technologies

Special Issue Reprint

The Future of Healthcare

Biomedical Technology and Integrated Artificial Intelligence 2nd Edition

Edited by
Juvenal Rodriguez-Resendiz, Gerardo I. Pérez-Soto, Karla Anhel Camarillo-Gómez and Saul Tovar-Arriaga

mdpi.com/journal/technologies

MDPI

The Future of Healthcare: Biomedical Technology and Integrated Artificial Intelligence 2nd Edition

The Future of Healthcare: Biomedical Technology and Integrated Artificial Intelligence 2nd Edition

Guest Editors

Juvenal Rodriguez-Resendiz
Gerardo I. Pérez-Soto
Karla Anhel Camarillo-Gómez
Saul Tovar-Arriaga

MDPI

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

Guest Editors

Juvenal Rodriguez-Resendiz
Facultad de Ingeniería
Universidad Autónoma
de Querétaro
Santiago de Queretaro
Mexico

Gerardo I. Pérez-Soto
Engineering Faculty
Autonomous University
of Queretaro
Queretaro
Mexico

Karla Anhel Camarillo-Gómez
Department of Mechanical
Engineering
Tecnologico Nacional de
Mexico en Celaya
Celaya
Mexico

Saul Tovar-Arriaga
Faculty of Engineering
Autonomous University
of Queretaro
Queretaro
Mexico

Editorial Office
MDPI AG
Grosspeteranlage 5
4052 Basel, Switzerland

This is a reprint of the Special Issue, published open access by the journal *Technologies* (ISSN 2227-7080), freely accessible at: https://www.mdpi.com/journal/technologies/special_issues/P18K600SD0.

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 978-3-7258-6093-7 (Hbk)
ISBN 978-3-7258-6094-4 (PDF)
https://doi.org/10.3390/books978-3-7258-6094-4

Contents

About the Editors

Juvenal Rodriguez-Resendiz

He was with West Virginia University as a Visiting Professor in 2012. Currently is the coordinator of the Engineering Ph.D. at Querétaro State University (UAQ), in México. Herein, he has taught more than 200 digital signal processing and research methodology courses. He belongs to the Mexican Academy of Sciences. He has the higher membership for researchers, called SNII level 3, given by the Mexican government. He has developed more than 50 industrial projects by linking UAQ and the government. His team has published 200 scientific papers. He patented more than 10 innovations. He has won national and international prizes because of his academic and innovation developments. He has been the advisor of more than 200 theses of undergraduate, master, and doctoral grades. He has been invited to give 30 conferences around the world. Because of his research, he collaborates with 30 national and international institutions.

Gerardo I. Pérez-Soto

Gerardo I. Pérez-Soto received the Ph.D. degree in Mechanical Engineering from the Universidad de Guanajuato, Mexico. In 2013, he joined the Universidad Autónoma de Querétaro (UAQ), where he is currently a Full Professor with the Facultad de Ingeniería. His current research focuses on theoretical kinematics, humanoid and mobile robotics, assembly automation, and the application of mechanical engineering to industrial processes. He is a member of the Mexican Association on Robotics and Industry and has received several fellowships and awards, including the A. T. Yang Memorial Award in Theoretical Kinematics for Best Paper by the American Society of Mechanical Engineers (ASME) in 2007 and 2011. Dr. Pérez-Soto is also a member of the National System of Researchers (SNII) of SECIHTI, Mexico.

Karla Anhel Camarillo-Gómez

Karla Anhel Camarillo-Gómez received her Ph.D. from the Tecnologico Nacional de Mexico en la Laguna. She joined the Tecnologico Nacional de Mexico en Celaya as a professor in 2009. She is currently a Professor at the Department of Mechanical Engineering. She leads the Robotics and Biomechanical Systems research group in the same department and collaborates with the Dynamic Systems and Control group. She has been a member of the Mexican Association on Robotics and Industry (AMRob) since 2006, the American Society of Mechanical Engineering from 2014 to 2023, and the Co-Chair of HuroCup in the Federation International of Robo-Sports Association (FIRA) since 2017. She was president of AMRob from 2012 to 2014. She was invited to the Scientific Research Honor Society in 2024. She has been the founder of the robotics team competition called LYNXbot since 2009. She received several robotics fellowships and awards from AMRob, FIRA, and Wiley, among other organizations. She is level 1 in the National Research System (SNII) of SECIHTI. Her current research focuses on modeling and control of robots, control of nonlinear systems, stability analysis of nonlinear systems, development of rehabilitation systems, technologies for drones, humanoid robots, and autonomous vehicles, and applications of assembly automation and vision control in industrial processes.

Saul Tovar-Arriaga

Dr. Saul Tovar-Arriaga received his PhD in Biomedical Sciences from the University of Erlangen-Nuremberg, Germany, his MSc in Mechatronics from the University of Siegen, Germany, and is an Electronics Engineer from the Queretaro Institute of Technology. He is a full-time professor at the Faculty of Engineering of the Autonomous University of Querétaro and currently serves as the coordinator of the Master of Science in Artificial Intelligence program. His research interests include automatic illness diagnosis, surgical robotics, and machine learning applications. He is a member of the National System of Researchers and a regular member of the Mexican Academy of Computing (AMEXCOMP). He has served as president of the IEEE Queretaro Section and the IEEE Computational Intelligence chapter.

technologies

MDPI

Editorial

Artificial Intelligence in Biomedical Technology: Advances and Challenges

Marcos Aviles [1,*], Saul Tovar-Arriaga [1], Gerardo Israel Pérez-Soto [1], Karla A. Camarillo-Gómez [2] and Juvenal Rodríguez-Reséndiz [1,*]

[1] Facultad de Ingeniería, Universidad Autónoma de Querétaro, Querétaro 76010, Mexico; saul.tovar@uaq.mx (S.T.-A.); israel.perez@uaq.mx (G.I.P.-S.)

[2] Departamento de Ingeniería Mecánica, Tecnológico Nacional de México en Celaya, Celaya 38010, Mexico; karla.camarillo@itcelaya.edu.mx

* Correspondence: marcosaviles@ieee.org (M.A.); juvenal@uaq.edu.mx (J.R.-R.)

1. Introduction

Artificial intelligence (AI) has had an increasingly widespread presence in biomedical technology in recent years. Its use has expanded to tasks such as diagnostic imaging, physiological signal analysis, clinical pattern classification, and the automation of medical processes. This expansion is due to the development of more accurate models, the increased availability of clinical data, and access to computing platforms capable of running algorithms in real-time. Recent studies have documented its implementation not only in hospitals but also in embedded systems, wearable devices, and mobile applications for field and home healthcare [1–4]. These advances reflect a transition from experimental models to functional solutions, with real potential for integration at different healthcare system levels.

One of the most developed areas is medical image processing using deep neural networks. Convolutional neural networks (CNNs) are used for tissue segmentation, lesion detection, and disease classification tasks. Models such as U-Net, YOLO, and VGG have been adapted and optimized for various modalities. These models have achieved accuracy comparable to, and even superior to, those of specialists under controlled conditions. In some cases, their performance has been validated in preliminary clinical trials, indicating that these tools have technical value and applicability in medical practice [2,5].

The development of lighter and more efficient versions of these models has allowed them to run on low-power platforms such as the Jetson systems. This line of work seeks to bring automated analysis closer to the point of care without relying on complex hospital infrastructure or permanent connectivity. This approach expands the scope of medical AI and addresses real needs in low-resource settings [6].

In parallel, interest has grown in using synthetic data to improve model training. Techniques such as generative adversarial networks (GANs) and diffusion models have generated artificial medical images and physiological signals, allowing data sets to be balanced or increased in size. These synthetic data have proven useful in improving model generalization and performance, especially when few real-world examples are available [7].

There has also been growing interest in making AI models interpretable. Explainability has become required for these tools to be accepted by clinicians and approved by regulatory bodies. Recent research has proposed methods for identifying which regions of an image influence a decision or how certain physiological features are weighted in a classification.

https://doi.org/10.3390/technologies13050208

These strategies increase medical user confidence and allow for the detection of errors, biases, or model failures before deployment [8].

In addition to its use in diagnosis and monitoring, AI has also begun to be applied to problems related to accessibility, prevention, and well-being. For example, systems have been developed for automatic sign language recognition, ergonomic assessment at work, and fall detection in older adults. These applications expand the field of biomedical AI beyond hospital settings and show that its potential impact can extend to social, community, and preventive problems [9,10].

