for automated pneumonia diagnosis from chest X-ray images through the integration of a novel module combining Inception blocks, Mish activation, and Batch Normalization (IncMB). Four state-of-the-art transfer learning models—InceptionV3, InceptionResNetV2, MobileNetV2, and DenseNet201—were evaluated in their base form and with the proposed IncMB extension. Comparative analysis based on standardized classification metrics reveals consistent performance improvements across all models with the addition of the IncMB module. The most notable improvement was observed in InceptionResNetV2, where the IncMB-enhanced model achieved the highest accuracy of 0.9812, F1-score of 0.9761, precision of 0.9781, recall of 0.9742, and strong specificity of 0.9590. Other models also demonstrated similar trends, confirming that the IncMB module contributes to better generalization and discriminative capability. These enhancements were achieved while reducing the total number of parameters, indicating improved computational efficiency. In conclusion, the integration of IncMB significantly boosts the performance of CNN-based pneumonia classifiers, offering a promising direction for the development of lightweight, high-performing diagnostic tools suitable for real-world clinical application, particularly in underserved healthcare environments.

Keywords: deep learning; convolutional neural network; medical imaging; image analysis; feature extraction; Mish activation function; optimization

1. Introduction

Pneumonia is an acute infection affecting the alveolar regions of the lungs and represents a leading cause of childhood morbidity and mortality worldwide [1,2]. It is typically caused by bacterial, viral, or fungal pathogens and manifests clinically with fever, cough, tachypnea, chest retractions, pleuritic pain, and abnormal breath sounds such as crackles or wheezing [3–5]. The severity of pneumonia in children can vary significantly, ranging from mild and self-limiting to life-threatening conditions involving respiratory failure and acute respiratory distress syndrome [6]. Prompt diagnosis and timely intervention are therefore essential for reducing complications and improving clinical outcomes [7]. Statistical data highlights the ongoing burden of pediatric pneumonia on global health systems. In 2019

alone, pneumonia accounted for approximately 740,000 deaths among children under five, representing 14% of all under-five mortality [8], and disproportionately affecting regions such as South Asia and sub-Saharan Africa [6,9]. Pneumonia remains a major cause of pediatric hospitalizations, contributing significantly to healthcare utilization and child mortality [10]. Risk factors that predispose children to pneumonia include malnutrition, incomplete immunization coverage, indoor air pollution, and limited access to quality healthcare services [11]. Treatment strategies for pediatric pneumonia mainly depend on the underlying etiology. Bacterial infections typically require antimicrobial therapy, with macrolides or beta-lactam antibiotics frequently used [12], depending on the age of the patient and the likely causative organism [13]. For example, macrolides are preferred in school-aged children when atypical pathogens such as Mycoplasma pneumoniae are suspected [14]. In contrast, viral pneumonia is managed primarily with supportive care, including oxygen therapy, hydration, and antiviral agents when appropriate [15,16]. Severe cases may necessitate hospitalization and intensive care support, including mechanical ventilation.

Radiological imaging is considered essential for the diagnosis of pneumonia, with chest radiography (CXR) recognized as the gold standard imaging modality [17]. The presence of new pulmonary infiltrates on a chest X-ray provides objective evidence of alveolar involvement and supports a definitive pneumonia diagnosis. Although limited in sensitivity and specificity, chest radiography is the most common instrument because of availability, speed, and the relative affordability to perform [18,19]. It enables differentiation between pneumonia and other respiratory conditions, and plays a central role in clinical decision-making, particularly in acute care settings [2]. While computed tomography (CT) of the chest offers higher sensitivity for detecting parenchymal changes, it is rarely employed in pediatric practice due to concerns about ionizing radiation and limited accessibility [17]. Lung ultrasound (LUS) has emerged as a potential alternative, offering the advantages of portability, absence of radiation, and suitability for point-of-care use [13,20]. However, despite its high sensitivity, LUS suffers from relatively low specificity and limited negative predictive value, making it unreliable as a standalone diagnostic tool. Studies have shown that although a positive LUS finding strongly suggests pneumonia, a negative result does not reliably exclude the disease [21,22]. Therefore, in cases with high clinical suspicion and negative ultrasound findings, additional imaging with CXR or CT is often warranted [17]. As such, chest radiography remains the reference standard for confirming pneumonia, especially in pediatric patients [23].

Because pneumonia diagnosis relies heavily on imaging, researchers have explored automating chest X-ray interpretation using artificial intelligence, particularly deep learning approaches. The CNNs have demonstrated great performance in the classification of thoracic pathologies such as pneumonia [24–26]. These models have demonstrated the ability to extract hierarchical image features that may not be apparent to human observers, facilitating more accurate and efficient diagnosis [27,28]. The application of CNNs in pediatric imaging has the potential to enhance diagnostic accuracy, reduce interpretation time, and alleviate the workload on radiologists [29,30]. In addition, deep learning has shown a powerful performance of processing high-dimensional data, noisy data, and non-stationary data in a wide range of fields. Multi-scale architectures are robust and generalizable as adaptive neural structures have been used in structural health monitoring to adapt to environmental variability and signal distortions using hierarchical feature optimization [31]. Despite notable progress, the broader clinical adoption of AI-driven models for pediatric pneumonia diagnosis remains constrained by several persistent limitations [32]. One primary concern is computational complexity. Advanced hybrid frameworks and transformer-based architectures, while delivering strong diagnostic performance, often rely on resource-intensive

components such as fuzzy inference systems, multi-head attention mechanisms, or ensemble learning strategies [24,33,34]. These introduce significant latency and memory consumption, which can limit deployment in low-resource clinical environments [35]. Even relatively streamlined models require sophisticated fine-tuning procedures and optimized activation functions, further complicating their practical implementation. Equally critical is the limited adaptability of many models to pediatric-specific imaging features [36,37]. Deep learning architectures trained predominantly on adult datasets may not generalize well to pediatric chest radiographs, which differ in anatomical scale, developmental variability, and disease manifestation patterns [34]. Pediatric pneumonia often presents with viral features that are more diffuse and less radiographically distinct, making the recognition task inherently more challenging [18]. Consequently, models optimized on adult populations may exhibit reduced sensitivity or specificity when applied to pediatric cases [34]. Furthermore, structural features of the medical imaging data present further complications. Unlike the natural RGB images, chest radiography is a monochromatic image with diagnostically valuable information encoded into subtle shades of grayscale level differences [38,39]. Many existing CNN-based solutions, including architectures like Inception and EfficientNet, were originally tailored for color image recognition and thus do not inherently accommodate the lower-dimensional and contrast-dependent nature of grayscale medical data [40]. This mismatch can diminish feature extraction efficiency, particularly in subtle or small lesion recognition scenarios. Furthermore, current models often fail to leverage the statistical and textural signatures that distinguish radiographic findings in pediatric pneumonia. Lesions in this population are frequently small, ill-defined, and distributed heterogeneously across the lung fields [41]. Standard modules lack adaptive mechanisms to integrate multi-scale contextual information from such complex grayscale inputs [42]. This gap suggests an underexplored need for architectural components specifically designed to enhance multi-scale feature learning in monochromatic images.

These challenges highlight the need for deep learning frameworks that are diagnostically accurate, computationally efficient, and specifically adapted to pediatric and grayscale imaging characteristics. In response to these challenges, we investigate the adaptation of the IncMB (Inception–Mish activation function–Batch Normalization) module—a lightweight and scalable architectural block originally designed for color image classification, in the context of pediatric pneumonia recognition using chest radiographs. The key contributions of this paper are as follows:

- This study adapts the IncMB module for grayscale pediatric chest X-ray analysis, overcoming the limitations of traditional CNNs designed for RGB images. By leveraging the Mish activation function for smoother gradients and Batch Normalization for robust regularization, the module achieves enhanced feature extraction in monochromatic medical images, improving pneumonia classification accuracy.
- This study rigorously evaluates the IncMB module's integration into established CNN architectures on a publicly available pediatric CXR dataset, demonstrating its balanced trade-off between computational efficiency and diagnostic performance. The module's design inherently reduces parameter growth compared to conventional multi-branch architectures while maintaining competitive accuracy.
- This work bridges the gap between adult-trained deep learning models and pediatric-specific diagnostic needs by optimizing feature learning for subtle, diffuse radiographic patterns unique to children. The adapted IncMB module enhances sensitivity to small lesions and viral pneumonia manifestations, supporting more accurate and accessible AI-driven diagnosis in pediatric healthcare.

This paper is organized into distinct sections. Section 2 surveys existing literature concerning deep learning approaches for diagnosing pediatric pneumonia. Section 3 details our proposed model and its methodologies. Subsequently, Section 4 presents experimental findings and a comparative analysis of the model's performance. The paper concludes in Section 5 with a summary of our findings and suggestions for future research.

2. Related Works

Advances in deep learning have significantly enhanced the diagnostic performance of CNNs in the analysis of pediatric chest radiographs, particularly for pneumonia recognition. These models have demonstrated promising results in terms of accuracy and sensitivity; however, the challenge of balancing diagnostic robustness, interpretability, and computational efficiency continues to hinder real-world deployment, especially in resource-constrained clinical environments that demand rapid and reliable diagnostic support.

Numerous innovative approaches have been proposed to address the challenges inherent in pediatric chest X-ray analysis. Sotirov et al. [43] introduced a hybrid framework that integrates CNNs with intuitionistic fuzzy estimators (IFEs) to improve robustness against data ambiguity and noise. By leveraging CNNs for feature extraction and IFEs for uncertainty modeling, their system achieved a classification accuracy of 0.9493, surpassing traditional CNNs in both sensitivity and specificity. Despite achieving higher accuracy, the inclusion of fuzzy inference layers increases computational cost, introducing latency and memory overhead that limit real-time clinical use. Ensemble learning approaches have also shown considerable promise. Prakash et al. [44] developed a stacking ensemble model incorporating five channel-attention-based CNNs—ResNetV2, Xception, and DenseNet169 among them. These feature representations were refined using kernel principal component analysis, with classification performed via support vector machines and XGBoost. The architecture yielded a high classification accuracy of 0.9615 and an AUC of 0.9624. However, the operational complexity associated with deploying multiple CNNs concurrently, along with the overhead from feature fusion and dimensionality reduction, imposes significant constraints on clinical integration, particularly in settings where computational resources and inference time are limited. Transfer learning has emerged as a viable strategy for improving diagnostic performance, particularly when pediatric data is scarce. Lan et al. [45] conducted a comparative analysis of ResNet50, DenseNet121, and EfficientNetV2-S for Mycoplasma pneumonia detection, finding ResNet50 to be the most effective with an accuracy of 0.8265. Their study highlighted the critical role of domain-specific pretraining on pediatric datasets over generic datasets such as ImageNet, underscoring the importance of contextual feature learning in medical imaging. However, the moderate AUC scores ranging from 0.822 to 0.758, and specificity ranging from 0.655 to 0.814 suggest ongoing difficulties in accurately distinguishing between different pneumonia subtypes and other thoracic pathologies. Further optimization of transfer learning was demonstrated by Khan et al. [46], who refined the EfficientNetB1 architecture through three principal innovations: the replacement of ReLU with Swish activation, extensive hyperparameter tuning, and classification head regularization. This approach yielded a 0.9613 accuracy in a four-class classification task involving COVID-19, viral pneumonia, lung opacity, and normal cases. Importantly, the regularization strategy effectively mitigated overfitting—a common issue in pediatric imaging due to limited dataset sizes. However, continuing to use RGB-based architectures for grayscale chest X-rays may constrain performance, suggesting potential benefits from grayscale-optimized designs. Architectural enhancements were also explored by Wang et al. [47], who improved DenseNet by integrating Squeeze-and-Excitation (SE) blocks, replacing average pooling with max-pooling to preserve lesion-level detail, and

employing PReLU activation to address neuron deactivation. Their modifications resulted in a 2.4% increase in accuracy over baseline DenseNet, reaching 0.928 with high precision of 0.926 and recall of 0.962. Nevertheless, their validation was limited to adult CXR datasets, raising concerns about direct applicability to pediatric cases due to notable anatomical and pathological differences. More recently, transformer-based models have pushed the boundaries of diagnostic performance. Chen et al. [33] introduced a Mixed-Scale Dynamic Attention Transformer for pediatric pneumonia diagnosis, incorporating dynamic local attention, hierarchical multi-scale feature integration, and large language model (LLM)-driven attention amplification. This architecture achieved state-of-the-art performance with 0.952 accuracy and 0.990 AUC. Despite these impressive metrics, the integration of LLMs and advanced attention mechanisms necessitates substantial computational resources, limiting deployment in real-world healthcare settings, particularly in low- and middle-income regions where pediatric pneumonia is most prevalent.

Studies have further explored the impact of domain-specific pretraining and foundation models on medical imaging performance. Models trained on large-scale radiology datasets such as RadiologyNET have demonstrated slightly higher accuracy than ImageNet-pretrained counterparts, particularly in data-limited scenarios [48]. Similarly, the RadImageNet initiative, trained on 1.35 million annotated medical images, reported measurable AUC gains across multiple classification tasks, including thyroid, breast, and pulmonary conditions [49]. Despite these improvements, both approaches remain limited by their high computational demands and reliance on extremely large pretraining datasets, which restricts their practical reproducibility and accessibility. In parallel, foundation models like BiomedCLIP have achieved state-of-the-art few-shot classification results in pediatric and thoracic imaging; however, their vast parameter counts and dependence on multimodal training data render them impractical for real-time clinical deployment [50].

Across approaches, several critical limitations persist. High computational complexity continues to hinder the integration of state-of-the-art models into real-time clinical workflows. Moreover, many architectures are not optimally tailored to the unique challenges of pediatric imaging, including subtle lesion presentation and age-dependent anatomical variability. In addition, a large proportion of existing models were originally developed for color image classification and may insufficiently exploit the structural and textural properties of grayscale chest radiographs, potentially limiting their diagnostic effectiveness in medical imaging contexts. In response to these challenges, the present study investigates the adaptation of the IncMB module, originally developed for color image classification, for application in pediatric chest radiograph analysis. This module integrates components from the Inception architecture, the Mish activation function, and Batch Normalization, all tailored to leverage the statistical and spatial properties of grayscale medical imaging. The proposed framework evaluates the incorporation of IncMB within standard CNN pipelines to enhance feature extraction and classification accuracy in pediatric pneumonia recognition, while maintaining computational efficiency suitable for potential integration into real-world diagnostic systems.

3. Materials and Methods

In this study, we evaluate the feasibility of adapting the IncMB module for grayscale chest radiographs to support pediatric pneumonia classification. Although originally developed for RGB image processing [51], we integrate IncMB into medical deep learning workflows and assess its effectiveness in enhancing feature extraction from monochromatic data. Our goal is to determine whether this lightweight, multi-scale module can improve diagnostic accuracy and efficiency when applied to pediatric chest X-rays.

To achieve this, we employ four widely used CNN architectures—InceptionV3, InceptionResNetV2, DenseNet201, and MobileNetV2—as both baseline models and modified versions incorporating the IncMB module. This dual evaluation allows us to directly compare performance gains introduced by IncMB. The module of IncMB is appended directly to backbone convolutional feature maps and precedes the classification head. This position enables the module to boost multi-scale properties without compromising the ability to extract features in the backbone network that is pretrained. The experimental workflow consists of three main stages, as illustrated in Figure 1: (1) Data preprocessing and preparation, using the publicly available Chest X-ray Images (Pneumonia) dataset to classify images into two categories: pneumonia and healthy. We apply preprocessing techniques and data augmentation to improve generalization and mitigate overfitting; (2) Model implementation, where we integrate the IncMB module into the selected CNN architectures and fine-tune both the base and modified models using transfer learning; (3) Performance evaluation, where we assess and compare both baseline and IncMB-enhanced models using classification metrics such as accuracy, precision, recall (sensitivity), F1 score and specificity.

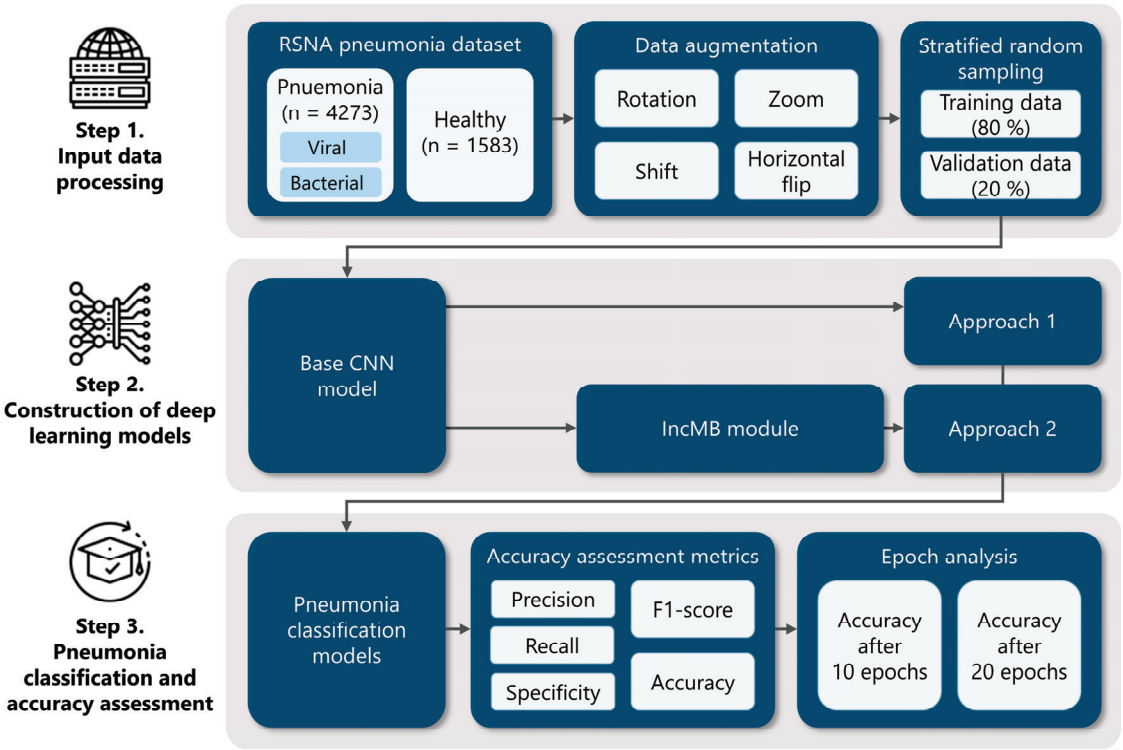


Figure 1. Comprehensive methodology for deep learning pneumonia classification: (1) Initial data preparation, encompassing augmentation techniques and the division of the dataset into training and validation subsets via stratified random sampling; (2) Development of deep learning models utilizing established architectures such as InceptionV3, InceptionResNetV2, DenseNet201, and MobileNetV2, integrated into two distinct modeling strategies, including an IncMB module; (3) Evaluation of pneumonia classification performance through various accuracy metrics and an analysis of model behavior across different training epochs.

3.1. Data Preparation and Experimental Setup

In the given study, we used publicly available Chest X-ray Images (Pneumonia) dataset [52], which consists of 5856 chest X-rays taken of pediatric patients (one to five years old) in Guangzhou Woman and Children Hospital. There are 4273 cases of pneumonia and 1583 healthy controls in this data [52]. While the dataset includes both viral and bacterial pneumonia labels, these were combined into a single pneumonia class, and all models

were trained using binary cross-entropy loss to distinguish pneumonia from healthy cases. Figure 2 shows some of the samples. The quality of the data and reliability of the diagnosis has been considered by screening of all X-ray images prior to their use to extract low-quality scans or those that are unreadable. Two expert physicians made the diagnoses, and a third expert would confirm the check to reduce the possibility of errors.

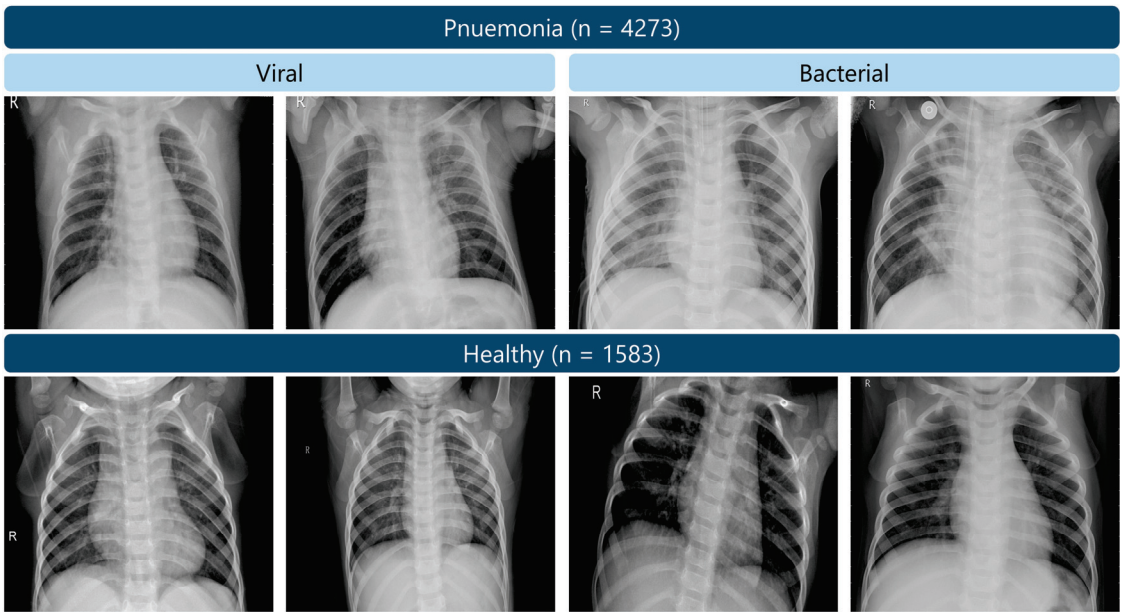


Figure 2. Examples of chest X-ray images from the RSNA pneumonia dataset.

The dataset was partitioned using a stratified random split method at an 80:20 ratio for training and validation, thereby preserving the original class distribution. Despite the inherent class imbalance, several strategies were implemented to mitigate its impact. Firstly, extensive data augmentation techniques, including rotation ($\pm 30^{\circ}$), width and height shifts ($\pm 10\%$), zooming (up to 20%), and horizontal flipping, were applied equally across both classes. This artificially expanded the training set and reduced overfitting, particularly benefiting the minority (healthy) class. Secondly, evaluation focused on metrics sensitive to class imbalance, such as precision, recall, and F1-score, rather than solely accuracy, providing a more reliable assessment of model performance across both classes. Thirdly, stratified splitting ensured consistent representation of the imbalance in both sets, preventing bias toward the majority class during evaluation. Due to computational resource limitations and the dataset's scale, a single 80:20 split was chosen over k-fold cross-validation, which would have significantly increased training time without proportional benefits. All chest X-ray images were resized to 224×224 pixels, matching the input dimensions commonly used by convolutional neural networks such as InceptionV3, DenseNet201, and MobileNetV2. This resizing provides a standardized input for deep learning models, while retaining sufficient detail for the detection of radiographic findings relevant to pneumonia. Although fine interstitial changes may be partially blurred at this resolution, larger opacities and consolidations, which are typically the focus of pneumonia diagnosis, remain discernible [9,18].

We developed and trained the deep learning models in a Python environment (Python 3.10; Python Software Foundation, Wilmington, DE, USA), leveraging the Keras-GPU [53] and TensorFlow-GPU [54] frameworks on the Google Colab platform. Google Colab facilitated efficient training through its NVIDIA Tesla K80 GPUs (NVIDIA Corporation, Santa Clara, CA, USA) with 12 GB of memory. Training details, optimizer, loss function, and other hyperparameters are summarized in Table 1.

Table 1. Summary of model hyperparameters.

Hyperparameter	Value
Number of epochs	20
Batch size	32
Optimizer	Adam
Adam β_1, β_2, ε	0.9, 0.999, 1×10^{-7}
Initial learning rate	0.001
Minimum learning rate	0.5×10^{-6}
Learning rate scheduler	Reduce on plateau

3.2. Model Architecture and Adapted IncMB Module

CNNs have been established as foundational in medical image analysis, primarily because of their capacity to hierarchically extract spatial and contextual features from input data [24,43,44]. In this study, four CNN architectures—InceptionV3, InceptionResNetV2, DenseNet201, and MobileNetV2—were adopted through transfer learning, utilizing pre-trained ImageNet weights. Instead of the fine-tuning of the entire layers, the pre-trained convolutional bases were left as fixed feature extraction modules, but custom classification heads were added and learnt on the pediatric pneumonia data. Although domain-specific pretraining can be beneficial, using ImageNet-pretrained models offers several advantages: they are extensively validated, widely available, and provide rich low-level visual representations that generalize well to medical imaging tasks. This approach also ensures methodological consistency and comparability across related studies [36,37].

A critical challenge arises from the fundamental difference between the grayscale nature of CXRs and the color RGB images on which traditional CNNs are typically trained. This distinction manifests in terms of dimensionality (single channel for grayscale vs. three channels for RGB), contrast sensitivity, and the intricate representation of texture. Consequently, directly applying CNNs trained on color images often leads to suboptimal feature extraction when confronted with grayscale medical images [38,39]. Pediatric CXRs present further complexities, including age-related anatomical variability, inherently lower contrast in pathological regions, and the subtle, diffuse, and often small-scale lesion presentations that are less radiographically distinct compared to those commonly observed in adult populations [34].

To address these challenges, we introduce the IncMB module, a lightweight architectural unit specifically designed to enhance multi-scale feature learning in grayscale medical images. Overall architecture is shown in Figure 3.

- The Inception architecture: This foundational design enables efficient multi-branch processing, allowing the network to concurrently extract features at different spatial scales. This is achieved through parallel convolutions employing various kernel sizes and a pooling branch [55,56]. Moreover, its inherent multi-scale capacity is particularly well-suited for pediatric pneumonia recognition, where pathologies can manifest as small, poorly defined opacities or as broader, more diffuse patterns [28].
- The Mish activation function: To further adapt the module for grayscale image processing, Mish is employed as the activation function. Mish is a non-monotonic, self-regularizing activation function [57,58] as mathematically defined in Equation (1):

$$Mish(x) = x \cdot tanh(softplus(x)) = x \cdot \tanh\left(\ln(1+ e^{x})\right) \quad (1)$$

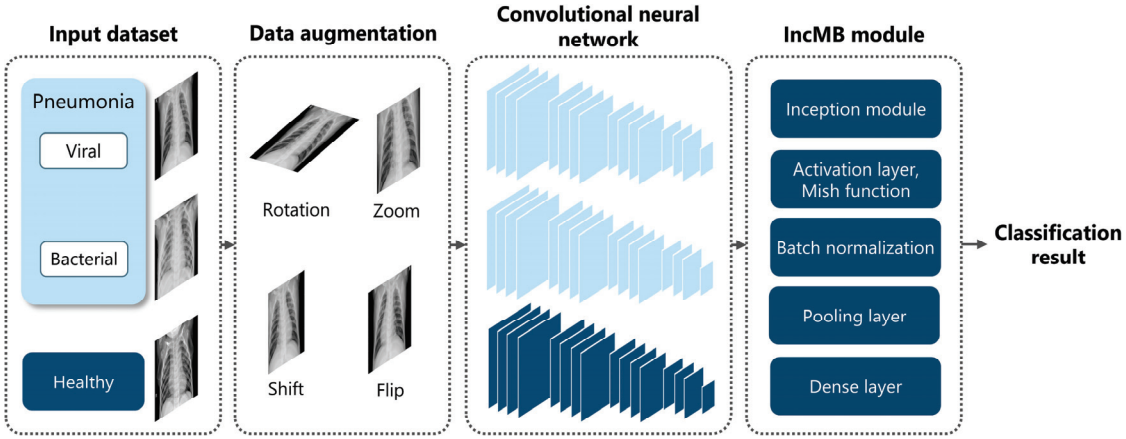


Figure 3. Overview of the proposed framework for pediatric pneumonia classification, highlighting the integration and role of the IncMB module.

Unlike traditional activation functions such as ReLU, Mish maintains a smooth, continuous gradient flow, even for negative input values, thereby preventing abrupt transitions that could impede network convergence [57]. Its formulation fosters better feature representation by enabling richer information propagation across layers, especially in deeper networks or when the input features contain subtle grayscale variations, as is characteristic of chest radiographs. Studies have compellingly demonstrated that Mish activation significantly outperforms both ReLU and Swish in tasks demanding the capture of fine-grained patterns, such as grayscale medical image classification [26,58]. Its smooth curvature and non-zero gradient across all real inputs contribute to more stable optimization and ultimately, higher classification accuracy. In the specific context of pediatric CXR analysis, where abnormalities may reside in low-contrast or texturally ambiguous regions, Mish ensures that these subtle diagnostic cues are meticulously preserved and amplified across convolutional layers, rather than being clipped or nullified as might occur with ReLU-based activations [58]. This property is particularly advantageous when combined with multi-scale feature extraction, as it empowers the network to retain crucial diagnostic details that might otherwise be lost through shallower or narrower activations.

- Batch Normalization: The IncMB module also incorporates Batch Normalization after the convolution and activation stages. This crucial regularization mechanism ensures that internal covariate shift is minimized and that feature distributions remain stable throughout the training process [59]. This leads to faster convergence, the ability to utilize higher learning rates, and a reduced risk of overfitting—all critical considerations when working with relatively small and potentially imbalanced pediatric datasets.

Structurally, the IncMB module comprises four distinct branches: a 1×1 convolutional pathway, a 1×1 followed by 3×3 convolutional pathway, a 1×1 followed by 5×5 convolutional pathway, and a 3×3 max-pooling followed by a 1×1 convolution. Each of these branches captures features at distinct receptive fields, which are subsequently concatenated to form a rich, multi-scale representation of the input image [55]. This composite feature map is then passed through the Mish activation function and Batch Normalization before being processed by the downstream CNN layers. A single IncMB module is positioned after the backbone feature extraction layers and before the global pooling and classification layers.

In adapting the IncMB module to the grayscale pediatric pneumonia dataset, it is strategically inserted into the selected CNN backbone, either replacing or complementing

traditional convolutional blocks. This deliberate integration allows the network to more effectively accommodate the unique radiological characteristics of pediatric CXRs, including heterogeneous lesion distribution and the often-subtle textural differences between pathological and healthy tissues. The overall methodology is outlined in Appendix A.

3.3. Pediatric Pneumonia Performance Evaluation

The assessment of deep learning model efficacy in the classification of pediatric pneumonia from chest radiographs necessitated the application of a comprehensive set of quantitative metrics. These indicators, including precision, recall (sensitivity), specificity, F1-score, and overall accuracy, were systematically derived from the confusion matrix to facilitate a granular evaluation of the models' diagnostic capabilities, as defined in Equations (2)–(6) [60,61].

$$Precision = \frac{TP}{TP + FP}, \tag{2}$$

$$Recall = \frac{TP}{TP + FN}, \tag{3}$$

$$Specificity = \frac{TN}{TN + FP}, \tag{4}$$

$$F1\text{-}score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}, \tag{5}$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}, \tag{6}$$

The model's classifications were precisely compared against the established ground truth. True Positives (TP) denoted instances where the presence of pneumonia was correctly identified. Conversely, True Negatives (TN) represented accurate determinations of the absence of pneumonia. Errors in classification were delineated as False Positives (FP), occurring when a non-pneumonic condition was erroneously assigned the pneumonia label, and False Negatives (FN), signifying a failure to detect actual pneumonia cases.

4. Results and Discussion

This section presents a comprehensive evaluation of the proposed pneumonia classification models using multiple transfer learning backbones—InceptionV3, InceptionResNetV2, MobileNetV2, and DenseNet201—under two approaches: a base model and the base model with the IncMB.

Across all networks, the integration of the IncMB module led to a consistent and notable improvement in classification performance metrics, underscoring its effectiveness in enhancing deep learning models for pneumonia recognition as detailed in Table 2. Among the evaluated models, InceptionResNetV2 with IncMB achieved the highest scores across several key metrics—accuracy of 0.9812, F1-score of 0.9761, and recall of 0.9742—demonstrating its superior ability to generalize and accurately classify pneumonia cases. These results highlight the synergy between the deep architecture of InceptionResNetV2 and the feature-enriching capabilities of IncMB. In addition to absolute performance gains, IncMB integration significantly boosted performance for lighter models. Notably, MobileNetV2 with IncMB exhibited the largest improvement in accuracy, with a jump of +4.69% from its base model from 0.9113 to 0.9582. This substantial gain reinforces the utility of IncMB, especially for resource-constrained settings where lightweight models are preferred. Moreover, improvements in precision and F1-score were consistently ob-

served across all models with IncMB. However, some trade-offs became apparent. For instance, MobileNetV2 with IncMB experienced a decrease in specificity from 0.9779 to 0.8675. Despite this, its recall increased significantly from 0.8865 to 0.9918, which is a desirable outcome in clinical scenarios where minimizing false negatives is paramount. In such contexts, especially in pneumonia diagnosis, capturing as many true cases as possible is often more critical than avoiding false positives [62]. Overall, the inclusion of IncMB led to improved robustness and diagnostic value across various deep learning backbones.

Table 2. Performance evaluation of the proposed pneumonia classification approaches based on overall accuracy metrics.

Transfer Deep Learning Model	Classification Approach	Accuracy	F1-Score	Precision	Recall	Specificity
InceptionV3	Base model	0.9633	0.9529	0.9592	0.9471	0.9117
	Base model with IncMB	**0.9710**	**0.9630**	**0.9659**	**0.9603**	**0.9369**
InceptionResNetV2	Base model	0.9676	0.9595	0.9536	0.9659	**0.9621**
	Base model with IncMB	**0.9812**	**0.9761**	**0.9781**	**0.9742**	0.9590
MobileNetV2	Base model	0.9113	0.8961	0.8763	**0.9322**	**0.9779**
	Base model with IncMB	**0.9582**	**0.9451**	**0.9640**	0.9297	0.8675
DenseNet201	Base model	0.9676	0.9579	**0.9704**	0.9470	0.9022
	Base model with IncMB	**0.9727**	**0.9656**	0.9629	**0.9684**	**0.9590**

The highest assessment metrics are bolded.

As detailed in Table 3, the base models generally demonstrated higher recall for the pneumonia class compared to the healthy class, indicating a tendency to better identify diseased cases while occasionally misclassifying healthy ones. This imbalance poses a challenge in clinical settings, where both false negatives and false positives can have serious implications. After incorporating the IncMB, recall for the healthy class improved notably across most models, suggesting enhanced discrimination between the two classes. For example, InceptionV3's healthy class recall increased from 0.9117 to 0.9369, and DenseNet201 saw a significant jump from 0.9022 to 0.9590. These improvements reflect a reduction in false positives—healthy patients incorrectly flagged as having pneumonia. The pneumonia class also showed improvements or maintained high recall values after IncMB integration. MobileNetV2 with IncMB achieved the highest pneumonia recall of 0.9918, marking a substantial improvement from its base recall of 0.8865. Interestingly, DenseNet201's base model already exhibited a high pneumonia recall of 0.9918, which remained robust after IncMB inclusion. These results suggest that IncMB not only improves the network's ability to detect healthy cases but also maintains or enhances its strength in identifying pneumonia. Overall, the IncMB contributes to a more balanced classification performance, reducing bias toward one class and improving the model's reliability for real-world clinical applications where accurate identification of both healthy and diseased cases is essential.

Table 3. Detailed recall scores for pneumonia classification for each CNN within the evaluated approaches.

Transfer Deep Learning Model	Classification Approach	Healthy	Pneumonia
InceptionV3	Base model	0.9117	0.9825
	Base model with IncMB	**0.9369**	**0.9836**
InceptionResNetV2	Base model	**0.9621**	0.9696
	Base model with IncMB	0.9590	**0.9895**
MobileNetV2	Base model	**0.9779**	0.8865

Table 3. *Cont.*

Transfer Deep Learning Model	Classification Approach	Healthy	Pneumonia
MobileNetV	Base model with IncMB	0.8675	**0.9918**
DenseNet201	Base model	0.9022	**0.9918**
	Base model with IncMB	**0.9590**	0.9778

The highest assessment metrics are bolded.

Training and validation trends over 10 and 20 epochs demonstrated smoother convergence and lower validation loss for all IncMB-enhanced models as presented in Table 4. A key observation is the marked reduction in validation loss, an indicator of improved model generalization and reduced overfitting. Most notably, InceptionResNetV2 with IncMB achieved the lowest validation loss of 0.0547 after 20 epochs, significantly outperforming its base counterpart, which recorded a validation loss of 0.0665. This suggests that the model not only converged more effectively but also retained better generalization capacity on unseen data. Similar positive trends were observed in InceptionV3 and DenseNet201. For instance, InceptionV3's validation loss dropped from 0.1235 for base model to 0.0739 for base model with IncMB after 20 epochs, reflecting more efficient learning and better representation of data distributions. DenseNet201 with IncMB also exhibited enhanced performance, with its validation loss reducing from 0.0857 to 0.0794, while validation accuracy increased from 0.9679 to 0.9731. Even for lightweight architectures like MobileNetV2, IncMB contributed to meaningful improvements. Validation loss decreased from 0.2459 to 0.1610 over 20 epochs, and validation accuracy improved from 0.9288 to 0.9601, demonstrating that IncMB facilitates more robust learning dynamics even in computationally efficient models. Overall, these results demonstrate that IncMB positively influences the training trajectory, enabling faster convergence, lower validation error, and higher validation accuracy across varying architectures. This enhanced training behavior positions IncMB as a valuable module for improving both performance and reliability in deep learning-based medical image classification tasks.

Table 4. Training dynamics illustrating the progression of accuracy and validation loss over 10 and 20 epochs for the different proposed classification approaches.

Transfer Deep Learning Model	Classification Approach	10 Epochs			20 Epochs		
		TA	VA	VL	TA	VA	VL
InceptionV3	Base model	0.9438	0.9479	0.1508	0.9640	0.9618	0.1235
	Base model with IncMB	**0.9700**	**0.9679**	**0.0910**	**0.9688**	**0.9740**	**0.0739**
InceptionResNetV2	Base model	**0.9811**	0.9505	0.1287	**0.9809**	0.9766	0.0665
	Base model with IncMB	0.9688	**0.9679**	**0.0887**	0.9688	**0.9818**	**0.0547**
MobileNetV2	Base model	0.9619	0.9392	0.2619	0.9616	0.9288	0.2459
	Base model with IncMB	**0.9656**	**0.9523**	**0.2056**	**0.9758**	**0.9601**	**0.1610**
DenseNet201	Base model	0.9540	0.9340	0.1590	**0.9688**	0.9679	0.0857
	Base model with IncMB	**0.9588**	**0.9609**	**0.1176**	0.9375	**0.9731**	**0.0794**

The highest assessment metrics are bolded.

As demonstrated in Figure 4, IncMB significantly decreased the number of pneumonia cases misclassified as healthy across all networks, for example, InceptionResNetV2's FNs dropped from 26 to 9, and MobileNetV2's from 97 to just 7, showing over 90% improvement in FN rate for the latter. True negative counts also improved for most models with IncMB, especially for DenseNet201 from 286 to 304 and InceptionV3 from 289 to 297, resulting in

improved specificity. While some models experienced a slight increase in false positives, this was often accompanied by a more clinically acceptable reduction in false negatives. For instance, MobileNetV2 with IncMB improved pneumonia recall from 0.8865 to 0.9918 at the cost of a moderate drop in healthy recall. Overall, the confusion matrices clearly demonstrate that the proposed IncMB module not only boosts aggregate performance metrics but also leads to a clinically meaningful redistribution of errors—favoring the reduction in false negatives, which is critical in pneumonia recognition.

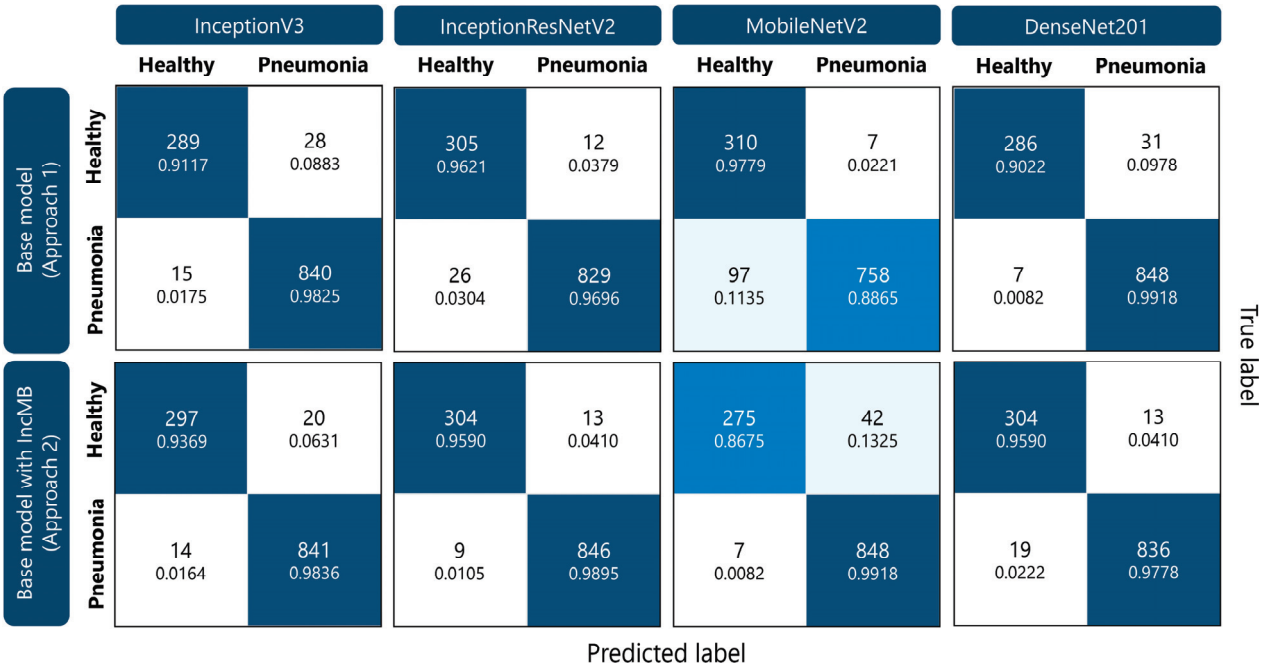


Figure 4. Confusion matrices illustrating the performance of different CNN models (InceptionV3, InceptionResNetV2, MobileNetV2, and DenseNet201) for pediatric pneumonia classification. Within each cell, the top number indicates the count of classified instances, while the lower number presents the corresponding percentage for that specific class.

Incorporating the IncMB module across different CNN architectures produced consistent improvements in accuracy, generalization, and computational efficiency. The observed gains are attributable to the synergistic interaction of three architectural enhancements: multi-branch Inception-style convolutions, Batch Normalization, and the Mish activation function. Together, these components provide a more expressive, stable, and generalizable feature extraction framework, particularly well-suited for the heterogeneous and fine-grained visual patterns encountered in pediatric chest radiographs. The Inception component of IncMB employs parallel convolutional paths with varying kernel sizes, allowing the network to simultaneously capture features at multiple spatial resolutions [55,63]. This capacity for multi-scale representation is critical in medical imaging, where pathological features such as infiltrates or consolidations may present with variable size, shape, and intensity. Furthermore, the use of 1×1 convolutions as bottleneck layers enables dimensionality reduction prior to computationally intensive operations, thereby lowering the total number of parameters without compromising feature richness [64]. Batch Normalization contributes to both training stability and generalization. By normalizing the intermediate feature distributions, Batch Normalization reduces internal covariate shift and smooths the loss landscape, which facilitates faster convergence and allows for the use of higher learning rates [65,66]. Moreover, its implicit regularization effect helps prevent overfitting, particularly in smaller or imbalanced datasets common in medical imaging [65].

The Mish activation function complements these benefits by introducing a smooth, non-monotonic, and self-regularizing nonlinearity [57]. Unlike traditional ReLU, Mish retains negative information while preserving gradient flow across layers. Its continuous differentiability and curvature near zero enable improved learning dynamics, particularly in deep architectures [57]. These properties have been shown to enhance representation learning in vision tasks and are reflected in the improved recall and specificity achieved across all IncMB-augmented models in the present study. The performance metrics support these architectural improvements. Models incorporating IncMB consistently demonstrated higher recall and specificity relative to their baseline counterparts, indicating improved sensitivity to pneumonia cases and enhanced discrimination of healthy instances. Such balanced performance is essential for clinical applicability, where both false negatives and false positives carry significant consequences [28,41]. Furthermore, training trajectories showed smooth and consistent convergence, with validation performance closely tracking training accuracy and loss. This behavior suggests an absence of significant overfitting and indicates that the models learned robust, generalizable features. The regularizing effects of Batch Normalization, the smoother optimization facilitated by Mish, and the use of data augmentation collectively contributed to this outcome.

The integration of the IncMB module resulted in more parameter-efficient network architectures across all evaluated models. As summarized in Table 5, each base convolutional neural network exhibited a modest reduction in total parameter count when modified with the IncMB block. For instance, the InceptionV3 model decreased from approximately 22.98 million parameters (87.67 MB) to 22.30 million parameters (85.09 MB). The MobileNetV2 architecture demonstrated a more substantial reduction, from 2.99 million parameters (11.43 MB) to 2.60 million (9.93 MB), representing a decrease of approximately 13%. InceptionResNetV2 and DenseNet201 also showed reductions of about 0.87% and 3.2%, respectively. These reductions can be attributed to the architectural design of the IncMB module, particularly the use of 1×1 bottleneck convolutions and efficient feature compression strategies characteristic of Inception-style modules [55]. Such mechanisms effectively reduce the number of trainable parameters while preserving the network's representational capacity. Importantly, this finding confirms that the improved classification performance observed with IncMB-enhanced models does not stem from increased model capacity or a higher number of trainable parameters. Instead, the gains primarily result from the module's multi-scale feature extraction, grayscale-adaptive processing, and refined feature re-encoding capabilities. Moreover, these gains in compactness were achieved without any degradation in classification performance. On the contrary, accuracy and other performance metrics consistently improved with the IncMB-enhanced models. This demonstrates that the IncMB contributes meaningful architectural efficiency and feature-level adaptability rather than serving as a source of additional trainable complexity. From a deployment perspective, the reduced parameter count implies lower memory requirements and potentially faster inference times, facilitating integration into real-time or resource-constrained clinical environments [67,68]. Additionally, the decreased model complexity may help mitigate overfitting, contributing to improved generalization in unseen data settings.

Table 5. Total parameters for transfer deep learning models across different classification approaches.

Transfer Deep Learning Model	Classification Approach	Total Parameters	Model Size (MB)
InceptionV3	Base model	22,982,561	87.67
	Base model with IncMB	22,304,653	85.09
InceptionResNetV2	Base model	55,221,601	210.65
	Base model with IncMB	54,734,157	208.79

Table 5. *Cont.*

Transfer Deep Learning Model	Classification Approach	Total Parameters	Model Size (MB)
MobileNetV2	Base model	2,995,393	11.43
	Base model with IncMB	2,603,181	9.93
DenseNet201	Base model	19,428,033	74.11
	Base model with IncMB	18,797,741	71.71

The technical enhancements introduced by the IncMB module hold practical relevance for pediatric pneumonia screening. High recall ensures that true pneumonia cases are reliably identified, reducing the risk of missed diagnoses, while high specificity minimizes false positives, preventing unnecessary interventions [69]. This balanced performance supports the use of IncMB-enhanced models as effective triage tools. Additionally, the reduction in model parameters enables deployment on resource-constrained devices, such as portable systems in rural or low-resource settings. These lightweight yet accurate models could facilitate faster, automated chest X-ray interpretation, assisting clinicians in prioritizing cases and improving diagnostic efficiency.

In comparative analysis with prior state-of-the-art approaches, the proposed IncMB-augmented models demonstrate performance that is both competitive and methodologically robust. Barakat et al. [70] evaluated a Quadratic SVM-based framework on a pediatric chest X-ray dataset comprising approximately 5856 images, achieving an accuracy of 0.9758. Although several performance measures were reported, the study emphasized accuracy as the primary indicator of model quality, without providing a deeper per-class breakdown or confusion matrix analysis to elucidate diagnostic trade-offs. Furthermore, no training–validation progression curves were presented, limiting the ability to assess potential overfitting or the stability of convergence. Similarly, Alsharif et al. [71] employed a deep learning–based model on 5852 pediatric X-ray images, attaining an accuracy of 0.9970, sensitivity of 0.9974, and an AUC of 0.9812. While Alsharif et al. reported marginally higher overall accuracy and sensitivity than our best-performing InceptionResNetV2 + IncMB configuration, its lower specificity and the absence of per-class recall metrics or confusion matrix analysis hinder a nuanced evaluation of diagnostic trade-offs [71]. Notably, neither Barakat et al. nor Alsharif et al. presented epoch-wise validation trends, whereas our results explicitly document smooth and stable convergence across training and validation values, with no evidence of significant overfitting [70,71]. These findings underscore that the observed performance gains in our models are not attributable to instability or over-optimization, but rather to the multi-scale, grayscale-adaptive feature enhancement capabilities introduced by the IncMB module. In addition to the aforementioned studies, several recent deep learning methodologies have reported commendable performance in pediatric pneumonia classification, employing diverse architectural and training paradigms. Ha Pham and Tran [72] implemented an ensemble of three pre-trained convolutional neural networks: InceptionResNetV2, DenseNet201, and VGG16, achieving an accuracy of approximately 95% and a 3% enhancement in the F1-score relative to single-model baselines. Their ensemble strategy effectively integrates complementary representational features, whereas the proposed IncMB-augmented InceptionResNetV2 architecture attains superior overall accuracy while preserving an equitable sensitivity–specificity balance, thereby exemplifying an alternative yet methodologically robust design approach. Kareem et al. [73] investigated a federated learning framework encompassing multiple pre-trained CNN architectures (AlexNet, DenseNet, ResNet50, Inception, and VGG19) trained on distributed pediatric chest X-ray datasets. The ResNet-50 model within their study achieved an ac-

curacy of approximately 93%, highlighting both the potential and inherent complexities of decentralized medical data learning. In contrast, the present centralized IncMB-based configuration exhibits marginally higher accuracy under controlled conditions, suggesting that multi-scale feature enrichment could serve as a complementary mechanism in prospective federated or hybrid learning paradigms. Patidar et al. [74] introduced a hybrid ensemble model combining MobileNetV2, ResNet152, and DenseNet169 for pediatric pneumonia detection, achieving 95.03% classification accuracy. Their results demonstrate the effectiveness of ensemble strategies in leveraging complementary network features. In comparison, our IncMB-augmented InceptionResNetV2 model attains 98.12% accuracy with stable convergence and detailed per-class evaluation, reflecting the complementary strengths of ensemble-based and feature-enhancement approaches.

As demonstrated in Figure 5, Grad-CAM visualizations illustrating the attention maps of four convolutional neural network architectures—InceptionV3, InceptionResNetV2, MobileNetV2, and DenseNet201—under two approaches: the base model and the base model augmented with the adaptive IncMB module. The example shown corresponds to a confirmed case of bacterial pneumonia, which on pediatric chest radiographs typically manifests as focal or lobar consolidation with increased opacity, often localized to one lung zone and associated with air bronchograms. In the base configurations, several models, particularly InceptionV3, MobileNetV2, and DenseNet201, exhibit partial or peripheral focus on non-pathological regions, such as the upper thoracic border and areas outside the lung fields. This misdirected attention is especially evident in MobileNetV2, where the heatmap concentrates along the image margins, suggesting suboptimal feature localization. Following the integration of IncMB, all four architectures demonstrate markedly improved focus, with activation maps more precisely covering the lung parenchyma and the region of consolidation, reflecting better alignment with radiological markers of bacterial pneumonia. This shift indicates that the IncMB module enhances the network's ability to attend to clinically relevant structures, reducing reliance on extraneous image features. The refined attention is consistent with the observed improvements in recall, specificity, and overall accuracy reported in Tables 1–3 and supports the hypothesis that IncMB facilitates more discriminative and pathology-centered feature extraction in pediatric pneumonia classification tasks.

As demonstrated in Figure 6, Grad-CAM visualizations for a representative healthy pediatric chest radiograph provide a comparative perspective on the distribution of model attention in non-pathological versus pathological cases. In the base configurations, several architectures display dispersed and irregular activation patterns, with emphasis along the lung apices and peripheral thoracic margins, suggesting nonspecific or contextually irrelevant feature attribution. After the integration of the adaptive IncMB module, the overall activation intensity markedly decreases and becomes confined to anatomically plausible pulmonary regions, with minimal focus on extraneous structures. Although IncMB-based activation maps may appear to encompass broader areas of the lung fields, this behavior reflects the module's multi-scale receptive design, which encourages contextual integration of bilateral and parenchymal patterns rather than a loss of spatial precision. This results in anatomically coherent, context-aware attention that enhances lesion localization in abnormal images (Figure 5) while concurrently suppressing false-positive responses in normal radiographs. Collectively, the comparison between Figures 5 and 6 demonstrates that the IncMB module promotes balanced and semantically meaningful feature attention across distinct clinical categories.

While our findings are promising, a significant limitation of this study is the dataset's scope. The chest X-ray images were sourced exclusively from a single publicly available pediatric dataset collected at one hospital. Although the dataset is extensive, it does

not fully represent the demographic, geographic, and clinical diversity found in real-world medical practice. The lack of variability in imaging equipment, patient populations, and acquisition settings may constrain the model's generalizability to different clinical environments. To improve robustness and applicability, future work should focus on evaluating these models using larger, multi-center datasets that include a wider range of age groups, disease presentations, and imaging protocols.

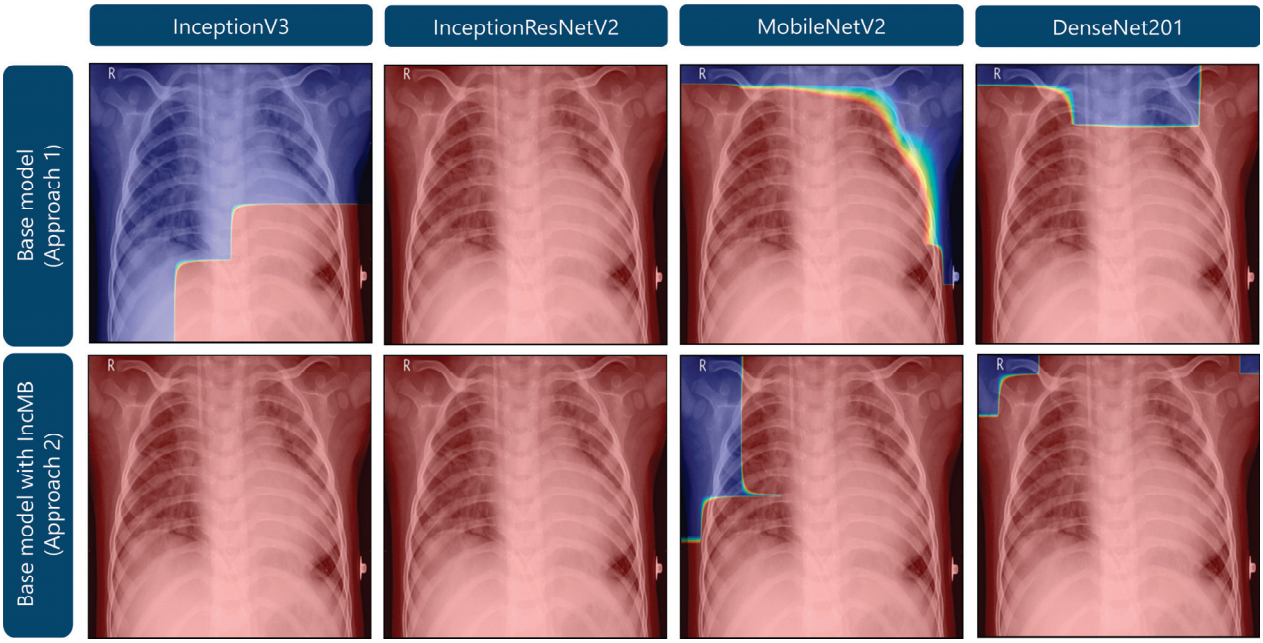


Figure 5. Grad-CAM visualizations highlighting areas of attention in images of bacterial pneumonia.

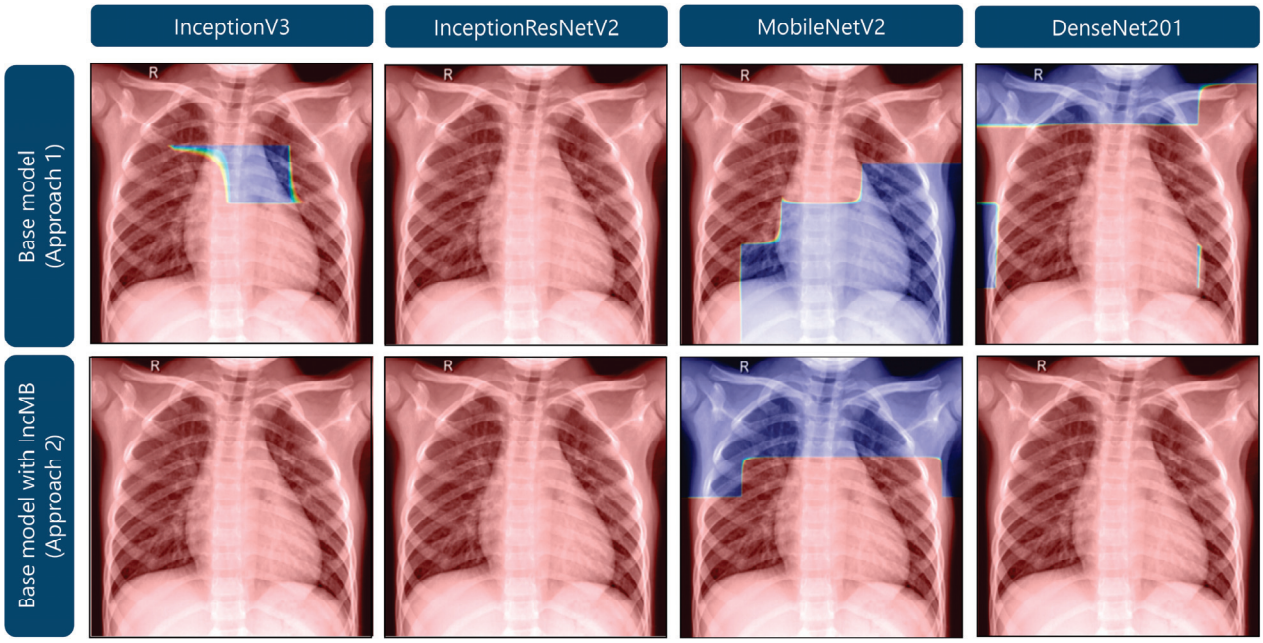


Figure 6. Grad-CAM visualizations highlighting areas of attention in images of healthy lungs.

5. Conclusions

In this study, we proposed an improved pneumonia classification framework based on transfer learning models enhanced with a custom IncMB module, which integrates

Inception blocks, the Mish activation function, and Batch Normalization. Among the evaluated architectures, the InceptionResNetV2 model with IncMB achieved the highest performance, recording an accuracy of 0.9812 and F1-score of 0.9761, outperforming all baseline counterparts. Additionally, the IncMB module consistently improved model generalization while reducing the number of parameters across all tested networks—for instance, lowering InceptionResNetV2's parameter counts by nearly half a million—thus contributing to computational efficiency without compromising predictive accuracy. These advances support the use of this approach in low-resource and point-of-care settings where compact, reliable diagnostic tools are required.

Despite these promising outcomes, several limitations must be acknowledged. The dataset, while sizable, may not fully capture the diversity encountered in global clinical practice, potentially limiting the generalizability of the models across different demographic, geographic, or equipment-specific contexts. In addition, the experimental design employed a single train–validation split without additional repetitions or cross-validation, which may limit the assessment of statistical variability and model stability. Nevertheless, consistent settings, standardized hyperparameters, and extensive data augmentation were applied to enhance reliability and mitigate potential bias.

Future research should focus on validating the proposed models using multi-center datasets that capture broader demographic, geographic, and equipment variability, ensuring robust generalization across different clinical settings. Beyond retrospective evaluation, prospective clinical trials are needed to assess real-world performance, including workflow integration and diagnostic impact in routine practice. Integrating additional imaging modalities, such as lung ultrasound, may further enhance clinical applicability by enabling multimodal diagnosis. Furthermore, integration with clinical decision support systems could facilitate seamless adoption into clinical workflows, improving interpretability and user trust. From a deployment perspective, efforts should include edge computing solutions and mobile platforms equipped with intuitive user interfaces tailored for healthcare professionals, allowing for real-time decision support in both hospital and resource-limited environments.

Author Contributions: Conceptualization, P.R.; methodology, P.R.; software, P.R.; validation, D.R. and G.M.; formal analysis, P.R.; investigation, P.R.; resources, P.R.; data curation, P.R.; writing—original draft preparation, P.R.; writing—review and editing, P.R., D.R. and G.M.; visualization, D.R. and P.R.; supervision, G.M.; project administration, G.M.; funding acquisition, D.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The code developed in this study is available on request from the corresponding author. The open access Chest X-Ray Images (Pneumonia) repository containing images collected from pediatric patients is divided into two categories—Pneumonia/Healthy—and is available at https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia (accessed on 9 October 2025).

Conflicts of Interest: Author Petra Radočaj was employed by the company Layer d.o.o. The remaining author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as potential conflicts of interest.

Appendix A. Detailed Pseudocode for CNN Training and Evaluation with Adapted IncMB Module

Algorithm A1: Training and Evaluation of CNN with Adapted IncMB Module

begin
  Input:
    - train_dir ← Path to training dataset
    - valid_dir ← Path to validation dataset
    - batch_size ← Mini-batch size for training
    - epochs ← Number of training epochs
    - Inception module parameters:
      f1, f2_in, f2_out, f3_in, f3_out, f4_out
  Output:
    - Trained CNN model with IncMB module
    - Classification metrics: Accuracy, Precision, Recall, F1-score, Specificity
    - Confusion matrix and classification report
    - Heatmap visualization of confusion matrix
  Step 1: Configure Environment
    Import required libraries (TensorFlow, Keras, NumPy, etc.);
    Set hyperparameters;
  Step 2: Define Activation Function and Metrics
    Define softplus(x) ← log(1 + exp(x));
    Define tanh(x) ← (exp(x) − exp(−x))/(exp(x) + exp(−x));
    Define mish(x) ← x × tanh(softplus(x));
    Register mish as a custom activation function;
    Define evaluation metrics using standard definitions;
  Step 3: Compute Dataset Properties
    nb_train_samples ← Count of images in train_dir;
    nb_val_samples ← Count of images in valid_dir;
    num_classes ← Count of distinct folders (class labels) in train_dir;
  Step 4: Data Preprocessing and Augmentation
    Define data preprocessing function;
    Initialize data generators with augmentation:
      - Rotation, zoom, shift, horizontal flip;
    Create train_generator and valid_generator with image resizing (224 × 224);
  Step 5: Define Inception Module
    Function Inception_Module (input, f1, f2_in, f2_out, f3_in, f3_out, f4_out):
      branch_1 ← 1 × 1 convolution with f1 filters;
      branch_2 ← 1 × 1 convolution (f2_in) → 3 × 3 convolution (f2_out);
      branch_3 ← 1 × 1 convolution (f3_in) → 5 × 5 convolution (f3_out);
      branch_4 ← 3 × 3 max pooling → 1 × 1 convolution (f4_out);
      return concatenation of all branches;
  Step 6: Construct CNN Model
    Initialize base CNN model with pretrained weights (excluding top layers);
    Apply Inception_Module with specified parameters;
    Apply Mish activation;
    Apply batch normalization;
    Apply global max pooling;
    Add fully connected layer with sigmoid activation for binary classification;

Algorithm A1: *Cont.*

```
        Define model from base input to final output;
    Step 7: Compile CNN Model
        Use optimization algorithm;
        Set loss function as binary cross-entropy;
        Set evaluation metrics;
    Step 8: Train the Model
        Calculate steps_per_epoch = nb_train_samples/batch_size;
        Calculate validation_steps = nb_val_samples/batch_size;
        Initialize learning rate scheduler and training logger;
        Train model with data generators and callbacks;
    Step 9: Save the Trained Model
        Save model architecture and weights to storage;
    Step 10: Evaluate Model Performance
        Generate predictions on validation set;
        Round predicted probabilities to binary labels;
        Calculate confusion matrix and classification report;
        Compute metrics;
    Step 11: Visualize Results
        Normalize confusion matrix values;
        Create labeled matrix for plotting;
        Generate annotated heatmap of confusion matrix;
        Label axes with class names (e.g., 'pneumonia', healthy);
end
```

References

1. Altalhi, A. Review: Pediatric Pneumonia, Etiology, Diagnosis and Management. 2020. Available online: https://www.iajps.com/March-2020/issue_20march_171.php (accessed on 8 October 2025).
2. Crame, E.; Shields, M.D.; McCrossan, P. Paediatric Pneumonia: A Guide to Diagnosis, Investigation and Treatment. *Paediatr. Child Health* **2021**, *31*, 250–257. [CrossRef]
3. Barson, W.J.; Kaplan, S.; Torchia, M. Pneumonia in Children: Epidemiology, Pathogenesis, and Etiology. UpToDate Walth. MAAccessed January 28 2018. 2014. Available online: http://bvndtp.org.vn/wp-content/uploads/2017/01/Pneumonia-in-childre1.doc (accessed on 8 October 2025).
4. de Benedictis, F.M.; Kerem, E.; Chang, A.B.; Colin, A.A.; Zar, H.J.; Bush, A. Complicated Pneumonia in Children. *Lancet* **2020**, *396*, 786–798. [CrossRef]
5. Jutzeler, C.R.; Bourguignon, L.; Weis, C.; Tong, B.; Wong, C.; Rieck, B.; Pargger, H.; Tschudin-Sutter, S.; Egli, A.; Borgwardt, K.; et al. Comorbidities, Clinical Signs and Symptoms, Laboratory Findings, Imaging Features, Treatment Strategies, and Outcomes in Adult and Pediatric Patients with COVID-19: A Systematic Review and Meta-Analysis. *Travel Med. Infect. Dis.* **2020**, *37*, 101825. [CrossRef] [PubMed]
6. Wilkes, C.; Bava, M.; Graham, H.R.; Duke, T. ARI Review group What Are the Risk Factors for Death among Children with Pneumonia in Low- and Middle-Income Countries? A Systematic Review. *J. Glob. Health* **2023**, *13*, 05003. [CrossRef]
7. Wu, Y.; Rocha, B.M.; Kaimakamis, E.; Cheimariotis, G.-A.; Petmezas, G.; Chatzis, E.; Kilintzis, V.; Stefanopoulos, L.; Pessoa, D.; Marques, A.; et al. A Deep Learning Method for Predicting the COVID-19 ICU Patient Outcome Fusing X-Rays, Respiratory Sounds, and ICU Parameters. *Expert Syst. Appl.* **2024**, *235*, 121089. [CrossRef]
8. Pneumonia in Children. Available online: https://www.who.int/news-room/fact-sheets/detail/pneumonia (accessed on 27 June 2025).
9. Frigati, L.; Greybe, L.; Andronikou, S.; Eber, E.; Sunder, B.; Venkatakrishna, S.; Goussard, P. Respiratory Infections in Low and Middle-Income Countries. *Paediatr. Respir. Rev.* **2025**, *54*, 43–51. [CrossRef]

10. Nisar, O.; Nisar, S.; Khattak Haroon Ur Rashid, S.; Ibne Ali Jaffari, S.M.; Haider, Z.; Fatima, F.; Zahra, S.E.; Ijaz, A.H.; Kaneez, M.; Shairwani, G.K. Clinical and Etiological Exploration of Ventilator-Associated Pneumonia in the Intensive Care Unit of a Developing Country. *Cureus* **2023**, *15*, e47515. [CrossRef] [PubMed]
11. Rudan, I.; O'brien, K.L.; Nair, H.; Liu, L.; Theodoratou, E.; Qazi, S.; Lukšić, I.; Walker, C.L.F.; Black, R.E.; Campbell, H. Epidemiology and Etiology of Childhood Pneumonia in 2010: Estimates of Incidence, Severe Morbidity, Mortality, Underlying Risk Factors and Causative Pathogens for 192 Countries. *J. Glob. Health* **2013**, *3*, 010401.
12. Wen, R.; Xu, P.; Cai, Y.; Wang, F.; Li, M.; Zeng, X.; Liu, C. A Deep Learning Model for the Diagnosis and Discrimination of Gram-Positive and Gram-Negative Bacterial Pneumonia for Children Using Chest Radiography Images and Clinical Information. *Infect. Drug Resist.* **2023**, *16*, 4083–4092. [CrossRef]
13. Guitart, C.; Bobillo-Perez, S.; Rodríguez-Fanjul, J.; Carrasco, J.L.; Brotons, P.; López-Ramos, M.G.; Cambra, F.J.; Balaguer, M.; Jordan, I. Lung Ultrasound and Procalcitonin, Improving Antibiotic Management and Avoiding Radiation Exposure in Pediatric Critical Patients with Bacterial Pneumonia: A Randomized Clinical Trial. *Eur. J. Med. Res.* **2024**, *29*, 222. [CrossRef]
14. Biagi, C.; Cavallo, A.; Rocca, A.; Pierantoni, L.; Antonazzo, D.; Dondi, A.; Gabrielli, L.; Lazzarotto, T.; Lanari, M. Pulmonary and Extrapulmonary Manifestations in Hospitalized Children with Mycoplasma Pneumoniae Infection. *Microorganisms* **2021**, *9*, 2553. [CrossRef]
15. Geanacopoulos, A.T.; Lipsett, S.C.; Hirsch, A.W.; Monuteaux, M.C.; Neuman, M. Impact of Viral Radiographic Features on Antibiotic Treatment for Pediatric Pneumonia. *J. Pediatr. Infect. Dis. Soc.* **2022**, *11*, 207–213. [CrossRef] [PubMed]
16. Wang, G.; Liu, X.; Shen, J.; Wang, C.; Li, Z.; Ye, L.; Wu, X.; Chen, T.; Wang, K.; Zhang, X.; et al. A Deep-Learning Pipeline for the Diagnosis and Discrimination of Viral, Non-Viral and COVID-19 Pneumonia from Chest X-Ray Images. *Nat. Biomed. Eng.* **2021**, *5*, 509–521. [CrossRef] [PubMed]
17. Al Nufaiei, Z.F.; Alshamrani, K.M. Comparing Ultrasound, Chest X-Ray, and CT Scan for Pneumonia Detection. *Med. Devices* **2025**, *18*, 149–159. [CrossRef]
18. Bayhan, G.İ.; Gülleroğlu, N.B.; Çetin, S.; Erat, T.; Yıldız, S.; Özen, S.; Konca, H.K.; Yahşi, A.; Dinç, B. Radiographic Findings of Adenoviral Pneumonia in Children. *Clin. Imaging* **2024**, *108*, 110111. [CrossRef]
19. Fancourt, N.; Deloria Knoll, M.; Barger-Kamate, B.; de Campo, J.; de Campo, M.; Diallo, M.; Ebruke, B.E.; Feikin, D.R.; Gleeson, F.; Gong, W.; et al. Standardized Interpretation of Chest Radiographs in Cases of Pediatric Pneumonia From the PERCH Study. *Clin. Infect. Dis.* **2017**, *64*, S253–S261. [CrossRef]
20. Balk, D.S.; Lee, C.; Schafer, J.; Welwarth, J.; Hardin, J.; Novack, V.; Yarza, S.; Hoffmann, B. Lung Ultrasound Compared to Chest X-Ray for Diagnosis of Pediatric Pneumonia: A Meta-Analysis. *Pediatr. Pulmonol.* **2018**, *53*, 1130–1139. [CrossRef]
21. Pereda, M.A.; Chavez, M.A.; Hooper-Miele, C.C.; Gilman, R.H.; Steinhoff, M.C.; Ellington, L.E.; Gross, M.; Price, C.; Tielsch, J.M.; Checkley, W. Lung Ultrasound for the Diagnosis of Pneumonia in Children: A Meta-Analysis. *Pediatrics* **2015**, *135*, 714–722. [CrossRef] [PubMed]
22. Pagano, A.; Numis, F.G.; Visone, G.; Pirozzi, C.; Masarone, M.; Olibet, M.; Nasti, R.; Schiraldi, F.; Paladino, F. Lung Ultrasound for Diagnosis of Pneumonia in Emergency Department. *Intern. Emerg. Med.* **2015**, *10*, 851–854. [CrossRef]
23. Delijani, K.; Price, M.C.; Little, B.P. Community and Hospital Acquired Pneumonia. *Semin. Roentgenol.* **2022**, *57*, 3–17. [CrossRef]
24. Lafraxo, S.; El Ansari, M.; Koutti, L. A New Hybrid Approach for Pneumonia Detection Using Chest X-Rays Based on ACNN-LSTM and Attention Mechanism. *Multimed. Tools Appl.* **2024**, *83*, 73055–73077. [CrossRef]
25. Dzhaynakbaev, N.; Kurmanbekkyzy, N.; Baimakhanova, A.; Mussatayeva, I. 2D-CNN Architecture for Accurate Classification of COVID-19 Related Pneumonia on X-Ray Images. *Int. J. Adv. Comput. Sci. Appl. IJACSA* **2024**, *15*, 905–917. [CrossRef]
26. Radočaj, P.; Radočaj, D.; Martinović, G. Pediatric Pneumonia Recognition Using an Improved DenseNet201 Model with Multi-Scale Convolutions and Mish Activation Function. *Algorithms* **2025**, *18*, 98. [CrossRef]
27. AlGhamdi, A.S. Efficient Deep Learning Approach for the Classification of Pneumonia in Infants from Chest X-Ray Images. *Trait. Signal* **2024**, *41*, 1245–1262. [CrossRef]
28. Radočaj, P.; Martinović, G. Interpretable Deep Learning for Pediatric Pneumonia Diagnosis Through Multi-Phase Feature Learning and Activation Patterns. *Electronics* **2025**, *14*, 1899. [CrossRef]
29. Pan, Z.; Wang, H.; Wan, J.; Zhang, L.; Huang, J.; Shen, Y. Efficient Federated Learning for Pediatric Pneumonia on Chest X-Ray Classification. *Sci. Rep.* **2024**, *14*, 23272. [CrossRef]
30. Stephen, O.; Sain, M.; Maduh, U.J.; Jeong, D.-U. An Efficient Deep Learning Approach to Pneumonia Classification in Healthcare. *J. Healthc. Eng.* **2019**, *2019*, 4180949. [CrossRef]
31. Liu, W.; Hu, J.; Lv, F.; Tang, Z. A New Method for Long-Term Temperature Compensation of Structural Health Monitoring by Ultrasonic Guided Wave. *Measurement* **2025**, *252*, 117310. [CrossRef]
32. De Oliveira, A.P.; Tadeu Braga, H.F. Artificial Intelligence: Learning and Limitations. *WSEAS Trans. Adv. Eng. Educ.* **2020**, *17*, 80–86. [CrossRef]