These results indicate that artificial intelligence in the biomedical field has reached a stage of applied maturity. While regulation, clinical validation, and operational integration challenges remain, accumulating evidence confirms that these technologies can play an important role in strengthening health systems and expanding access to medical services.

2. Emerging Trends

Recent developments in artificial intelligence for biomedical technology not only offer specific solutions, but also allow it to identify lines of evolution that outline the direction the field is heading. These trends do not emerge from a stated intention by the authors, but from observable patterns in the nature of the problems addressed, the techniques selected, and the conditions under which they are designed and validated.

2.1. Compact and Executable Models in Real Time

A clear trend is the search for models that can be executed outside of traditional hospital environments. Priority is given to the design of efficient architectures, optimized for embedded platforms such as Raspberry Pi, or even to run directly in mobile applications. This orientation responds to the need to address low-infrastructure contexts, as well as decentralized or emergency care scenarios. Local execution eliminates constant connection to servers and guarantees real-time responses, which is critical in applications such as field monitoring, emergency ultrasound, or mass screening. Furthermore, this type of implementation favors adoption in healthcare systems with limited resources, where AI can extend diagnostic capabilities without requiring expensive equipment or highly specialized personnel [11].

2.2. Use of Synthetic Data for Training and Validation

Another emerging area is the generation of synthetic data using models such as GANs or diffusion models to solve problems of scarcity or imbalance in biomedical data sets. These techniques allow the generation of images, signal sequences, or even structured data that simulate real physiological properties. Synthetic data has proven useful not only for improving model performance but also for representing rare classes that are often underrepresented in traditional clinical databases. Empirical evidence shows that models trained with these data generalize better, especially in multiclass classification or precise segmentation tasks. Although its adoption is still incipient in formal clinical settings, the technical results are consistent and open up new possibilities for expansion in scenarios with ethical or logistical restrictions on collecting real data [12].

2.3. Explainability as a Functional Component of Design

Interest in explainable models is not new, but it has moved from a theoretical ideal to a practical requirement. Integrating real-time interpretation methods, such as activation maps, filter visualization, or variable weighting, is gaining ground in real-world applications as a tool for subsequent analysis and a practical component during clinical use.

This evolution reflects the need for systems to be accurate but also understandable and auditable by medical personnel. Explainability is a necessary bridge between algorithmic engineering and responsible clinical practice [13].

2.4. Expansion of Functional Scope Toward Social and Preventive Problems

Another relevant trend is the expansion of the field to address problems that are not strictly medical but directly impact people's health, well-being, and autonomy. This includes automatic sign language recognition or fall detection in older adults. These types of applications show that biomedical AI is not limited to diagnosis but can contribute to overcoming communication barriers, environmental monitoring, and risk prevention. In all cases, the design of the solutions reflects a sensitivity to real-life conditions and the diversity of users, suggesting an ethical and social evolution in the field [5].

These trends reveal a shift in the field toward applicability, operational efficiency, and clinical accountability. Far from focusing solely on technical metrics, current developments point to practical, explainable, and functional systems in real-world settings, marking a new stage in the evolution of artificial intelligence in medicine.

3. Persistent Gaps and Challenges

Although recent advances in biomedical artificial intelligence are technically sound, their clinical adoption faces significant obstacles. Many models are validated only under controlled conditions or with well-labeled public databases but are rarely tested in real-world settings. This limits their usefulness outside the laboratory and creates uncertainty about their behavior in the face of clinical, demographic, or technical variability [14].

Another key limitation is the lack of integration with existing clinical systems. Many solutions do not consider interoperability standards or privacy regulations, making them difficult to implement on hospital platforms or public health systems. Added to this is the limited attention paid to model traceability and auditability [15].

Explainability remains a missing component in most proposals. While some techniques allow for visualizing regions of interest or internal weights, few are designed with interpretability as a central criterion. This directly affects the trust of medical professionals and hampers regulatory validation. There also persists a heavy reliance on clinical data with limited diversity. Many models are trained on homogeneous populations, which limits their generalization capacity and can reproduce biases that affect diagnostic equity. While synthetic data helps mitigate this problem, its use still requires rigorous validation [16].

These challenges reflect the fact that technical effectiveness is not enough. The responsible adoption of artificial intelligence in healthcare requires validated, explainable, and adaptable models aligned with the clinical context for which they are intended.

4. Future Perspectives

The future of artificial intelligence in biomedicine depends less on algorithmic improvements and more on its effective integration into real-world contexts. Moving forward, clinical validation needs to be expanded to diverse settings, with studies that reflect heterogeneous operating conditions and populations. This will allow for the evaluation of technical accuracy and clinical utility.

Explainability must be incorporated from the design stage. Models must offer clear and understandable justifications to facilitate their acceptance by medical personnel and ensure their traceability. Progress in the area of interoperability with clinical systems is also needed, considering their integration with existing platforms and current regulations from the outset.

Furthermore, it is urgent to improve the representativeness of the data used for training, incorporating population variability and regional contexts. This is key to building solutions that work equitably in different healthcare settings.

Finally, the impact needs to be evaluated beyond technical metrics. Analyzing costs, operational benefits, and acceptance by real users will allow for the prioritization of developments with tangible clinical value and long-term sustainability.

5. Conclusions

Artificial intelligence has taken on an increasingly relevant role in the development of biomedical solutions, with applications ranging from automated diagnosis to real-time monitoring and support for vulnerable populations. Recent advances show a shift toward more functional, explainable models adapted to the environment of use.

However, the clinical value of these technologies depend not only on their accuracy, but also on their validation under real-world conditions, their integration with existing healthcare systems, and their user acceptance. Significant challenges remain in terms of regulation, data diversity, explainability, and impact assessment.

The field is at a point of transition. It is no longer enough to design effective algorithms; it is necessary to build technically viable, clinically useful, and socially responsible solutions. The future of artificial intelligence in biomedicine depends on the ability to maintain this balance.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Szilágyi, L.; Kovács, L. Special issue: Artificial intelligence technology in medical image analysis. *Appl. Sci.* **2024**, *14*, 2180. [CrossRef]
2. Frid-Adar, M.; Diamant, I.; Klang, E.; Amitai, M.; Goldberger, J.; Greenspan, H. GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification. *Neurocomputing* **2018**, *321*, 321–331. [CrossRef]
3. Tjoa, E.; Guan, C. A survey on explainable artificial intelligence (XAI): Toward medical XAI. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 4793–4813. [CrossRef] [PubMed]
4. Alharbi, H.A.; Alharbi, K.K.; Hassan, C.A.U. Enhancing elderly fall detection through IoT-enabled smart flooring and AI for independent living sustainability. *Sustainability* **2023**, *15*, 15695. [CrossRef]
5. Gou, F.; Liu, J.; Xiao, C.; Wu, J. Research on artificial-intelligence-assisted medicine: A survey on medical artificial intelligence. *Diagnostics* **2024**, *14*, 1472. [CrossRef] [PubMed]
6. Kolosov, D.; Kelefouras, V.; Kourtessis, P.; Mporas, I. Contactless camera-based heart rate and respiratory rate monitoring using AI on hardware. *Sensors* **2023**, *23*, 4550. [CrossRef] [PubMed]
7. Müller-Franzes, G.; Niehues, J.M.; Khader, F.; Arasteh, S.T.; Haarburger, C.; Kuhl, C.; Wang, T.; Han, T.; Nolte, T.; Nebelung, S.; et al. A multimodal comparison of latent denoising diffusion probabilistic models and generative adversarial networks for medical image synthesis. *Sci. Rep.* **2023**, *13*, 12098. [CrossRef] [PubMed]
8. Hildt, E. What is the role of explainability in medical artificial intelligence? A case-based approach. *Bioengineering* **2025**, *12*, 375. [CrossRef] [PubMed]
9. García-Gil, G.; López-Armas, G.d.C.; Sánchez-Escobar, J.J.; Salazar-Torres, B.A.; Rodríguez-Vázquez, A.N. Real-time machine learning for accurate Mexican sign language identification: A distal phalanges approach. *Technologies* **2024**, *12*, 152. [CrossRef]
10. Villa, M.; Casilari, E. Wearable fall detectors based on low power transmission systems: A systematic review. *Technologies* **2024**, *12*, 166. [CrossRef]
11. Baraneedharan, P.; Kalaivani, S.; Vaishnavi, S.; Somasundaram, K. Revolutionizing healthcare: A review on cutting-edge innovations in Raspberry Pi-powered health monitoring sensors. *Comput. Biol. Med.* **2025**, *190*, 110109. [CrossRef] [PubMed]