33. Chen, Q.; Chen, L.; Nie, W.; Li, X.; Zheng, J.; Zhong, J.; Wei, Y.; Zhang, Y.; Ji, R. A Mixed-Scale Dynamic Attention Transformer for Pediatric Pneumonia Diagnosis. *Displays* **2025**, *87*, 102953. [CrossRef]
34. Morcos, G.; Yi, P.H.; Jeudy, J. Applying Artificial Intelligence to Pediatric Chest Imaging: Reliability of Leveraging Adult-Based Artificial Intelligence Models. *J. Am. Coll. Radiol. JACR* **2023**, *20*, 742–747. [CrossRef] [PubMed]
35. Hung, C.-L.; Hsin, C.; Wang, H.-H.; Tang, C.Y. Optimization of GPU Memory Usage for Training Deep Neural Networks. In *Pervasive Systems, Algorithms and Networks*; Esposito, C., Hong, J., Choo, K.-K.R., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 289–293.
36. Liang, G.; Zheng, L. A Transfer Learning Method with Deep Residual Network for Pediatric Pneumonia Diagnosis. *Comput. Methods Programs Biomed.* **2020**, *187*, 104964. [CrossRef]
37. Luján-García, J.E.; Yáñez-Márquez, C.; Villuendas-Rey, Y.; Camacho-Nieto, O. A Transfer Learning Method for Pneumonia Classification and Visualization. *Appl. Sci.* **2020**, *10*, 2908. [CrossRef]
38. Burns, J.L.; Zaiman, Z.; Vanschaik, J.; Luo, G.; Peng, L.; Price, B.; Mathias, G.; Mittal, V.; Sagane, A.; Tignanelli, C.; et al. Ability of Artificial Intelligence to Identify Self-Reported Race in Chest X-Ray Using Pixel Intensity Counts. *J. Med. Imaging Bellingham Wash* **2023**, *10*, 061106. [CrossRef]
39. Lee, P.; Tahmasebi, A.; Dave, J.K.; Parekh, M.R.; Kumaran, M.; Wang, S.; Eisenbrey, J.R.; Donuru, A. Comparison of Gray-Scale Inversion to Improve Detection of Pulmonary Nodules on Chest X-Rays Between Radiologists and a Deep Convolutional Neural Network. *Curr. Probl. Diagn. Radiol.* **2023**, *52*, 180–186. [CrossRef]
40. Bhatt, D.; Patel, C.; Talsania, H.; Patel, J.; Vaghela, R.; Pandya, S.; Modi, K.; Ghayvat, H. CNN Variants for Computer Vision: History, Architecture, Application, Challenges and Future Scope. *Electronics* **2021**, *10*, 2470. [CrossRef]
41. Neuman, M.I.; Lee, E.Y.; Bixby, S.; Diperna, S.; Hellinger, J.; Markowitz, R.; Servaes, S.; Monuteaux, M.C.; Shah, S.S. Variability in the Interpretation of Chest Radiographs for the Diagnosis of Pneumonia in Children. *J. Hosp. Med.* **2012**, *7*, 294–298. [CrossRef]
42. An, J.; Kpeyiton, K.G.; Shi, Q. Grayscale Images Colorization with Convolutional Neural Networks. *Soft Comput.* **2020**, *24*, 4751–4758. [CrossRef]
43. Sotirov, S.; Orozova, D.; Angelov, B.; Sotirova, E.; Vylcheva, M. Transforming Pediatric Healthcare with Generative AI: A Hybrid CNN Approach for Pneumonia Detection. *Electronics* **2025**, *14*, 1878. [CrossRef]
44. Prakash, J.A.; Asswin, C.R.; Kumar, K.S.D.; Dora, A.; Ravi, V.; Sowmya, V.; Gopalakrishnan, E.A.; Soman, K.P. Transfer Learning Approach for Pediatric Pneumonia Diagnosis Using Channel Attention Deep CNN Architectures. *Eng. Appl. Artif. Intell.* **2023**, *123*, 106416. [CrossRef]
45. Lan, X.; Zhang, Y.; Yuan, W.; Shi, F.; Guo, W. Image-Based Deep Learning in Diagnosing Mycoplasma Pneumonia on Pediatric Chest X-Rays. *BMC Pediatr.* **2024**, *24*, 720. [CrossRef] [PubMed]
46. Khan, E.; Rehman, M.Z.U.; Ahmed, F.; Alfouzan, F.A.; Alzahrani, N.M.; Ahmad, J. Chest X-Ray Classification for the Detection of COVID-19 Using Deep Learning Techniques. *Sensors* **2022**, *22*, 1211. [CrossRef]
47. Wang, K.; Jiang, P.; Meng, J.; Jiang, X. Attention-Based DenseNet for Pneumonia Classification. *IRBM* **2022**, *43*, 479–485. [CrossRef]
48. Napravnik, M.; Hržić, F.; Urschler, M.; Miletić, D.; Štajduhar, I. Lessons Learned from RadiologyNET Foundation Models for Transfer Learning in Medical Radiology. *Sci. Rep.* **2025**, *15*, 21622. [CrossRef]
49. Mei, X.; Liu, Z.; Robson, P.M.; Marinelli, B.; Huang, M.; Doshi, A.; Jacobi, A.; Cao, C.; Link, K.E.; Yang, T.; et al. RadImageNet: An Open Radiologic Deep Learning Research Dataset for Effective Transfer Learning. *Radiol. Artif. Intell.* **2022**, *4*, e210315. [CrossRef]
50. Woerner, S.; Baumgartner, C.F. Navigating Data Scarcity Using Foundation Models: A Benchmark of Few-Shot and Zero-Shot Learning Approaches in Medical Imaging. In *Foundation Models for General Medical AI*; Deng, Z., Shen, Y., Kim, H.J., Jeong, W.-K., Aviles-Rivero, A.I., He, J., Zhang, S., Eds.; Springer Nature Switzerland: Cham, Switzerland, 2025; pp. 30–39.
51. Radočaj, P.; Radočaj, D.; Martinović, G. Image-Based Leaf Disease Recognition Using Transfer Deep Learning with a Novel Versatile Optimization Module. *Big Data Cogn. Comput.* **2024**, *8*, 52. [CrossRef]
52. Kermany, D.S.; Goldbaum, M.; Cai, W.; Valentim, C.C.S.; Liang, H.; Baxter, S.L.; McKeown, A.; Yang, G.; Wu, X.; Yan, F.; et al. Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning. *Cell* **2018**, *172*, 1122–1131. [CrossRef] [PubMed]
53. Team, K. Keras Documentation: Keras 3 API Documentation. Available online: https://keras.io/api/ (accessed on 29 June 2025).
54. Module: Tf | TensorFlow v2.16.1. Available online: https://www.tensorflow.org/api_docs/python/tf (accessed on 29 June 2025).
55. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going Deeper with Convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
56. McNeely-White, D.; Beveridge, J.R.; Draper, B.A. Inception and ResNet Features Are (Almost) Equivalent. *Cogn. Syst. Res.* **2020**, *59*, 312–318. [CrossRef]

57. Misra, D. Mish: A Self Regularized Non-Monotonic Activation Function. Misra, Diganta. In Proceedings of the British Machine Vision Conference 2020, Online, 7–10 September 2020.
58. Radočaj, P.; Radočaj, D.; Martinović, G. Optimizing Convolutional Neural Network Architectures with Optimal Activation Functions for Pediatric Pneumonia Diagnosis Using Chest X-Rays. *Big Data Cogn. Comput.* **2025**, *9*, 25. [CrossRef]
59. Batch Normalization | Proceedings of the 32nd International Conference on International Conference on Machine Learning—Volume 37. Available online: https://dl.acm.org/doi/10.5555/3045118.3045167 (accessed on 30 June 2025).
60. Diallo, R.; Edalo, C.; Awe, O.O. Machine Learning Evaluation of Imbalanced Health Data: A Comparative Analysis of Balanced Accuracy, MCC, and F1 Score. In *Practical Statistical Learning and Data Science Methods: Case Studies from LISA 2020 Global Network, USA*; Awe, O.O., Vance, E.A., Eds.; Springer Nature Switzerland: Cham, Switzerland, 2025; pp. 283–312, ISBN 978-3-031-72215-8.
61. Naidu, G.; Zuva, T.; Sibanda, E.M. A Review of Evaluation Metrics in Machine Learning Algorithms. In *Artificial Intelligence Application in Networks and Systems*; Silhavy, R., Silhavy, P., Eds.; Springer International Publishing: Cham, Switzerland, 2023; pp. 15–25.
62. Srivastava, R.; Pandey, D.R.; Singh, A.K. Comparative Analysis of Radiological and Machine Learning-Based Interpretations for Differentiating COVID-19 and Pneumonia. *Recent Adv. Electr. Electron. Eng.* **2025**, *18*, 850–861. [CrossRef]
63. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning | Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence. Available online: https://dl.acm.org/doi/10.5555/3298023.3298188 (accessed on 30 June 2025).
64. Lin, B.; Su, H.; Li, D.; Feng, A.; Li, H.; Li, J.; Jiang, K.; Jiang, H.; Gong, X.; Liu, T. PlaneNet: An Efficient Local Feature Extraction Network. *PeerJ Comput. Sci.* **2021**, *7*, e783. [CrossRef]
65. Segu, M.; Tonioni, A.; Tombari, F. Batch Normalization Embeddings for Deep Domain Generalization. *Pattern Recognit.* **2023**, *135*, 109115. [CrossRef]
66. Understanding Batch Normalization | Proceedings of the 32nd International Conference on Neural Information Processing Systems. Available online: https://dl.acm.org/doi/10.5555/3327757.3327868 (accessed on 30 June 2025).
67. Merkow, J.; Soin, A.; Long, J.; Cohen, J.P.; Saligrama, S.; Bridge, C.; Yang, X.; Kaiser, S.; Borg, S.; Tarapov, I.; et al. CheXstray: A Real-Time Multi-Modal Monitoring Workflow for Medical Imaging AI. In *Medical Image Computing and Computer Assisted Intervention—MICCAI 2023*; Greenspan, H., Madabhushi, A., Mousavi, P., Salcudean, S., Duncan, J., Syeda-Mahmood, T., Taylor, R., Eds.; Springer Nature Switzerland: Cham, Switzerland, 2023; pp. 326–336.
68. Characterizing Parameter Scaling with Quantization for Deployment of CNNs on Real-Time Systems | ACM Transactions on Embedded Computing Systems. Available online: https://dl.acm.org/doi/full/10.1145/3654799 (accessed on 30 June 2025).
69. Singh, M.P.; Singh, J.; Ravi, V.; Gupta, A.D.; Alahmadi, T.J.; Shivahare, B.D.; Diwakar, M.; Tayal, M.; Singh, P. A Healthcare System Employing Lightweight CNN for Disease Prediction with Artificial Intelligence. *Open Public Health J.* **2024**, *17*, e18749445302023. [CrossRef]
70. Barakat, N.; Awad, M.; Abu-Nabah, B.A. A Machine Learning Approach on Chest X-Rays for Pediatric Pneumonia Detection. *Digit. Health* **2023**, *9*, 20552076231180008. [CrossRef] [PubMed]
71. Alsharif, R.; Al-Issa, Y.; Alqudah, A.M.; Qasmieh, I.A.; Mustafa, W.A.; Alquran, H. PneumoniaNet: Automated Detection and Classification of Pediatric Pneumonia Using Chest X-Ray Images and CNN Approach. *Electronics* **2021**, *10*, 2949. [CrossRef]
72. Pham, N.H.; Tran, G.S. Apply a Cnn-Based Ensemble Model to Chest-x Ray Image-Based Pneumonia Classification. *J. Adv. Inf. Technol.* **2024**, *15*, 1205–1214. [CrossRef]
73. Kareem, A.; Liu, H.; Velisavljevic, V. A Federated Learning Framework for Pneumonia Image Detection Using Distributed Data. *Healthc. Anal.* **2023**, *4*, 100204. [CrossRef]
74. Patidar, M.; Pandey, G.; Koolagudi, S.G.; S, K.K.; Chandra, V. Enhancing Paediatric Healthcare: Deep Learning-Based Pneumonia Diagnosis from Children's Chest X-Rays. In Proceedings of the 2024 Sixteenth International Conference on Contemporary Computing, Noida, India, 8–10 August 2024; Association for Computing Machinery: New York, NY, USA, 2024; pp. 128–135.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

applied sciences

MDPI

Article

Deep Learning Based Correction Algorithms for 3D Medical Reconstruction in Computed Tomography and Macroscopic Imaging

Tomasz Les [1], Tomasz Markiewicz [1,2], Malgorzata Lorent [2], Miroslaw Dziekiewicz [2] and Krzysztof Siwek [1,*]

[1] Faculty of Electrical Engineering, Warsaw University of Technology, Plac Politechniki 1, 00-661 Warsaw, Poland; tomasz.les@pw.edu.pl (T.L.); tomasz.markiewicz@pw.edu.pl (T.M.)
[2] Department of Vascular and Endovascular Surgery, Military Institute of Medicine-National Research Institute, Szaserów 128, 04-141 Warsaw, Poland; mlorent@wim.mil.pl (M.L.); dziekiewicz@wim.mil.pl (M.D.)
* Correspondence: krzysztof.siwek@pw.edu.pl

Abstract

This paper introduces a hybrid two-stage registration framework for reconstructing three-dimensional (3D) kidney anatomy from macroscopic slices, using CT-derived models as the geometric reference standard. The approach addresses the data-scarcity and high-distortion challenges typical of macroscopic imaging, where fully learning-based registration (e.g., VoxelMorph) often fails to generalize due to limited training diversity and large nonrigid deformations that exceed the capture range of unconstrained convolutional filters. In the proposed pipeline, the Optimal Cross-section Matching (OCM) algorithm first performs constrained global alignment—translation, rotation, and uniform scaling—to establish anatomically consistent slice initialization. Next, a lightweight deep-learning refinement network, inspired by VoxelMorph, predicts residual local deformations between consecutive slices. The core novelty of this architecture lies in its hierarchical decomposition of the registration manifold: the OCM acts as a deterministic geometric anchor that neutralizes high-amplitude variance, thereby constraining the learning task to a low-dimensional residual manifold. This hybrid OCM + DL design integrates explicit geometric priors with the flexible learning capacity of neural networks, ensuring stable optimization and plausible deformation fields even with few training examples. Experiments on an original dataset of 40 kidneys demonstrated that the OCM + DL method achieved the highest registration accuracy across all evaluated metrics: NCC = 0.91, SSIM = 0.81, Dice = 0.90, IoU = 0.81, HD95 = 1.9 mm, and volumetric agreement DC_{Vol} = 0.89. Compared to single-stage baselines, this represents an average improvement of approximately 17% over DL-only and 14% over OCM-only, validating the synergistic contribution of the proposed hybrid strategy over standalone iterative or data-driven methods. The pipeline maintains physical calibration via Hough-based grid detection and employs Bézier-based contour smoothing for robust meshing and volume estimation. Although validated on kidney data, the proposed framework generalizes to other soft-tissue organs reconstructed from optical or photographic cross-sections. By decoupling interpretable global optimization from data-efficient deep refinement, the method advances the precision, reproducibility, and anatomical realism of multimodal 3D reconstructions for surgical planning, morphological assessment, and medical education.

Keywords: deep learning; 3D reconstruction; computed tomography; macro imaging; kidney model correction; hough transformation; Bézier curves; Cross-section Matching

1. Introduction

In recent decades, advances in the field of medical imaging have significantly contributed to the development of diagnostics and treatment of kidney diseases. Technologies such as computed tomography (CT) and macroscopic imaging have become indispensable tools in identifying and analyzing kidney pathologies. However, despite the advanced capabilities of these methods, challenges related to geometric differences between 3D models obtained from various imaging sources exist. These differences can significantly impact the accuracy of diagnostics, surgical treatment planning, and medical education.

The aim of this study is to develop and evaluate geometric correction methods for 3D kidney models reconstructed from macroscopic imaging, using CT-based models as the reference standard.

In this work, computed tomography (CT) data is used as the anatomical reference standard, while the proposed correction methods are applied exclusively to 3D kidney models reconstructed from macroscopic imaging. The goal is to improve the accuracy and anatomical consistency of macroscopic models by aligning them with CT-based reconstructions.

In particular, 3D kidney models obtained via computed tomography offer high accuracy and detail thanks to the use of advanced image reconstruction algorithms. They allow for precise representation of the anatomical and pathological structure of organs. On the other hand, macroscopic models, created based on a series of photographic images of organ sections, can be less accurate due to potential tissue deformations, errors in matching sections, and physiological shrinkage of the extracted organs—for example, due to water loss. The combined use of translation, rotation, and scaling allows effective 3D alignment, compensating for their global effects and improving anatomical consistency. These differences can lead to errors in the interpretation of imaging data, directly affecting the quality of diagnostics and planning of surgical interventions.

In response to the geometric inconsistencies observed between both modalities, this article presents a developed method for the geometric correction of 3D kidney models obtained from macroscopic imaging. This method aims to minimize the differences between models obtained from different imaging techniques, through precise matching and geometric correction of macroscopic models.

By applying advanced image processing techniques and optimization algorithms, it is possible to obtain macroscopic models with significantly better accuracy and anatomical fidelity. Such an approach opens up new possibilities in the more accurate modeling of anatomical structures, which is crucial not only in diagnostics and surgery but also in scientific research and medical education.

This work introduces a hybrid OCM + DL framework that decomposes registration into deterministic global alignment and learned residual refinement. This strategy ensures high accuracy and data efficiency on small clinical datasets ($N = 40$), bridging the gap between macroscopic photography and CT standards while maintaining physical metric consistency.

2. Problem Statement

The primary issue addressed in this article concerns the discrepancy in geometric accuracy between 3D kidney models obtained from computed tomography (CT) and those derived from macroscopic imaging. These differences can lead to inconsistencies in the interpretation of imaging data, which directly impact the effectiveness of diagnostics, surgical planning, and medical education in the field of kidney anatomy. This issue is particularly significant in the context of precisely modeling pathological changes, such as kidney tumors, where the accuracy of the model is crucial for treatment success [1–6].

https://doi.org/10.3390/app16041954

The challenge lies in developing a method capable of effectively correcting these differences, enabling the creation of accurate and anatomically faithful 3D kidney models based on macroscopic data. Such a method must be able to address potential tissue deformations and errors in matching sections, which are typical for the process of creating macroscopic models. This solution must also be sufficiently flexible to be applied to modeling different types and sizes of kidneys, as well as varying degrees of pathology progression [7,8].

In the context of this challenge, we present a CT image of a kidney and a macroscopic scale image of a kidney in Figure 1. Due to differences in scale, sample preparation methods, and the inherent nature of both imaging techniques, these sections can vary significantly.

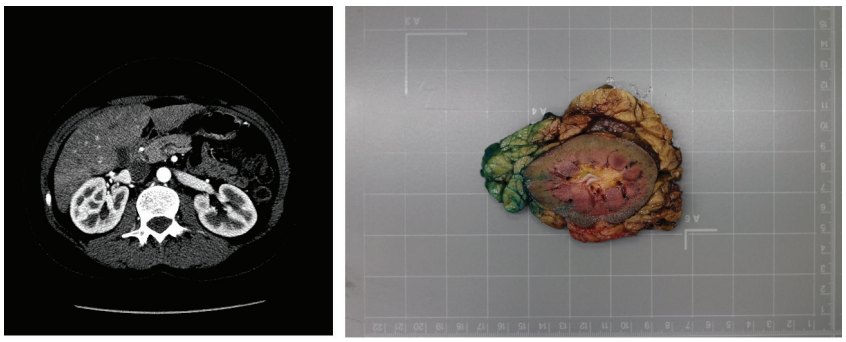


Figure 1. On the (**left**), a CT scan section of a kidney is shown, displaying the organ's detailed internal structure. On the (**right**), a macroscopic image of a kidney is presented, illustrating its external appearance and shape, providing information on the organ's surface condition.

These differences are not limited to visual aspects but also have a profound impact on the process of creating 3D models. In particular, when constructing 3D kidney models from both CT and macroscopic data, the differences can amount to up to 30% in volume and in specific dimensions, such as length, width, and thickness. Recent comparative studies confirm that macroscopic observation and digital 3D reconstructions (e.g., CT or surface scans) often yield inconsistent measurements and interpretations due to imaging artifacts, handling differences [9,10]. Such discrepancies pose a significant challenge to the accuracy of anatomical modeling, directly affecting the reliability of the models used in medicine, which underscores the importance of developing an effective method for correcting these differences.

Previous approaches to macroscopic imaging quantification have typically relied on manual or semi-manual scaling using visible reference markers such as rulers or calibration grids placed in the photographic background. These methods often involve estimating pixel-to-millimeter conversion factors by hand, which introduces variability and is sensitive to camera angle, lighting, and slice deformation. Despite their simplicity, such techniques remain widespread due to the lack of standardized, automated tools. Our approach enhances this process by introducing automated detection of scale features using the Hough Transform and by integrating slice-wise geometric alignment to improve consistency and accuracy across reconstructed 3D volumes.

Recent advancements in computational methods and deep learning algorithms have catalyzed a significant evolution in the realm of 3D reconstruction from computed tomography (CT) scans. Contrary to earlier limitations, contemporary research highlights the integration of sophisticated image processing techniques and machine learning models, marking a pivotal enhancement in the precision of anatomical modeling. There are numerous studies on automated CT organ detection systems, which exemplify the integration of advanced computational techniques to improve the accuracy and efficiency of medical diagnostics [11–15]. This trend is reinforced by comprehensive reviews summarizing the state

https://doi.org/10.3390/app16041954

of the art in 3D medical reconstructions, including both classical and deep learning-based methods [16].

The development of CNN-based segmentation for cardiac micro-CT data, for instance, exemplifies the push towards utilizing deep learning for enhancing the granularity and accuracy of 3D reconstructions, underscoring the method's capability in high-throughput phenotyping applications [17]. Similarly, innovations in scaffolding methods for cartilage restoration further illustrate the multifaceted applications of 3D CT reconstructions in regenerative medicine, emphasizing its role in fabricating conducive chondrogenic environments [18].

A critical point in recent research is the correction of geometric artifacts—especially relevant in symmetric organ structures—where sophisticated geometric modeling approaches have been proposed for improving alignment between cross-sectional and full volumetric representations [19]. Additionally, learning-based frameworks that incorporate geometric constraints are being explored to optimize model consistency and anatomical fidelity [20,21].

In contrast to the well-explored domain of CT-based reconstruction, research into macroscopic imaging reconstructions remains under-represented. Despite the critical insights offered by macroscopic imaging, particularly in surface detail and texture analysis of organs, its potential is yet to be fully leveraged within anatomical modeling [22]. The manual delineation of organs, a gold standard in micro-computed tomography (μCT) data processing, highlights the challenges of time-intensive, error-prone methodologies that lack scalability for extensive data analysis [22]. Automated segmentation solutions, crucial for processing the extensive field-of-view in CT images, have seen advancements through multi-atlas segmentation (MAS) strategies. However, these often fail to preserve spatial information among adjacent organs, leading to segmented models that may lack coherence [23].

Addressing these challenges, recent studies propose innovative approaches for both manual and automated segmentation techniques. For instance, the development of organ-specific insert phantoms through 3D printing technology based on volumetric CT image datasets offers a novel pathway for simulating anatomical features in CT images, providing a cost-effective alternative for organ-specific studies [6]. Furthermore, the integration of the Internet of Health Things (IoHT) with deep learning for 3D reconstruction presents a forward-thinking solution to obtain high-accuracy diagnosis results, showcasing the potential for super-resolution CT (SRCT) images through advanced computational models [7].

Other emerging platforms, such as Nextmed, enable automatic segmentation and visualization of 3D models from imaging data, facilitating reproducible and streamlined workflows in clinical environments [24]. At the same time, methods utilizing residual encoder–decoder architectures and directional wavelet transforms have proven effective in reducing noise in low-dose CT reconstructions [25,26], while deep learning continues to push the frontier of tomographic reconstruction with frameworks designed for image fidelity, generalization, and inference speed [27–30]. Finally, developments in real-time 3D endoscopic reconstruction methods provide additional pathways for extending anatomical modeling into intraoperative and minimally invasive settings.

These advancements underline the ongoing need to refine and enhance anatomical modeling and segmentation techniques, emphasizing the critical role of precision and efficiency in medical imaging and diagnostics. The synergy between CT and macroscopic imaging, supported by cutting-edge computational methodologies, offers a compelling direction for future research in kidney anatomy modeling and beyond.

The proposed strategy of coupling deterministic optimization with deep learning also aligns with a broader emerging paradigm in computational modeling. Recent studies have

https://doi.org/10.3390/app16041954

demonstrated that hybridizing meta-heuristic or geometric optimization with learning-based ensembles can significantly enhance predictive stability and accuracy in complex structural modeling tasks [31]. By leveraging optimization to constrain the learning manifold, these hybrid frameworks overcome the limitations of standalone models, much like our OCM + DL approach for anatomical reconstruction.

Unlike standard DL-only registration, our framework addresses the physical discontinuities and domain gaps inherent in macroscopic imaging. By combining OCM for global topological consistency with DL for local flexibility, we provide a robust solution for cases where fully learned models typically fail to generalize.

Our work integrates slice-wise registration and cross-modal alignment. 3D reconstruction from physical sections is challenged by tissue deformation and "z-axis" discontinuities, similar to issues in gross pathology. Aligning 2D photography with 3D CT further requires bridging significant domain gaps. Our hybrid framework addresses these by combining deterministic geometric anchors with learned residual refinements for multimodal consistency.

3. Methods

The proposed methodology employs a two-stage hybrid approach integrating global geometric optimization with local neural refinement. The framework decomposes the registration process to bridge the gap between macroscopic imaging and CT standards, ensuring high data efficiency on small clinical datasets. The overall architecture of the proposed framework is illustrated in Figure 2.

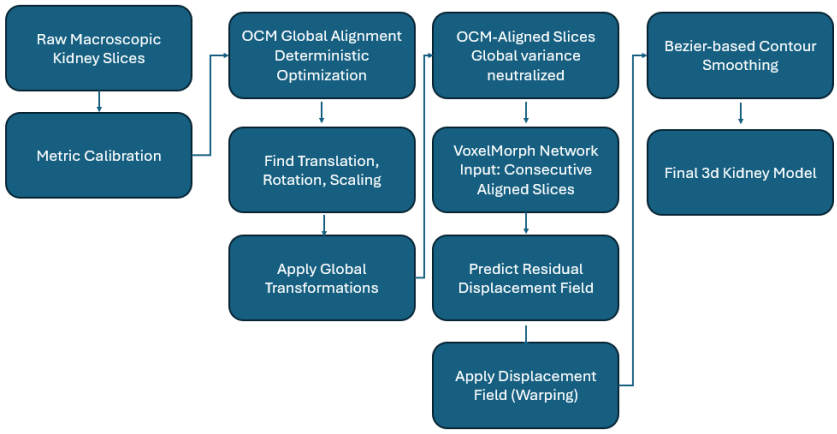


Figure 2. Schematic representation of the proposed hybrid OCM + DL framework. The pipeline illustrates the progression from raw macroscopic slices and CT reference data through the deterministic Optimal Cross-section Matching (OCM) and the subsequent deep learning (DL) refinement stage to achieve the final 3D reconstruction.

In the first stage, the Optimal Cross-section Matching (OCM) algorithm establishes a global coordinate system by addressing:

- Metric Inconsistencies: Utilizing the Hough Transform for consistent grid-based calibration.
- Rigid Displacements: Determining optimal translation and rotation to neutralize variance from physical handling.
- Global Scaling: Adjusting for size discrepancies via constrained similarity transforms.

The subsequent deep learning (DL) refinement stage addresses residual non-linearities through:

https://doi.org/10.3390/app16041954

- Residual Displacement Prediction: Using a VoxelMorph-based architecture to predict dense displacement fields.
- Apply Displacement Field (Warping): Physically shifting pixels to compensate for local tissue distortions and physiological shrinkage.
- Data Efficiency: Training on pre-aligned data to focus on fine-grained adjustments and enhance stability.

The process concludes with Bezier-based contour smoothing to improve 3D mesh continuity. This strategy ensures high anatomical fidelity to the CT reference while minimizing volumetric and dimensional discrepancies.

3.1. Data Used in the Study

The data from 40 patients diagnosed with kidney diseases, including kidney tumors, were used in this study. For each patient, CT images of the kidneys were available, from which a total of 2157 scans were obtained. The number of CT scans per patient ranged from 44 to 65, with an average number of scans per patient being 54.

Additionally, a set of photographs of the excised kidney was collected for each patient. The number of these photographs also varied from 11 to 20 for each patient, with an average number of photos being 16.

The macroscopic images depicting the excised kidneys were taken immediately after surgery, where each kidney was sliced into 1 cm thick sections and then photographed. The cutting was performed using a calibrated slicing guide with 10 mm slot spacing to ensure consistent section thickness. Additionally, the thickness of each slice was verified post-sectioning using a digital caliper.

The data were collected at the Military Institute of Medicine-National Research Institute in Warsaw, which specializes in the treatment of kidney tumors. This center is equipped with modern CT imaging equipment and experienced staff, ensuring high quality and consistency of the collected data.

Characteristics of kidney tumors in patients: The study included patients diagnosed with various types of kidney tumors or other kidney diseases. Data analysis revealed the following characteristics: Most cases were clear cell renal carcinoma, which is consistent with the general distribution of this type of tumor. The classification of cancers according to the WHO/ISUP (International Society of Urological Pathology) enables determining the tumor's grade of advancement. Tumors varied in size, indicating their diverse stages of development at the time of diagnosis. For instance, the dimensions of one of the kidneys were $12.5 \times 8.5 \times 7.5$ cm, while others had smaller dimensions, such as $11 \times 4.5 \times 7$ cm. Tumors were located in both the right and left kidneys. The provided histopathological descriptions offer information on the tumor's stage of advancement. For example, one description indicates clear cell renal carcinoma, WHO/ISUP G-4, with infiltration of the renal vein wall and perinephric fat, denoting a high degree of malignancy. These characteristics underscore the diversity of kidney tumor cases in the study group. This diversity significantly impacts the treatment process, as well as the accuracy and efficiency of using CT imaging and macroscopic analysis in diagnostics and surgical planning. The tumor descriptions and classifications are detailed and based on actual clinical data.

Quality and parameters of CT scans: The images used in the study, recorded in DICOM standard version 3, are characterized by a resolution of 512×512 pixels and a 16-bit depth, offering a rich grayscale range essential for precise identification of tissue density differences. Grayscale images are standard for computed tomography, allowing for detailed analysis of anatomical structures. The equipment used in our studies is GE Medical Systems Discovery CT750 HD (GE Healthcare, Chicago, IL, USA). The study focused on

https://doi.org/10.3390/app16041954

detailed analysis of the abdominal cavity and pelvis, conducted in two phases, which is typical for studies requiring high-precision imaging.

The layer thickness, measuring 660 μm, highlights the high precision of scanning, enabling the detection of minute structures. The kilovolt peak (KVP) set to 120 influences penetration and image contrast, while the exposure time of 6676 milliseconds determines the scanning duration. These technical parameters play a crucial role in obtaining high-resolution and contrast images, essential for accurate diagnostics. The slice thickness of 2.5 mm represents an optimal balance between accuracy and scanning time and radiation dose for the patient. Thinner slices allow for more detailed analysis and reconstruction.

Kidney segmentation masks in CT were created manually by experienced radiologists using diagnostic workstations and previously validated annotations. These masks were drawn on axial slices and reviewed for consistency. As such, they serve as a high-quality reference standard for the evaluation of macroscopic models.

Method of matching scans: Macroscopic images, such as the provided kidney photo after excision, were taken using a Canon EOS 80D DSLR camera (Canon Inc., Tokyo, Japan) under controlled lighting conditions to ensure uniform presentation of tissues and their colors. The camera was equipped with a 50 mm lens at f/2.8 aperture and a 1/60 s exposure time with ISO 200, specifically chosen to capture detailed images crucial for the study's accuracy and clarity. Each kidney slice was meticulously placed on a standardized background featuring clear scale markers, facilitating later accurate digitization and scaling of images for 3D modeling.

The images were saved in JPEG format, a standard digital format, with a resolution of 6000 pixels by 4000 pixels. To minimize the appearance of natural postoperative changes such as minor deformations or color shifts, specific photography protocols were adopted. These protocols, combined with the use of an appropriate depth of field and high-quality photographic equipment, ensured the acquisition of sharp and detailed images. The careful setting of these parameters is pivotal to ensuring that the imaging technique used is reproducible in further studies.

To evaluate the performance of the proposed OCM + DL framework against the baseline DL-only method, a statistical analysis was performed. Given the paired nature of the data and the non-normal distribution of Dice scores, the Wilcoxon signed-rank test was utilized to determine the significance of the improvements. A p-value of less than 0.05 was considered statistically significant.

3.2. Implementation Details and Computational Efficiency

The deep learning refinement was performed by a 2D U-Net with a depth of 4 resolution levels. The feature maps followed a $16, 32, 64, 128$ filter sequence, resulting in approximately 1.2 million trainable parameters. This lightweight architecture was chosen to ensure high generalization despite the limited dataset size. Training was conducted for 100 epochs using the Adam optimizer with a learning rate of 10^{-4} and a batch size of 8. The loss function combined Binary Cross-Entropy and Dice Loss to ensure both pixel-wise accuracy and regional overlap. All computations were executed on a high-performance server equipped with an NVIDIA GPU (33.70 GB VRAM, Compute Capability 12.0) running CUDA 12.2. A critical advantage of the proposed hybrid framework is its computational profile, as summarized in Table 1. While the deterministic OCM reconstruction serves as the primary bottleneck due to its iterative optimization (taking 12–15 s per slice), the U-Net inference is nearly instantaneous on the GPU ($<$0.1 s). This allows the entire 3D kidney reconstruction to be completed in approximately 3 min, making it viable for time-sensitive clinical or educational assessments.

https://doi.org/10.3390/app16041954

Table 1. Computational runtime comparison for different stages of the reconstruction pipeline.

Process Stage	Device	Average Time per Slice
OCM Reconstruction	CPU	12.4 ± 2.1 s
U-Net Inference	CPU	0.72 ± 0.1 s
U-Net Inference	GPU	0.05 ± 0.01 s
Total Pipeline (Hybrid)	GPU	$\approx$12.5 s

4. Method of Matching Scans

The method for matching kidney sections employs an algorithmic approach to register each successive section relative to the previous one, aiming to create a continuous sequence of images that will enable three-dimensional reconstruction. This registration framework is built upon a slice-wise 2D formulation and a non-diffeomorphic deformation assumption. This scope is chosen for practical reasons, as physical macroscopic sections often exhibit discontinuous tissue distortions and tears that violate the topology-preserving constraints of diffeomorphic models. While these assumptions prioritize robustness in handling physical artifacts, the framework could be extended in future work to include 3D volumetric interpolation or diffeomorphic displacement fields (e.g., stationary velocity fields) to enhance topological consistency. This process requires precise correction of spatial distortions resulting from tissue preparation non-uniformities.

1. Process objective: Create a continuous sequence of kidney sections suitable for accurate 3D reconstruction.
2. Sequence initiation: Set the first slice I_1 as the baseline (no transformation).
3. Objective function: For each subsequent slice I_i, minimize the sum of squared pixel differences
$$\text{difference}(I_i, T) = \sum_{x,y} (I_{i-1}(x,y) - T(I_i)(x,y))^2.$$
4. Optimisation: For $i = 2, \ldots, n$, find $T^* = \arg\min_T \text{difference}(I_i, T)$ under realistic bounds.
5. Transformation application: Apply the constrained similarity transform $T = \{s, \theta, t_x, t_y\}$ (uniform scale, rotation, translation) to align I_i to I_{i-1}.
6. Residual Deep Learning Refinement: A CNN predicts a local displacement field ϕ_{res} to compensate for non-linear tissue distortions. Training on OCM-pre-aligned outputs minimizes the learned degrees of freedom, ensuring robust alignment and high data efficiency.
7. Result: A uniform, anatomically consistent sequence of sections enabling robust 3D reconstruction.

Sequence Initialization: The first section is taken as the base (I_1) and undergoes no transformation. This becomes the starting point for registering all subsequent sections.

Section Registration: For each section I_i for $i = 2, 3, \ldots, n$ (where n is the number of sections), we apply an optimization algorithm to find the best fit to the previous section I_{i-1}. The goal is to minimize the difference between the section and its registered predecessor.

Objective Function: The difference between the registered section I_{i-1} and the currently transformed section I_i is expressed as the sum of squared pixeldifferences (1).

$$\text{difference}(I_i, T) = \sum_{x,y} (I_{i-1}(x,y) - T(I_i)(x,y))^2 \quad (1)$$

where T denotes the spatial transformation consisting of scaling, rotation, and translation.

https://doi.org/10.3390/app16041954

Optimization: Finding the optimal transformation T^* for section I_i involves minimizing the objective function (2).

$$T^* = \underset{T}{\text{argmin}}\, \text{difference}(I_i, T) \tag{2}$$

where T is constrained to realistic scaling values (s), rotation angles (θ), and translations (t_x, t_y).

Applying the Transformation: The OCM algorithm applies a series of transformations to achieve optimal registration of successive image sections. Starting from the initial image, the algorithm progressively aligns each subsequent image using transformations such as translation, rotation, and scaling. These steps are visualized in Figure 3, illustrating the dynamic nature of the OCM method in achieving a coherent and continuous image sequence for accurate three-dimensional reconstruction.

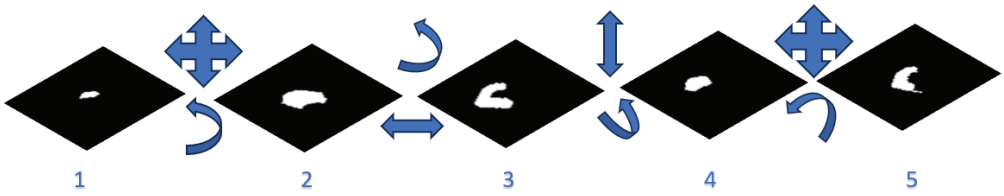


Figure 3. Symbolic illustration of the OCM algorithm's processing steps. Beginning with the first image (1), each subsequent image is aligned through translation, rotation, and scaling transformations to produce a seamless sequence (2–5).

The transformation T is applied to section I_i through the following operations:

1. Scaling:
   Scaling is accomplished by resizing the section. If $s < 1$, the section is reduced, and for $s > 1$, it is enlarged.
2. Rotation: The rotation of section I_i by angle θ is described by the rotation matrix $R(\theta)$ on Equation (3)

$$R(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \tag{3}$$

3. Translation: Translation is performed by shifting the section's pixels by vector $[t_x, t_y]$.

Final Outcome: The registration process concludes with a series of sections that, after applying optimal transformations, form a unified sequence, allowing for the creation of an accurate 3D kidney model.

This method combines image processing theory with advanced numerical techniques to provide precise and consistent data for creating 3D models in medicine and research.

Limitations in the Method of Matching Kidney Sections:In the process of matching kidney sections, we apply a specialized optimization technique that allows for the consideration of lower and upper bounds for transformation parameters. This method is essential for maintaining realistic transformation parameters, such as scaling, rotation, and translation.

Variables subject to constraints are transformed from a bounded space to an unbounded space. This transformation is achieved through appropriate mathematical functions, ensuring that the variables remain within acceptable limits.

- For variables with a lower bound (4)

$$x_{\text{trans}} = \sqrt{x - LB} \tag{4}$$

https://doi.org/10.3390/app16041954

- For variables with an upper bound (5)

$$x_{\text{trans}} = \sqrt{UB - x} \tag{5}$$

- For variables with double-sided constraints (6)

$$x_{\text{trans}} = \arcsin\left(\frac{2(x - LB)}{UB - LB} - 1\right) \tag{6}$$

- x: This is our variable that we wish to transform. In the context of our problem, this could be, for example, the degree of rotation of a kidney section or the scaling factor.
- LB and UB: These are the lower (LB-Lower Bound) and upper (UB-Upper Bound) limits for our variable x. They define the range within which the variable x can vary. For example, if the rotation can only be within the range of -45 to +45 degrees, then LB would be -45, and UB would be +45.
- x_{trans}: This is the transformed variable x, adjusted to account for the constraints. After transformation, x_{trans} can be used in a standard optimization algorithm without direct constraints.

After transforming the bounded variables into an unbounded domain, the optimization algorithm searches for values that minimize the objective function (e.g., the difference between consecutive kidney sections). Once the optimum is found, the parameters are inversely mapped back to the original bounded space to retrieve physically meaningful transformation values. Variables with identical lower and upper bounds are treated as constants and excluded from the optimization, thereby reducing the dimensionality of the problem.