12. Goyal, M.; Mahmoud, Q.H. A systematic review of synthetic data generation techniques using generative AI. *Electronics* **2024**, *13*, 3509. [CrossRef]
13. Antoniadi, A.M.; Du, Y.; Guendouz, Y.; Wei, L.; Mazo, C.; Becker, B.A.; Mooney, C. Current challenges and future opportunities for XAI in machine Learning-based Clinical Decision Support Systems: A systematic review. *Appl. Sci.* **2021**, *11*, 5088. [CrossRef]
14. Nagendran, M.; Chen, Y.; Lovejoy, C.A.; Gordon, A.C.; Komorowski, M.; Harvey, H.; Topol, E.J.; Ioannidis, J.P.A.; Collins, G.S.; Maruthappu, M. Artificial intelligence versus clinicians: Systematic review of design, reporting standards, and claims of deep learning studies. *BMJ* **2020**, *368*, m689. [CrossRef] [PubMed]
15. Petersson, L.; Larsson, I.; Nygren, J.M.; Nilsen, P.; Neher, M.; Reed, J.E.; Tyskbo, D.; Svedberg, P. Challenges to implementing artificial intelligence in healthcare: A qualitative interview study with healthcare leaders in Sweden. *BMC Health Serv. Res.* **2022**, *22*, 850. [CrossRef] [PubMed]
16. Ali, S.; Akhlaq, F.; Imran, A.S.; Kastrati, Z.; Daudpota, S.M.; Moosa, M. The enlightening role of explainable artificial intelligence in medical & healthcare domains: A systematic literature review. *Comput. Biol. Med.* **2023**, *166*, 107555.

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.

technologies

MDPI

Article

AI Diffusion Model-Based Technology for Automating the Multi-Class Labeling of Electron Microscopy Datasets of Brain Cell Organelles for Their Augmentation and Synthetic Generation

Nikolay Sokolov , Alexandra Getmanskaya and Vadim Turlapov *

Research Center for Artificial Intelligence, Institute of Information Technologies, Mathematics, and Mechanics, Lobachevsky University, 603022 Nizhny Novgorod, Russia; nikolay.sokolov@unn.ru (N.S.); alexandra.getmanskaya@itmm.unn.ru (A.G.)

* Correspondence: vadim.turlapov@itmm.unn.ru

Abstract: A technology for the automatic multi-class labeling of brain electron microscopy (EM) objects needed to create large synthetic datasets, which could be used for brain cell segmentation tasks, is proposed. The main research tools were a generative diffusion AI model and a U-Net-like segmentation model. The technology was studied on the segmentation task of up to six brain organelles. The initial dataset used was the popular EPFL dataset labeled for the mitochondria class, which has training and test parts having 165 layers each. Our mark up for the EPFL dataset was named EPFL6 and contained six classes. The technology was implemented and studied in a two-step experiment: (1) dataset synthesis using a diffusion model trained on EPFL6; (2) evaluation of the labeling accuracy of a multi-class synthetic dataset by the segmentation accuracy on the test part of EPFL6. It was found that (1) the segmentation accuracy of the mitochondria class for the diffusion synthetic datasets corresponded to the accuracy of the original ones; (2) augmentation via geometric synthetics provided a better accuracy for underrepresented classes; (3) the naturalization of geometric synthetics by the diffusion model yielded a positive effect; (4) due to the augmentation of the 165 layers of the original EPFL dataset with diffusion synthetics, it was possible to achieve and surpass the record accuracy of Dice = 0.948, which was achieved using 3D estimation in Hive-net (2021).

Keywords: diffusion neural network; automatic multi-class labeling; electron microscopy; synthetic dataset; dataset augmentation; geometric augmentation; semantic segmentation

1. Introduction

As the title suggests, this article proposes a technology for the automatic multi-class labeling of brain electron microscopy (EM) objects based on a generative diffusion model. Despite the recent emergence of diffusion models, according to the review [1], many new methods for segmentation and diagnostics in biomedicine have already been developed based on them. But for the technology to emerge, it is also necessary to automate unlimited-volume synthetic EM dataset creation for training foundation models [2] to replicate technologies in downstream applications. Such technologies are also relevant for augmenting existing datasets. Specifically, the area of segmentation and classification of brain cell organelles based on brain EM data has many of its own features that complicate the problem. The complexity of this problem stems from the fact that EM operates at the limit of its resolution capabilities (units of nanometers) when capturing organelle images. This leads to a high level of noise in the images, significantly impacting the accuracy and feasibility of segmenting individual organelles. Another challenge is the significant variation in the representation of different organelles per unit area of the EM layer, as well as their underrepresentation in the dataset as a whole. The traditional solution of this problem is an augmentation of real data datasets using classical methods (reflection, rotation, tiling, etc.). This solution is simple and cost-effective but does not solve the problem completely. The development of neural networks, particularly generative models, has given rise to data augmentation (DA) methods leveraging these technologies. One of the earliest studies to use serial block scanning electron microscopy (EM) as a source of high-resolution three-dimensional nanohistology for cells and tissues was ref. [3]. A subsequent series of works was aimed at creating datasets for training deep learning networks for EM data segmentation designed for the binary segmentation of brain cell organelles–neural membranes [4] and the supervoxel segmentation of mitochondria [5]. Simultaneously, the problem of 3D reconstruction of the brain neural network and the problem of brain connectomics on the basis of neuron organelles and connections between neurons (synapses) is stated in [6]. In this problem, of particular importance is the segmentation of postsynaptic densities (PSDs), vesicles, and axons.

The invention of U-Net in 2015 [7] gave rise to numerous novel models and adaptations for segmenting brain EM data. The reason for U-Net's success is related to the contextual information of an input image at all levels of processing. Almost immediately, the publication [8] experimentally confirmed that the skip connection of the U-Net architecture is effective in solving segmentation problems in biomedicine.

U-Net also provided a basis for creating numerous models: [9,10], 3DU-Net [11], V-Net [12], DeepMedic [13], HighRes3DNet [14], Inception U-Net [15], R2U++ [16].

The application of artificial intelligence methods for EM data processing is significantly hindered by the limited availability of labeled data for training and testing deep neural networks (DNNs). Open EM data as a whole are represented by only a few labeled datasets, both due to the laboriousness of preparing samples for an electron microscope, and due to the lack of specialists for manual labeling. Labeling electron microscopy data remains a time-consuming and labor-intensive task, with the annotation of a single experiment requiring up to six months of manual effort. We found four open EM datasets, the earliest and most popular of which were only labeled for one class (mitochondria or membranes). In the two other datasets, several classes were distinguished. As a result, the majority of neural networks used in EM processing are only trained to perform binary segmentation.

According to articles by [17,18], DA is a recognized effective solution to expand the training dataset. The cheapest and most effective of the traditional augmentation methods are horizontal and vertical flipping, translation, and cropping. This means that the relative arrangement of the compartments remains unchanged. With the development of neural networks, there is now the possibility of using deep learning models to generate synthetic images for DA goals.

One of the popular solutions was the use of Generative Adversarial Networks (GANs) [19–22] to augment data for classes that are underrepresented in the dataset. Employing GANs for data generation [23] can yield images practically indistinguishable from real ones. However, training these neural networks requires a large volume of data, and this process may be unstable [24].

However, despite the simplicity of classical augmentation methods, according to a study [25] on the classification task, cropping is more effective than WGAN.

Using variational autoencoders (VAEs) [26] enables training on unlabeled data; however, the quality of generation may be inferior to real data.

Neural diffusion (ND) models are an advanced approach in the field of artificial intelligence that models information diffusion in neural networks to achieve realistic results. They define a Markov chain of diffusion steps to gradually add random noise to the data and then learn to reverse the diffusion process to generate desired data samples from the noise. This method finds widespread application in generative modeling, texture synthesis, and image restoration [27]. Recently, the probabilistic model of diffusion denoising (DDPM) [28] has garnered significant attention due to its excellent quality in generating synthetic images [24,29,30].

The main advantages of neural diffusion (ND) models can be summarized as follows:

1. ND enables the generation of high-quality images with rich details, realistic textures, and smooth transitions. This approach is capable of eliminating noise and artifacts, resulting in the generation of clean and natural images.
2. The ND approach provides control over the image generation process. Through diffusion parameters, one can adjust the level of detail and blur or the degree of preserving the original information. This allows users to customize generated images according to specific requirements.
3. ND can be applied to various types of images, including photographs, drawings, textures, and others. This makes it a versatile tool for generating and processing diverse kinds of visual data.