Figure 4 presents a set of kidney masks in CT imaging, showing a clear and uniform structure of sections. This continuity of sections is crucial for effective three-dimensional reconstruction and allows for the precise alignment of successive layers.

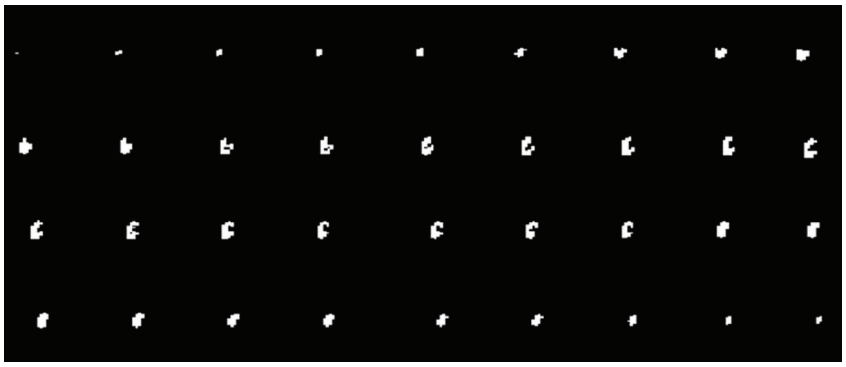


Figure 4. A set of kidney masks in CT imaging, showing a uniform structure of sections.

Figure 5 shows a set of kidney masks in macroscopic imaging before the section matching process. At this stage, the heterogeneities and shifts between individual sections are clearly visible, complicating direct 3D reconstruction.



Figure 5. A set of kidney masks in macroscopic imaging before the section matching process, showing visible heterogeneities and shifts.

https://doi.org/10.3390/app16041954

Figure 6 demonstrates the result of our section matching process, where the described optimization and transformation methods were applied. In this image, it can be seen that the kidney sections in macroscopic imaging have been effectively aligned, significantly improving the uniformity and continuity of the sequence, allowing for a more accurate 3D reconstruction.



Figure 6. A set of kidney masks in macroscopic imaging after applying the section matching method, showing significantly improved continuity and uniformity of sections.

By applying advanced image processing techniques and optimization, significant improvements in section matching were achieved, which is crucial for accurately replicating the structure of the kidneys and precise 3D reconstruction. Figures 4–6 illustrate the evolution of the process from raw sections to the final matching outcome, highlighting the efficiency of the applied method.

The integration of this optimization technique into the section matching process allows for the accurate and realistic adjustment of each kidney section, which is essential for creating precise and consistent 3D models. It also ensures control over the transformation parameters, ensuring that transformations are physically possible and consistent with the actual properties of the imaged tissues.

To better demonstrate the effectiveness of our kidney section matching method, we include visualizations of 3D kidney models before and after applying our process. Figure 7 on the left side displays a 3D kidney model obtained before applying our matching method. In this image, some discontinuity and inconsistency in the model's structure can be observed, resulting from suboptimal matching of individual sections.

Figure 7 on the right side demonstrates a 3D kidney model after applying our section matching process. Thanks to the use of advanced optimization techniques and precise section matching, the 3D model shows a significant improvement in uniformity and structural consistency. As a result, we obtain a model that is much more faithful to the actual anatomy of the kidney, which is crucial for medical and research applications.

The comparison of both 3D models, presented in Figure 7, unequivocally illustrates that our kidney section matching method brings noticeable and positive results. The improvement in the quality of the 3D reconstruction is evident, confirming the effectiveness of our approach in accurately replicating the structure of the kidneys.

An important implementation detail concerns the selection of the initial reference slice I_1, which serves as the anchor for registering all subsequent layers. In our dataset, the first macroscopic slice used for reconstruction was typically not the true apex of the kidney, but rather the second physical section (approximately at 1 cm depth). This choice was deliberate, as the apical slice was often too thin or deformed to provide a reliable anatomical base.

Two fundamentally distinct processes contribute to the dimensional accuracy of the reconstructed 3D kidney models. First, metric calibration is performed using the Hough Transform, which detects the calibration grid embedded in each macroscopic image. This grid provides a physical pixel-to-millimeter conversion ratio specific to each slice. The resulting scale factor is constant per image and is exclusively used to convert pixel-based dimensions and volumes into real-world physical units (e.g., millimeters or cubic centimeters). Importantly, this calibration step is entirely independent of any optimization procedure and

https://doi.org/10.3390/app16041954

is not modified throughout the reconstruction process. Second, the Optimal Cross-section Matching (OCM) algorithm applies geometric alignment transformations—namely, translation, rotation, and a local mask-based scaling—to improve spatial consistency between adjacent slices. The purpose of this internal scaling is not to modify physical units, but to enhance anatomical continuity by allowing slight morphological adjustments to the kidney contours. These adjustments are implemented using morphological operations such as imerode and imdilate, which affect the shape of the masks without altering image resolution or affecting real-world measurements. The geometric transformation parameters are constrained to realistic anatomical bounds, ensuring that local modifications remain plausible.

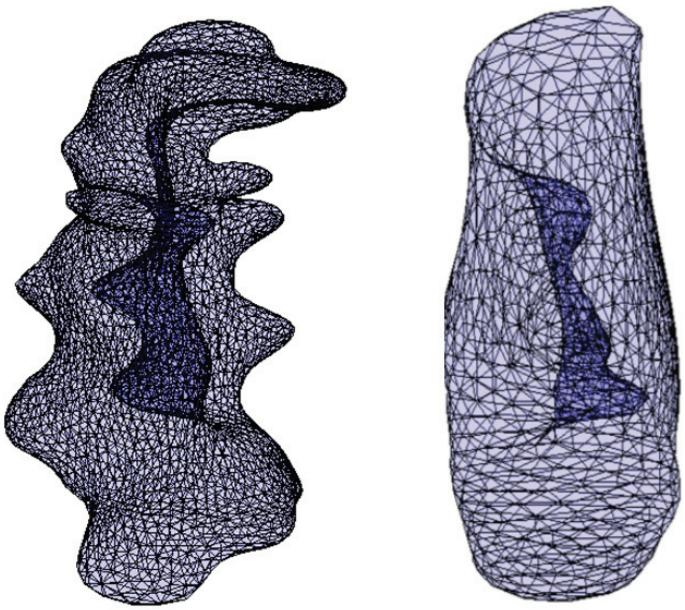


Figure 7. A 3D kidney model after applying the section matching method, showing significant improvement in uniformity and structural consistency.

5. Application of the Hough Transform for Metric Grid Calibration in Macroscopic Kidney Image Analysis

The Hough Transform [32], utilized in our study, served to precisely detect metric grid lines on macroscopic images of kidneys. This method enabled accurate measurement of the metric grid lines' lengths, allowing for the assessment of the kidneys' actual sizes.

The Hough Transform is defined as a function that transforms points from the image space (x, y) into a new parameter space, where each point in the image is represented by a set of curves in the parameter space. Lines in the image are identified as points of intersection of these curves, mathematically described by Equation (7).

$$\rho = x \cdot \cos(\theta) + y \cdot \sin(\theta) \tag{7}$$

where ρ is the distance from the origin to the line and θ is the angle of inclination of the line to the x-axis.

Peak Detection in the Hough Transform

After applying the Hough Transform to a binary image, the 'houghpeaks' function is used to identify peaks in the parameter space, corresponding to the detection of lines

https://doi.org/10.3390/app16041954

on the image. These peaks are determined based on the values in the Hough Transform accumulator matrix, and their detection is described as (8).

$$P = \text{houghpeaks}(H, N, \text{'threshold'}, \tau) \tag{8}$$

where H is the Hough Transform accumulator matrix, N is the maximum number of peaks to detect, and τ is the threshold above which values in the matrix H are considered as potential lines.

Using the 'houghlines' function, lines are extracted based on the detected peaks and line parameters in the ρ and θ space. These lines are then mapped back to the image space, enabling accurate determination of their position and orientation.

Through the application of the Hough Transform for detecting metric grid lines on macroscopic images, precise measurements of kidney dimensions were possible, which is crucial for assessing their health condition. The Hough Transform, by enabling accurate calibration of macroscopic images, significantly enhances the diagnostic value of these images, allowing for precise comparisons with data obtained using more advanced imaging techniques, such as computed tomography. This technique successfully pinpointed lines, as illustrated in Figure 8. The presented image demonstrates the effectiveness of the Hough Transform in identifying and extracting lines from the image, even in the presence of noise or background inhomogeneities.

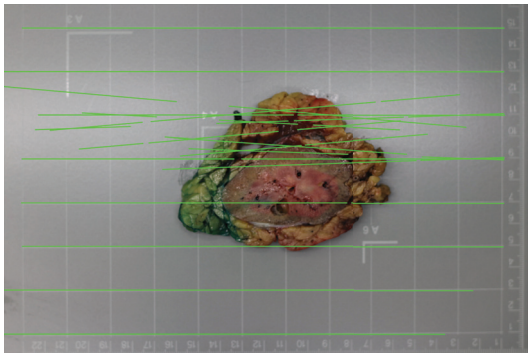


Figure 8. Effect of finding lines using the Hough Transform. These lines were identified based on accumulations in the parameter space (ρ, θ), where each point represents a potential line on the image. The resultant lines are then mapped back to the image space, illustrating the detected edges of objects.

6. Utilization of Bézier Curves in the Analysis of Anatomical Structures

To ensure geometrically accurate and visually smooth representations of kidney contours and other anatomical structures, we employed Bézier curves as an intermediate interpolation technique within each 2D image slice. Owing to their ability to generate continuous and flexible shapes from discrete control points, Bézier curves are particularly well-suited to modeling the complex and irregular boundaries commonly encountered in medical images. Beyond mere visualization, this interpolation significantly contributes to the subsequent volumetric reconstruction by providing a dense, regularized set of points that improve the continuity and fidelity of the resulting 3D meshes.

6.1. Application of Bézier Curves

Bézier curves [33] were used to interpolate and smoothly connect key points that define the shape and boundaries of the kidneys and potential pathological changes within them. Through the use of these curves, it was possible to create clear and intuitive visualizations that facilitate the identification and assessment of the morphology of the examined structures.

https://doi.org/10.3390/app16041954

A Bézier curve of degree n is defined by control points $P_0, P_1, \ldots, P_n$ and Equation (9).

$$B(t) = \sum_{i=0}^{n} P_i B_{i,n}(t) \quad \text{for} \quad t \in [0,1] \tag{9}$$

where $B_{i,n}(t)$ are Bernstein polynomials, given by (10),

$$B_{i,n}(t) = \binom{n}{i} t^i (1-t)^{n-i} \tag{10}$$

and $\binom{n}{i}$ denotes the binomial coefficient, equal to the number of combinations of i elements chosen from n elements.

In the context of our application, the control points P_i correspond to selected points on the contours of the kidneys and other structures, and the Bézier curves are used to create smooth lines that connect these points. Thanks to the flexibility of Bézier curves, it is possible to precisely model anatomical shapes, which is crucial in medical visualization applications.

In the implementation, for each point on the curve, the values of x, y (and in the case of 3D data, z) are independently calculated using the above formula, where t is a parameter running from 0 to 1, determining the position of the point on the curve. As a result, we obtain a smooth and accurate representation of the contours of the examined structures.

The implementation of Bézier curves in our system significantly contributed to the improvement of the quality of medical data visualization, which is invaluable in diagnosis and treatment planning.

6.2. Deep Learning-Based Registration with VoxelMorph

The novelty lies in a residual learning strategy where OCM neutralizes global variance, allowing the CNN to focus exclusively on local tissue deformations. This decoupling overcomes the limitations of standard VoxelMorph when handling the large spatial inconsistencies of macroscopic imaging, resulting in superior stability and faster convergence.

We adopt a two-stage residual learning paradigm. The total transformation Φ is theoretically decomposed as $\Phi(p) = \phi_{res} \circ T_{OCM}(p)$, where T_{OCM} is the deterministic global alignment and ϕ_{res} is the learned local displacement. By neutralizing global variance via OCM, we constrain the CNN to a small, well-conditioned deformation manifold. This decoupling reduces sample complexity and enhances training stability compared to fully-learned coarse-to-fine frameworks, as the model focuses exclusively on residual non-linearities.

Following Balakrishnan et al. [34], the network predicts a dense displacement field ϕ_i to warp the pre-aligned slice $I_i' = T_{OCM}(I_i)$ to I_{i-1}. We use a compact 2D U-Net with the unsupervised loss:

$$L_{\text{us}} = L_{\text{sim}}(I_{i-1}, I_i' \circ \phi) + \lambda L_{\text{smooth}}(\phi) \tag{11}$$

where L_{sim} is the local normalized cross-correlation. This hybrid positioning ensures high data efficiency, as the network does not need to internalize rigid-body physics already optimized by OCM.

7. Automated Kidney Reconstruction from CT and Macroscopic Image Analysis: Experimental Validation

In this chapter, we present the results of experiments conducted to assess the effectiveness of our developed method for the automatic reconstruction of three-dimensional kidney models based on computed tomography (CT) images and macroscopic images, supported by appropriate organ masks. These studies are crucial for evaluating the reliability

https://doi.org/10.3390/app16041954

and precision of the transformation used to create 3D models, representing an innovative approach to the analysis and modeling of anatomical structures in medicine.

The experiments were based on a dataset consisting of CT images and macroscopic images of kidneys from 40 patients, each of whom underwent nephrectomy due to kidney disease. For each case, we had both CT images and masks, which served as a reference standard, as well as macroscopic images with masks. Our research focused on comparing the dimensions of the kidneys (length, width, thickness) and volume before and after applying the developed transformation, aimed at optimizing the matching and reconstruction process of 3D models.

7.1. Results

This section presents a structured evaluation of the proposed pipeline in three configurations: (i) OCM-only alignment, (ii) deep learning (DL)-only refinement, and (iii) the final OCM + DL configuration. Anatomical mask agreement is assessed with Dice, IoU, and HD95, and geometric consistency with difference coefficients (DC) for length (L), thickness (T), width (W), and volume (Vol). When comparing continuous images (e.g., macroscopic vs. CT intensities), we also report NCC and SSIM.

Given the cohort size ($n = 40$), we adopt patient-level k-fold cross-validation (here $k = 5$). Splits are performed at the patient level to prevent slice leakage; in each fold, 80% of patients form the training set and 20% the held-out test set. The DL model is re-trained in every fold, while OCM is re-run per split without learning.

The quantitative results for all 40 patients are summarized in Table 2. The OCM + DL framework achieved a mean Dice similarity coefficient of 0.900 ($\pm$0.010), significantly outperforming the DL-only baseline, which yielded a mean of 0.783 ($\pm$0.029). The Wilcoxon signed-rank test confirmed the statistical significance of this improvement ($p < 0.001$). As shown in the boxplot (Figure 9), the OCM + DL method not only improved the overall accuracy but also substantially reduced the variability across the dataset, demonstrating higher consistency compared to the DL-only approach.

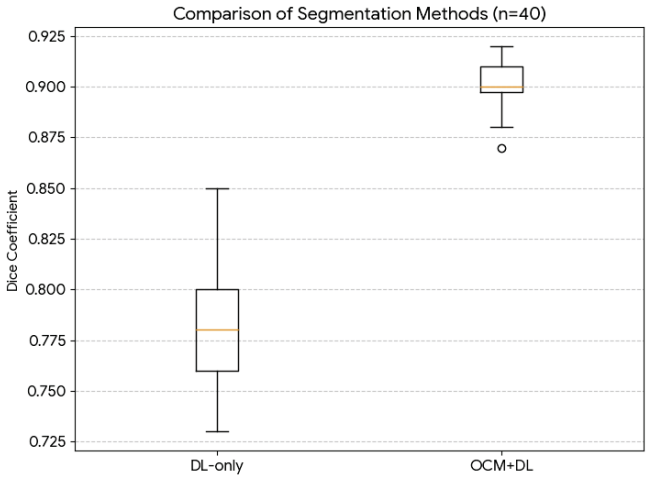


Figure 9. Comparison of Dice similarity coefficients between DL-only and OCM + DL methods across 40 patients.

https://doi.org/10.3390/app16041954

Table 2. Overall registration accuracy across methods (mean ± std over 40 patients). Higher NCC/SSIM/Dice/IoU indicate better alignment; lower HD95/DC indicate lower geometric error.

Method	NCC	SSIM	Dice	IoU	HD95 (mm)	DC(Vol)
OCM-only	0.79 ± 0.03	0.67 ± 0.08	0.79 ± 0.07	0.71 ± 0.06	3.5 ± 1.3	0.32 ± 0.07
DL-only	0.85 ± 0.07	0.65 ± 0.06	0.78 ± 0.06	0.68 ± 0.04	2.9 ± 1.6	0.25 ± 0.05
OCM + DL (final)	0.91 ± 0.04	0.81 ± 0.04	0.90 ± 0.01	0.81 ± 0.05	1.9 ± 0.7	0.11 ± 0.07

7.1.1. Evaluation Metrics

Overlap-Based Metrics (Binary Masks)

Given binary masks A (macroscopic after a given method) and B (CT reference):

$$\mathrm{Dice}(A,B) = \frac{2|A \cap B|}{|A| + |B|}, \qquad \mathrm{IoU}(A,B) = \frac{|A \cap B|}{|A \cup B|}, \quad \mathrm{IoU} = \frac{\mathrm{Dice}}{2 - \mathrm{Dice}}.$$

We also report $\mathrm{HD95}(A,B)$, the 95th percentile Hausdorff distance (mm) between mask boundaries.

Geometric Agreement (Binary Masks)

Let principal axes (o_1, o_2, o_3) be obtained by PCA on the CT mask B. For any mask $M \in \{A, B\}$ and axis o_k, define the extent as

$$E_k(M) = \max_{p \in M} \langle p, o_k \rangle - \min_{p \in M} \langle p, o_k \rangle,$$

yielding $L = E_1$, $W = E_2$, $T = E_3$ (with physical spacing). The difference coefficient for a quantity $q \in \{L, W, T, \mathrm{Vol}\}$ is

$$\mathrm{DC}(q) = 1 - \frac{q_{\mathrm{macro}}}{q_{\mathrm{CT}}},$$

so $\mathrm{DC} = 0$ indicates perfect agreement and values closer to 0 are better.

Intensity-Based Similarity (Optional)

When comparing continuous images (not masks), we use local NCC and SSIM averaged over the kidney region.

7.1.2. Experimental Setup and OCM Parameter Bounds

We analyzed macroscopic images from 40 patients and compared them with CT as the reference. OCM was applied to macroscopic slices using the following bounds:

$$s \in [0.8, 1.2], \quad \theta \in [-45^\circ, +45^\circ], \quad t_x \in [-w/2, +w/2], \quad t_y \in [-h/2, +h/2],$$

where w and h denote the image width and height. Bounds were selected based on typical anatomical misalignments observed during visual inspection.

For CT-based volume, a voxel volume was computed as $V_{\mathrm{voxel}} = p_x \cdot p_y \cdot d$ (mm^3), converted to cm^3 and multiplied by the number of kidney voxels. For macroscopic images, physical scaling was obtained from a metric calibration grid (Hough-based), with the global factor $S = \frac{L_{\mathrm{real}}}{L_{\mathrm{pixel}}}$ applied consistently to all slices.

7.1.3. OCM-Only: Dimensional and Volumetric Validation

Per-Patient Dimensional Comparison

Table 3 illustrates L/T/W measured on transformed macroscopic images (OCM) versus CT for a representative subset of five patients.

https://doi.org/10.3390/app16041954

Table 3. Comparison of kidney dimensions between transformed macroscopic images and CT images (reference standard) for a representative subset of 5 patients.

Patient	Macro L	Macro T	Macro W	CT L	CT T	CT W
1	6.75	2.87	2.64	9.45	3.54	3.56
2	7.56	2.56	2.85	11.85	3.65	4.01
3	8.54	2.75	2.71	12.665	3.23	3.64
4	8.42	3.03	3.45	10.34	3.64	5.85
5	8.85	3.23	3.28	11.43	3.84	4.74

Cohort-Level Averages of Dimensions

Table 4 summarizes cohort averages across all 40 patients.

Table 4. Average kidney dimensions across all patients obtained from CT images (reference) and from transformed macroscopic images (after applying OCM).

Image Type	Average Length (cm)	Average Thickness (cm)	Average Width (cm)
Macroscopic (OCM)	7.53	2.74	3.14
CT	10.46	3.24	3.95

Difference Coefficients for L/T/W

Per-patient coefficients are shown in Table 5, with cohort-level averages in Table 6.

Table 5. Dimension difference coefficients between transformed macroscopic images (OCM) and CT for 5 selected patients.

Patient	Length Coeff.	Thickness Coeff.	Width Coeff.
1	0.29	0.19	0.26
2	0.36	0.30	0.29
3	0.33	0.15	0.26
4	0.19	0.17	0.41
5	0.23	0.16	0.31

Table 6. Average dimension difference coefficients across all patients before and after transformation of macroscopic images. indicates improved accuracy after OCM.

	Length Coeff.	Thickness Coeff.	Width Coeff.
No Transform	0.32	0.41	0.49
OCM (After)	0.29	0.25	0.35

Volumes and Volume DC

Per-patient volumes (subset) are shown in Table 7; the volume difference coefficient before/after OCM is in Table 8.

Table 7. Kidney volumes (cm^3) from CT (reference) and transformed macroscopic images after OCM for 5 selected patients. Macroscopic volumes were scaled by Hough-based calibration.

Image Type	P1	P2	P3	P4	P5
Macro (OCM)	110.02	111.14	126.34	98.53	99.56
CT	136.65	147.53	126.47	135.63	146.62

https://doi.org/10.3390/app16041954

Table 8. Volume difference coefficients before and after OCM. Lower values indicate better alignment with CT volumes.

Condition	Volume Diff. Coeff.
Before OCM	0.57
After OCM	0.34

7.1.4. Deep Learning (DL)-Only: Mask-Based Agreement

To isolate the learning contribution, we evaluate a VoxelMorph-style model applied without OCM pre-alignment (identity initialization). We compare the warped macroscopic masks against CT masks (reference), reporting Dice/IoU/HD95 and geometric DCs.

7.1.5. Final Alignment: OCM + DL (Two-Stage)

We next evaluate the two-stage pipeline, where OCM provides global alignment and DL performs residual local refinement between consecutive slices. We compare *OCM-only*, *DLhl-only*, and *OCM + DL* using mask-based metrics and geometric DCs (primary analysis), as well as runtime/data efficiency.

Ablation of the smoothness weight λ (range 0.001–0.1) showed that $\lambda = 0.01$ provides the optimal balance between alignment accuracy and deformation plausibility. Lower values ($\lambda = 0.001$) caused non-physical folding artifacts, while higher values ($\lambda = 0.1$) overly restricted local refinements.

8. Discussion

The proposed two-stage pipeline combining the deterministic *Optimal Cross-section Matching (OCM)* and a lightweight *VoxelMorph-style deep learning (DL)* refinement represents a practical and robust strategy for macroscopic-to-CT registration of kidney models. The integration of these two complementary approaches allows for precise geometric alignment while maintaining computational efficiency and interpretability. OCM provides global normalization by correcting scale, rotation, and translation errors, thereby establishing a consistent anatomical frame of reference. The DL-based component then refines local slice-to-slice deformations that cannot be captured by global affine transformations alone. This combination substantially reduces residual geometric discrepancies and yields smoother and more anatomically faithful 3D reconstructions.

The evaluation results confirm that the hybrid OCM + DL approach achieves the highest geometric and visual accuracy across all tested configurations. Quantitatively, it delivers higher Dice and IoU scores, lower HD95 boundary errors, and smaller dimension and volume difference coefficients (DC) compared with the OCM-only and DL-only variants. These findings indicate that combining interpretable deterministic registration with learned residual refinement yields both accuracy and efficiency gains. The OCM-only method efficiently removes large-scale misalignments, while the DL module focuses on fine local corrections, leading to faster convergence, shorter inference times, and reduced training data requirements. In particular, the OCM + DL model requires significantly fewer training pairs than DL-only, since the network operates on pre-aligned input with limited residual deformation. This design aligns with the principles of residual learning and amortized optimization observed in recent registration frameworks.

From a clinical perspective, these improvements have direct implications for organ-level modeling, where precise geometric alignment is essential for assessing pathology, planning nephron-sparing surgeries, or evaluating postoperative morphology. Higher Dice/IoU and lower HD95 values translate to better correspondence of organ boundaries, while reduced dimensional and volumetric errors (DC) ensure that reconstructed kidneys preserve realistic proportions. The resulting 3D models can be integrated into interactive

https://doi.org/10.3390/app16041954

surgical planning systems or medical training simulators, where both anatomical fidelity and quantitative consistency are critical for usability and trust. Furthermore, the computational efficiency and data economy of the proposed approach make it suitable for deployment in clinical and research environments with limited training data availability.

Despite its strengths, several limitations remain. The current framework operates in 2D slice space, which may not fully account for out-of-plane distortions or missing tissue in thick macroscopic sections. The DL module, while regularized for smoothness, does not explicitly guarantee diffeomorphic transformations; future work could integrate velocity-based or invertible deformation fields to preserve topology. Additionally, small inaccuracies in metric calibration grids can propagate into volumetric estimates, suggesting a need for uncertainty quantification and robust calibration procedures. Further research should also investigate spline-based or learned 3D interpolation to enhance volumetric continuity along the z-axis and extend the approach to other organs or modalities.

Overall, this study demonstrates that a synergistic combination of interpretable geometric correction (OCM) and data-driven refinement (DL) provides a powerful and generalizable framework for multimodal anatomical registration. The findings reinforce the broader concept that hybrid models integrating deterministic and learning-based components can effectively balance accuracy, interpretability, and efficiency in medical image analysis.

A key advantage of the OCM + DL framework is its robustness in challenging anatomical scenarios. In an analysis of the "difficult cases" (defined as the five patients with the lowest baseline performance), the mean Dice score increased from 0.738 to 0.902. This average improvement of 16.4% suggests that the integration of OCM effectively compensates for limitations in standard deep learning models when dealing with high-variability cases.

From a clinical and anatomical perspective, achieving a volumetric difference coefficient of $DC_{Vol} \approx 0.11$ represents a significant milestone in macroscopic reconstruction. Given that excised tissues naturally undergo physiological shrinkage and deformation during slicing, a residual error of 11% is well within the acceptable margin for surgical simulation and medical education. More importantly, the achieved boundary accuracy (HD95 = 1.9 mm) aligns with the precision required for nephron-sparing surgery, where identifying tumor margins within a 2 mm threshold is critical for preserving healthy renal parenchyma. These improvements suggest that the OCM + DL framework provides a sufficiently realistic anatomical representation for both preoperative planning and high-fidelity educational models.

The results confirm that the hybrid OCM + DL framework outperforms standalone models by effectively decoupling global geometric alignment from local tissue refinement. While DL-only registration often fails due to the large rotations and translations inherent in macroscopic photography, the OCM stage provides a stable anchor that allows the neural network to focus exclusively on non-linear residual deformations. The persistent 11% volumetric difference (DC_{Vol}) is attributed to physiological tissue shrinkage and loss of turgor after excision, rather than algorithmic error, as the high Dice scores (0.90) demonstrate excellent anatomical shape recovery. Clinically, the achieved boundary accuracy (HD95 = 1.9 mm) is highly significant, as it falls within the 2–5 mm safety margins required for nephron-sparing surgery. This synergy makes the framework particularly robust for complex pathological cases where purely data-driven methods typically struggle with convergence.

From a computational perspective, the hybrid framework offers a favorable trade-off between accuracy and speed, which is critical for clinical adoption in surgical planning. As detailed in Table 1, the total reconstruction time for a complete 3D kidney model is approximately 3 min on a standard GPU-enabled workstation. While the OCM stage

https://doi.org/10.3390/app16041954

remains the primary bottleneck due to its iterative nature (about 90 s per image), the near-instantaneous inference of the DL refinement network ($<$0.1 s) ensures that the pipeline remains viable for time-sensitive diagnostic workflows. This efficiency allows for rapid intraoperative or postoperative morphological assessment, bridging the gap between high-fidelity 3D modeling and the fast-paced requirements of modern surgical environments without requiring massive computational clusters.

Despite the observed improvements, several limitations of the current framework should be acknowledged. First, the registration is conducted in a slice-wise 2D space, which may not fully account for complex 3D out-of-plane deformations or missing tissue volume between thick macroscopic sections. Second, while the DL refinement is regularized for smoothness, it does not explicitly enforce diffeomorphic constraints; therefore, extreme tissue tears or non-physical distortions might still result in local topological inconsistencies. Third, the accuracy of the reconstruction remains inherently sensitive to the quality of the initial macroscopic photography and the precision of the physical slicing process. Future research will focus on integrating 3D volumetric interpolation and invertible deformation fields to further enhance the topological realism of the reconstructed kidney models.

9. Conclusions

This paper presented a novel hybrid registration framework that integrates Optimal Cross-section Matching (OCM) with a VoxelMorph-style deep learning refinement to align macroscopic kidney images with CT-based anatomical references. The two-stage design leverages OCM for global alignment and deep learning for residual local correction, resulting in improved overlap (Dice, IoU), reduced boundary errors (HD95), and enhanced geometric consistency (dimension and volume DC). Quantitative evaluation across 40 patients demonstrated that OCM alone effectively corrects large-scale misalignments, while the addition of DL refinement yields further gains in anatomical accuracy and spatial coherence.

The OCM + DL configuration outperformed both the OCM-only and DL-only baselines, achieving the best trade-off between precision, runtime, and data efficiency. This outcome highlights the benefit of combining deterministic and learned components in a unified framework: OCM provides interpretability and stability, whereas the DL module captures subtle nonrigid deformations in a data-efficient manner. Such hybrid registration methods are particularly valuable in clinical and educational contexts where high anatomical fidelity and computational reliability are required.

In conclusion, the proposed OCM + DL approach enables high-quality, quantitatively consistent 3D reconstructions of kidneys from heterogeneous imaging sources. By bridging classical geometric registration with modern deep learning, it offers a reproducible, efficient, and extensible solution for multimodal anatomical modeling. Future work will focus on diffeomorphic extensions, 3D volumetric interpolation, and generalization to other organs and modalities, supporting broader adoption in computer-assisted diagnosis, surgical planning, and medical visualization systems.

Author Contributions: Conceptualization, T.L.; Methodology, T.L. and M.L.; Software, T.L. and K.S.; Validation, T.M., K.S. and M.D.; Formal analysis, M.L. and M.D.; Investigation, M.L. and M.D.; Resources, M.L. and M.D.; Data curation, T.L.; Writing—original draft preparation, T.L.; Writing—review & editing, T.L. and K.S.; Visualization, T.L.; Supervision, T.M.; Project administration, T.M. and K.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

https://doi.org/10.3390/app16041954

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki and approved by the Ethics Committee of the Military Institute of Medicine - National Research Institute in Warsaw, Poland (protocol code 38/WIM/2017 on 21 June 2017).

Informed Consent Statement: Patient consent was waived due to the retrospective nature of the study and the use of previously archived, anonymized data.

Data Availability Statement: The data that support the findings of this study are available from the Military Institute of Medicine-National Research Institute in Warsaw, but restrictions apply to the availability of these data, which were used under license for the current study, and so are not publicly available. Data are however available from the authors upon reasonable request and with permission of the Military Institute of Medicine-National Research Institute.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Pennarossa, G.; Arcuri, S.; De Iorio, T.; Gandolfi, F. Current advances in 3D tissue and organ reconstruction. *Int. J. Mol. Sci.* **2021**, *22*, 830. [CrossRef] [PubMed]
2. Wang, K.; Ho, C.; Zhang, C.; Wang, B. A review on the 3D printing of functional structures for medical phantoms and regenerated tissue and organ applications. *Engineering* **2017**, *3*, 652–663. [CrossRef]
3. Soler, L.; Nicolau, S.; Pessaux, P.; Mutter, D. Real-time 3D image reconstruction guidance in liver resection surgery. *Hepatobiliary Surg. Nutr.* **2014**, *3*, 284–291.
4. Hu, M.; Penney, G.; Edwards, P.; Figl, M. 3D reconstruction of internal organ surfaces for minimal invasive surgery. In Proceedings of the International Conference on Medical Imaging, Brisbane, QLD, Australia, 29 October–2 November 2007; Springer: Berlin/Heidelberg, Germany, 2007.
5. Geiger, B. *Three-Dimensional Modeling of Human Organs and Its Application to Diagnosis and Surgical Planning*; Technical Report; INRIA: Le Chesnay-Rocquencourt, France, 1993.
6. Abdullah, K.A.; McEntee, M.F.; Reed, W.; Kench, P.L. Development of an organ-specific insert phantom generated using a 3D printer for investigations of cardiac computed tomography protocols. *J. Med. Radiat. Sci.* **2018**, *65*, 175–183. [CrossRef] [PubMed]
7. Zhang, J.; Gong, L.R.; Yu, K.; Qi, X.; Wen, Z.; Hua, Q.; Myint, S.H. 3D reconstruction for super-resolution CT images in the Internet of health things using deep learning. *IEEE Access* **2020**, *8*, 121513–121525. [CrossRef]
8. Sarmah, M.; Neelima, A.; Singh, H.R. Survey of methods and principles in three-dimensional reconstruction from two-dimensional medical images. *Vis. Comput. Ind. Biomed. Art* **2023**, *6*, 15. [CrossRef]
9. Abegg, C.; Balbo, I.; Dominguez, A.; Grabherr, S.; Campana, L.; Moghaddam, N. Virtual anthropology: A preliminary test of macroscopic observation versus 3D surface scans and computed tomography (CT) scans. *Forensic Sci. Res.* **2021**, *6*, 34–41. [CrossRef]
10. Viero, A.; Biehler-Gomez, L.; Cappella, A.; Messina, C.; Montisci, M.; Cattaneo, C. The potential of micro-CT for dating post-cranial bone fractures: A macroscopic, radiographic, and microtomography study of fractures of known post-traumatic ages. *Int. J. Leg. Med.* **2021**, *135*, 1913–1921. [CrossRef]
11. Hagen, F.; Mair, A.; Bitzer, M.; Bösmüller, H.; Horger, M. Fully automated whole-liver volume quantification on CT-image data: Comparison with manual volumetry using enhanced and unenhanced images as well as two different radiation dose levels and two reconstruction kernels. *PLoS ONE* **2021**, *16*, e0255374. [CrossRef]
12. Chun, M.; Choi, J.; Kim, S.; Ahn, C.; Kim, J. Fully automated image quality evaluation on patient CT: Multi-vendor and multi-reconstruction study. *PLoS ONE* **2022**, *17*, e0271724. [CrossRef]
13. Wang, Z.; Zhang, C.; Jiao, T.; Gao, M.; Zou, G. Fully automatic segmentation and three-dimensional reconstruction of the liver in CT images. *J. Healthc. Eng.* **2018**, *2018*, 6797102. [CrossRef]
14. Rister, B.; Yi, D.; Shivakumar, K.; Nobashi, T.; Rubin, D. CT-ORG, a new dataset for multiple organ segmentation in computed tomography. *Sci. Data* **2020**, *7*, 381. [CrossRef]
15. Armato, S.; Altman, M.; La Rivière, P. Automated detection of lung nodules in CT scans: Effect of image reconstruction algorithm. *Med. Phys.* **2003**, *30*, 461–472. [CrossRef]
16. Zhou, L.; Wu, G.; Zuo, Y.; Chen, X.; Hu, H. A comprehensive review of vision-based 3d reconstruction methods. *Sensors* **2024**, *24*, 2314. [CrossRef]
17. Badea, C.T.; Han, Z.; Clark, D.P.; Badea, A. Whole heart CNN-based segmentation for phenotypical analysis of APOE mouse models using photon counting cine cardiac micro-CT data. In Proceedings of the Medical Imaging 2024: Clinical and Preclinical Optical Diagnostics, San Diego, CA, USA, 18–22 February 2024.

https://doi.org/10.3390/app16041954

18. Zhao, J.; Yan, Z.; Ding, Y.; Dai, Y.; Feng, Z.; Li, Z.; Ma, L.; Diao, N.; Guo, A.; Yin, H. A hybrid scaffold induces chondrogenic differentiation and enhances in vivo cartilage regeneration. *Tissue Eng. Part A* **2024**, *30*, 488–499. [CrossRef]
19. Wang, Z.; Sun, Y.; Li, S.; Lin, L.; Wang, S.; Cui, J. Geometric Artifact Correction for Symmetric Multi-Linear Trajectory CT: Theory, Method, and Generalization. *arXiv* **2024**, arXiv:2408.15069. [CrossRef]
20. Shen, L.; Zhao, W.; Capaldi, D.; Pauly, J.; Xing, L. A geometry-informed deep learning framework for ultra-sparse 3D tomographic image reconstruction. *Comput. Biol. Med.* **2022**, *148*, 105710. [CrossRef] [PubMed]
21. Feng, Z.; Yang, L.; Guo, P.; Li, B. Cvrecon: Rethinking 3d geometric feature learning for neural reconstruction. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 1–6 October 2023; pp. 17750–17760.
22. Rosenhain, S.; Magnuska, Z.A.; Yamoah, G.G.; Rawashdeh, W.; Kiessling, F.; Gremse, F. A preclinical micro-computed tomography database including 3D whole body organ segmentations. *Sci. Data* **2018**, *5*, 180294. [CrossRef]
23. Oliveira, B.; Queirós, S.; Morais, P.; Torres, H.R.; Gomes-Fonseca, J.; Fonseca, J.C.; Vilaça, J.L. A novel multi-atlas strategy with dense deformation field reconstruction for abdominal and thoracic multi-organ segmentation from computed tomography. *Med. Image Anal.* **2018**, *45*, 108–120. [CrossRef] [PubMed]
24. González Izard, S.; Sánchez Torres, R.; Alonso Plaza, O.; Juanes Mendez, J.A.; García-Peñalvo, F.J. Nextmed: Automatic imaging segmentation, 3D reconstruction, and 3D model visualization platform using augmented and virtual reality. *Sensors* **2020**, *20*, 2962. [CrossRef]
25. Chen, H.; Zhang, Y.; Kalra, M.K.; Lin, F.; Chen, Y.; Liao, P.; Zhou, J.; Wang, G. Low-dose CT with a residual encoder-decoder convolutional neural network. *IEEE Trans. Med. Imaging* **2017**, *36*, 2524–2535. [CrossRef] [PubMed]
26. Kang, E.; Min, J.; Ye, J.C. A deep convolutional neural network using directional wavelets for low-dose X-ray CT reconstruction. *Med. Phys.* **2017**, *44*, e360–e375. [CrossRef]
27. Wang, G. A perspective on deep imaging. *IEEE Access* **2016**, *4*, 8914–8924. [CrossRef]
28. Hsieh, J.; Nett, B.; Yu, Z.; Sauer, K.; Thibault, J.B.; Bouman, C.A. Recent advances in CT image reconstruction. *Curr. Radiol. Rep.* **2013**, *1*, 39–51. [CrossRef]
29. Kofler, A.; Haltmeier, M.; Schaeffter, T.; Kachelrieß, M.; Dewey, M.; Wald, C.; Kolbitsch, C. Neural networks-based regularization for large-scale medical image reconstruction. *Phys. Med. Biol.* **2020**, *65*, 135003. [CrossRef] [PubMed]
30. Wang, G.; Ye, J.C.; Mueller, K.; Fessler, J.A. Image reconstruction is a new frontier of machine learning. *IEEE Trans. Med. Imaging* **2018**, *37*, 1289–1296. [CrossRef]
31. Bouhacina, O.; Khatir, A.; Matoug, S.A.; Tamine, T. A Novel Hybrid TOARS-Optimized Ensemble of Tree-Based Models for Predicting Soil Temperature at Shallow Depths. *Sustainability* **2026**, *18*, 294. . [CrossRef]
32. Illingworth, J.; Kittler, J. A survey of the Hough transform. *Comput. Vision Graph. Image Process.* **1988**, *44*, 87–116. [CrossRef]
33. Han, X.A.; Ma, Y.; Huang, X. A novel generalization of Bezier curve and surface. *J. Comput. Appl. Math.* **2008**, *217*, 180–193. [CrossRef]
34. Balakrishnan, G.; Zhao, A.; Sabuncu, M.R.; Guttag, J.; Dalca, A.V. VoxelMorph: A Learning Framework for Deformable Medical Image Registration. *IEEE Trans. Med. Imaging* **2019**, *38*, 1788–1800. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

https://doi.org/10.3390/app16041954

applied sciences

MDPI

Article

Identifying and Predicting Changes in Behavioral Patterns for Temporal Data in Treatment of Neonatal Respiratory Failure

Adam Szczur [1], Jan G. Bazan [1], Urszula Bentkowska [1,*], Piotr Kruczek [2] and Stanislawa Bazan-Socha [3]

[1] Institute of Computer Science, University of Rzeszów, Pigonia 1, 35-310 Rzeszów, Poland; aszczur@ur.edu.pl (A.S.); jbazan@ur.edu.pl (J.G.B.)
[2] Deparment of Neonatalogy, R. Czerwiakowski Hospital, Kraków, Siemiradziego 1, 31-137 Kraków, Poland
[3] Department of Internal Medicine, Faculty of Medicine, Jagiellonian University Medical College, Jakubowskiego 2, 30-668 Kraków, Poland; stanislawa.bazan-socha@uj.edu.pl
* Correspondence: ubentkowska@ur.edu.pl

Abstract: In this paper, we present the findings of a study focused on discovering process models and tracking their evolution over time. The research specifically targets a distinct category of these models known as behavioral patterns. Consequently, the challenges and techniques addressed here involve temporal data. To demonstrate the issues and methodologies associated with identifying process patterns and their changes, we use illustrative data from the treatment of respiratory failure in premature infants. The main achievement of the paper is to successfully model behavioral patterns using machine learning models and predict changes in a neonate's state. That was justified by comparing the classifier's sensitivity for cases of deterioration, improvement, and stability. Classification quality is best when the pattern remains constant. However, in the proposed model, when the pattern deteriorates, the classification quality decreases only slightly.