The flexible DA method based on diffusion models proposed in ref. [31] outperforms the standard DA baseline by about 0.3 accuracy points for the classification task.

The achievements of expanding the dataset for the segmentation task are not so impressive. Ref. [32] proposed expanding the dataset using a diffusion neural network and improved the segmentation results by 0.01–0.03 values in the Dice metric.

The most interesting publication in recent years for us was a review [33] devoted to the study of the capabilities of diffusion models in medical problems. These tasks include anomaly detection; medical image segmentation; noise suppression; classification, generation, and others. Three main approaches to diffusion modeling are characterized as follows: Denoising Diffusion Probabilistic Models (DDPMs), Noise Conditioned Score Networks (NCSNs), and Stochastic Differential Equations (SDEs). The greatest attention is paid to the DDPM approach; so-far unresolved problems are also considered.

In our previous papers [34,35], we considered the problem of recognition by neural networks of classes that are underrepresented in the training dataset. To achieve this, we proposed using a geometric parametric algorithm that allows us to create the required number of missing images and markings for them. This approach was effective, but the generated images lacked realism. The idea for the further development of this approach to augmentation was suggested by an article linking the modeling of nonequilibrium thermodynamics with modern diffusion models [36]. Such models are capable of turning any simple geometric models into realistic images with noise to the required degree.

Thus, the above review ultimately inspired us not only to explore the possibilities of diffusion neural network models that would provide the augmentation of real data simultaneously with their labeling, but also to increase the similarity of simple geometric models to real data, also through diffusion models.

Additionally, we aimed to develop and explore a multi-class technology for the automatic labeling of EM data, invariant to the number and diversity of organelles. This was demonstrated using six organelles as examples: mitochondria, mitochondria boundaries, vesicles, postsynaptic densities (PSDs), cell membranes, and axon sheaths (including their contents).

2. Materials and Methods

2.1. Segmentation Task, Metric, and Network

Semantic Segmentation is a computer vision task in which the goal is to produce a dense pixel-wise segmentation map of an image, where each pixel is assigned to a specific class or object.

We used the Dice–Sørensen coefficient (DSC, Dice), a metric commonly used for evaluating the segmentation of biomedical images. The DSC values ranged from zero to one. Let the number of pixels correctly classified as belonging to the target class be defined as the true positive (TP), the number of correctly classified background pixels as the true negative (TN), the number of pixels erroneously classified as belonging to the target class as the false positive (FP), and the number of erroneously classified background pixels as the false negative (FN). The DSC metric is then defined as follows:

$$DSC = \frac{2TP}{2TP + FP + FN} \tag{1}$$

Since we consider multi-class segmentation in this study, we are interested in multi-class evaluation metrics. Since the Dice metrics compare two sets, in the case of multi-class classification, the result will be a vector of Dice metrics for each class. For training a neural network for multi-class segmentation, we should transform DSC_i metrics to the scalar error or loss function. For this purpose, we use the linear combination defined in Equation (2):

$$Loss = \sum_{i=1}^{N} \alpha_i (1 - DSC_i), \alpha_i \geqslant 0, \sum_{i=1}^{N} \alpha_i = 1 \tag{2}$$

where $Loss$ is a scalar total multi-class loss function, N is the number of classes, α_i is a weighting coefficient, and $(1 - DSC_i)$ is the loss value for the i-th class. The weight coefficients α_i were chosen equal to $1/N$.

Segmentation Network Architecture

U-Net is widely regarded as a standard convolutional neural network architecture for biomedical image segmentation tasks. The architecture comprises two main components: a contracting path, which captures global context and feature hierarchies, and a symmetric expanding path, which enables precise localization of structures. The basis of the network of our study is the U-Net project (https://github.com/zhixuhao/unet, accessed on 2 December 2024). In the original project, U-Net was used for the binary classification of membranes.

We built compact modifications of the U-Net model and present here the tiny-U-Net model. The architecture of our simplified model is illustrated in Figure 1.

The tiny-U-Net has the following differences from the previous architecture:

- Our network input is an image of size 256 × 256 × 1 instead of 512 × 512 × 1.
- Our network output is 256 × 256 × N instead of 512 × 512 × 1, where N is the number of classes.
- We added batch normalization after each ReLU, convolution, and activation layers.
- Number of channels in the original U-Net convolution blocks: 64 → 128 → 256 → 512 → 1024; number of channels in our architecture: 32 → 32 → 64 → 128 → 256.

- The resulting model contains 15.7 times fewer parameters than the original model and takes up 15.2 times less memory (24 MB instead of 364 MB).

As in the original project, we use the output activation function sigmoid. We extend the application of this function not only for binary but also for multi-class segmentation. Using this activation function guarantees that each mask is in the range of [0, 1] and also provides the independence of masks. In this approach, one pixel of the layer can be associated with several classes at once, unlike the standard approach with the softmax function.

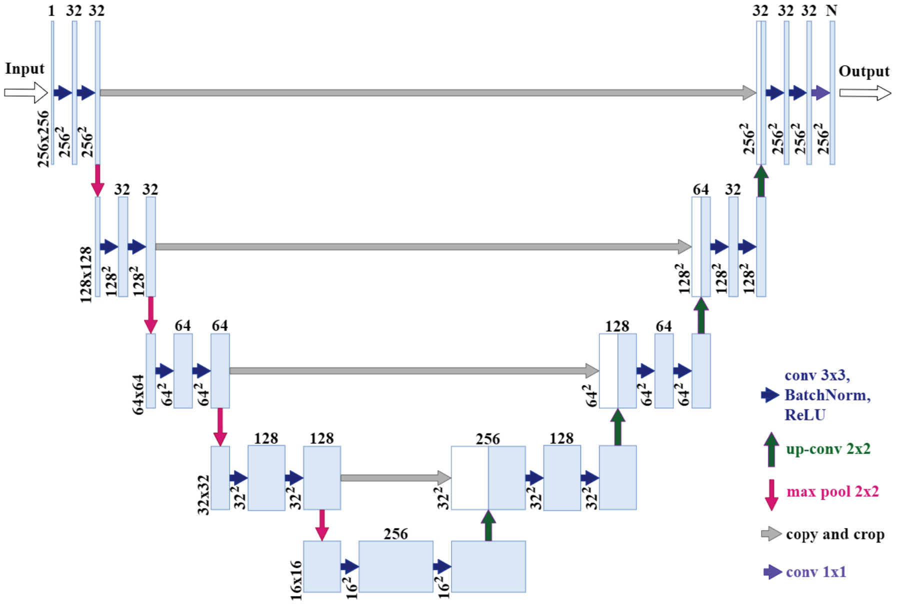


Figure 1. The architecture of the tiny-U-Net model.

2.2. Datasets and Their Markups

In this section, we provide an overview of publicly available datasets for electron microscopy (EM) segmentation task (Table 1). Among the most widely used datasets for this task are those collected by Lucchi et al. in [37]. These datasets have become a benchmark for evaluating segmentation performance in EM images.

Table 1. Open labeled electron microscopy datasets.

No.	Name	Data Volumes	Labeled Data Volumes	Labeled Classes	Resolution (nm/voxel)
1	AC4, ISBI 2013 [38]	4096 × 4096 × 1850	1024 × 1024 × 100	membranes	6 × 6 × 30
2	EPFL [b], Lucchi++ [c] [5]	1065 × 2048 × 1536	2 datasets 1024 × 768 × 165	mitochondria	5 × 5 × 5
3	Kasthuri et al. [39]		1334 × 1553 × 75 1463 × 1613 × 85	mitochondria	3 × 3 × 30
4	UroCell [a] [40]	1366 × 1180 × 1056	5 datasets 256 × 256 × 256	mitochondria, endolysosomes, fusiform vesicles	16 × 16 × 15

[a] Data are available on GitHub: https://github.com/MancaZerovnikMekuc/UroCell, accessed on 2 December 2024. [b] EPFL dataset is available at https://www.epfl.ch/labs/cvlab/data/data-em/, accessed on 2 December 2024. [c] Data are available at https://casser.io/connectomics/, accessed on 2 December 2024.

It is seen that in three of the four labeled open datasets, only one class is labeled. Only one dataset contains more than one labeled class. For this reason, the vast majority of neural networks in EM are trained to classify only two classes (object and background).