Keywords: machine learning; classification; temporal data; behavioral pattern

1. Introduction

In this contribution, we describe the results of research aimed at developing a new data mining method for identifying process models and their changes. Specifically, a special type of such models, called *behavioral patterns*, is discussed. Therefore, the problems and methods considered here concern temporal data, i.e., data that takes into account the observation time of the represented objects. The presented results are based on exemplary data on the treatment of respiratory failure in premature infants, illustrating the problems and methods related to detecting process patterns and their changes.

Advances in medical science and technology over the past few decades have created new opportunities for intensive care. This makes it possible to keep premature neonates alive, including the smallest ones, those born at 20–24 weeks of gestation, and those weighing more than 500 g. Since premature infants experience several disorders in the first weeks of life, managing them is essential for avoiding severe multi-organ complications and ensuring survival. Prematurity is characterized by the incomplete maturity of various systems and organs, which contributes to the functional impairments observed after birth. Among them, breathing abnormalities occurring in the first hours of life are critical to the development of respiratory failure, the leading cause of death in those patients.

Efficient complex dynamical system monitoring very often requires the identification of so-called behavioral patterns or a specific type of such patterns called *high-risk patterns* or *emergent patterns* (see, e.g., [1] for more details). They are complex concepts concerning the dynamic properties of complex objects, which are dependent on time and space and expressed in natural language. Examples of behavioral patterns may include overtaking on a road, the behavior of a patient faced with a serious life-threatening situation, and ineffective behavior of a robot team. These types of concepts are much more difficult to approximate than complex concepts, for which approximation does not require following object changes over time. Identifying certain behavioral patterns can be crucial for recognizing or predicting the behavior of a complex dynamical system. Suppose specific risk patterns are identified in a certain situation; then the control object (a driver of a vehicle, a physician, a pilot of an aircraft, etc.) can use this information to adjust selected parameters and obtain the desirable behavior of the complex dynamical system. That can make it possible to overcome dangerous or uncomfortable situations (see, e.g., [1] for more details).

However, sometimes the identification of a behavioral pattern in relation to a monitored complex system may come too late to take action that could counteract the undesirable behavior. For example, if a patient undergoing treatment starts behaving according to a life-threatening pattern and we identify it at that point, it may already be too late to prevent the patient's death. Therefore, it would be beneficial to predict in advance that the complex system is about to change its behavior and begin to match a different behavioral pattern. Therefore, recognizing changes in complex objects over time, also known as temporal change detection, has broad applications in many real-life areas. The goal of such a process is to detect, analyze, and sometimes predict how objects or systems evolve in time. This prediction is important, for example, in industrial systems (predictive maintenance—anticipating machine failure by analyzing sensor data; process optimization—predicting how production parameters will affect output over time), autonomous vehicles (trajectory prediction—forecasting the future positions of pedestrians, cars, or other agents), healthcare (disease progression modeling, patient monitoring—anticipating crises from continuous vital signs), climate and environmental systems (climate modeling, ecosystem change), and business and finance (customer behavior, market evolution). Diverse scientific tools were proposed in the literature to resolve these problems: time-series forecasting, Recurrent Neural Networks (RNNs), LSTMs, GRUs, transformers for time series, dynamical system modeling, and Graph Neural Networks (GNNs) (cf. [2,3]). In the case of healthcare and patient monitoring in the treatment of neonatal respiratory failure, machine learning methods have also previously been demonstrated (cf. [4–7]).

However, it is challenging to find methods in the literature for predicting changes in behavioral patterns that are manually defined by experts, possibly because this requires substantial support from domain experts. This study is dedicated to developing and exploring such methods.

The primary objective of this work is to identify and predict changes in the behavioral patterns of neonates in the treatment of respiratory failure. However, the proposed methodology may also be applied to other areas. The goal of this paper is to predict early (while the object matches the specific pattern) whether an object will later match another pattern. This method can be considered more advanced than the usually performed experiments, which are focused solely on identifying behavioral patterns. The results presented in this work are based on the concepts presented in [1]. The nodes in behavior graphs represent temporal concepts defined in time windows and requiring approximation. In this approach, the behavior graph of a complex object is interpreted as a complex classifier, enabling the identification of the behavior pattern described by this graph. That is achieved by

observing the behavior of the complex object over time and verifying whether it aligns with a selected path of the behavior graph. If so, the behavior is determined to fit the behavior pattern represented by this graph, which enables the detection of specific behaviors of complex objects. It is worth adding that the structure of such a behavior graph (nodes and edges) is proposed by domain experts. In this paper, the proposed approach to defining and identifying behavior patterns differs slightly, although it also leverages the knowledge of domain experts. In particular, a behavior pattern is defined as a logical formula defined for a time window, describing the specific behavior of a complex object within that window. If the formula is true for the behavior of a given complex object, this means that the object behaves according to the pattern described by this formula.

The principal achievement of the paper is successfully modeling behavioral patterns and predicting changes in a neonate's state. This was enabled by the use of machine learning methods, which utilized the classifier's sensitivity for cases of deterioration, improvement, and stability. The analysis reveals that classification quality is best when the pattern remains constant. When the patient's condition improves, the classification quality of such a change is also high. However, in the proposed model, when the pattern deteriorates, the classification quality decreases only slightly. Accurately predicting changes in these behaviors is complex due to noise, nonlinearity, and context dependency. The obtained results are excellent. They were obtained from a dataset collected at the Neonatal Pathology and Intensive Care Unit of the University Children's Hospital in Krakow, Poland, from 2002 to 2004 using the Neonatal Information System (NIS).

However, the goal of this paper is not to build a computer-based system to support the treatment of infants. That would be practically impossible, as the data used is already quite old. Since their collection, these treatment methods have undergone significant evolution. Therefore, to develop a system supporting actual treatment, it would be necessary to collect current data. In this paper, we use the medical data solely to illustrate the method for predicting changes in behavior patterns.

The main results of this paper are as follows:

- Development of a method for defining behavior patterns and a method for identifying behavior patterns;
- Development of a method for perceiving changes in behavior patterns;
- Illustration of the proposed methods using medical data related to the treatment of respiratory failure in neonates;
- Verification of the quality of the proposed methods on medical data.

The structure of the paper is as follows: In Section 2, we provide the theoretical foundations of the work. In Section 3, we describe the methodology used in the experiments. In Section 4, we detail the characterization of the dataset, and in Section 5, we provided an overview of the proposed experimental approach. The classification procedure is described in Section 6. Finally, a discussion of the results is presented in Section 7.

2. Theoretical Foundations

An information system [8,9] is a pair of the form $\mathbf{IS} = (U, A)$, where

- U is a finite, non-empty set, called universum, and the elements of U are called objects: $U = \{u_1, u_2, \ldots, u_n\}$;
- A is a finite, non-empty set of attributes (properties, features): $A = \{a_1, a_2, \ldots, a_m\}$.

If in an information system one of the attributes indicates to which category each object belongs, then such a system is called a decision table.

A decision table is an information system of the form $\mathbf{DT} = (U, A \cup \{d\})$, where

- $d \notin A$ is a decision attribute;
- Elements $a \in A$ are called conditional attributes, and A is called the set of conditional attributes.

If there is a need to represent the complex states of objects observed in complex dynamic systems, the standard concept of an information system needs to be extended. Therefore, a temporal information system is defined [1,10].

A temporal information system is a four-element tuple $\mathbf{TIS} = (U, A, a_{id}, a_t)$, where

- (U, A) is an information system;
- a_{id} and a_t are distinguished numerical attributes from set A.

We say that object $u \in U$ represents the current parameters of a complex object with identifier $a_{id}(u)$ at time point $a_t(u)$ in the temporal information system **TIS**. It should be additionally noted that attribute a_{id} is understood here as the identifier of a complex object that can be observed at multiple time points. Each such time point is represented in the system **TIS** by a single object (a row in the table). Then, for each object, the value of attribute a_{id} is the same. For example, consider a complex object corresponding to a patient with a fixed identifier p_{id}. Information about this patient can be represented in **TIS** by a list of objects, where each object represents information about the patient at a single time point. In each of these objects, the value of attribute a_{id} is the same and equal to p_{id}.

Moreover, we say that an object $u_1 \in U$ precedes an object $u_2 \in U$ in the temporal information system **TIS** if and only if

$$u_1 \neq u_2 \ \wedge \ a_{id}(u_1) = a_{id}(u_2) \ \wedge \ a_t(u_1) < a_t(u_2). \tag{1}$$

Objects from the temporal information system are grouped into time windows. *For a time window* w in a temporal information system **TIS** is any sequence $(u_1, \ldots, u_k)$ of objects from U, it follows that

- $k > 1$;
- for any $i, j \in \{1, \ldots, k\}$, it holds that $a_{id}(u_i) = a_{id}(u_j)$;
- $a_t(u_i) < a_t(u_{i+1})$, for $1 \leq i < k$.

The visualization of a time window is presented in Figure 1.

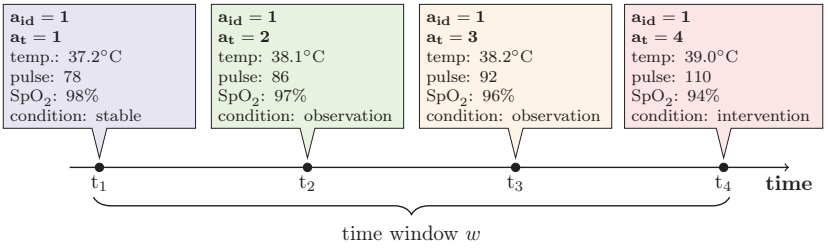


Figure 1. Visualization of exemplary time window.

If w is a time window and u is one of the objects in that window, then we denote this fact as $u \sqsubset w$. We denote the family of all time windows from the temporal information system **TIS** as $\mathbf{W}(\mathbf{TIS})$.

For a given complex object (e.g., a patient undergoing treatment), a window of full size represents the entire history of changes in attribute values from the set $A \setminus \{a_{id}\}$ for that complex object (it is the entire history of a given complex object). The family of all such time windows from the temporal information system **TIS** is denoted as $\mathbf{W}_{tot}(\mathbf{TIS})$ (where $\mathbf{W}_{tot}(\mathbf{TIS}) \subseteq \mathbf{W}(\mathbf{TIS})$).

It is easy to see that for a given $u \in U$, there exists only one time window $w = (u_1, \ldots, u_k) \in \mathbf{W}_{tot}(\mathbf{TSI})$ such that $u = u_i$ for exactly one $i \in 1, \ldots, k$. For a given

$u \in U$, we denote such a time window by $w(u)$. In practice, there is a need to consider a number of decision problems that require the approximation of temporal concepts. In a temporal information system, these concepts are usually associated with time windows, and therefore the relationship of time windows with concepts must be represented in some way. As in the case of classical decision tables, we can use a special attribute here, called the decision attribute.

In principle, each time point within a given time window can have a distinct decision attribute value. However, in this paper, what is of particular interest to us is the situation in which each time point within the window has the same decision attribute value. In other words, a specific decision attribute value is associated with all time points representing information about a given time window. Therefore, the concept of a temporal information system requires yet another extension. Therefore, we define a temporal decision table.

A temporal decision table (TDT) is a seven-element tuple $\mathbf{TDT} = (U, A, a_{id}, a_t, W, \widetilde{A}, \widetilde{d})$, where

- $\mathbf{TIS} = (U, A, a_{id}, a_t)$ is a temporal information system;
- $W \subseteq \mathbf{W}(\mathbf{TIS})$ is a set of time windows selected by an expert;
- $\widetilde{A}$ is a set of time window attributes (proposed by an expert or extracted using an automatic feature extraction method);
- $\widetilde{d} \in \widetilde{A}$ is a distinguished attribute of the time window, called the decision attribute of the time window.

The first four elements of the TDT tuple, i.e., U, A, a_{id}, and a_t, constitute the temporal information system and have been discussed earlier. The elements W, $\widetilde{A}$, and $\widetilde{d}$ require further explanation. W is the set of time windows prepared for the experiments. These windows were selected from the original dataset. In general, many different time windows of varying lengths and starting points can be selected from the original data. For this paper, a method for selecting time windows for experimental purposes had to be proposed. The proposed method generates time windows in such a way that, for a given patient, windows starting from any time point are created, but only windows of a fixed length are considered. Each window consists of eight time points, with the first five points forming the conditional part of the window (conditional subwindow) and the last three points forming the decision part of the window (decision subwindow). Thus, the length of each window in the set W is eight time points, with the lengths of the conditional and decision parts being five and three points, respectively. These lengths were chosen for practical reasons: with such lengths, the proposed method can be applied in a realistic manner. Extending the conditional part could reduce the classifier's sensitivity to sudden changes at the end of the conditional part, while shortening it could reduce sensitivity due to insufficient context for the patient. Similarly, extending the decision part could lead to decision classes that are impossible to predict based on the conditional part (sudden changes at the end of the decision part could not be anticipated from the conditional observations). Shortening the decision part, on the other hand, would make pattern prediction less interesting, as it would concern only the very near future. These observations were confirmed by auxiliary experiments. $\widetilde{A}$ is the set of temporal attributes calculated for the time windows in W. These attributes are necessary to approximate the decision attribute $\widetilde{d}$. The values of the decision attribute $\widetilde{d}$ are calculated based on the definition of behavior patterns (see the next subsection). By computing the decision attribute in this way, it is possible to construct a classifier that can predict the membership of a time window, represented by its conditional part, to the appropriate behavior pattern defined on the decision part of the window from the set W.

For a given temporal decision table **TDT**, we also define an additional attribute d for objects from U (time points), called the decision attribute. It is computed for any $u \in U$ as follows: $d(u) = \widetilde{d}(w(u))$. It is easy to see that the value of the decision attribute d is always the same for all time points belonging to a given time window.

If $\mathbf{TDT} = (U, A, a_{id}, a_t, W, \widetilde{A}, \widetilde{d})$ is a temporal decision table (where $\mathbf{TIS} = (U, A, a_{id}, a_t)$ is a temporal information system) and $W = \mathbf{W}_{tot}(\mathbf{TIS})$, then we call the temporal decision table **TDT** a *temporal decision table with full time windows*, and to simplify the description, we denote such a table as $\mathbf{TDT}_t$ instead of $\mathbf{TDT}_t = (U, A, a_{id}, a_t, \mathbf{W}_{tot}(\mathbf{TIS}), \widetilde{A}, \widetilde{d})$.

3. Methodology

3.1. Behavior Patterns Based on Domain Knowledge

In this paper, we concentrate on the problem of recognizing changes in a neonate's behavioral patterns related to the treatment of respiratory failure. We will refer to it as the MED problem. In this paper, we understand a behavioral pattern as a description of the behavior of a complex object (e.g., a patient undergoing treatment) that characterizes a state of that object that cannot be immediately identified (i.e., at a single point in time) but requires observation of the object's behavior over a period of time, i.e., observation of the object in a time window containing a sequence of time points. This refers to a situation where, in the available data, we have a recorded sequence of time points for each complex object, and at each time point, we have a description of the object represented by a fixed number of attributes, which we will call sensory attributes, as they are typically collected using specific sensors.

In the remainder of this paper, for the requirements of the experiments, we assume that there exists a complex object *id* and a sequence of time points from the time window $w = (t_1, \ldots, t_n)$ describing the situation of an object *id*.

Furthermore, at each time point, we have available m sensory attributes $s_{a1}, \ldots, s_{am}$ describing the state of object *id* at individual time points. In addition to sensory attributes, additional attributes based on domain knowledge can be defined at time points to describe the behavior of the complex object at that time point. We will refer to these as expert attributes. In our case, let us say there are k attributes $e_{a1}, \ldots, e_{ak}$. These additional attributes often require approximation based on sensory attributes, or their values are calculated using analytical formulas based on sensory attributes.

We often define a behavior pattern based on observing changes in expert attributes, but we must do this based on sequences of expert attribute values from individual time points. For example, suppose we are interested in an expert attribute $e \in \{e_{a1}, \ldots, e_{ak}\}$, which we want to use to describe behavior patterns. In time window w, there is a sequence of attribute values e, which we denote by $ve_1, \ldots, ve_n$. Theoretically, we could label each such sequence as a behavior pattern characteristic of a specific complex object, but the data can contain many sequences that may correspond to different complex objects, or even different time windows for the same object. Therefore, to define temporal patterns that describe interesting behaviors of multiple objects (e.g., high-risk patterns, negative behavior patterns), we may consider grouping sequences of time points into clusters so that individual clusters have an interesting interpretation related to domain knowledge.

The problem of clustering sequences of time points has long been intensively studied in data mining. Existing methods often use sequence features computed as specific aggregations of sensory or expert characteristics from individual time points. For example, if time points have a numerical attribute a, then the aggregate features for this parameter could be, for example, the minimum value of a, the average value of a, the maximum value of a, etc. These aggregated features can then be used to cluster sequences of time points. Time points are typically clustered into fixed-length sequences, which are often called time windows. Therefore, clustering

sequences is essentially clustering time windows. After clustering windows, each cluster can be labeled, associating with this cluster a specific behavior of the complex object.

In our example, if we have a time window with a sequence of values $e_1, \ldots, e_n$ of attribute e and we use the four aggregation functions avg (average), min (minimum), max (maximum), and $stddev$ (standard deviation), then the attribute values of this window will be denoted as $avg(e1, \ldots, en)$, $min(e1, \ldots, en)$, $max(e1, \ldots, en)$, and $stddev(e1, \ldots, en)$, respectively. In this way, the calculated values of new attributes can be grouped, obtaining clusters of time windows.

Unfortunately, in practice, for specific datasets, fully automatic clustering of behavioral patterns encounters significant challenges. One such problem is that the data may not be sufficiently representative. As a result, automatically determined patterns may differ significantly from one data sample to another. This problem is particularly acute when discovering behavioral patterns related to phenomena that occur rarely (e.g., severe patient conditions). If the data contains few sequences describing such situations, it is difficult to predict the construction of clustering methods that would allow for the construction of classifiers that effectively classify test cases into such groups. Fortunately, domain experts (e.g., doctors) possess knowledge that can aid in the discovery of behavioral patterns. This means that instead of automatically clustering time windows, one can construct special formulas based on domain knowledge that allow for determining the behavioral pattern.

For example, in the MED problem, we have a sensory attribute a_w that describes the patient's current ventilation method, characterizing the patient's current state during respiratory failure treatment. This attribute takes values in the range $[0,\ 1]$, where 0 represents the most invasive form of mechanical ventilation used in the most severe cases of respiratory failure, and 1 represents the patient's spontaneous breathing (without the need for respiratory support).

We therefore have values for this attribute for all time points in the window $a_{w1}, \ldots, a_{wn}$. Let us also assume that we have calculated the values of the aggregated attributes: $avg(a_{w1}, \ldots, a_{wn})$, $min(a_{w1}, \ldots, a_{wn})$, and $max(a_{w1}, \ldots, a_{wn})$.

As a result of an interview with an expert, it was determined that the following three patient behavior patterns should be monitored during respiratory failure treatment.

- Behavior pattern 0: This pattern is characterized by the formula

$$avg(a_{w1}, \ldots, a_{wn}) < 0.3 \wedge max(a_{w1}, \ldots, a_{wn}) < 0.4 \wedge min(a_{w1}, \ldots, a_{wn}) \geq 0.0 \quad (2)$$

  —belonging to this pattern means that the patient's condition is serious; their life may be in danger.
- Behavior pattern 2: This pattern is characterized by the formula

$$avg(a_{w1}, \ldots, a_{wn}) > 0.9 \wedge max(a_{w1}, \ldots, a_{wn}) \leq 1.0 \wedge min(a_{w1}, \ldots, a_{wn}) > 0.8 \quad (3)$$

  —belonging to this pattern means the patient is in a good condition, which promises successful completion of treatment for respiratory failure.
- Behavior pattern 1: This is an intermediate pattern between patterns 0 and 2, characterized by the formula

$$avg(a_{w1}, \ldots, a_{wn}) \geq 0.3 \wedge avg(a_{w1}, \ldots, a_{wn}) \leq 0.9 \wedge \quad (4)$$

$$max(a_{w1}, \ldots, a_{wn}) < 1.0 \wedge min(a_{w1}, \ldots, a_{wn}) \geq 0.0$$

  —belonging to this pattern indicates that the patient is in an average condition, requiring further treatment, especially in the event of complications, such as a bacterial, viral, or fungal infection.

The behavior patterns defined above were proposed based on domain knowledge. The physicians participating in the described studies indicated that, from a domain knowledge perspective, the most important parameter describing a patient's condition in the context of respiratory failure is the parameter a_w described earlier. Furthermore, the physicians proposed several thresholds for this parameter, which divide the patient's states into better or worse conditions. These thresholds have the following values: 0.0, 0.3, 0.4, 0.8, 0.9, and 1.0. After defining the thresholds, the experts formulated three logical expressions describing three significantly different patient states. These formulas were then used to define three patient behavior patterns.

Once the above patterns are defined, it is easy to check whether a patient belongs to a particular pattern in a given time window. Certainly, in subsequent time windows, the patient may change their behavior pattern. For example, they may move from pattern 1 to pattern 0, indicating a significant deterioration in the patient's condition.

Although this is not necessary, for practical applications related to this work, we define behavior patterns such that each time window matches exactly one behavior pattern (the sets of objects matching individual patterns are disjoint, non-empty, and together constitute the entire set).

3.2. Method for Predicting Changes in Behavior Patterns

In this paper, we present results related to the development of new methods for predicting the transition of a complex object from one behavioral pattern to another. This involves a situation where a complex object first matches one pattern and, after some time, another behavioral pattern. The goal is to predict early (while the object matches the first pattern) whether the object will later match the second pattern. This method can be considered more advanced than methods that are focused solely on identifying behavioral patterns [1]. For example, in the MED problem, a patient may transition from pattern 1 to pattern 0, indicating a significant deterioration in the patient's condition. From a technical perspective, we consider a dataset consisting of a sequence of pairs of time windows (w_1, w_2). Pairs of time windows may originate from different composite objects, but in a given pair (w_1, w_2) both windows (w_1 and w_2) originate from the same composite object, and window w_1 describes the situation of the object immediately before the situation in window w_2. Using the method described in detail in [1], we can determine for each pair of windows (w_1, w_2) the behavior pattern to which window w_2 belongs, denoted by $W(w_2)$. This yields a dataset that is a sequence of pairs $(w_1, W(w_2))$. A visualization of a dataset created in the above described manner is given in Figure 2.

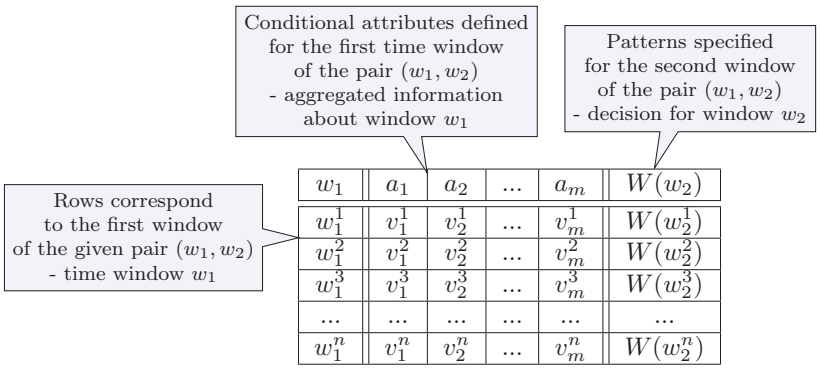


Figure 2. Visualization of decision table where objects are time windows w_1 for which a decision is a pattern for time window w_2.

For this dataset, we construct classifiers that learn to make decisions for window w_2 regarding its behavior pattern membership based on the data in window w_1.

4. Dataset

Respiratory failure, which dominates the clinical picture of a premature infant, is not the only factor determining their recovery. Effective care for this type of patient requires consideration of all coexisting disorders, such as congenital and acquired infections, fluid and electrolyte imbalances, acid–base, circulatory, and renal disorders, etc. All of these factors are interconnected and interact with one another. Therefore, caring for premature infants in the first days of life requires continuous analysis of numerous vital signs and additional test results. These can be divided into stationary data (e.g., gestational age, birth weight, Apgar score) and continuous data (variables that change over time). Continuous variables can be examined ad hoc (e.g., arterial blood gas values) or monitored continuously, for example, using monitoring devices (hemoglobin oxygen saturation—SAT, heart rate, blood pressure, body temperature, and mechanical parameters of artificial ventilation). When caring for premature neonates, doctors also evaluate the results of imaging tests (e.g., brain ultrasound, echocardiography, chest X-ray). In addition to parameters that define the patient's condition, comprehensive analysis also considers the treatment methods used. These can be qualitative (e.g., drug administration) or quantitative (e.g., ventilator settings). Daily analysis of parameters requires extensive theoretical knowledge and practical experience of physicians. Additionally, this analysis must be quick and precise. Assessments are often made in a rushed and stressful environment. A critical element of this analysis is the accurate assessment of the young patient's risk of death due to respiratory failure in the coming hours and days. Therefore, the correct conclusions regarding life-threatening situations are based not only on current clinical condition, laboratory tests, and imaging studies of the neonate, but also on the recently observed dynamics and nature of changes in health status (e.g., deterioration in blood gas indicators of respiratory failure). Given the difficulties associated with analyzing all the necessary information at a given moment, IT methods can prove to be extremely helpful.

The analyzed dataset was collected at the Neonatal Pathology and Intensive Care Unit of the University Children's Hospital in Krakow, Poland, from 2002 to 2004 using the Neonatal Information System (NIS) computer system. The dataset provides detailed information on the treatment of 340 neonates (row data). Detailed information was collected for each infant, including the mother's pregnancy history, birth weight and age, laboratory and imaging test results, detailed diagnoses made during follow-up, medical procedures performed, and medications administered. The study group consisted of premature neonates with a birth weight of $\leq$1500 g, admitted to the hospital within 2 days of life. Additionally, for the purposes of the performed experiments, data from neonates were selected, removing those diagnosed with respiratory failure without a diagnosis of neonatal respiratory distress syndrome (RDS), patent ductus arteriosus (PDA), sepsis, or ureaplasma. In the original dataset obtained from NIS, complete information on cases of RDS, PDA, sepsis, and ureaplasma was represented. Therefore, it was possible to exclude patients who did not have these conditions. This approach was justified by the fact that the excluded cases were much less relevant to physicians from the perspective of developing computer-based methods to support treatment. The data of neonates who died in the hospital but did not have breathing problems (they died from another cause) were also removed.

A train and test split methodology was applied to perform the experiments. With a sufficiently large size, a dataset consisting of a sequence of pairs $(w_1, W(w_2))$ can be randomly divided into two parts to implement the train and test methods. However, when

dividing the data into training and testing samples, it is necessary to ensure that the training part contains windows from different complex objects than the test sample. If this is not the case, the classifier may be trained and tested on data from the same complex object, which is inappropriate since it may lead to data leakage.

In the performed experiments, each patient's data is used to generate pairs of windows: a conditional window w_1 (five time points were used) and a decision window w_2 (three time points were used). Visualization of the process of time window generation is presented in Figure 3.

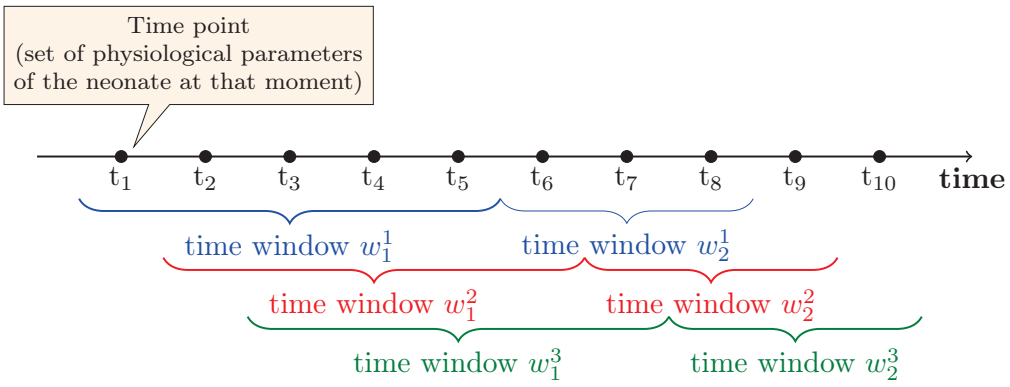


Figure 3. Visualization of the process of generating time windows w_1 and w_2, with 5 time points for w_1 and 3 time points for w_2.

As a result the data of neonates with less than eight observations (time points) were removed, which means that the neonate stayed in the hospital for too short a time to be examined using time-based methods (e.g., a neonate was transferred to another hospital or died almost immediately). Finally, for the purpose of the experiments performed in this paper, 173 neonates were chosen. The IDs were extracted (without repeats), and then the IDs were randomly divided in a 50% to 50% ratio. Therefore, there were 87 training patients and 86 test patients. These data served as the basis for determining window pairs for generating training and test data for the experiments.

The total number of pairs $(w_1, W(w_2))$ is 7186 (this refers to the total number of training and testing time windows). The decision window allowed for generating one of three decision values: 0 (bad state), 1 (medium state), or 2 (good state).

For the training portion, the class sizes were 0: 122 pairs $(w_1, W(w_2))$; 1: 1419 pairs $(w_1, W(w_2))$; 1: 2126 pairs $(w_1, W(w_2))$; and total: 3667 pairs $(w_1, W(w_2))$.

For the test portion, the class sizes were 0: 247 pairs $(w_1, W(w_2))$; 1: 1156 pairs $(w_1, W(w_2))$; 2: 2116 pairs $(w_1, W(w_2))$; and total: 3519 pairs $(w_1, W(w_2))$.

A short summary of the dataset details is provided as follows:

- Total data size: 7186 objects (this is the number of pairs $(w_1, W(w_2))$).
- Class 0: 360 objects (bad).
- Class 1: 2575 objects (medium).
- Class 2: 4242 objects (good).

This data is therefore unbalanced. This fact will influence the selection of the classification quality metric, which will be described in Section 6.

5. Proposed Approach Overview

In this section, we present a comprehensive schematic of the proposed method. Figure 4 illustrates all stages of the proposed method for predicting changes in behav-

ioral patterns, along with the relationships between them and the data flow. The individual objects in the diagram represent the main components of the method, while the arrows depict the flow of data and the corresponding actions.

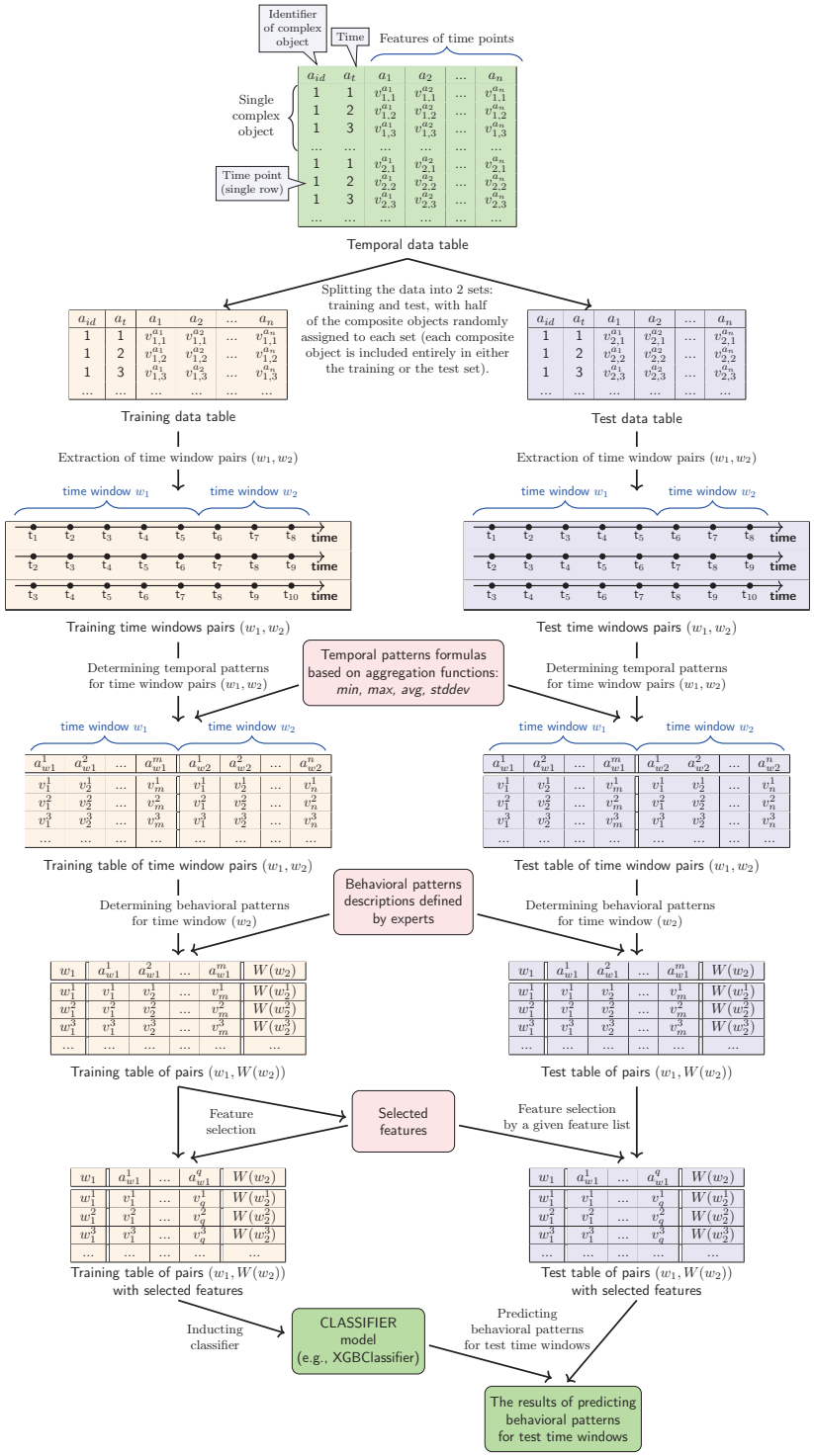


Figure 4. Illustration of the proposed method for behavioral pattern prediction.

The process begins with a temporal data table, where the rows correspond to time points of complex objects and the attributes describe the state of these objects at each time point. To enable model training and evaluation, the data are divided into two disjoint subsets: training and test sets, in a 50:50 ratio. The split is performed at the level of

complex objects, ensuring that records belonging to the same object do not appear in both subsets simultaneously.

In the second step, pairs of time windows are generated separately for the training and test data using a sliding window mechanism, as illustrated in Figure 3. Each window covers a defined time interval, and sliding the window along the time axis enables capturing changes in behavior in a continuous and localized manner. For each complex object, successive pairs of windows (w_1, w_2) are created, representing two adjacent segments of observations.

For each created pair of time windows (w_1, w_2), temporal patterns are determined—a set of features describing the complex object within the given time interval. These patterns are computed using aggregation functions such as *min*, *max*, *avg*, and *stddev*, applied separately to windows w_1 and w_2. The result is a table of time window pairs (w_1, w_2), where each pair is represented by a feature vector describing the properties of both windows.

Next, for each time window w_2 in the pair (w_1, w_2), a corresponding behavioral pattern $W(w_2)$ is determined, using behavioral pattern descriptions defined by experts (see Section 3.1). This pattern reflects the characteristic behavior of the observed object within the given time window. The outcome of this process is a table of pairs $(w_1, W(w_2))$, containing a set of features for window w_1 along with the corresponding behavioral label for window w_2.