For our study, we utilized the EPFL dataset, which is publicly available at https://www.epfl.ch/labs/cvlab/data/data-em/, accessed on 2 December 2024. The data used in this work were acquired by a focused ion beam scanning electron microscope (FIB-SEM, Zeiss NVision40), which uses a focused beam of gallium ions to mill the surface of a sample and an electron beam to image the milled face. The milling process removes approximately 5 nm of the surface, while the scanning beam produces images with a pixel size of 5 × 5 nm [5]. This dataset comprises images obtained from the CA1 region of the hippocampus in the brain, with a voxel resolution of approximately 5 × 5 × 5 nm. The training set includes 165 image stack fragments, each with a size of 1024 × 768 pixels. This dataset was acquired by Graham Knott and Marco Cantoni at EPFL. Notably, the original EPFL dataset provides annotations exclusively for mitochondria. The test set of the EPFL dataset consists of 165 full-size images (1024 × 768 pixels) with corresponding ground truth annotations for mitochondria. These test images are used to evaluate the generalization capability of trained models on unseen data. An example of the dataset is illustrated in Figure 2.

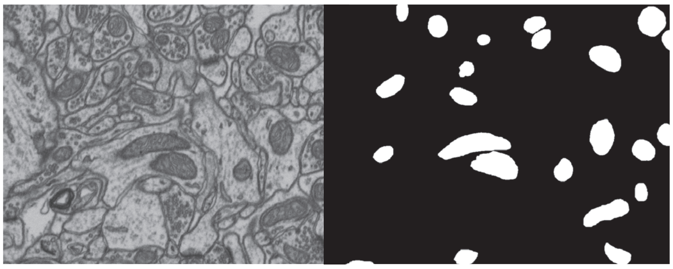


Figure 2. The electron microscopy 1024 × 768 image from the EPFL dataset of a mouse brain and mitochondrial annotation: (**a**) EPFL layer; (**b**) annotation mask.

For the EPFL dataset, an enhanced annotation called Lucchi++ is available [41]. The research team re-annotated two EPFL Hippocampus image stacks to enhance the consistency and accuracy of mitochondrial membrane labeling. The process involved a senior biologist manually refining the annotations, which were then independently reviewed by two neuroscientists. Disagreements were resolved through iterative corrections, ensuring consensus and high-quality annotations. In other words, the Lucchi++ dataset provided an enhanced labeling for both the training and test sets of the EPFL dataset, with each stack having dimensions of 165 × 1024 × 768 pixels.

But in addition to mitochondria, the EPFL dataset can be marked with 4 types of compartments: PSD, vesicles, membranes, and axons. And if the first three are presented in the dataset as well as mitochondria, axons, on the contrary, are very poorly represented. The axons' sheath in the training dataset is present only in the first 36 layers and looks completely different from the axon sheath in the test dataset Figure 3. In the test dataset, axons appear in the first 70 layers, transitioning from an elongated to a more rounded shape. Additionally, they exhibit a darker interior and an inner ring, further distinguishing them from the training set.

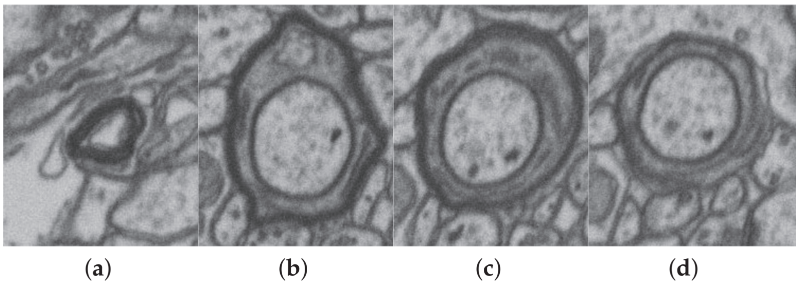


Figure 3. Axon sheath in the training and test EPFL datasets: (**a**) axon sheath in the training set; (**b**) axon sheath in the test set, first layer; (**c**) axon sheath in the test set, 35th layer; (**d**) axon sheath in the test set, 70th layer.

Since we wanted to deal with multi-class segmentation, we had to make our own markup for the existing dataset [42]. We marked 55 training layers and 5 test ones, on which we tested multi-class models. Also, we tested our models on the full EPFL test volume and the complete Lucci++ markup to compare our results with other studies.

Thus, in our work, we used three markups of the same dataset (EPFL, Lucchi++, and ours markups); the difference between the markups is shown in the Figure 4. We calculated the differences between markups using the formula

$$difference = \frac{count\_different\_areas\_in\_pixels}{count\_intersection\_in\_pixels} \tag{3}$$

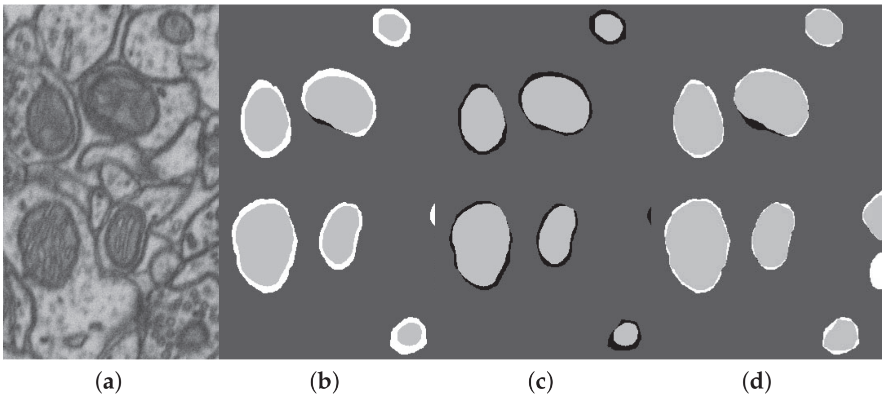


Figure 4. Markup differences (background—dark gray, intersection areas—light gray): (**a**) original EPFL EM fragment; (**b**) EPFL (black)/our (white) markup; (**c**) Lucchi++ (black)/EPFL (white) markup; (**d**) Lucchi++ (black)/our (white) markup.

The difference in the results for 42 layers is as follows: Lucchi++ vs. ours, 0.09; EPFL vs. ours, 0.21; Lucchi++ vs. EPFL, 0.19. This is a significant difference. This explains the fact that our test results are better for the Lucci++ markup. At the same time, it can be noted that Lucci++, due to enhanced markup checking compared to EPFL, like our dataset, more correctly solves the problem as a whole.

2.3. Segmentation Algorithm Stability

We investigated how stable both are: the process of training the multi-class segmentation of the tiny-U-Net model and the assessment of the segmentation accuracy by this model. It was interesting to obtain such an assessment both on our EPFL6 dataset and on the Lucchi++ dataset to ensure the correctness of using the segmentation assessment as an assessment of the quality of the generated synthetic datasets. It is also important that the dataset we are studying also contains underrepresented classes, which can happen in many cases of technology applications. Therefore, it is also interesting to see how the underrepresentation of a class would look like in such assessments. As a result, the EPFL6 dataset is represented by 42 layers from the training part and 5 layers from the test part (taken for marking from 165 layers of the EPFL test part). The results are presented in Table 2. The Lucchi++ column contains test results for model that trained on 100 (1st line) and 165 (2nd line) layers from the training set and 165 layers from the test set. Note also that the abbreviation "Mit.boundary" stands for "Mitochondrial boundary".

We trained the model 20 times and averaged the results. From Table 2, one can see that the metrics for mitochondria and membranes are quite stable from run to run. The standard deviation is ~0.007 and ~0.002. On the contrary, the results for the axon and PSD vary greatly from run to run.

Table 2. Stability of segmentation estimation by tiny-U-Net model on EPFL6 and Lucchi++ datasets.

Metric	Mitochondria	PSD	Vesicles	Axon	Membrane	Mit.boundary	Lucchi++
5 classes, Mean	0.925	0.800	0.727	0.128	0.872		0.928
5 classes, Std	0.007	0.022	0.004	0.152	0.002		0.006
6 classes, Mean	0.927	0.775	0.725	0.125	0.872	0.798	0.934
6 classes, Std	0.007	0.064	0.005	0.169	0.002	0.007	0.003

2.4. Technology for the Automatic Labeling of Synthetic Classes Based on the Diffusion Model

If you take an image and start applying Gaussian noise to it, after a sufficient number of iterations of adding noise, the original image will turn into a pattern of pure noise. The core concept of the diffusion models is to learn how to reverse the described process, gradually removing noise from a noisy image and eventually obtaining a clear image. Based on this, the network architecture should satisfy the following requirement—the input and output should have the same dimensions. An architecture that fulfills these requirements is, for example, the standard U-Net architecture. This approach has been successful in the field of image generation, and models using this method are starting to compete with and even surpass other types of generative models. For instance, such models already outperform Generative Adversarial Networks (GANs) in terms of perceptual quality metrics [28].