The next step involves feature selection from the set of attributes determined for each window w_1. The goal of this stage is to reduce data dimensionality and retain only those features that are most relevant for predicting behavioral patterns. Three alternative feature selection methods, described in Section 6.2, are employed, and separate experiments are conducted using each of these methods to evaluate their impact on classification performance.

Finally, for the training data, the table of pairs $(w_1, W(w_2))$ with the selected features serves as a decision table, where the pattern $W(w_2)$ is the decision attribute and the features of window w_1 are the conditional attributes. Based on this table, a classifier is induced (e.g., an XGBClassifier model). The trained classifier is then applied to the test data to predict behavioral patterns in time window w_2 for each test time window w_1. The outcome is a set of predicted patterns for each test window.

6. Classification Procedure

In this section, we describe the steps involved in selecting the optimal classifier model for data related to the MED problem. In the experiments, the PyCommoDM library, developed in Python (version 3.12.1) by researchers at the Institute of Computer Science, University of Rzeszów, was used. This library [11] was implemented using the Pandas [12] and Scikit-learn [13] packages in versions 2.2.3 and 1.5.2, respectively, as well as the Xgboost package in version 2.1.2. In cases where a pseudo random number generator was applied, no seed was set, relying instead on the generator's default seeding. However, the experiments were repeated 10 times with the standard deviation calculated, which allows for an assessment of the reproducibility of the results. The classifiers used in the experiments were constructed based on the parameterless constructor of the class creating the given classifier.

6.1. Performance Metrics and Classifier Choice

We propose the $F1$-score as a measure of classifier quality. It combines *precision* and *recall* (*sensitivity*) into a single value and is especially useful when the data is unbalanced (it

helps balance the impact of false positives and false negatives in classification). However, before we define the $F1$-score, we will define precision and sensitivity.

For the case of two decision classes (binary classification), precision tells us how many of the predicted positive cases were actually correct, namely

$$precision = \frac{True\ positives\ (TP)}{True\ positives\ (TP) + False\ positives\ (FP)} \tag{5}$$

where

- *True positives* (TP)—the number of cases that were classified as positive (so-called true positives),
- *False positives* (FP)—the number of cases that were incorrectly classified as positive (so-called false positives).

Sensitivity, for the case of two decision classes, tells us how many of the actually positive cases the model detected. Namely,

$$sensitivity = \frac{True\ positives\ (TP)}{True\ positives\ (TP) + False\ negatives\ (FN)}, \tag{6}$$

where

- *False negatives* (FN)—the number of cases that were incorrectly classified as negative (omissions).

Precision and *sensitivity* are in conflict because increasing *precision* often reduces *sensitivity*. For example, we might have a model that detects cancer only in very obvious cases, resulting in high *precision*, but misses many less obvious cases (low *sensitivity*). On the other hand, increasing *sensitivity* often reduces *precision*, for example, a model detects cancer in almost all patients, but it leads to many false positives (low *precision*).

The $F1$-score for the case of two decision classes is defined as the harmonic mean of precision and sensitivity. Therefore, the $F1$-score combines both metrics into a single value that balances the trade-off between precision and sensitivity. Namely,

$$F1 = 2 \cdot \frac{precision \cdot sensitivity}{precision + sensitivity}. \tag{7}$$

The harmonic mean is more stringent than the arithmetic mean because it penalizes a large difference between *precision* and *sensitivity*. If one of these values is very low, the $F1$ will also be low. This provides a good balance between *precision* and *sensitivity*, even for unbalanced data.

However, in this work, the classification problem is a multi-class one. Therefore, we need to generalize the *precision*, *sensitivity*, and $F1$-score defined above. Several methods for such generalization can be found in the literature. In this work, we use the approach that is probably most frequently used. When it comes to calculating *precision* and *sensitivity*, we calculate these measures separately for each class, treating each as "positive" and the others as "negative" (i.e., the so-called one-vs-rest strategy). Now we need to choose a method for aggregating the $F1$-score measures calculated for all decision classes. In the literature, three main aggregation approaches are typically distinguished:

- Micro $F1$—computes global TP, FP, and FN values for all classes and calculates a single $F1$-score (favors majority classes),
- Weighted $F1$—a weighted average of the $F1$-scores for each class, with weights based on class sizes (majority classes have a greater influence on the result),

- Macro $F1$—the arithmetic mean of the $F1$-scores for each class, where all classes are treated equally (majority classes do not have a greater impact on the result).

Since the data analyzed in this study are imbalanced, we use the macro $F1$ method, which handles the influence of dominant classes on the classification result best among the three approaches mentioned above.

The results of the experiment performed to select the best classifier model for the MED problem are based on the $F1$-score. In Table 1, the results of nine well-known classification methods are presented.

Table 1. The results of the experiment performed to select the best classifier model for the MED problem.

Classifier	F1-Score	Standard Deviation
XGBClassifier	0.970	0.024
DecisionTree	0.964	0.021
GradientBoosting	0.954	0.022
RandomForest	0.909	0.047
AdaBoost	0.806	0.119
GaussianNB	0.745	0.105
MLPClassifier	0.714	0.054
LogisticRegression	0.666	0.064
k-NN (k = 3, Euclidean distance)	0.633	0.066

As we can see, XGBClassifier is the best, but it is not far ahead of several next-ranked classifiers.

6.2. Feature Selection Method

To justify the feature selection method used in this work, we compare the following three feature selection methods available in scikit-learn:

1. f_classif—this is a function used in feature selection that uses one-way analysis of variance (ANOVA) to determine how strongly each feature is related to a decision variable (decision); it is a statistical method that examines whether the values of a feature differ significantly between different classes.
2. mutual_info_classif—this is a function that calculates the *mutual information* between each conditional feature and the decision feature. It is a nonlinear measure of dependence and works well even when the dependencies between variables are more complex (it measures how much information about the decision variable a given feature provides).
3. Feature selection using the RandomForestClassifier classifier—the features selected are those that have high importance from the point of view of the classifier structure.

In Tables 2–4, the experimental results of the above three feature selection methods are shown for different numbers of top features. In each experiment, we use XGBClassifier and perform 10 experiments, from which we calculate the mean and standard deviation values.

As can be seen, all three feature selection methods achieve very similar results. However, the random forest-based method achieved the best result for 1000 features. Therefore, for the purposes of this work, we use the random forest-based feature selection method.

Table 2. Results of the feature selection experiment using the selection method f_classif.

Number of Features	F1-Score	Standard Deviation
500	0.978	0.010
1500	0.975	0.019
1000	0.968	0.023
100	0.948	0.025
200	0.948	0.041
20	0.946	0.013
50	0.934	0.028
10	0.900	0.028
5	0.884	0.100

Table 3. Results of the feature selection experiment using the selection method mutual_info_classif.

Number of Features	F1-Score	Standard Deviation
1000	0.978	0.011
1500	0.972	0.021
500	0.960	0.026
200	0.957	0.023
100	0.934	0.023
50	0.922	0.045
20	0.910	0.025
10	0.855	0.061
5	0.810	0.041

Table 4. Results of the feature selection experiment using the selection method RandomForestClassifier.

Number of Features	F1-Score	Standard Deviation
1000	0.979	0.006
500	0.972	0.019
1500	0.971	0.022
100	0.971	0.012
200	0.970	0.014
50	0.956	0.018
20	0.937	0.025
10	0.935	0.019
5	0.870	0.059

The selected 1000 attributes describe all essential characteristics of the patient in the context of respiratory failure treatment. In particular, these include the following attributes or groups of attributes:

- information about the method of mechanical ventilation or spontaneous breathing,
- Information about the neonate's birth weight.

- Attributes describing FiO2 values (FiO2—fraction of inspired oxygen; exactly the percentage of oxygen in the breathing mixture that the patient inhales), the PaO2/FiO2 ratio (assessment of the patient's respiratory function, where PaO2 is the partial pressure of oxygen in arterial blood), and the results of other laboratory tests.
- Information about culture results for sepsis and ureaplasma.
- Information on the presence of PDA (patent ductus arteriosus) and whether it has been closed.
- Information about the administration of medications such as antibiotics, macrolides, steroids, surfactant, etc.
- Information about the presence of RDS (respiratory distress syndrome).

All these parameters are analyzed at each time point, and the conditional attributes in the decision table aggregate their values within a time window (e.g., the minimum, maximum, or average value of a numerical attribute within the window, the number of occurrences of a symbolic attribute value within the window, etc.).

6.3. Adjusting the Sensitivity Level to Decision Classes

Although the XGBClassifier achieves a very high $F1$-score, in practical applications it is often necessary to achieve high sensitivity (recall) for individual decision classes. For example, in the MED problem, doctors would prefer the highest-quality classifier for classes 0 and 1, because these are associated with a worse patient condition than class 2. Therefore, in this section, we describe an experiment related to adjusting the classifier's sensitivity, performing experiments for different weight thresholds for classes 0 and 1. In these experiments, unlike the classical approach (where the class with the highest weight calculated for the test object is classified), we use a special approach based on the following Algorithm 1.

Algorithm 1: Classification rule for sensitivity adjustment

Input:
w_0, w_1 - weights generated by the classifier for classes 0 and 1,
θ_0, θ_1 - experimentally selected thresholds for weights of classes 0 and 1.
Output: Selected decision class from the set $\{0, 1, 2\}$

```
if w_0 > θ_0 then
    Return class 0;
end
else if w_1 > θ_1 then
    Return class 1;
end
else
    Return class 2;
end
```

7. Results of the Experiments and Discussion

7.1. Results of Experiments with the Selection of Weight Thresholds

It is easy to see that the approach proposed in Section 6 allows the sensitivity to both classes (class 0 and class 1) to be adjusted. In Table 5, we present the experimental results for different weight thresholds for class 0 and class 1. The method uses 1000 selected features.

Table 5 shows several interesting rows showing specific result configurations for both decision classes. It shows that by adjusting the weight threshold for classes 0 and 1,

the sensitivity of the classifier can be fine-tuned. It is worth noting that in the table, columns Recall0, Recall1, and Recall2 correspond to sensitivities for classes 0, 1, and 2, respectively. For example, in order to achieve high sensitivity in detecting the improvement of a patient's condition to class 2, the parameters should be set to $\theta_0 = 0.2$ and $\theta_1 = 0.9$. In this case, the sensitivity for class 2 recognition is 0.999, but the sensitivity for class 0 recognition is 0.958. On the other hand, to achieve high sensitivity in detecting the deterioration of a patient's condition to class 0, the parameters should be set to $\theta_0 = 0.1$ and $\theta_1 = 0.3$. In this case, the sensitivity for class 0 recognition is 0.992, but the sensitivity for class 2 recognition is 0.996.

Table 5. Results of the experiments with the selection of weight thresholds for classes 0 and 1.

θ_0	θ_1	Recall0	StdDev	Recall1	StdDev	Recall2	StdDev
0.1	0.3	0.992	0.009	0.977	0.009	0.996	0.002
0.1	0.7	0.987	0.030	0.971	0.011	0.998	0.001
0.1	0.8	0.987	0.017	0.950	0.023	0.999	0.001
0.1	0.5	0.976	0.032	0.975	0.005	0.996	0.002
0.4	0.3	0.974	0.014	0.974	0.016	0.995	0.002
0.2	0.3	0.968	0.060	0.981	0.007	0.994	0.002
0.1	0.2	0.967	0.060	0.976	0.016	0.993	0.003
0.2	0.2	0.965	0.039	0.978	0.007	0.994	0.002
0.3	1.0	0.963	0.057	0.000	0.000	1.000	0.000
0.2	0.5	0.961	0.045	0.974	0.021	0.997	0.001
0.4	0.2	0.961	0.025	0.983	0.006	0.992	0.004
0.1	0.4	0.959	0.073	0.968	0.017	0.996	0.002
0.7	0.5	0.959	0.025	0.980	0.007	0.996	0.002
0.2	0.1	0.958	0.058	0.986	0.006	0.991	0.004
0.2	0.9	0.958	0.066	0.962	0.015	0.999	0.001
0.2	0.8	0.952	0.069	0.968	0.012	0.998	0.001
0.3	0.4	0.952	0.064	0.980	0.005	0.995	0.002

7.2. Summary of the Quality of Recognizing Deterioration, Improvement, and Permanent Transitions in Behavior Patterns

Although the classifier's sensitivity results presented in Table 5 are very good and satisfactory from an application perspective, one might suspect that the classifier only performs well in cases where the patterns are stable, i.e., transitions from pattern 2 to 2, 1 to 1, or 0 to 0. If this is the case, the classifier's quality would indeed be questionable. However, in medical practice, the most interesting transitions are those related to the deterioration of the patient's condition, i.e., transitions from 2 to 1, 2 to 0, and 1 to 0. Therefore, we present the results of an experiment comparing the classifier's sensitivity for cases of deterioration, improvement, and stability. In particular, four experiments were conducted with thresholds $\theta_0 = 0.2$ and $\theta_1 = 0.2$ on the test data used in the experiments from Table 5, where in each of these experiments the test objects were restricted to a specific subset:

- In the first experiment, the entire test set was evaluated (as in Table 5).
- In the second experiment, only those test cases representing a deterioration of the patient's condition were evaluated.

- In the third experiment, only those test cases representing an improvement in the patient's condition were evaluated.
- In the fourth experiment, only those test cases representing a stable patient condition were evaluated.

Table 6 shows that classification quality is indeed best when the pattern remains constant. When the pattern deteriorates, the classification quality decreases slightly, especially for class 0. However, the quality remains good, especially for class 1. Interestingly, when the patient's condition improves, the classification quality of such a change is also high.

Table 6. Results of the experiment comparing the sensitivity of the classifier for the cases of deterioration, improvement, and stable level of the behavior pattern.

Direction	Recall0	STD	Recall1	STD	Recall2	STD
All	0.965	0.039	0.978	0.007	0.994	0.002
Worse	0.864	0.116	0.922	0.064	—	—
Better	—	—	0.921	0.062	0.947	0.018
Stable	0.998	0.004	0.982	0.006	1.000	0.001

7.3. Discussion on Main Achievements

The goal of the study presented was to predict early on (while the object matches one behavioral pattern) whether the object will later match another behavioral pattern. This method can be considered more advanced than usually performed experiments, which are focused solely on identifying behavioral patterns. The overview of the literature on the treatment of neonate respiratory failure shows that many studies are retrospective. Open datasets specific to neonatal continuous monitoring for respiratory failure are less common, especially those including high-frequency signals (ventilator flow, oxygenation over time, etc.). Prediction horizons (i.e., how far ahead one can predict failure) are often not very short (minutes to an hour) in neonates and generally several days for more general outcomes (BPD, mortality).

There exist papers devoted to studying the treatment of neonatal respiratory failure and predicting neonatal health conditions. However, in these studies, the change in behavioral patterns is not predicted. In the paper [5], the authors demonstrated that machine learning models such as Random Forest and bagged CART can more accurately predict the mortality of critically ill neonates than conventional scoring systems such as the Neonatal Therapeutic Intervention Scoring System (NTISS) and Score for Neonatal Acute Physiology Perinatal Extension II (SNAPPE-II). The study in [6] identified the independent risk factors of respiratory failure in Neonatal Respiratory Distress Syndrome (NRDS) patients and used them to construct and evaluate a respiratory failure risk prediction model for NRDS, which may help clinicians to identify and intervene in the early stage. In [4], the authors used multiple ML methods (Random Forest, SVM, etc.) to predict respiratory distress syndrome (RDS) occurrence in premature infants. The best model, Random Forest, achieved an AUC of approx. 0.843 and an accuracy of approx. 0.815. In [7], the authors developed a decision-support tool for predicting extubation failure (EF) in neonates with bronchopulmonary dysplasia (BPD) using a set of machine learning algorithms. The study indicated that the XGBoost model was significant in predicting EF in BPD neonates with mechanical ventilation, which is helpful in determining the right extubation time among neonates with BPD to reduce the occurrence of complications.

The goal of this paper was achieved successfully (cf. Table 6). Comparing the classifier's sensitivity for cases of deterioration, improvement, and stability shows that classification quality is the best when the pattern remains constant. When the patient's condition improves, the classification quality of such a change is also high. Importantly, in the proposed model, when the pattern deteriorates, the classification quality decreases only slightly. Despite the good research results, it is worth noting that the presented method also has certain limitations. Firstly, as already mentioned, the data is outdated and therefore cannot be used to build a treatment-supporting system. Moreover, the number of time points where the patient's condition is severe is small, which results in lower-quality prediction of the transition from a better pattern to a worse one. The number of behavioral patterns is small (only three), which certainly does not account for possible more detailed behavioral patterns that would be interesting from a practical perspective. Finally, the constructed classifier, which predicts changes in behavioral patterns, is of high quality, but no methods for explaining the decisions have been tested for them (e.g., methods known from XAI).

7.4. Linking Pattern Change Recognition with Concept Drift

The problem of predicting changes in process models can be treated as a special case considered in a subfield of machine learning called *concept drift* (cf. [14,15]). Generally speaking, it considers a situation where the properties of objects belonging to particular decision classes change over time. This makes it practically impossible to construct a classifier that operates effectively over time. Methods for fast and efficient adaptation of classifiers are needed. Therefore, if we treat a behavior pattern as the definition of a certain concept represented by a binary decision attribute (an object matches or does not match the pattern), then a change in the behavior pattern means a change in the concept, which in a sense "drifts" from one behavior pattern to another. Note that calculating the value of such an attribute requires knowledge of the behavioral pattern, which in the approach discussed here must be defined by domain experts. Incidentally, domain experts also define other patterns to which an object can fit. Following this line of thought, a more natural analogy is a decision table with a decision attribute with multiple values. Such a table would often be contradictory, because an object can fit multiple behavioral patterns simultaneously. Given this representation, the problem of predicting changes in the process may be treated as the problem of predicting changes in the decision class to which an object belongs. Nevertheless, the so-called active methods for dealing with concept drift discussed in the literature can be considered potential solutions to the problem in question. However, the proposed approach is not about recognizing that a behavioral pattern has changed (which requires adaptation, which is often the goal of methods developed in approaches related to concept drift), but about predicting that a behavioral pattern change will occur. This property significantly distinguishes the methods described in this paper from those typically used in concept drift. This is particularly evident in the case of predicting a potential change in a behavioral pattern caused by an important event that may trigger a change in the pattern.

8. Conclusions

The paper aimed to propose a model that can predict early on (while the object matches the first pattern) whether the object will match the second pattern in the near future. Our method can be considered more advanced than the experiments usually presented in the literature, which focus solely on identifying behavioral patterns. A crucial aspect of the study was domain knowledge, irreplaceable in defining patterns. In medical practice, the most interesting transitions are those related to the deterioration of the patient's

condition, i.e., transitions from 2 to 1, 2 to 0, and 1 to 0. Comparing the classifier's sensitivity for cases of deterioration, improvement, and stability reveals that classification quality is best when the pattern remains constant. However, in the proposed model, when the pattern deteriorates, the classification quality decreases only slightly. When the patient's condition improves, the classification quality of such a change is also high.

The approach to predicting changes in behavioral patterns can be applied in various domains. However, the prerequisite is that the behavioral patterns must be defined by experts based on selected parameters of complex objects. This relatively high level of involvement from domain experts may be regarded, on the one hand, as a drawback of the method. On the other hand, it enables the injection of domain knowledge into machine learning models, which can improve their quality, especially when the available datasets are relatively small. Moreover, such an approach may lead to the development of so-called AI agents, which have recently been gaining significant popularity. One can easily imagine an AI agent that interacts with a user (e.g., a physician) and answers questions regarding the possibility of a given patient transitioning to a different behavioral pattern. Furthermore, if, during such an interaction, the user wishes to change the definition of the behavioral patterns being used, the AI agent can recalculate all the models used in the AI framework in real time and then continue the dialogue with the user based on the new models. The authors of this study plan to develop an AI assistant to support the treatment of respiratory failure, including the prediction of patient transitions to different behavioral patterns.

Author Contributions: A.S.: conceptualization, methodology, software, investigation, data curation, writing draft preparation; J.G.B.: conceptualization, methodology, validation, investigation, supervision, funding acquisition; U.B.: methodology, validation, investigation, writing draft preparation; P.K.: resources, data curation, methodology; S.B.-S.: resources, data curation, methodology. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: The medical data used in the experiments described in this publication were not collected as part of a medical experiment but rather during routine clinical activities in a hospital. Therefore, according to current Polish law (Legal justification: Act of 5 December 1996 on the Professions of Physician and Dentist—the act is still in force today) https://isap.sejm.gov.pl/isap.nsf/download.xsp/WDU19970280152/U/D19970152Lj.pdf (accessed on 5 November 2025) their analysis does not require approval from a bioethics committee.

Informed Consent Statement: The data were obtained from the Jagiellonian University Medical College (CM UJ) in 2005 and were anonymized (i.e., none of the members of the mathematical and computer science research team had access to the patients' identities). Consequently, under the personal data protection law in force in Poland in 2005 (Act of 29 August 1997 on the Protection of Personal Data, in force from 30 April 1998 to 25 May 2018) https://isap.sejm.gov.pl/isap.nsf/download.xsp/WDU19971330883/O/D19970883.pdf (accessed on 5 November 2025) the use of such data for scientific research was permitted and consent from participants was not required.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to patients origin.

Acknowledgments: This work was partially supported by the Centre for Innovation and Transfer of Natural Sciences and Engineering Knowledge of University of Rzeszów, Poland. The authors are very grateful to the Reviewers for the valuable comments which helped to improve the final version of the paper.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Bazan, J.G. Hierarchical classifiers for complex spatio-temporal concepts. In *Transactions on Rough Sets IX*; Peters, J.F., Skowron, A., Eds.; Springer: Berlin/Heidelberg, Germany, 2008; pp. 474–750.
2. Pedrycz, W.; Chen, S.-M. *Time Series Analysis, Modeling and Applications. A Computational Intelligence Perspective*; Springer: Berlin/Heidelberg, Germany, 2013.
3. Pokkuluri, K.S.; Khang, A. Deep Learning for Identification of Behavioral Changes. In *Behavioral Economics and Neuroeconomics of Health and Healthcare*; Jayasankara Reddy, K., Ed.; IGI Global Scientific Publishing: Hershey, PA, USA, 2025; pp. 65–78.
4. Farshid, P.; Mirnia, K.; Rezaei-Hachesu, P.; Maserat, E.; Samad-Soltani, T. Developing a model to predict neonatal respiratory distress syndrome and affecting factors using data mining: A cross-sectional study. *Int. J. Reprod. Biomed.* **2023**, *21*, 909–920 [CrossRef] [PubMed]
5. Hsu, J.F.; Yang, C.; Lin, C.Y.; Chu, S.M.; Huang, H.R.; Chiang, M.C.; Wang, H.C.; Liao, W.C.; Fu, R.H.; Tsai, M.H. Machine Learning Algorithms to Predict Mortality of Neonates on Mechanical Intubation for Respiratory Failure. *Biomedicines* **2021**, *9*, 1377. [CrossRef] [PubMed]
6. Lei, Y.; Qiu, X.; Zhou, R. Construction and evaluation of neonatal respiratory failure risk prediction model for neonatal respiratory distress syndrome. *BMC Pulm. Med.* **2024**, *24*, 8. [CrossRef] [PubMed]
7. Tao, Y.; Ding, X.; Guo, W.L. Using machine-learning models to predict extubation failure in neonates with bronchopulmonary dysplasia. *BMC Pulm. Med.* **2024**, *24*, 308. [CrossRef] [PubMed]
8. Pawlak, Z. *Rough Sets: Theoretical Aspects of Reasoning About Data*; Kluwer: Norwell, MA, USA, 1991.
9. Pawlak, Z.; Skowron, A. Rudiments of rough sets. *Inf. Sci.* **2007**, *177*, 3–27. [CrossRef]
10. Synak, P. Temporal Aspects of Data Analysis: A Rough Set Approach. Ph.D. Thesis, The Institute of Computer Science of the Polish Academy of Sciences, Warsaw, Poland, 2003. (In Polish)
11. Available online: https://github.com/aszczur/PyCommoDM/tree/main/bpatterns (accessed on 31 October 2025).
12. Available online: https://pandas.pydata.org/ (accessed on 31 October 2025).
13. Available online: https://scikit-learn.org/stable/ (accessed on 31 October 2025).
14. Kurian, J.F.; Allali, M. Detecting drifts in data streams using Kullback-Leibler (KL) divergence measure for data engineering applications. *J. Data Inf. Manag.* **2024**, *6*, 207–216. [CrossRef]
15. Lu, J.; Liu, A.; Dong, F.; Gu, F.; Gama, J.; Zhang, G. Learning under Concept Drift: A Review. *IEEE Trans. Knowl. Data Eng.* **2019**, *31*, 2346–2363. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

applied sciences

MDPI

Article

Possibilities of Reflecting the Mechanical Properties of Non-Absorbable Surgical Meshes in an AI-Based Model in the Context of Industry 4.0/5.0

Marek Andryszczyk, Izabela Rojek, Tomasz Bednarek and Dariusz Mikołajewski *

Faculty of Computer Science, Kazimierz Wielki University, Chodkiewicza 30, 85-064 Bydgoszcz, Poland; marek.andryszczyk@ukw.edu.pl (M.A.); izabela.rojek@ukw.edu.pl (I.R.); tomasz.bednarek@ukw.edu.pl (T.B.)
* Correspondence: dariusz.mikolajewski@ukw.edu.pl

Abstract: Non-absorbable surgical meshes are key biomedical materials used for tissue reinforcement, designed for durability, biocompatibility, and mechanical stability in clinical applications. The mechanical properties of these meshes, such as tensile strength, elasticity, and porosity, are crucial for their long-term performance and integration with host tissue. In the context of Industry 4.0/5.0, emphasis is placed on integrating intelligent technologies, such as real-time data acquisition and advanced computational modeling, to improve the design and production of surgical meshes. Computational models simulate the mechanical behavior of meshes under physiological conditions, enabling precise optimization of their material properties and design. In this article, we propose potential artificial intelligence (AI)-based approaches for future research, such as machine learning (ML), for analyzing large datasets from computational and experimental studies to identify optimal mesh configurations. The direction of tensile loading significantly influences the mechanical response of the mesh. Transversely stretched specimens demonstrated higher maximum failure forces and greater fatigue resistance than longitudinally stretched specimens, both in sutured and unsutured conditions. Suturing the mesh to biological tissue significantly reduced its mechanical strength and stiffness, demonstrating a weakening effect at the mesh-tissue interface. Cyclic loading revealed a gradual decrease in strength in all specimens, suggesting fatigue, but transversely stretched meshes maintained higher forces for >1000 cycles than longitudinally stretched meshes. The observed differences in mechanical behavior can be attributed to the anisotropic mesh structure and mechanical suturing effects, which introduce stress concentrations and structural discontinuities. These results emphasize the importance of considering both directionality and surgical technique when selecting and implementing mesh implants. Both AI-based models achieved scores above 80%, demonstrating their clinical utility and the potential for development toward prediction accuracy above 85–90% in clinical settings. Future research should incorporate AI-based computational models to improve predictive capabilities, ultimately leading to the development of more effective, patient-specific surgical meshes.

Keywords: computer science; computational analysis; artificial intelligence; machine learning; automation; surgical meshes; Industry 4.0; Industry 5.0

1. Introduction

Nonabsorbable surgical meshes are medical implants designed for long-term or permanent soft tissue reinforcement. They are made of synthetic or biological materials that remain in

the body indefinitely without being absorbed or degraded [1]. These meshes serve critical purposes in a variety of surgical procedures, particularly in the repair of hernias, prolapses, and other structural soft tissue defects. The primary purpose of nonabsorbable surgical meshes is to provide permanent mechanical support, ensuring stability while the body heals and adapts. Unlike absorbable meshes, which dissolve over time, these materials maintain their integrity and tensile strength for the life of the patient [2]. Their durability makes them particularly suitable for cases where permanent reinforcement is necessary, such as large or recurrent hernias. Nonabsorbable surgical meshes are most commonly used in abdominal wall repair, but their applications also include urology, gynecology, and reconstructive surgery [3]. Advances in imaging and materials science have improved preoperative planning and mesh performance, allowing surgeons to achieve better outcomes. The global market for these devices continues to grow, reflecting the increasing incidence of hernias and other soft tissue defects. Nonabsorbable meshes originated in the mid-20th century with the development of polymers such as polypropylene and expanded polytetrafluoroethylene. These materials were chosen for their biocompatibility, strength, and resistance to infection or degradation. Over time, innovations in mesh design have led to lighter, more flexible products with improved pore sizes that promote tissue in growth while minimizing complications. Modern nonabsorbable meshes use advanced manufacturing technologies, including 3D knitting and laser cutting, to create customizable structures tailored to specific clinical needs [4]. Some meshes are coated with antimicrobials or hydrophilic layers to reduce the risk of infection and improve biocompatibility. Others incorporate composite designs that combine different materials to optimize strength and integration with the surrounding tissue [5,6]. Despite their advantages, nonabsorbable meshes are not without their challenges, including the risk of infection, chronic pain, and mesh erosion. Addressing these complications has led to ongoing research to improve mesh properties and surgical techniques. The use of robotic and laparoscopic technologies has also increased the precision of mesh placement, reducing recovery time and complications. With increasing regulatory scrutiny, manufacturers are now focusing on rigorous testing and clinical evaluation of new mesh designs [5,6]. This has led to the development of meshes that balance strength with patient comfort, including ergonomic designs that conform to anatomical structures [7,8]. Nonabsorbable surgical meshes play a critical role in modern medicine, providing long-term solutions for difficult surgical repairs.

The mechanical properties of these meshes, such as tensile strength, elasticity, and porosity, are essential for their long-term performance and integration with host tissue [9,10]. Towards Industry 4.0/5.0, the emphasis is on integrating intelligent technologies, such as real-time data acquisition and advanced computational modeling, to improve the design and production of surgical meshes [11]. Computational models simulate the mechanical behavior of meshes under physiological conditions, enabling precise optimization of their material properties and structural designs [12]. Advanced simulations to date use finite element analysis to predict stress distribution, strain, and failure mechanisms in the mesh under dynamic loading conditions. In this article, we propose the use of data-driven approaches such as artificial intelligence and machine learning to analyze large datasets from computational and experimental studies to identify optimal mesh configurations. Industry 4.0 emphasizes connectivity and automation, facilitating the integration of sensor technologies into the manufacturing process to monitor material quality and performance in real time. Additive manufacturing (3D printing) combined with computational models enables the production of custom-designed meshes with patient-specific properties, allowing for improved clinical outcomes [5,13]. Moving toward Industry 5.0, a human-centric approach focuses on incorporating human expertise with computational advances to develop more sustainable, efficient, and user-centric surgical meshes [14]. Computational modeling in the context of Industry 4.0/5.0 revolutionizes the design and evaluation

of nonresorbable surgical meshes, supporting innovation and improving patient care through precision and intelligent manufacturing [6].

Nonabsorbable surgical meshes are evaluated for mechanical properties such as tensile strength, elasticity, pore size, and fatigue resistance to ensure long-term biocompatibility and structural integrity. In the context of Industry 4.0, advanced manufacturing and sensor integration enable real-time monitoring and quality control of these mechanical attributes. New AI-based models contribute to predictive analysis and optimization of mesh performance based on massive clinical and material testing datasets. Industry 5.0 introduces a human-centric paradigm, where ergonomic feedback and clinician-in-the-loop mechanisms drive iterative improvements in mesh design. These systems aim to improve user interaction, comfort, and implant outcomes through continuous feedback loops involving surgeons and patients. However, the incorporation of such human-centric innovations into current AI-based models remains largely conceptual and underdeveloped in practice. While these ideas suggest a promising future for co-adaptive medical devices, current methodologies often lack the depth to justify true adaptation to Industry 5.0. Clinician feedback is sometimes used to refine models, but the integration remains minimal and is not systematically validated. Ergonomics assessments are rarely standardized or quantitatively linked to mesh performance metrics in AI systems. Thus, despite references to Industry 5.0, this link has not yet materially influenced the empirical framework or results.

The mechanical properties of nonabsorbable surgical meshes, such as tensile strength, pore size, and fatigue resistance, are also fundamental to their long-term clinical performance. Traditionally, challenges in this area include manufacturing variability, lack of standardized testing, and limited in vivo predictability. These fundamental concerns must be clearly defined before moving to the opportunities offered by digital and AI-assisted approaches. Advanced computational tools within established materials science frameworks are needed, as well as a combination of classical mechanical testing with AI-based modeling to help explain how data from physical experiments can most effectively be fed into predictive systems. The leap into Industry 4.0/5.0 paradigms requires grounding in traditional engineering workflows to fully convey its significance. In the AI-based model, Industry 4.0 is reflected in data acquisition, simulation, and pattern recognition. In the case of Industry 5.0, concepts such as personalization, ergonomic design, and feedback loops for physicians remain largely conceptual and currently under-implemented. So, while the study suggests innovation, its structural flow and depth of integration limit its effectiveness in fully capturing the Industry 4.0/5.0 transition.

This manuscript introduces a novel study of the mechanical properties of nonabsorbable surgical meshes, uniquely integrating their characterization with Industry 4.0/5.0 paradigms. The study innovatively uses AI-based modeling to predict and optimize mesh performance, combining material science and smart manufacturing. Unlike previous work, it contextualizes surgical mesh evaluation within cyber-physical systems, enabling real-time feedback and intelligent decision-making in the future. The manuscript approach uses ML to discriminate between complex, nonlinear relationships between mesh structure, composition, and biomechanical behavior. It pioneers the incorporation of digital twins into the medical textile field, enhancing predictive maintenance and customization capabilities. By aligning mesh production and evaluation with digital transformation goals, it offers scalable, data-driven solutions for quality assurance and personalized medicine. This fusion of traditional biomedical engineering with cutting-edge AI and Industry 4.0/5.0 technologies represents a significant and original contribution to both the medical and manufacturing research communities.

This study addresses Industry 4.0/5.0 by demonstrating how data from physical and mechanical tests of non-absorbable surgical meshes can be transformed into predictive

AI-based digital models. It aligns with smart manufacturing principles by combining laboratory measurements of material behavior with computational tools that can support automated quality assessment and design optimization. By integrating the physical and digital domains, this work contributes to establishing biomedical materials DTs as a key element of cyber-physical Industry 4.0 systems. The study also reflects Industry 5.0 values, enabling more human-centric, customizable, and precise medical device development, supported by AI-assisted design rather than simple automation. The study positions biomedical materials testing within a broader smart manufacturing ecosystem, where data, models, and expertise work together to enhance product reliability and innovation.

The aim of this study is to experimentally investigate the influence of tensile loading direction and suture integration on the mechanical behavior and fatigue resistance of nonabsorbable polypropylene surgical meshes.

2. Materials and Methods

2.1. Material

The study used the Optomesh Macropore surgical mesh (TRICOMED SA, Łódź, Poland) (Figure 1) [15]. The study uses Optomesh Macropore as a representative mesh, clearly describing its structural and mechanical properties. The macroporous architecture of this mesh and the polypropylene composition make it suitable for assessing both biomechanical performance and long-term durability. The rationale for selecting Optomesh Macropore is its availability, popularity, and representativeness of parameters. In the context of Industry 4.0, Optomesh's AI-based modeling enables detailed simulations and predictions based on mechanical test data. These models help refine design parameters and identify potential failure modes under physiological loads. The well-documented properties of the mesh allow for more accurate data input into the computational framework, combining material science with digital innovation. The company's extensive offer includes primarily medical implants, specialist dressings and compression products for scar rehabilitation. Optomesh is a non-resorbable surgical mesh recommended for reconstructive procedures to fill soft tissue defects. It is made using a knitting technique from transparent, monofilament polypropylene yarn with high biocompatibility with the body. It is fully synthetic, does not contain any allogenic or animal additives. These meshes are characterized by:

- High mechanical strength ensuring a permanent connection of the mesh to the tissue;
- Multidirectional shape memory;
- Low surface mass;
- Edge fraying and resistance to thread pulling during sewing close to the edge;
- High resistance to seam tearing;
- Minimal risk of bacterial infection, thanks to the use of a monofilament structure.

Optomesh surgical meshes are used in cases such as abdominal hernias, postoperative hernias, inguinal hernias, femoral hernias, periumbilical hernias, and hernias in the postoperative scar. Main physical parameters:

- Surface mass of the meshes–varies depending on the mesh variant from 60 to 85 g/m^2, ensuring comfort of use for the patient;
- Mesh thickness–similarly to the above parameter, this value varies depending on the mesh variant and ranges from about 0.4 mm to about 0.8 mm, ensuring high mechanical strength;
- Thread thickness–the meshes are made of monofilament yarn suitable for large-pore meshes, therefore ensuring high resistance to tearing out of the sutures with which the meshes are attached to the tissues.

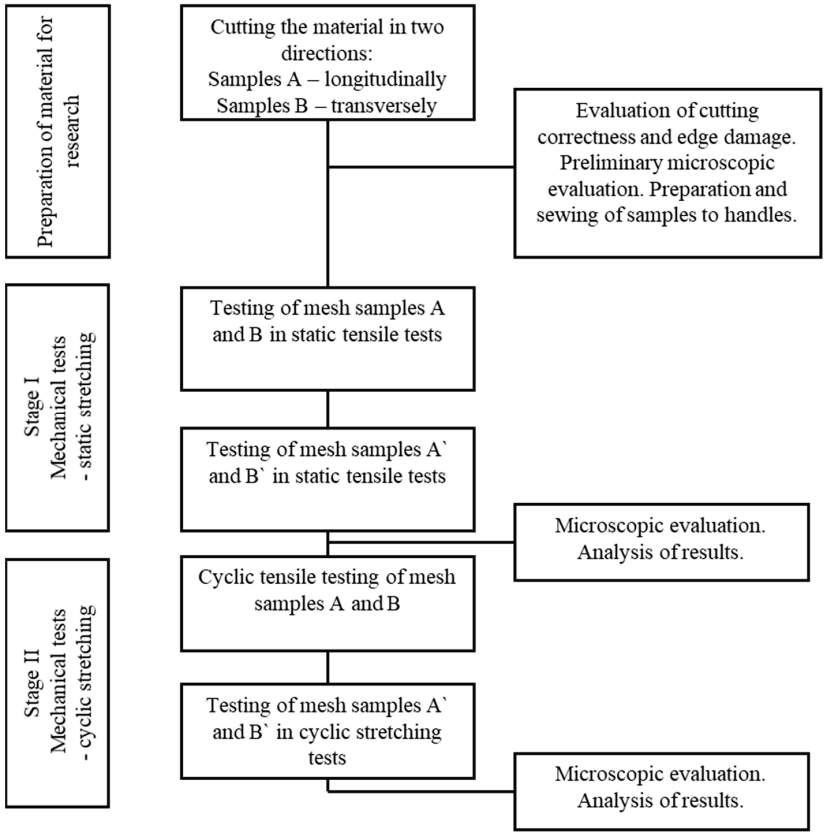


Figure 1. Protocol of the study.