The idea of training a model capable of generating both an image and its corresponding annotation is based on creating a training dataset for a diffusion model where each input tensor contains information about the original image and the annotation for that image.

The EPFL dataset was used only as the source data for our own multiclass labeling. The new dataset was generated as follows:

1. The data with their annotations, layer-by-layer (in order of layer sequence), are combined into one common tensor of size H × W × C, where H and W are the image sizes, and C is the number of channels. The C value determines the number of channels in the original layer image (in our case, the image is grayscale, that is, one channel) plus the number of classes, for each of which a single-channel mask should be generated.
2. The layer image is divided into parts of size 256 × 256 pixels. This is performed to avoid scaling the image during the dataset preprocessing before model training, which could lead to loss of information about the original image and its details as well as blurring of images.
3. For layers, standardization (or Z-Score normalization) is used, and mask normalization (or Min-Max scaling or division by 255) is used. The raw dataset data are converted using the following formulas:

$$layer_{normalized} = \frac{layer - mean_{layer}}{std_{layer}},\ mask_{normalized} = \frac{mask - min_{mask}}{max_{mask} - min_{mask}} = \frac{mask}{255} \quad (4)$$

U-Net from the python library diffusers (https://pypi.org/project/diffusers/, accessed on 2 December 2024) was used as the architecture of the diffusion model. The model was created using a function called UNet2DModel and allows one to obtain a model with additional modifications, such as embedding a position for diffusion time and internal attention blocking.

To create the model, the UNet2DModel function was used from diffusers with the following parameters: $sample\_size = 256$—sets the width and height of the input data; $in\_channels$ and $out\_channels$ = number of classes + 1 (adding 1 channel for a layer)—set the number of input and output channels; $layers\_per\_block = 2$—the number of repetitions convolution/deconvolution per block; $block\_out\_channels$ = (128, 128, 256, 256, 512, 512)—list from the number of channels after exiting the blocks (the list has an equal encoder or decoder block); the encoder block contains 4 "DownBlock2D", after them "AttnDownBlock2D" and "DownBlock2D"; the decoder block contains "UpBlock2D", "AttnUpBlock2D", and 4 more blocks "UpBlock2D".

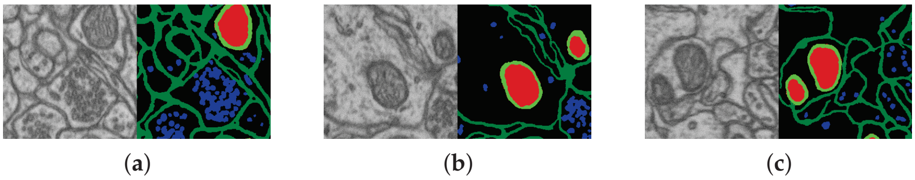
The resulting model is shaped like U-Net, contains 6 convolutional and deconvolutional layers (or blocks) with skip connections, and also has an attention model (block with Attn) and an input to indicate the current step of the diffusion (or denoising) process.

To return the normalized layers to the previous pixel intensity range of 0–255, the reverse method was applied. For this, the following formula was applied:

$$image = image_{after\_model} * std_{train\_dataset} + mean_{train\_dataset} \quad (5)$$

where $mean_{train\_dataset} = 138.84$ and $std_{train\_dataset} = 29.68$ are the average and std of the pixel intensities of 42 layers of the EPFL training dataset. This method allows us to avoid shifting the average and std intensity values across channels, which can be observed with other methods of returning the image to the 0–255 range. To return the masks, the pixel values were multiplied by 255.

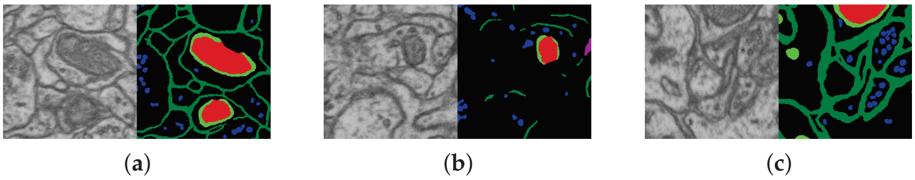
Figure 5 shows examples of good generation using the diffusion model. Layers of masks are translated into one image, where different classes are marked with different colors: red—the inside area of the mitochondrion, light green—the border of the mitochondrion, green—membranes, blue—vesicles.



(**a**) (**b**) (**c**)

Figure 5. Successful examples for the 6-class-labeled synthetic dataset generated by a diffusion model; (**a**–**c**)—3 synthetic images and their labeling; the training dataset includes 30 EPFL6 layers.

Examples of unsuccessful generations are also shown (Figure 6) as 3 synthetic images and their labeling masks.



(**a**) (**b**) (**c**)

Figure 6. Examples of mistakes for the 6-class-labeled synthetic dataset generation via a diffusion model. The mask shows mitochondria boundaries having gaps (**a**,**b**); the membrane mask does not quite match the layer (**b**); mitochondria consisting of only the boundary class (**c**).

Examples of axon generations are shown in (Figure 7).

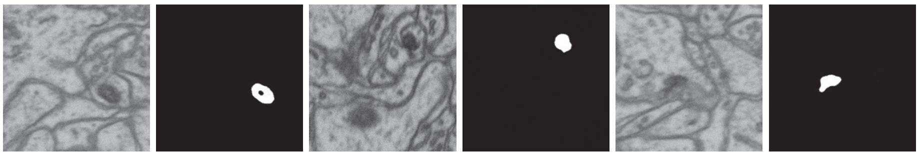


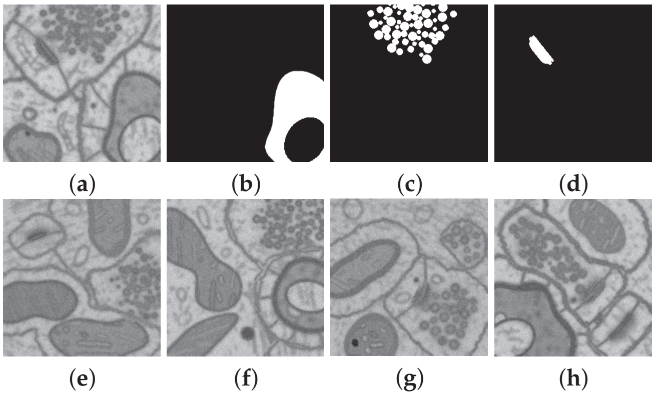
Figure 7. Example of synthesized data by DIFF 6-class (42-layer) diffusion model. Synthetic tile (fragment) and it's axon masks.

2.5. Geometric Models for Dataset Synthesis

The research presented in this section is a development of our work [34,35]. We present new samples of synthetic images generation as the main method of augmentation for classes that are underrepresented in the dataset. In the EPFL dataset, this is an axon class (see Figure 3).

The essence of the geometric algorithm is that organelles are drawn using simple geometric primitives such as lines, splines, and area fills. The sizes and gray levels for organelles are set based on data from the target dataset. A complex internal structure is also applied by adding lines, circles, spline pieces, and blurs to the selected area. Membranes are calculated using a region-growing algorithm. We used Gauss filtering and Poisson noise to simulate image blurring and noise from the registration device. Drawing masks repeats the algorithm for drawing organelles, only without blurring and adding noise. One can see and download the implementation of our algorithms for geometric synthesis here: https://github.com/GraphLabEMproj/Synthetics/, accessed on 2 December 2024. A description of the geometric synthesis algorithms can be found in [35].

This is why, for the geometric synthesized dataset, we generated 2000 images of size 256 × 256 pixels; half of this dataset contained the axon area. An example of one labeled sample and an example of several samples are shown in Figure 8. The shape, size, and gray levels of compartments are chosen to be similar to the shape, size, and gray levels of the EPFL dataset. The advantage of a synthetic set is that you can obtain any number of images you need along with their labeling automatically.



(**a**) (**b**) (**c**) (**d**)

(**e**) (**f**) (**g**) (**h**)

Figure 8. Example of geometric synthesized data: (**a**) layer, (**b**) mask of axons' sheaths, (**c**) mask of vesicles, (**d**) PSD mask. (**e**–**h**) examples of synthetic layers.

2.6. Naturalization of Geometric and Other Synthetic Datasets by Diffusion Models

The Table 3 shows that geometric synthetics have a significant drawback due to their schematic and complexity of the brain cell shapes. The column best results are highlighted by the bold font, and the best for left or right halftable by the gray background.

Diffusion networks do a great job of synthesizing textured objects such as mitochondria and clusters of vesicles, but diffusion neural networks cannot confidently model properties that geometric synthetics can guarantee: membrane continuity and PSD placing. We decided to combine the strengths of diffusion and geometric synthetics by naturalizing the result of geometric synthetics with a diffusion model. To do this, an experiment was conducted to determine both the possibility of such a naturalization method and adequate proportions for mixing the models. In this experiment, images of geometric synthetics were noisy to a level determined by the coefficient α and then fed for denoising to the input of a diffusion model trained on the original data. For a number of values of α, the naturalization of the geometric model reaches a level equal to the result of applying the diffusion model to the original dataset.

We used a geometric synthetic image as the initial image and made it noisy as required by the diffusion model. The noisy image is fed to the input of the diffusion model in Section 2.4 and trained on images of the original dataset. The input of the model forms one common tensor from the layers of the original layer and the mask; this entire tensor is noisy and restored by diffusion. For masks, the operations are the same as for a layer. Therefore, if diffusion corrects the layer, it also corrects the masks. We tested 20 levels of added noise (see Figure 9).

Table 3. Dice coefficient of electron microscopy data segmentation for the tiny-U-Net model for the original EPFL6 dataset (ORIG), the dataset enriched with *diffusion*-synthesized images (MIX), and the fully diffusion-synthesized dataset (DIFF).

	Training Dataset											
Num	**MIX**		**DIFFUSION**		**ORIGINAL**		**MIX**		**DIFFUSION**		**ORIGINAL**	
	MIX 5	**MIX 6**	**DIFF 5**	**DIFF 6**	**ORIG 5**	**ORIG 6**	**MIX 1**	**MIX 6**	**DIFF 1**	**DIFF 6**	**ORIG 1**	**ORIG 6**
	Mitochondria						Mitochondria/Mitochondrial boundaries					
5	0.860	0.886	0.850	0.871	0.882	0.885	0.880	0.760	0.835	0.741	0.877	0.752
10	0.943	0.935	**0.936**	0.906	**0.934**	**0.928**	0.921	0.797	0.910	0.752	0.916	0.787
15	**0.945**	**0.942**	0.930	0.937	0.930	0.926	0.918	0.793	0.876	**0.762**	**0.914**	0.790
20	0.939	0.938	0.931	**0.941**	0.924	0.925	**0.925**	0.795	**0.924**	0.754	0.912	0.794
30	0.932	**0.942**	0.895	0.928	0.922	0.924	0.924	**0.799**	0.916	0.751	0.913	**0.796**
	MIX 5	MIX 6	DIFF 5	DIFF 6	ORIG 5	ORIG 6	MIX 5	MIX 6	DIFF 5	DIFF 6	ORIG 5	ORIG 6
	PSD						Membranes					
5	0.672	0.654	0.513	0.582	0.556	0.566	0.868	0.868	0.849	**0.852**	0.861	0.862
10	0.820	0.783	0.735	0.532	**0.755**	0.754	0.869	**0.872**	**0.854**	0.847	**0.872**	**0.873**
15	0.723	0.755	0.621	0.643	0.685	0.697	0.869	0.864	0.844	0.814	0.871	0.872
20	**0.821**	0.773	**0.748**	0.731	**0.755**	0.726	*0.865*	0.864	0.831	0.813	0.871	**0.873**
30	0.816	**0.810**	0.472	**0.783**	0.723	**0.766**	**0.872**	0.864	0.849	0.791	**0.872**	**0.873**
	Vesicles						Axon					
5	0.692	0.697	0.687	0.693	0.689	0.686	0.094	0.043	0.023	0.036	0.144	0.282
10	**0.734**	0.719	0.727	0.704	**0.720**	0.714	0.009	**0.062**	0.013	**0.178**	**0.304**	0.265
15	0.730	0.728	0.735	0.725	*0.716*	0.714	0.030	0.007	0.013	0.024	0.122	0.274
20	**0.734**	0.728	**0.736**	0.729	0.714	0.713	0.000	0.000	0.022	0.097	0.243	0.181
30	0.725	**0.735**	0.731	**0.730**	**0.720**	**0.720**	**0.274**	0.010	**0.376**	0.007	0.192	**0.312**

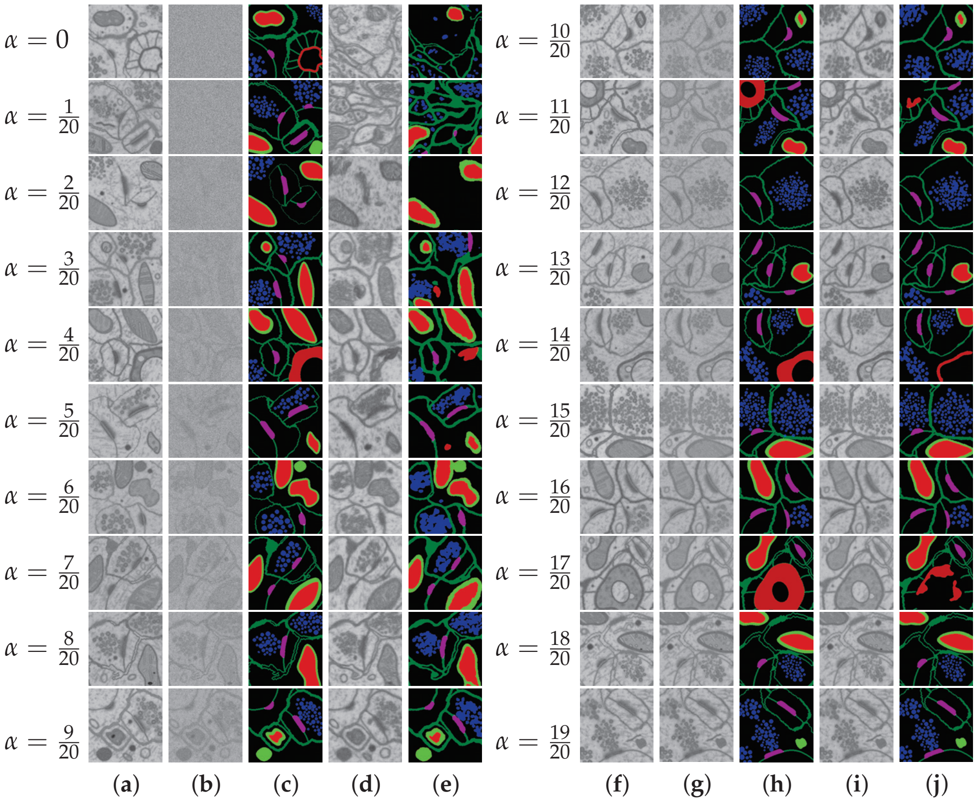


Figure 9. Naturalization of geometric synthetics using a diffusion model. Columns: (**a**,**f**)—geometrically synthesized image; (**b**,**g**)—α-noised synthetic; (**c**,**h**) mask of geometric source; (**d**,**i**)—diffusion refinement; (**e**,**j**)—diffusion refinement mask. The α parameter controls the noise of the geometric source: $\alpha = 0$—the source is completely noisy; $\alpha = 1$—no noise is added.

3. Results

The experiment was designed as a two-stage one: Stage 1—synthesis of a multi-class dataset by a diffusion model trained on EPFL6; Stage 2—evaluation on the test part of EPFL6 of the accuracy of multi-class segmentation trained on its training part.

Before discussing the obtained results and to understand them better, it is worth clarifying how the training datasets for the diffusion model and for the segmentation model were formed in each experiment, on whose test dataset the segmentation accuracy was checked.

First of all, we note that only two original datasets were used in the experiments: EPFL, marked up for one class of mitochondria, in the refined Lucchi++ markup; EPFL6 with our markups for one, five, and six classes. Only one designation is used for EPFL6. EPFL with the refined markup of mitochondria is designated as Lucchi++ or Lucchi for short in the tables and explanatory texts. Lucchi++ has two parts, training and testing, with 165 layers each. From EPFL6, from 5 to 42 layers from the training part and 5 layers from the testing part were used in the experiment. For the augmentation of 5 EPFL6 test layers, regular tiling with a 256 × 256 window with an offset of 128 was used, giving 35 [tiles/layer] instead of 12 [tiles/layer] (without overlapping). Regardless of how many layers are used to train the model, all test layers of the corresponding datasets are used for testing. All tables show Dice values averaged over several (from 5 to 20) implementations.