Nonabsorbable surgical meshes must demonstrate high tensile strength, elasticity, and dimensional stability to withstand the long-term mechanical stresses of the human body without degradation. These meshes are typically tested for uniaxial tensile loading, cyclic fatigue, and fracture toughness to simulate physiological conditions and ensure reliable performance. Key mechanical properties such as ultimate tensile strength (UTS), Young's modulus, elongation at break, and suture retention strength are critical benchmarks for biocompatibility and mechanical integrity. Industry 4.0/5.0 enables advanced AI-based modeling to analyze these test results, identify patterns, and optimize mesh design for specific clinical scenarios. The AI-based model reflects feedback from high-throughput mechanical testing, using machine learning algorithms to predict material behavior under a variety of anatomical and surgical conditions. These models confirm that the meshes meet fundamental criteria for medical materials, including mechanical compatibility, minimal strain under stress, and fatigue resistance. As a result, the integration of AI into mesh testing and analysis confirms their suitability for long-term implantation and supports precise design for patient-specific surgical applications.

Representative values for key mechanical properties of nonresorbable surgical meshes, typically used in hernia repair and other soft tissue reinforcement applications can vary depending on the material (e.g., polypropylene, polyester, ePTFE), weave pattern, and manufacturing method, but they serve as critical benchmarks:

1. Ultimate tensile strength:
    - Range: 15–120 MPa;
    - Typical polypropylene mesh: 30–50 MPa;
    - A higher UTS ensures the mesh can withstand internal stresses without failure;

2. Young's modulus:
    - Range: 0.1–1.0 GPa (100–1000 MPa);
    - Typical polypropylene mesh: 200–500 MPa;
    - Reflects stiffness: must balance support and flexibility to avoid discomfort or damage to surrounding tissue;
3. Elongation at Break:
    - Range: 50–150%;
    - Typical polypropylene mesh: 70–100%;
    - Indicates mesh elasticity; essential to accommodate dynamic physiological movements without tearing;
4. Suture Retention Strength:
    - Range: 5–20 N;
    - Typical polypropylene mesh: 10–15 N;
    - Ensures that sutures do not tear through the mesh during or after implantation, maintaining fixation.

These mechanical properties are aligned with ISO 13934-1 [16] and ASTM D5035 [17] testing standards and are considered critical indicators of mechanical integrity, biocompatibility, and long-term performance. AI-based models using these values can accurately predict in vivo behavior and suitability, verifying that the meshes meet the stringent requirements for permanent implantation in medical applications.

2.2. Methods

The study was conducted on Optomesh Macropore surgical meshes made of monofilament polypropylene. The aim was to determine their mechanical properties in different stretching directions and in conditions simulating real clinical application (suturing with tissue). The study was conducted according to the protocol presented in Figure 1.

Preparation of the material for testing included cutting the each mesh into four smaller fragments in two perpendicular directions:

- Samples A—cut longitudinally (Figure 2),
- Samples B—cross-cut (Figure 3).

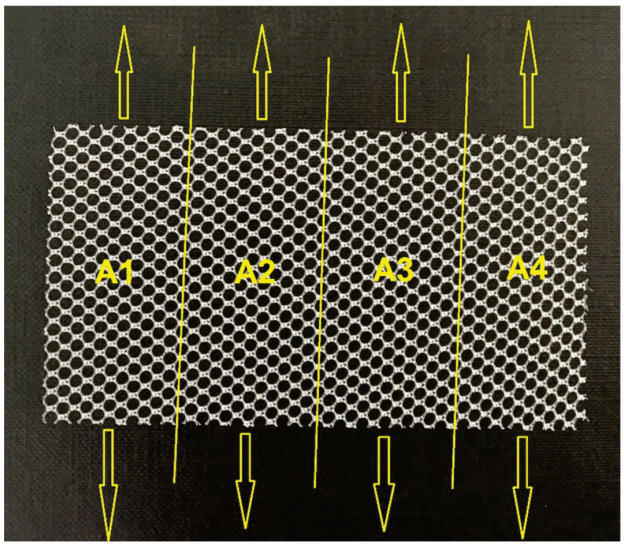


Figure 2. Optomesh Macropore surgical mesh. Lines show where the mesh was cut into 50 mm × 25 mm samples. Arrows show the direction in which samples A were stretched.

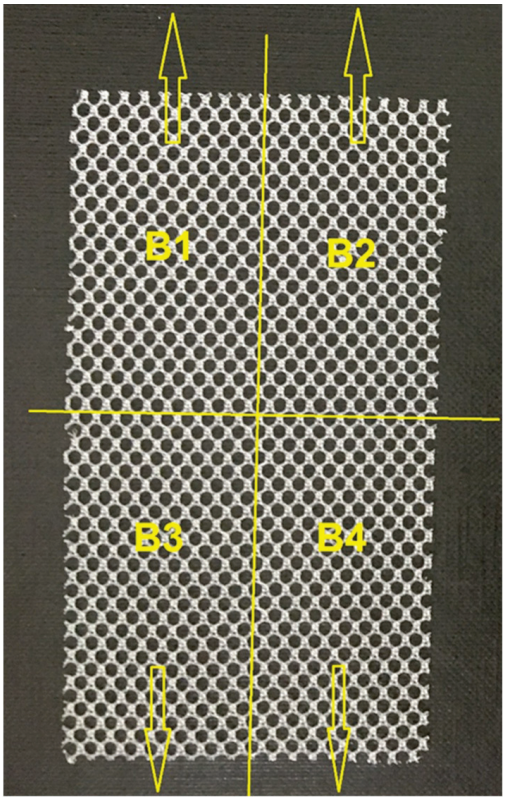


Figure 3. Optomesh Macropore surgical mesh: lines show where the mesh was cut into 50 mm × 25 mm samples. Arrows show the direction in which the B samples were stretched.

156 samples were used in the study. The mesh samples were statically and cyclically tested in two directions (Figures 2 and 3) to check for differences in the weave and thus in the mesh production and consequently in the tensile strength. The samples were rectangular in shape, 50 mm long and 25 mm wide. At this stage, the quality of the cut and any damage to the edges were assessed, and a preliminary microscopic analysis of the mesh structure was performed (Figure 4) (sample data set in the Supplementary Materials).

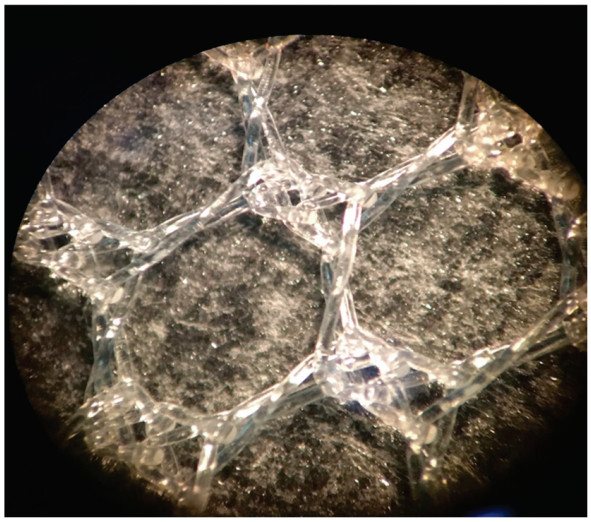


Figure 4. Microscopic image of Optomesh Macropore surgical mesh.

The samples were sewn together with prepared fragments of bovine skin (2 mm thick), which were intended to simulate anatomical conditions. This type of sample allows for an approximate determination of how the meshes together with surgical threads will behave in the human body. The difference between the mesh samples placed directly in the forceps of the testing machine and the samples shown in Figure 5 is that in the case of the former, the force is distributed over the entire width of the sample. The situation is different in the case of meshes sewn between the skin, because they are placed in the forceps only with the skin part, and the tensile force acts only in the places where the seams occur (two points on each side).

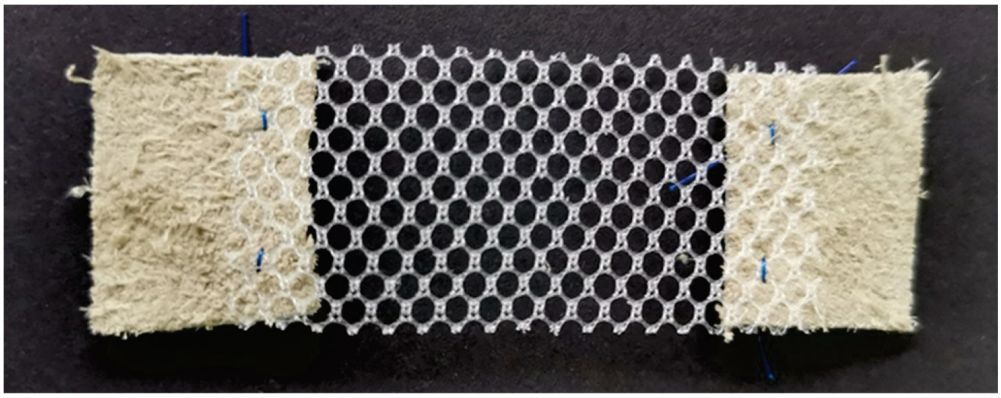


Figure 5. A mesh sample sewn with a single seam onto two pieces of cowhide.

The suture used in the study was Prolene [8] (Figure 6). This is a synthetic, nonabsorbable suture made of polypropylene, dyed blue to increase its visibility in the surgical field. It is indicated for soft tissue approximation and ligation, and is used in cardiology, ophthalmology, and neurology. Prolene suture strengths:

- Used in over 100 million people worldwide, including 8 out of 10 coronary artery bypass grafting procedures;
- High tensile strength;
- Unique manufacturing process that ensures a uniform suture diameter, eliminating weak points in the suture.



Figure 6. Single pack of Prolene suture.

Prolene is a nonabsorbable suture that causes minimal inflammatory reactions in the tissue. It is a monofilament thread, which makes it resistant to infections, so it can be used in contaminated areas and infected wounds, in order to minimize the subsequent displacement of the thread above the skin surface. Threads with metric numbering 2-0 were used for the study [18].

In the first phase of the study, static mechanical tests were performed using an Instron E3000 (Instron, Norwood, MA, USA) testing machine. The tensile speed was 5 mm/min, in accordance with the ASTM D638 [19] standard:

- For unstitched mesh: Samples A and B were tested separately, according to the tensile direction–half samples each were used in the test;
- For meshes sewn to the skin (imitating a connection with tissues): Samples marked as A‘ and B‘ were tested in an analogous manner–half samples were used in the study.

All samples were stretched to failure and data were recorded in terms of failure force and strain.

In the second phase of the study, cyclic tests (reproducing dynamic loads) were carried out, which were to simulate repeated body movements. The tests were also carried out on the Instron E3000 machine. The amplitude of the material stretching was 2 mm (10% of the sample length), and the cycle frequency was 0.5 Hz. The tests were repeated in the same arrangement for unstitched meshes (A and B) and stitched to the skin (A‘ and B‘). The aim was to determine the effect of dynamic loading on the mechanical durability of the meshes and their behavior in different stretching directions. The study used 3 samples for each of the tests.

After the tests were completed, a microscopic analysis of the mesh structure and damage (destruction) locations was performed. A Delta Optical ZS-430 (Delta Optical, Gdańsk, Poland) optical microscope with an 8 Mpa digital camera was used for microscopic examinations. The microscopic evaluation was performed at 10× magnification. The collected data were subjected to statistical analysis: average values of destructive force and deformation were calculated, standard deviations and coefficients of variation were taken into account, and comparison charts were prepared.

2.3. Statistical and Computational Analysis

The collected data were subjected to basic statistical analysis. For each group of samples (A, B, A‘, B‘) the following were calculated: arithmetic mean, standard deviation, coefficient of variation and quartiles (minimum value, Q1, median, Q3, maximum value). These indices allowed for the comparison of mechanical properties of the meshes depending on the direction of stretching and the use of stitching with the skin. The Student *t* test was used for statistical comparison of results, and significant differences between the results were considered when the *p*-value was <0.05. Calculations and visualizations were performed in Microsoft Excel (Microsoft, Redmond, WA, USA) and Statistica 13 (StatSoft Inc., Tulsa, OK, USA).

Statistical comparisons based on *t*-tests were performed with the awareness that some groups had relatively small sample sizes, which may challenge the assumptions underlying parametric tests. Therefore, normality of distribution was checked both using the Shapiro–Wilk test and by visual inspection of Q-Q plots to ensure that the data did not deviate significantly from a Gaussian distribution. Homogeneity of variance was assessed using Levene’s test to verify that group variances were sufficiently comparable for *t*-tests of the pooled variance. In cases where these assumptions were not met or when sample sizes were too small to reliably test, nonparametric alternatives, such as the Mann–Whitney U

test, were used. This approach ensured that statistical inferences were based on methods appropriate to the data structure and sample size limitations.

Computational models were performed in ML.NET, Visual Studio 2022 (Microsoft, Redmond, WA, USA). The choice of ML.NET in Visual Studio 2022 for modeling the strength of surgical sutures can be justified by several technical and practical arguments. ML.NET is a machine learning library developed by Microsoft, designed to work in the.NET environment. This allows you to seamlessly integrate ML models with medical applications written in C#, e.g., image analysis systems or electronic medical records. Visual Studio 2022 offers ML.NET Model Builder tools, which allow users with less experience in ML to quickly train models without writing code from scratch. In addition, modeling the strength of surgical sutures is usually a classic case of regression, where we predict a numerical value (e.g., the force of thread breakage). ML.NET supports many regression algorithms, such as FastTree (Gradient Boosting), SDCA (Stochastic Dual Coordinate Ascent) or LightGBM. This allows you to automatically compare multiple algorithms, choose the best one, and assess accuracy using metrics such as R^2 or Root Mean Squared Error (RMSE). Strength inputs can come from a variety of sources, including biomechanical experiments, thread characteristics (thickness, material, structure), and operating conditions (temperature, humidity), and ML.NET allows you to easily import data from CSV files, SQL databases, or directly from applications, making it easy to transform research lab data into predictive models. In clinical applications, understanding the model is key, and ML.NET allows you to closely track the training, feature engineering, and evaluation processes of the model. Open source and the absence of a "black box" (like AutoML without controls) increase transparency and trust in the results—important for medical certification. ML.NET allows you to export the model as a C# or API file, load the model into a desktop, web, or mobile application, and run the model offline, locally in the lab or within the hospital network, without the need for the cloud, which increases the security of medical data.

The input data was audited for completeness, repeatability, and lack of outliers, and processed into a csv file. Data normalization was performed within the MS Visual Studio 2022 environment. The result was computational models ranked from best, with the main selection criteria being model accuracy and the smallest error.

The model was developed by pre-splitting the full dataset into separate training and test subsets, ensuring that all measurements from the same physical sample were placed in only one subset to prevent overlap. A typical 70%/30% split was used, allowing the model to learn underlying patterns from a larger portion while retaining a portion for independent performance evaluation. To increase robustness, k-fold cross-validation was performed on the training set, allowing the algorithm to be repeatedly trained and validated on different internal partitions without exposing the test set during tuning. Hyperparameters were optimized exclusively within these cross-validation loops to ensure that no information from the test set influenced the model configuration. Data leakage was further prevented by standardizing or normalizing the input data after the training and test splits, fitting preprocessing parameters only to the training data, and applying them to the test data. methodological safeguards reinforce the belief that reported predictive accuracy reflects true generalization and not random reuse of information.

3. Results

The tested samples were subjected to a static and cyclic stretching test. This was to reflect the behavior of the meshes after their implementation in the abdominal cavity. The movements and behaviors that people perform every day affect the forces acting on surgical

meshes connecting the edges of damaged tissues. The study aimed to check whether the direction of the mesh during its sewing into the damaged tissue site is important. All samples cracked, both when stretched longitudinally (samples referred to hereinafter as A) and transversely (samples referred to hereinafter as B), and in none of them did the mesh come undone. In the case of meshes sewn between two pieces of cowhide, a difference in the way the meshes stretched could be seen. In the case of tests on the meshes themselves, the action of the tensile force was visible along the entire length of the mesh attachment in the forceps of the testing machine, whereas in the case of sewn meshes, the force acted only at the seams (Figure 7). Strength parameters were analyzed in terms of values such as mean parameter value, standard deviation, coefficient of variation and quartiles (minimum value, Q1, median, Q3, maximum value).

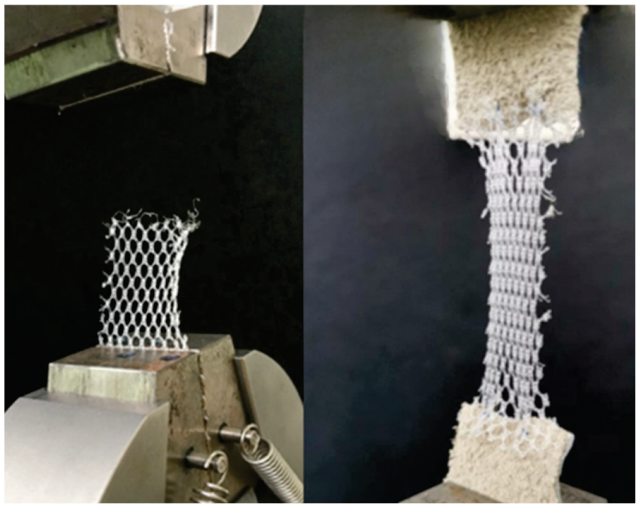


Figure 7. Mesh sample, type A (**left**) and A‘ (**right**) at the moment of failure during the static tensile test.

Microscopic images allowed us to take a closer look at the structure and weave of the meshes, and also to see how the meshes were destroyed (Figure 8).

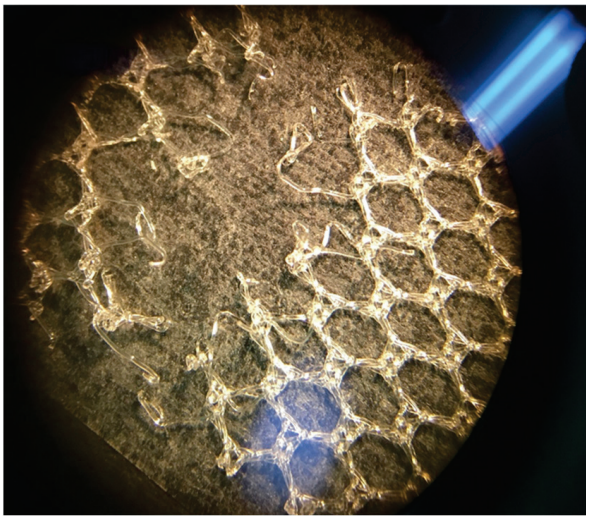


Figure 8. Microscopic image of Optomesh Macropore surgical mesh that fractured in a static tensile test.

3.1. Results from the Monotonic Tensile Test

The following four graphs (Figure 9) show the results of static tensile tests. The individual graphs were compared, taking into account the type of samples and their direction of stretching. The samples with the best individual graphs were selected, i.e., those that did not show any signs of artifacts during the strength test. Artifacts are disturbances created during the test as a result of the machine operation and abrupt changes in force and mesh displacement.

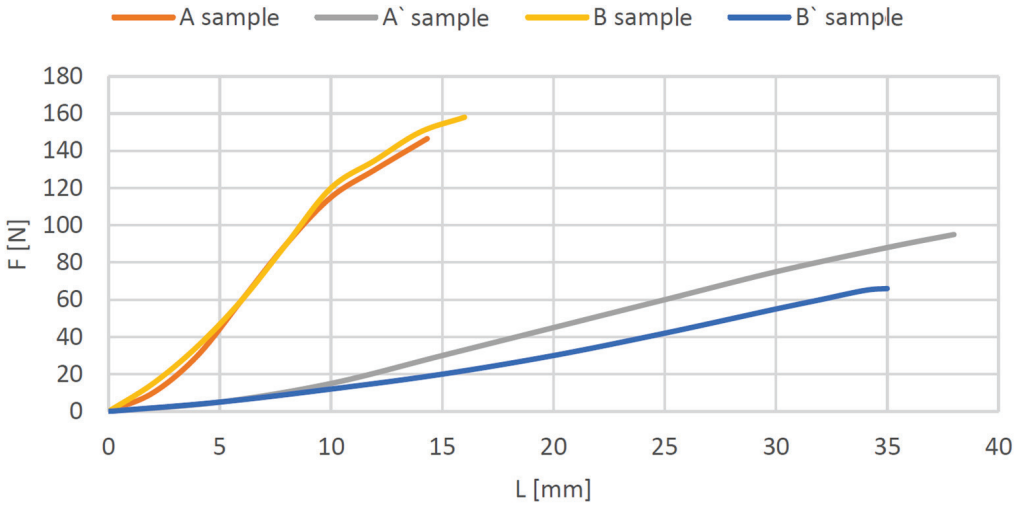


Figure 9. Tensile test graph of the meshes themselves, divided into the tensile direction (A—longitudinally, B—transversely).

3.1.1. Force

The average breaking force for unsewn surgical mesh samples was significantly higher than for samples sewn to the skin. For meshes tested longitudinally (samples A), the average value was 146.49 N, while for samples sewn in the same direction (A‘) it was only 75.52 N, which is a decrease of almost 48%.

For meshes stretched transversely, the breaking force for unsewn samples (B) was 149.14 N, and for sewn samples (B‘) it was 52.33 N, which is a decrease of as much as 65%. This means that sewing the mesh significantly reduces its mechanical strength, regardless of the direction of stretching.

The dispersion of results in each group was assessed using the standard deviation and the coefficient of variation. The standard deviation was the lowest for B samples (7.87 N) and the highest for A‘ (21.48 N), which may be related to differences in seam quality and local stress. The coefficient of variation for sutured samples was clearly higher (A‘: 28.4%, B‘: 36.0%) than for unsutured samples (A: 13.2%, B: 5.3%), indicating greater instability of the results in the sutured groups. To verify the significance of these differences, statistical analysis was performed using the Student (Welch) *t*-test. The test results are as follows:

- A vs. A‘: statistically significant difference ($p < 0.001$); stitching significantly reduces the strength of longitudinally stretched meshes.
- B vs. B‘: statistically very significant difference ($p < 0.001$); suturing significantly weakens transversely stretched meshes.
- A vs. B and A‘ vs. B‘: the difference was not statistically significant.

The above data confirm that the greatest influence on reducing the destructive force is the very fact of sewing the mesh, regardless of the direction of its arrangement. The

differences between the directions (A vs. B, A‘ vs. B‘) did not reach the level of statistical significance (Table 1).

Table 1. Descriptive analysis of the force parameters required to destroy the surgical mesh in the static stretching test, divided into the type of samples and the direction of their stretching.

Stretching Direction	M	SD	V	Min	Q1	Me	Q3	Max
Just the mesh								
Lengthwise (A)	146.49	19.29	13.2%	122.2	133.3	142.3	161.2	172.0
Crosswise (B)	149.14	7.87	5.3%	137.6	145.3	147.6	154.5	159.1
Sewn mesh								
Lengthwise (A‘)	75.52	21.48	28.4%	36.9	69.8	82.0	87.0	97.7
Crosswise (B‘)	52.33	18.83	36.0%	30.6	37.3	52.9	63.5	78.5

Where M—mean value, SD—standard deviation, V—variability, Min—minimum value, Q1—first quartile, Me—median, Q3—third quartile, Max—maximum value.

The results of the force required to break the meshes are presented in the following graph with the division into the direction of sample stretching marked (Figure 10).

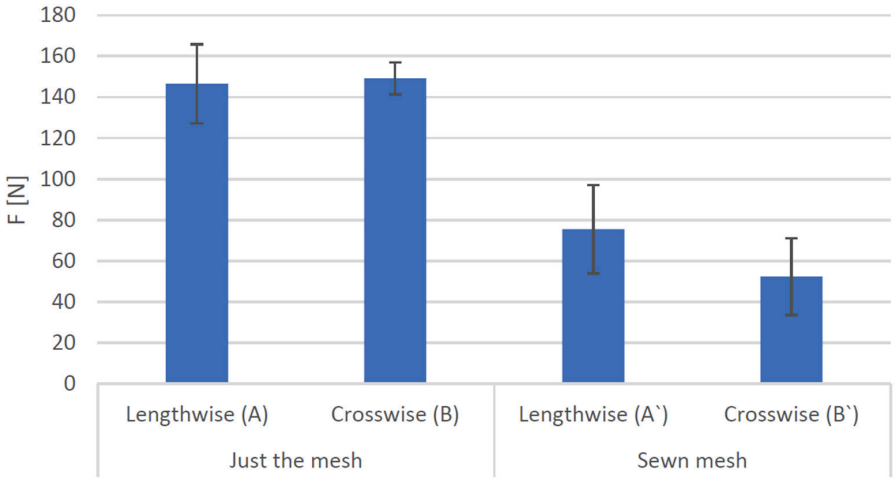


Figure 10. Graphical representation of the maximum force required to break samples of the meshes themselves, taking into account the division into the direction of their stretching.

3.1.2. Deformation

The results presented in Table 2 show the values of the maximum deformation of surgical meshes at the moment of failure, taking into account the direction of stretching and the configuration of the sample (sutured or not). The average deformation values for unsutured samples were 14.32 ± 1.53 mm (A—longitudinally) and 14.77 ± 0.80 mm (B—transversely), which indicates very similar mechanical compliance of the material in both directions. Statistical analysis did not show significant differences between these two groups ($p = 0.894$). In turn, the deformation of samples sutured with a fragment of bovine leather was significantly greater, reaching an average of 30.51 ± 5.80 mm for A‘ and 31.90 ± 6.35 mm for B‘. The difference between these values did not reach the level of statistical significance ($p = 0.532$), which suggests that the direction of loading does not affect the maximum deformation in the sutured configuration. Significant differences were revealed when comparing the sutured and unsutured samples. Both in the case of longitudinal (A vs. A‘, $p < 0.001$) and transverse (B vs. B‘, $p < 0.001$) stretching, highly statistically

significant increases in strain were noted after sewing the mesh. These results confirm that the connection of the mesh with a fragment of biological tissue significantly affects the change in its mechanical properties, increasing its susceptibility to deformation. This may result from a decrease in the local stiffness of the system or the effect of redistribution of stresses resulting from sewing.

Table 2. Descriptive analysis of the deformation parameters [mm] of the surgical mesh, resulting from the static stretching test, divided into the type of samples and their stretching direction.

Stretching Direction	M	SD	V	Min	Q1	Me	Q3	Max
Just the mesh								
Lengthwise (A)	14.32	1.53	10.7%	12.6	13.1	13.7	15.7	16.3
Crosswise (B)	14.77	0.80	5.4%	13.5	14.4	14.8	15.2	16.0
Sewn mesh								
Lengthwise (A‘)	30.51	5.80	19.0%	19.9	29.6	31.6	33.7	36.7
Crosswise (B‘)	31.90	6.35	19.9%	20.1	31.3	33.4	35.3	38.0

Where M—mean value, SD—standard deviation, V—variability, Min—minimum value, Q1—first quartile, Me—median, Q3—third quartile, Max—maximum value.

The results of mesh deformation are presented in the graph below with the division into the direction of sample stretching indicated (mesh itself and sewn mesh—Figure 11).

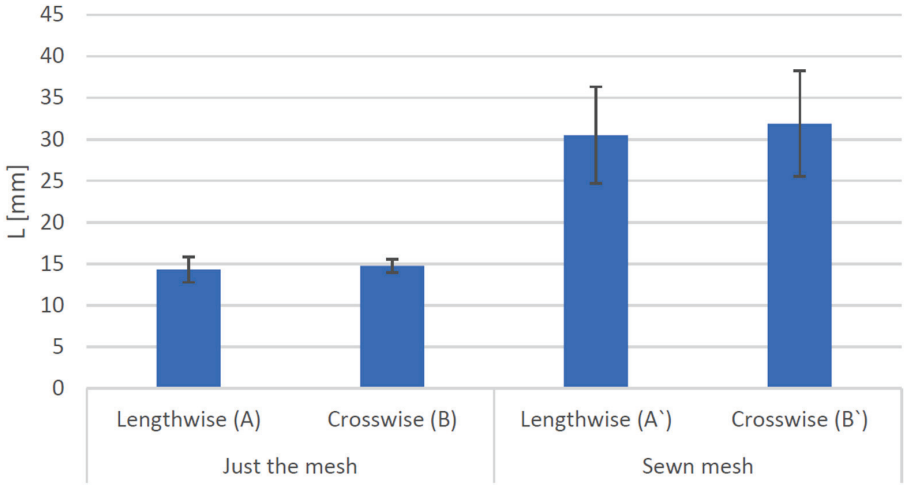


Figure 11. Graphical representation of the deformation of mesh samples, taking into account the division into the direction of their stretching.

3.2. Results from the Cyclic Stretching Test

The next four graphs (Figures 12 and 13) show the results from the cyclic tensile tests. As in the case of the static tensile test, the samples with the best individual graphs were selected, i.e., no signs of failure during the strength test.

The tests were carried out at a constant stretching amplitude of 2 mm and a frequency of 0.25 Hz, which was intended to reproduce fatigue conditions similar to those occurring in vivo, e.g., during the patient's daily activity.

Each point on the graph represents the value of the force recorded in a single cycle for one sample—corresponding to the mechanical resistance of the mesh at a deformation of 2 mm. The range of cycles is from 1 to 1000, and the horizontal axis is presented on a

logarithmic scale, which allows for a detailed analysis of changes occurring both at the beginning and in the final phase of loading.

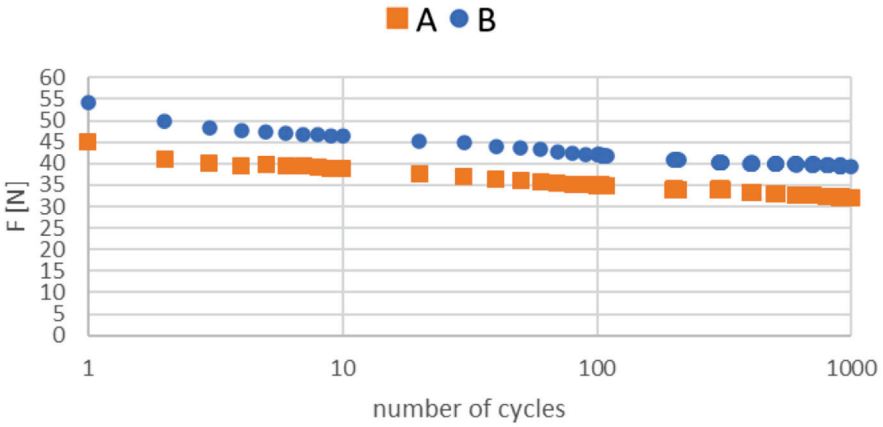


Figure 12. Cyclic tensile test graph of the meshes themselves, divided into the tensile direction (A—longitudinally, B—transversely).

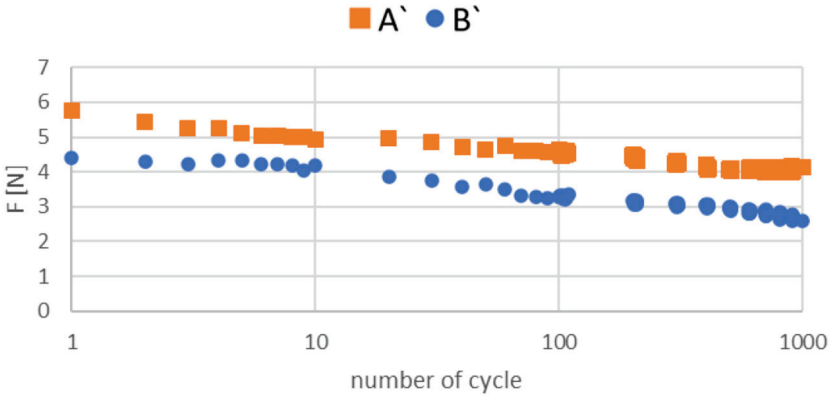


Figure 13. Cyclic tensile test graph of sewn meshes, divided into the tensile direction (A‘—longitudinally, B‘—transversely).

The mesh samples stretched in the longitudinal direction (A) showed an initial value of the force required for deformation at the level of 45 N. With the increase in the number of cycles, a gradual, almost linear decrease in force was observed towards a value of 31 N after 1000 cycles. This means a loss of about 31% of the initial stiffness of the material during the test, which may indicate slight damage to the micropore structure or displacement of the mesh fibers. For samples stretched transversely (B), the initial force values were higher and amounted to about 55 N, and the final force values after 1000 cycles dropped to about 41 N. Despite the drop, the B meshes maintained higher force values throughout the test, which suggests that the material in the transverse direction is characterized by greater resistance to mechanical fatigue.

The results presented in Figure 11 for the sewn samples indicate a significantly lower level of force required for deformation. For sample A‘ the initial values of the force required for deformation were 5.8 N and during the test they decreased gradually, reaching a level of about 4.1 N after 1000 cycles. The decrease in force was visible. Despite the fatigue effect, the mesh sewn to the skin retained some of its stiffness. The values of the initial force required for deformation in group B‘ samples were clearly lower, amounting to 4.5 N at the beginning of the test, and decreased faster than in group A‘, reaching a level of 2.9 N after 1000 cycles.

In the unstitched samples (A and B), the initial force values were higher than in the stitched samples (A‘ and B‘), confirming the negative effect of stitching on the integrity and stiffness of the system. In group A, the initial force was 45 N and decreased to 31 N after 1000 cycles, while in group B the values ranged from 55 N to 41 N. Sample B showed better fatigue resistance than A, suggesting that the transverse direction of stretching provides greater mechanical stability of the mesh. In the stitched samples, the forces were significantly lower. In group A‘, the force decreased from about 5.8 N to 4.1 N, while in B‘–from 4.5 N to 2.9 N. Stitching the material increased susceptibility to fatigue degradation, which may result from local weakening of the structure due to the presence of stitches and uneven stress distribution. In conclusion, both the direction of stretching and the presence of suture have a significant influence on the fatigue behavior of surgical meshes. A decrease in stiffness during the test was observed in all groups, but the rate and extent of the force decrease were different and dependent on the mechanical arrangement of the sample.

3.3. Results of Computational Modeling

The comprehensive experimental dataset provided here offers valuable insights and foundational data, which could be used in future artificial intelligence and machine learning analyses for predictive modeling of surgical mesh behavior. In the study, two models were created to answer the following research questions:

- how to predict the connection forces,
- what thread to choose to obtain a given connection force.

After the analyses, the number of model parameters was reduced to the following (for 156 samples):

1. Input parameters:
    - Skin connection,
    - Stretching direction,
    - Fixing,
    - Thread Strength,
2. Output parameter:
    - Strength of entire connection.

The results for entire strength connection prediction are presented in Table 3 and Figure 14—a good prediction accuracy of 84.12% was achieved, giving a chance for clinical application of the model after tuning it and expanding the database.

The results for thread selection to obtain a given connection strength are presented in Table 4 and Figure 15—a good prediction accuracy of 82.47% was achieved, also giving the prospects for clinical use of the model after tuning it and expanding the database.

Table 3. Top results for entire strength connection prediction (of 42 models).

Algorithm	Accuracy	Absolute-Loss	Squared-Loss	RMS-Loss
SdcaRegression	84.12	13.63	286.19	16.68
LbfgsPoissonRegressionRegression	83.98	12.99	273.83	16.33
LightGbmRegression	83.94	13.21	282.11	16.56
FastForestRegression	83.86	13.07	275.63	16.40
LightGbmRegression	83.84	13.11	279.22	16.49

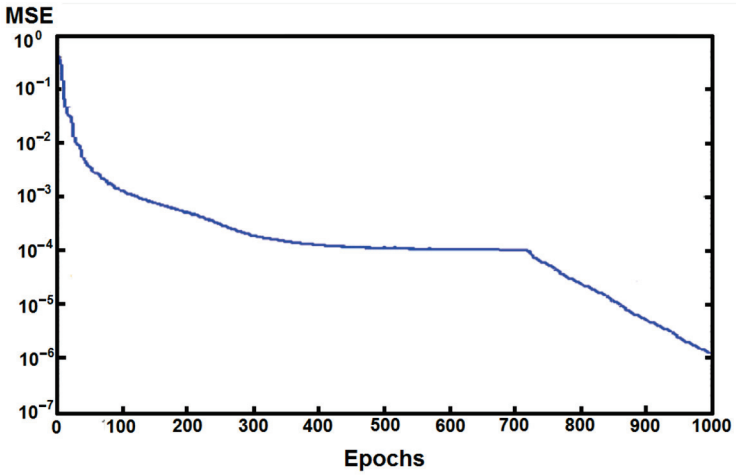


Figure 14. Values of MSE during learning for entire strength connection prediction (print screen from ML software VisualStudio 2022).

Table 4. Top results for thread selection to obtain a given connection strength (of 47 models).

Algorithm	Accuracy	Absolute-Loss	Squared-Loss	RMS-Loss
SdcaRegression	82.47	13.57	285.32	16.65
LbfgsPoissonRegressionRegression	81.77	12.26	283.95	16.21
LightGbmRegression	80.52	13.11	279.13	16.97
LightGbmRegression	79.14	13.37	277.54	16.13
FastForestRegression	78.22	13.23	276.13	16.44

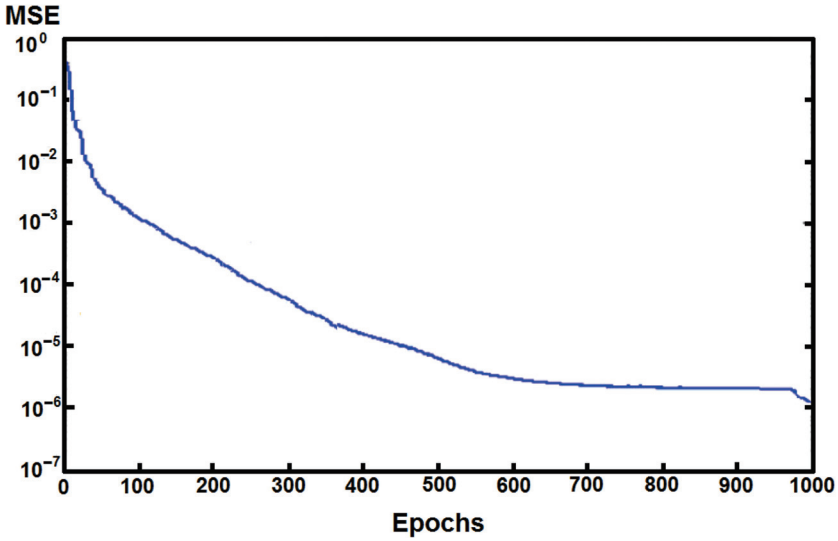


Figure 15. Values of MSE during learning for thread selection to obtain a given connection strength (print screen from ML software).

In Tables 3 and 4 we presented only the top five models/algorithms out of 42 and 47 tested ML models, respectively. SdcaRegression and FastForest Regression are the names of the ML models.

The TRIPOD model compliance report showed that the model was developed using a multivariate regression approach with predictors selected based on both prior literature and

exploratory correlation analysis. Internal validation was performed with cross-validation to assess model stability. Missing data were noted as less than 5% missing and were treated using mean or median imputation, depending on the variable type. Calibration showed a calibration slope close to 1.0, indicating good agreement between predicted and observed results.

The study effectively investigates the mechanical properties of nonabsorbable surgical meshes by comparing sutured and unsutured specimens in a standardized test framework. Sutured specimens exhibit reduced elasticity and increased local stress concentrations, leading to earlier mechanical failure compared to unsutured counterparts. This difference is clearly described and reflects the mechanical effects of needle penetration and suture tension on mesh integrity. Suturing introduces discontinuities and stress increases in the mesh structure, which weakens its ability to evenly distribute loads. From a mechanical perspective, these changes disrupt the native strain behavior of the mesh, increasing stiffness and reducing fatigue resistance. An AI-based model captures these changes, using empirical data to simulate performance outcomes under various clinical fixation scenarios. In the context of Industry 4.0, this integration supports predictive modeling and virtual prototyping of mesh placement techniques. While Industry 5.0 promotes clinician feedback in the loop, the current model only implicitly addresses user interaction in surgical decision-making. Nevertheless, recognizing the mechanical impact of sutures provides valuable insights for both design optimization and surgical guidance. The study's clear presentation of sutured and unsutured performance enhances the validity of its findings within a digitally assisted, patient-specific mesh planning framework.

Suturing nonabsorbable surgical meshes introduces local stress concentrations at the needle entry points, which can act as initiation points for mechanical failure. These stress spikes disrupt the uniform load distribution across the mesh, reducing its overall flexibility and fatigue resistance. An AI-based model captures these effects by analyzing strain patterns and identifying zones of mechanical compromise due to suturing. As part of Industry 4.0, such modeling supports virtual testing and optimization of surgical fixation techniques. Industry 5.0 adds a human-centric layer where surgeon feedback can guide improvements in mesh design and application strategies. Therefore, understanding how suturing compromises performance is critical to developing smarter, more resilient meshes that are based on both data and clinical insights.

4. Discussion

The mechanical properties of nonabsorbable surgical meshes, including tensile strength, stiffness, and fatigue resistance, provide the foundation for AI-based modeling to optimize mesh design and performance. These Industry 4.0-ready models enable predictive simulations and virtual prototyping to predict clinical outcomes. However, several practical barriers hinder seamless implementation of AI integration into clinical workflows. Regulatory requirements pose significant challenges, as AI tools used in healthcare must meet stringent standards for safety, transparency, and repeatability. Clinical validation is another critical hurdle, requiring extensive in vivo and multi-center studies to ensure AI predictions translate into real-world performance. Additionally, integration with existing hospital IT systems is often complex due to concerns about compliance, data security, and interoperability. While Industry 5.0 emphasizes human-centered innovation, its implementation remains limited unless these systemic barriers are addressed. Overcoming these challenges is essential to fully realize the potential of AI-assisted mesh technology in routine surgical practice.

The mechanical behavior and fatigue resistance of nonabsorbable polypropylene surgical meshes are fundamental to their clinical efficacy, especially in procedures such as hernia repair and pelvic organ prolapse (POP) surgery [20–23]. Recent experimental studies have highlighted the importance of factors such as fiber orientation, suture integration, and load direction, which directly affect the mechanical integrity, degradation profile, and long-term functionality of mesh implants [24–27].

Historically, polypropylene has been the material of choice for surgical meshes due to its favorable mechanical properties, such as high tensile strength, flexibility, and structural stability under loading conditions [28,29].

The mechanical properties of nonabsorbable surgical meshes—such as tensile strength, pore stability, and fatigue resistance—are critical to ensuring long-term implant success. AI-based models within Industry 4.0 offer powerful tools to simulate and predict these properties in a variety of clinical settings. However, for clinical implementation, these models must undergo extensive validation and robustness testing to ensure reliability across a variety of patient populations and surgical scenarios. Validation must include in vitro, in vivo, and potentially multicenter clinical studies to confirm that the AI predictions are consistent with real-world outcomes. Robustness testing is also necessary to assess the model's sensitivity to changes in mesh type, surgical technique, and patient-specific factors. While Industry 5.0 envisions a human-centric, personalized approach to healthcare, it is highly dependent on the reliability of such AI systems. Without comprehensive validation, integrating AI into clinical decision-making remains speculative and potentially risky. Therefore, before widespread adoption, AI tools for evaluating surgical meshes must meet rigorous clinical and regulatory standards.

In this study, the maximum tensile forces observed for seamless polypropylene meshes were 146.49 N (longitudinal direction, group A) and 149.14 N (transverse direction, group B), while for the stitched samples the maximum tensile forces decreased significantly to 75.52 N (A') and 52.33 N (B'), respectively. These values are consistent with reports from the literature, where similar polypropylene meshes tested under monotonic tensile loading showed maximum failure forces in the range of about 120–180 N.

In particular, meshes made of isotactic polypropylene (i-PP) exhibit improved fatigue resistance, maintaining structural integrity under cyclic loading conditions [28,30]. However, the inherent anisotropy of the mesh design often leads to direction-dependent mechanical responses. This was evident in our findings, where the transverse samples (B) consistently outperformed the longitudinal samples (A) in terms of both tensile strength and fatigue life. This is consistent with the literature that attributes such behavior to fiber orientation and the resulting stress distribution on the mesh surface [31].

Seam integration further complicates the mechanical behavior of the meshes. Our study confirmed that the stitched samples (A' and B') exhibited lower mechanical performance than their unstitched counterparts, with a more pronounced loss of strength over time. The observed reduction in tensile strength after suturing was approximately 48–65%, confirming previous studies indicating that suturing significantly reduces mechanical integrity by introducing stress concentrations.

The sutured meshes showed significantly greater strain (approximately 30.51 mm (A') and 31.90 mm (B')), highlighting the significant increase in mesh compliance after biointegration, consistent with the results of [31] who noted increased elongation and decreased stiffness due to material–tissue interactions.

While nonabsorbable sutures provide mechanical stability, they may also increase the risk of postoperative pain due to local stiffness and stress concentrations [32]. Furthermore, while absorbable sutures may reduce postoperative discomfort, they may also compromise

long-term fixation and the ability of the mesh to share loads. Therefore, the choice of suture material remains a balance between mechanical performance and clinical outcome.

Cyclic tensile testing demonstrated progressive loss of stiffness in all mesh types, with sutured samples degrading more rapidly. Our samples showed a measurable decrease in force from initial values (45 N in A samples to 34 N after 1000 cycles; 55 N in B samples to approximately 41 N). This progressive mechanical degradation confirms previous cyclic fatigue studies that report force reductions of approximately 15–30% under similar testing conditions [28]. The higher residual strength of transverse (B) meshes compared to longitudinal (A) meshes is inconsistent (larger) with previous observations that anisotropic fiber orientations significantly affect fatigue resistance [31].

This phenomenon is compounded by the potential of polypropylene to undergo oxidative degradation and surface cracking over time, which may further contribute to structural weakening and inflammatory reactions [33]. These effects have prompted the development of modified meshes, including bioactive coatings and composite structures that incorporate resorbable components to reduce adverse tissue reactions [33]. Our study further supports recent trends in mesh design facilitated by modern manufacturing techniques such as 3D printing and material functionalization. Studies investigating innovative composite geometries and designs have demonstrated improved mechanical resistance and tailored compliance to specific surgical needs [34,35]. This approach may contribute to personalized surgical solutions and potentially lower complication rates associated with conventional polypropylene meshes [36].

Overall, our results reinforce the concept that polypropylene surgical meshes have directional dependence and are significantly affected by sutures, which are essential factors to consider when planning surgery and selecting a mesh. This is quantitatively consistent with recent literature [37–43].

While our study has demonstrated significant differences in mesh behavior related to loading direction and suture integration, a potential direction for future research involves the use of artificial intelligence (AI)-based methods [44,45]. AI techniques, such as ML and deep neural networks, can leverage extensive experimental datasets like those presented here to develop predictive models. These models could significantly enhance mesh design by optimizing material composition, geometry, and structural properties, potentially improving surgical outcomes and reducing postoperative complications. Furthermore, advanced AI simulations could enable personalized mesh selection tailored to patient-specific anatomical and physiological conditions, thus aligning surgical mesh development with principles of precision medicine.

The improved fatigue resistance of cross-link meshes has important clinical implications, suggesting that their microstructure is inherently better suited to the multidirectional cyclic loading encountered in vivo. This orientation likely distributes stress more evenly across the fibers, reducing the concentration of local strains that typically initiate fatigue or fiber breakage. A tighter or more connected cross-link arrangement may also limit progressive strain, helping the mesh maintain its structural integrity during prolonged physiological loading, such as breathing, coughing, or abdominal wall movement. Clinically, this means that cross-link meshes may maintain mechanical stability longer, potentially reducing the risk of mesh stretching, pore collapse, or recurrent hernias. Increased durability may also translate into more stable support for surrounding tissues, improving postoperative comfort and long-term outcomes. mechanistic interpretation indicates that the transverse fiber orientation provides an intrinsic fatigue advantage, making these meshes more suitable for stressed anatomical sites where cyclic stresses cannot be avoided.

Industry 4.0 emphasizes connectivity and automation, facilitating the integration of sensor technologies into the manufacturing process to monitor material quality and performance in real time. Additive manufacturing (3D printing) combined with computational models enables the production of custom-designed meshes with patient-specific properties, allowing for improved clinical outcomes. Moving toward Industry 5.0, a human-centric approach focuses on incorporating human expertise with computational advances to develop more sustainable, efficient, and user-centric surgical meshes. Computational modeling in the context of Industry 4.0/5.0 revolutionizes the design and evaluation of nonresorbable surgical meshes, supporting innovation and improving patient care through precision and intelligent manufacturing.

4.1. Limitations of Current Studies

Current research on nonabsorbable surgical meshes faces several limitations that impact their development and clinical application. Many studies lack long-term follow-up data, making it difficult to assess the durability and safety of meshes over decades [46–48]. Variability in clinical practice and study designs, including differences in patient populations, surgical techniques, and evaluation criteria, often leads to inconsistent results that make generalizability difficult [49–51]. Another limitation is the underrepresentation of certain patient groups, such as those with comorbidities or unusual anatomical changes/challenges, limiting the applicability of findings to diverse populations [20,21]. Preclinical studies are often based on animal models that may not accurately represent human tissue responses, making it difficult to translate findings into clinical practice [52,53]. Furthermore, the mechanisms underlying mesh-related complications, such as chronic pain and fibrosis, remain poorly understood, limiting the ability to effectively predict and mitigate risks [21,22].

There is also a lack of standardized protocols for evaluating and comparing new mesh materials, which complicates comparisons of outcomes between products or their successive generations [54,55]. The cost-effectiveness of advanced mesh designs, particularly in low-resource settings (developing countries), has not been adequately addressed and compared [56]. Despite the emphasis on a holistic biopsychosocial approach and patient comfort, most studies focus on initial surgical outcomes, ignoring the psychological and quality-of-life impact that potential mesh-related changes and complications may have on patients over time (e.g., limitations in some activities such as demanding sports or even vigorous and prolonged dancing) [57,58]. These limitations underscore the need for more robust, multidisciplinary research to improve the safety and efficacy of nonresorbable surgical meshes.

The mechanical properties of nonresorbable surgical meshes, such as stiffness, tensile strength, and fatigue resistance, are critical to their clinical performance and durability. In the context of Industry 4.0/5.0, AI-based models can help analyze and predict these properties using data-driven insights and intelligent manufacturing techniques. However, the current study's reliance on a single commercial polypropylene mesh limits the generalizability of its conclusions. Drawing broad conclusions about all nonresorbable meshes without comparative analysis reduces the external validity of the findings. Industry 4.0 promotes comprehensive comparative testing and data integration, which were clearly lacking in this approach. Comparative testing with multiple commercial meshes would provide critical benchmarks for assessing mechanical and fatigue behavior. In addition, Industry 5.0 principles emphasize personalization and collaboration, which require diverse inputs to support clinician-in-the-loop decision-making. The lack of mesh diversity also makes it difficult for the AI model to adapt or learn to different material profiles. As a

result, although the model reflects some conceptual aspects of Industry 4.0/5.0, it lacks the methodological rigor needed to fully comply with their standards, which we will address in subsequent studies, considering the present research as preliminary.

The use of ML algorithms such as FastForest regression may seem disproportionate when the dataset is small, as these models are designed to capture complex, nonlinear patterns that typically require large amounts of data for reliable generalization. With a limited number of observations, the algorithm can overfit, exploiting learning noise or group-level patterns rather than underlying material-property relationships. In such cases, the model's "predictions" may reflect little more than statistical correlations between group means or predefined categories, rather than actual learned behavior. In this study, we minimized the aforementioned risk of the model appearing to perform well in the sample while lacking significant predictive power for new or unfamiliar grid configurations. Consequently, the sophistication of the ML technique does not exceed the capabilities of the dataset, leading to results closer to descriptive statistics than robust predictive modeling.

Fatigue testing provides valuable data because it captures the response of surgical meshes to repeated loading cycles that mimic physiological conditions. Raw fatigue life measurements offer only a snapshot of performance, not describing the progression of degradation over time, so we consider this a preliminary study. We plan to incorporate a degradation rate constant in a future study, which would determine the rate of deterioration of mechanical properties, enabling more accurate modeling of long-term behavior. Similarly, S–N curve fitting would enable prediction of fatigue life across a range of stress amplitudes, not just under the tested conditions. Together, these additional elements would transform fatigue test results derived from isolated observations into a framework for extrapolating the long-term durability of the mesh.

The ML component relies entirely on off-the-shelf ML.NET algorithms, as the study focuses on demonstrating feasibility within the Industry 4.0/5.0 pipeline rather than developing new model architectures or interpretability frameworks. Using AutoML-derived models streamlines the process but inherently limits insight into how individual features influence predictions. This is important because regression results in this context can influence material or suture selection, and decisions without clear rationale can be difficult to justify in biomedical applications. Recognizing this limitation helps explain the current work's emphasis on proof-of-concept modeling rather than fully interpretable decision support systems. Future work could integrate model-agnostic interpretability tools (permutation-based feature significance, partial dependency analysis, or SHAP explanations) to reveal how mechanical parameters contribute to predictions. Implementing such methods would strengthen clinical and engineering trust by making the AI component more transparent, auditable, and aligned with responsible Industry 5.0 principles.

4.2. Directions for Further Research

Future research on nonabsorbable surgical meshes should focus on developing advanced biomaterials that minimize immune response and chronic inflammation. Personalized and patient-specific meshes, potentially using 3D printing technologies, can be tailored to individual anatomical and clinical needs to improve integration and outcomes [23]. Exploring the use of nanotechnology and bioactive coatings to deliver antimicrobial agents or promote tissue regeneration is another promising direction [24]. Long-term studies evaluating the durability, safety, and efficacy of new mesh designs (including changes in mechanical parameters) are crucial to understanding their performance over the lifespan of the patient. Research into the underlying mechanisms of complications, such as chronic

pain, fibrosis, and adhesion formation, could lead to innovations in mesh structure and surface properties [25]. Standardization of evaluation protocols and clinical trial designs would provide better comparability across mesh products and streamline regulatory processes. Further research into robotic and minimally invasive surgical techniques may optimize mesh placement and reduce postoperative complications [26]. Integration of computational modeling and imaging technologies could improve preoperative planning and help predict the most appropriate type of mesh for individual cases [27]. Research into the cost-effectiveness of advanced mesh technologies across healthcare settings is also essential to ensure accessibility and equitable distribution [59,60]. Research into the psychological and qualitative effects of mesh complications and procedures can help improve patient care and provide holistic solutions [61,62]. Together, these directions aim to create safer, more effective, and patient-centered approaches to the use of nonresorbable surgical mesh [63–66].

5. Conclusions

In this study, the mechanical behavior of nonabsorbable polypropylene surgical meshes under monotonic and cyclic tensile loading was evaluated, taking into account the direction of stretching (longitudinal vs. transverse) and the presence of biological integration via suturing to porcine skin. The experimental results allow for the following conclusions:

- The direction of tensile loading significantly affects the mechanical response of the mesh. Transversely stretched samples (B) showed higher maximum failure forces and greater fatigue resistance than longitudinally stretched samples (A), both in sutured and unsutured conditions.
- Suturing the mesh to biological tissue significantly reduces its mechanical strength and stiffness. In both directions, the sutured samples (A‘ and B‘) showed lower tensile strength and a larger range of strains under loading, suggesting a weakening effect at the mesh-tissue interface.
- Cyclic loading revealed a gradual degradation of strength in all samples, indicating material fatigue. However, transversely stretched meshes maintained higher forces over 1000 cycles than longitudinal meshes, confirming the improved fatigue life of this configuration.
- The observed differences in mechanical behavior can be attributed to the anisotropic mesh structure and the mechanical effects of suturing, which introduce stress concentrations and structural discontinuities.
- These findings underscore the importance of considering both directionality and surgical technique when selecting and implementing mesh implants, especially in applications where repeated loading and long-term durability are critical.
- Both AI-based models achieved results above 80%, which shows their clinical utility and the possibility of development towards prediction accuracy above 85–90% (i.e., at least at the level of a human expert) available in each hospital and adapted to local conditions (e.g., type of thread, etc.).

Future research should incorporate computational and artificial intelligence models to refine predictive capabilities, ultimately leading to the development of more effective, patient-specific surgical meshes.

The experimental data obtained in this study can serve as a basis for AI-assisted predictive modeling and simulation, contributing to the development of intelligent surgical materials and decision support tools in the framework of Industry 4.0 and 5.0.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/app152412894/s1, sample.dataset.xls.

Author Contributions: M.A., I.R. and D.M.; methodology, M.A., I.R. and D.M.; software, M.A., I.R. and D.M.; validation, M.A., I.R. and D.M.; formal analysis, M.A., I.R. and D.M.; investigation, M.A., I.R., T.B. and D.M.; resources, M.A., I.R. and D.M.; data curation, M.A.; writing—original draft preparation, M.A., I.R., T.B. and D.M.; writing—review and editing, M.A., I.R., T.B. and D.M.; visualization, M.A., I.R. and D.M.; supervision, I.R.; project administration, I.R.; funding acquisition, I.R. All authors have read and agreed to the published version of the manuscript.

Funding: The work presented in the paper has been financed under a grant to maintain the research potential of Kazimierz Wielki University.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Dataset is available on request from the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

3D	Three-dimensional
AI	Artificial intelligence
ML	Machine learning

References

1. Saxena, A.K.; Hayward, R.K. Patches in Congenital Diaphragmatic Hernia: Systematic Review. *Ann. Surg.* **2024**, *280*, 229–234. [CrossRef]
2. Poli, P.P.; Beretta, M.; Maiorana, C.; Souza, F.Á.; Bovio, A.; Manfredini, M. Therapeutic Strategies in the Management of Nonresorbable Membrane and Titanium Mesh Exposures Following Alveolar Bone Augmentation: A Systematic Scoping Review. *Int. J. Oral Maxillofac. Implants* **2022**, *37*, 250–269. [CrossRef]
3. Patel, K.; Brandstetter, K. Solid Implants in Facial Plastic Surgery: Potential Complications and How to Prevent Them. *Facial Plast. Surg.* **2016**, *32*, 520–531. [CrossRef]
4. Heesakkers, J.P.; Vierhout, M.E. Prolapse surgery: Which technique and when? *Curr. Opin. Urol.* **2011**, *21*, 281–285. [CrossRef]
5. Rojek, I.; Mikołajewski, D.; Dostatni, E.; Kopowski, J. Specificity of 3D Printing and AI-Based Optimization of Medical Devices Using the Example of a Group of Exoskeletons. *Appl. Sci.* **2023**, *13*, 1060. [CrossRef]
6. Rojek, I.; Dostatni, E.; Mikołajewski, D.; Pawłowski, L.; Wegrzyn-Wolska, K. Modern approach to sustainable production in the context of Industry 4.0. *Bull. Pol. Acad. Sci. Tech. Sci.* **2022**, *70*, e143828. [CrossRef]
7. Martire, M.V.; Marazzi, D.; Lancia, E.; Vesco, S.; Apa, L.; Del Prete, Z.; Trovalusci, F.; Rizzuto, E. Measuring Mechanical Properties of 3D Printed Specimens for Creating Cerebral Aneurysm Mock-Ups for Surgical Training. In Proceedings of the 2024 IEEE International Symposium on Medical Measurements and Applications (MeMeA), Eindhoven, The Netherlands, 26–28 June 2024; pp. 1–6.
8. Yang, J.; Hu, M.; Shi, X.; Zhao, D.; Yu, L. Deformation modeling based on mechanical properties of liver tissue for virtuanormal vectors of trianglesl surgical simulation. *Int. J. Comput. Assist. Radiol. Surg.* **2021**, *16*, 253–267. [CrossRef]
9. He, W.; Liu, X.; Wu, S.; Liao, J.; Cao, G.; Fan, Y.; Li, X. A numerical method for guiding the design of surgical meshes with suitable mechanical properties for specific abdominal hernias. *Comput. Biol. Med.* **2020**, *116*, 103531. [CrossRef]
10. Lim, Y.J.; De, S.; Jones, D.B.; Singh, T.P. In Situ Measurement of Cadaveric Soft Tissue Mechanical Properties and Fulcrum Forces for Use in Physics-Based Surgical Simulation. In Proceedings of the 2006 14th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, Alexandria, VA, USA, 25–26 March 2006; p. 37.
11. Naderi, A.; Zhang, B.; Belgodere, J.A.; Sunder, K.; Palardy, G. Improved Biocompatible, Flexible Mesh Composites for Implant Applications via Hydroxyapatite Coating with Potential for 3-Dimensional Extracellular Matrix Network and Bone Regeneration. *ACS Appl. Mater. Interfaces* **2021**, *13*, 26824–26840. [CrossRef]

12. Ortenzi, M.; Rapoport Ferman, J.; Antolin, A.; Bar, O.; Zohar, M.; Perry, O.; Asselmann, D.; Wolf, T. A novel high accuracy model for automatic surgical workflow recognition using artificial intelligence in laparoscopic totally extraperitonealinguinal hernia repair (TEP). *Surg. Endosc.* **2023**, *37*, 8818–8828. [CrossRef]
13. Ballard, D.H.; Jammalamadaka, U.; Tappa, K.; Weisman, J.A.; Boyer, C.J.; Alexander, J.S.; Woodard, P.K. 3D printing of surgical hernia meshes impregnated with contrast agents: In vitro proof of concept with imaging characteristics on computed tomography. *3D Print Med.* **2018**, *4*, 13. [CrossRef]
14. Shiroud Heidari, B.; Dodda, J.M.; El-Khordagui, L.K.; Focarete, M.L.; Maroti, P.; Toth, L.; Pacilio, S.; El-Habashy, S.E.; Boateng, J.; Catanzano, O.; et al. Emerging materials and technologies for advancing bioresorbable surgical meshes. *Acta Biomater.* **2024**, *184*, 1–21. [CrossRef]
15. Optomesh. Available online: https://tricomed.com/products/optomesh/ (accessed on 27 January 2025).
16. *ISO 13934-1:2013*; Textiles — Tensile properties of fabrics Part 1: Determination of maximum force and elongation at maximum force using the strip method. Edition 2. Available online: https://www.iso.org/standard/60676.html (accessed on 4 December 2025).
17. *ASTM D5035-11 (2019)*; Standard Test Method for Breaking Force and Elongation of Textile Fabrics (Strip Method). Available online: https://store.astm.org/d5035-11r19.html (accessed on 4 December 2025).
18. Prolene. Available online: https://spacemed.pl/pl/c/Prolene/185 (accessed on 27 January 2025).
19. *ASTM D638-14*; Standard Test Method for Tensile Properties of Plastics. Available online: https://store.astm.org/d0638-14.html (accessed on 4 December 2025).
20. Maier, W. Biomaterials in reconstructive skull base surgery. *Laryngorhinootologie* **2009**, *88* (Suppl. S1), S64–S75. [CrossRef]
21. Jank, M.; Boettcher, M.; Keijzer, R. Surgical management of the diaphragmatic defect in congenital diaphragmatic hernia: A contemporary review. *World J. Pediatr. Surg.* **2024**, *7*, e000747. [CrossRef]
22. Di Spirito, F.; Giordano, F.; Di Palo, M.P.; Ferraro, C.; Cecere, L.; Frucci, E.; Caggiano, M.; Lo Giudice, R. Customized 3D-Printed Mesh, Membrane, Bone Substitute, and Dental Implant Applied to Guided Bone Regeneration in Oral Implantology: A Narrative Review. *Dent. J.* **2024**, *12*, 303. [CrossRef]
23. Rojek, I.; Studziński, J. Comparison of different types of neuronal nets for failures location with in water-supply networks. *Eksploat. Niezawodn.-Maint. Reliab.* **2014**, *16*, 42–47.
24. McDowell, D.T.; Cheng, T.; Darani, A.; Dye, R.; Arbuckle, S.; Cohen, R.C. Thoracotomy patch repair of large diaphragmatic herniae in a porcine model: A tale of two patches. *Pediatr. Surg. Int.* **2024**, *40*, 305. [CrossRef]
25. Rojek, I.; Jagodziński, M. Hybrid Artificial Intelligence System in Constraint Based Scheduling of Integrated Manufacturing ERP Systems. In *Hybrid Artificial Intelligent Systems*; Lecture Notes in Computer Science; Corchado, E., Snášel, V., Abraham, A., Woźniak, M., Graña, M., Cho, S.B., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; Volume 7209, pp. 229–240.
26. Saxena, A.K. Surgical perspectives regarding application of biomaterials for the management of large congenital diaphragmatic hernia defects. *Pediatr. Surg. Int.* **2018**, *34*, 475–489. [CrossRef]
27. de Haro Jorge, I.; Prat Ortells, J.; Martín-Solé, O.; Muñoz Fernandez, E.; Pertierra, A.; Martin-Lluis, A.; Tarrado, X. Porcine dermal patches as a risk factor for recurrence after congenital diaphragmatic hernia repair. *Pediatr. Surg. Int.* **2021**, *37*, 59–65. [CrossRef]
28. Rojek, I.; Dostatni, E. Machine learning methods for optimal compatibility of materials in Ecodesign. *Bull. Pol. Acad. Sci.-Tech. Sci.* **2020**, *68*, 199–206. [CrossRef]
29. Fontana-Escartín, A.; Hauadi, K.E.; Lanzalaco, S.; Pérez-Madrigal, M.M.; Armelín, E.; Turón, P.; Alemán, C. Preparation and characterization of functionalized surgical meshes for early detection of bacterial infections. *ACS Biomater. Sci. Eng.* **2023**, *9*, 1104–1115. [CrossRef]
30. Harman, M.K.; Champaigne, K.; Cobb, W.S.; Lu, X.; Chawla, V.; Wei, L.; Nagatomi, J. A novel bio-adhesive mesh system for medical implant applications: In vivo assessment in a rabbit model. *Gels* **2023**, *9*, 372. [CrossRef]
31. Wang, H.; Klosterhalfen, B.; Klinge, U.; Müllen, A.; Jockenhoevel, S. Influence of polypropylene mesh degradation on tissue inflammatory reaction. *J. Biomed. Mater. Res. Part A* **2023**, *111*, 1110–1119. [CrossRef]
32. Dahesh, M.B.; Asayesh, A.; Jeddi, A.A.A. A novel method for evaluating the tensile behaviorisotropy of surgical meshes based on the analysis of the polar graph of tensile properties using image processing method. *J. Ind. Text.* **2022**, *51* (Suppl. S1), 569S–586S. [CrossRef]
33. Agarwal, P.; Sutrave, T.; Kaushal, D.; Vidua, R.; Malik, R.; Maurya, A.P. Comparison of postoperative chronic groin pain after repair of inguinal hernia using nonabsorbable versus absorbable sutures for mesh fixation. *Cureus* **2023**, *15*, e35562. [CrossRef]
34. Abuzayed, B.; Rawagah, F.; Alrashdan, M.S.; Al-Abadi, H.; Jamous, O.; Said, A. Cranioplasty for skull defects with polypropylene abdominal wall mesh. *J. Craniofacial Surg.* **2022**, *33*, e764–e767. [CrossRef]
35. Sterk, S.; Silva, M.E.; Fernandes, A.A.; Huss, A.; Wittek, A. Development of new surgical mesh geometries with different mechanical properties using the design freedom of 3d printing. *J. Appl. Polym. Sci.* **2023**, *140*, 46. [CrossRef]

36. Lanzalaco, S.; Weis, C.; Traeger, K.A.; Turón, P.; Alemán, C.; Armelín, E. Mechanical properties of smart polypropylene meshes: Effects of mesh architecture, plasma treatment, thermosensitive coating, and sterilization process. *ACS Biomater. Sci. Eng.* **2023**, *9*, 3699–3711. [CrossRef]
37. Sakuma, I.; Nishimura, Y.; Chui, L.K.; Kobayashi, E.; Inada, H.; Chen, X.; Hisada, T. *In Vitro Measurement of Mechanical Properties of Liver Tissue Under Compression and Elongation Using a New Test Piece Holding Method with Surgical Glue*, 4th ed.; Springer: Berlin/Heidelberg, Germany, 2003; pp. 284–292.
38. Todros, S.; Pavan, P.G.; Natali, A.N. Biomechanical properties of synthetic surgical meshes for pelvic prolapse repair. *J. Mech. Behav. Biomed. Mater.* **2015**, *55*, 271–285. [CrossRef]
39. Lombardi, J.; Stec, E.; Edwards, M.; Connell, T.; Sandor, M. Comparison of mechanical properties and host tissue response to OviTex™ and Strattice™ surgical meshes. *Hernia* **2023**, *27*, 987–997. [CrossRef]
40. Bokhari, N.; Ali, A.; Nausheen, S.; Yasmeen, A.; Anjum, M.A.; Sharif, F. Biomimetic silk fibroin meshes for regenerative soft tissue repair: Multiscale evaluation of a spatiotemporally active scaffold. *Biomater. Adv.* **2026**, *178*, 214435. [CrossRef] [PubMed]
41. Todros, S.; Pavan, P.G.; Natali, A.N. Synthetic surgical meshes used in abdominal wall surgery: Part I—materials and structural conformation. *J. Biomed. Mater. Res. B Appl. Biomater.* **2017**, *105*, 689–699. [CrossRef]
42. Todros, S.; Pavan, P.G.; Pachera, P.; Natali, A.N. Synthetic surgical meshes used in abdominal wall surgery: Part II—Biomechanical aspects. *J. Biomed. Mater. Res. B Appl. Biomater.* **2017**, *105*, 892–903. [CrossRef]
43. Tomaszewska, A.; Lubowiecka, I.; Szymczak, C. Mechanics of mesh implanted into abdominal wall under repetitive load. Experimental and numerical study. *J. Biomed. Mater. Res. B Appl. Biomater.* **2019**, *107*, 1400–1409. [CrossRef] [PubMed]
44. Liu, W.; Xie, Y.; Zheng, Y.; He, W.; Qiao, K.; Meng, H. Regulatory science for hernia mesh: Current status and future perspectives. *Bioact. Mater.* **2020**, *6*, 420–432. [CrossRef]
45. Żywicka, B.; Struszczyk, M.H.; Paluch, D.; Kostanek, K.; Krucińska, I.; Kowalski, K.; Kopias, K.; Rybak, Z.; Szymonowicz, M.; Gutowska, A.; et al. Design of New Concept of Knitted Hernia Implant. *Materials* **2022**, *15*, 2671. [CrossRef] [PubMed]
46. Song, X.; Hu, M.; Li, B.; Zhang, K.; Zhang, X.; Wang, L. Advancing medical device regulatory reforms for innovation, translation and industry development in China. *J. Orthop. Transl.* **2022**, *37*, 89–93. [CrossRef]
47. Tian, J.; Song, X.; Wang, Y.; Cheng, M.; Lu, S.; Xu, W.; Gao, G.; Sun, L.; Tang, Z.; Wang, M.; et al. Regulatory perspectives of combination products. *Bioact. Mater.* **2021**, *10*, 492–503. [CrossRef]
48. Zamkowski, M.; Tomaszewska, A.; Lubowiecka, I.; Śmietański, M. Biomechanical causes for failure of the Physiomesh/Securestrap system. *Sci. Rep.* **2023**, *13*, 17504. [CrossRef]
49. Zhou, H.; Shen, Y.; Zhang, Z.; Liu, X.; Zhang, J.; Chen, J. Comparison of outcomes of ventral hernia repair using different meshes: A systematic review and network meta-analysis. *Hernia* **2022**, *26*, 1561–1571. [CrossRef]
50. Zamkowski, M.; Tomaszewska, A.; Lubowiecka, I.; Karbowski, K.; Śmietański, M. Is mesh fixation necessary in laparoendoscopic techniques for M3 inguinal defects? An experimental study. *Surg. Endosc.* **2023**, *37*, 1781–1788. [CrossRef] [PubMed]
51. Fieot, M.; Maliszewska, I.; Rac-Rumijowska, O.; Suchorska-Woźniak, P.; Lewińska, A.; Teterycz, H. The Relationship between the Mechanism of Zinc Oxide Crystallization and Its Antimicrobial Properties for the Surface Modification of Surgical Meshes. *Materials* **2017**, *10*, 353. [CrossRef] [PubMed]
52. Liu, P.; Chen, N.; Jiang, J.; Wen, X. Preparation and in Vitro Evaluation of New Composite Mesh Functionalized with Cationic Antimicrobial Peptide. *Materials* **2019**, *12*, 1676. [CrossRef] [PubMed]
53. Serrano-Aroca, Á.; Pous-Serrano, S. Prosthetic meshes for hernia repair: State of art, classification, biomaterials, antimicrobial approaches, and fabrication methods. *J. Biomed. Mater. Res. Part A* **2021**, *109*, 2695–2719. [CrossRef]
54. Nosheen, S.; Mukhtar, H.; Haider, S.; Khan, R.; Sharif, F. Tissue engineered multifunctional chitosan-modified polypropylene hernia mesh loaded with bioactive phyto-extracts. *Int. J. Biol.* **2024**, *271*, 132282. [CrossRef]
55. Santiago-Castillo, K.; Del Angel-López, D.; Torres-Huerta, A.M.; Domínguez-Crespo, M.A.; Palma-Ramírez, D.; Willcock, H.; Brachetti-Sibaja, S.B. Effect on the process ability, structure and mechanical properties of highly dispersed in situ ZnO: CS nanoparticles into PVA electrospun fibers. *J. Mater. Res. Technol.* **2021**, *11*, 929–945. [CrossRef]
56. Xu, D.; Fang, M.; Wang, Q.; Qiao, Y.; Li, Y.; Wang, L. Latest trends on the attenuation of systemic foreign body response and infectious complications of synthetic hernia meshes. *ACS Appl. Bio Mater.* **2021**, *5*, 1–19. [CrossRef]
57. Liu, P.; Ren, J.; Zhao, Y.; Zhang, K.; Guo, C.; Wang, L.; Xie, J. Mimicking nature: Shish-kebab topology in nanofibers enhances antibacterial and mechanical properties. *Text. Res. J.* **2020**, *95*, 00405175251317211. [CrossRef]
58. Pasculli, A.; Gurrado, A.; De Luca, G.M.; Mele, A.; Marzullo, A.; Mangone, A.; Testini, M. Bridging repair of the abdominal wall in a rat experimental model. Comparison between uncoated and polyethylene oxide-coated equine pericardium meshes. *Sci. Rep.* **2020**, *10*, 6959. [CrossRef]
59. Mourtzis, D. The Metaverse in Industry 5.0: A Human-Centric Approach towards Personalized Value Creation. *Encyclopedia* **2023**, *3*, 1105–1120. [CrossRef]

60. Marinelli, M. From Industry 4.0 to Construction 5.0: Exploring the Path towards Human–Robot Collaboration in Construction. *Systems* **2023**, *11*, 152. [CrossRef]
61. Mladineo, M.; Celent, L.; Milković, V.; Veža, I. Current State Analysis of Croatian Manufacturing Industry with Regard to Industry 4.0/5.0. *Machines* **2024**, *12*, 87. [CrossRef]
62. Czeczot, G.; Rojek, I.; Mikołajewski, D.; Sangho, B. AI in IIoT Management of Cybersecurity for Industry 4.0 and Industry 5.0 Purposes. *Electronics* **2023**, *12*, 3800. [CrossRef]
63. Hassan, M.A.; Zardari, S.; Farooq, M.U.; Alansari, M.M.; Nagro, S.A. Systematic Analysis of Risks in Industry 5.0 Architecture. *Appl. Sci.* **2024**, *14*, 1466. [CrossRef]
64. Tzampazaki, M.; Zografos, C.; Vrochidou, E.; Papakostas, G.A. Machine Vision — Moving from Industry 4.0 to Industry 5.0. *Appl. Sci.* **2024**, *14*, 1471. [CrossRef]
65. Kabir, M.H.; Chowdhury, J.A.; Fahim, I.M.; Hasan, M.N.; Hasnat, A.; Mahdi, A.J. Design and Simulation of AI-Enabled Digital Twin Model for Smart Industry 4.0. *Eng. Proc.* **2023**, *58*, 119. [CrossRef]
66. Charalampous, P.; Pelekoudas, A.; Kostavelis, I.; Ioannidis, D.; Tzovaras, D. An Integrated Modeling Framework for Automated Product Design, Topology Optimization, and Mechanical Simulation. *J. Manuf. Mater. Process.* **2024**, *8*, 285. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

MDPI AG
Grosspeteranlage 5
4052 Basel
Switzerland
Tel.: +41 61 683 77 34

Applied Sciences Editorial Office
E-mail: applsci@mdpi.com
www.mdpi.com/journal/applsci



Disclaimer/Publisher's Note: The title and front matter of this reprint are at the discretion of the Guest Editors. The publisher is not responsible for their content or any associated concerns. The statements, opinions and data contained in all individual articles are solely those of the individual Editors and contributors and not of MDPI. MDPI disclaims responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.





Academic Open
Access Publishing

mdpi.com

ISBN 978-3-7258-9274-7