The size of the synthetic datasets is practically unlimited, but for resource conservation reasons, 1008 disjoint images of size 256 × 256 were generated for any size of the training dataset, equivalent to 84 standard EPFL 1024 × 768 layers, i.e., 12 images per layer. Synthetic datasets were generated once, a separate dataset was generated for each combination of the number of classes and number of training layers. Their naming is more diverse and is tied to the result tables. To increase the volume of the training dataset, regular tiling was applied to its layers with a tile size of 256 × 256 and an offset of 64 (or 128), selected based on the experiment. The number of tiles in a layer when tiling without tile intersections is 12 (4 × 3). With an offset of 128, the number of tiles k is determined by the formula k = (2n − 1) × (2m − 1), where n = 4, m = 3, and k = 35 = 7 × 5. The next division of the offset by 2 gives, according to the same formula, k = 117 = (2 × 7 – 1)(2 × 5 – 1). That is, when training on L layers, we have the following pairs in the training dataset: [number of layers, number of intersecting 256 × 256 tiles]: [5, 585], [10, 1170], [15, 1755], [20, 2340], [30, 3510], [42, 4914]. In this case, the number of diffusion synthetic images generated on each training dataset, as indicated above, remained constant in all experiments and was equal to 1008 tiles. This option is interesting because it allows us to compare the segmentation accuracy when training on an increasing and constant dataset volume. The only exception is made for the five-layer dataset, in which the DIFF synthetic training dataset is represented by 595 out of 1008 images.

For the computational experiment, the nodes of the supercomputer (SC) Lobachevsky (Lobachevsky University, Nizhny Novgorod) (https://hpc-education.unn.ru/en/resources, accessed on 2 December 2024), equipped with eight graphic processors (GPUs) A100, 40 GB, were used. A total of 100 epochs were allocated for the training process of the diffusion model on the original datasets formed from the EPFL6 composition, which took about 2.5 h A100, 40 GB. The process of generating a synthetic dataset of 1008 images of size 256 × 256 for each of the markings (1, 5, 6 classes) by the trained model took about 5 h GPU A100, 40 GB. The time on the computing node together with loading and unloading was about 8 h for one synthetic dataset of 1008 images of the above size. A total of 200 epochs were allocated for the training process of the tiny-U-Net segmentation model.

3.1. Software and Technologies for Training and Testing

We used Python as the main programming language to conduct experiments. The main libraries for working with deep learning are PyTorch and Diffusers. PyTorch is chosen as one of the most popular libraries for DL. The Diffusers library, based on PyTorch, contains functions for creating and training models using diffusion technology.

Training deep learning models requires a large amount of data to reduce the possibility of overfitting. At the same time, the larger the model, the more data it requires. To increase the amount of data for training, we sliced the training layers into tiles of 256 × 256 pixels. In our previous works, we used 256 × 256 slicing with an offset of 128 and 512 × 512 with an offset of 256, which allowed us to obtain 41 tiles per layer. Thus, for 42 layers, 1722 tiles of training data were obtained. When conducting initial experiments with the standard diffusion model, which is large in size, the generation results after training were unsatisfactory, so it was decided to increase the training dataset by reducing the slicing bias to 64. Therefore, in this work, slicing the layer into 256 × 256 tiles with an offset of 64 was used to train the models. This allowed for obtaining 117 tiles from a single 1024 × 768 pixel layer. It is worth noting that these data are not diverse due to multiple overlaps. From one 1024 × 768 layer, only 12 tiles of a size of 256 × 256 can be obtained without intersections, so expanding the dataset by slicing with an offset increases its diversity nonlinearly and loses its power at small values.

U-Net, despite its architecture, may still face difficulties when processing image edges. To avoid artifacts at the edges of tiles, we cut the test image into overlapping tiles.

We prepared tiles with a size of 256 × 256 pixels and an overlap of 128 pixels for testing. Only the centers of the 128 × 128 pixel regions within these tiles are included in the final mask (see Figure 10).

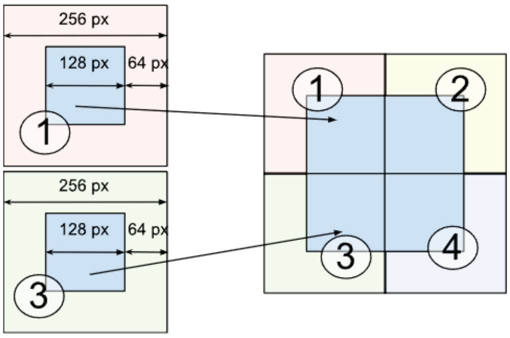


Figure 10. Process of splitting a test image into tiles.

3.2. Generation of Synthetic Multi-class Datasets by Diffusion Model

The U-Net model was trained with the following hyperparameters: input and output image sizes = 256 × 256 × (N + 1) pixels, where N is the number of classes; batch size for training set = 12; number of training epochs = 100; learning rate for optimizer = 0.0004; number of denoising steps = 700.

With these parameters, we generated the next 1000-tiles training volumes:

- DIFF 6-class 42 layers: 42 layers from the EPFL6 training dataset, all six classes (the example: see Figure 11). The diffusion model was trained on this dataset, and this model synthesized the input datasets DIFF 6 and MIX 6 (with the addition of the EPFL6 dataset images) for the segmentation task in Tables 3 and 4.
- DIFF 5-class 42 layers: 42 layers from the EPFL6 training dataset, five classes of markup. The diffusion model was trained on this dataset, and this model synthesized the input datasets DIFF 5 and MIX 5 (with the addition of images from the EPFL6 dataset) for the segmentation task in Tables 3 and 4.
- DIFF 1-class 42 layers: 42 layers from the EPFL6 training dataset, one class of markup. The diffusion model was trained on this dataset, and this model synthesized the input datasets DIFF 1 and MIX 1 (as a fusion with the images from the EPFL6 dataset) for the segmentation task in Table 4.
- DIFF 1-class 165 layers: 165 images from the EPFL one-class training dataset in Lucchi++ labeling. The diffusion model trained on this dataset synthesized the input dataset for the combination: Lucchi++ plus DIFF 1(165), 84 (Table 5).

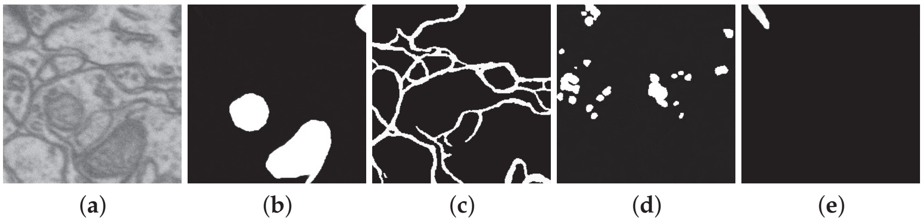


Figure 11. Example of synthesized dataset DIFF 6-class 42 layers (only nonzero masks are shown): (**a**) the synthetic layer, (**b**) mitochondria mask, (**c**) membranes mask, (**d**) vesicles mask, (**e**) PSD mask.

Table 4. Dice coefficient for EPFL6-based synthetic data segmentation by tiny-U-Net model. The table has 3 main sections: Original datasets; Synthetic datasets; Mixed two firsts, - for 3 types of synthetics.

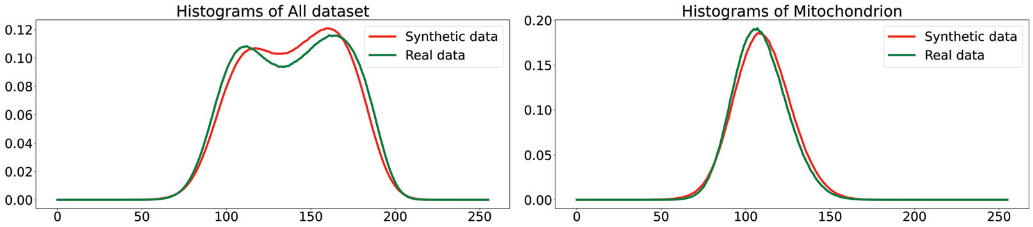
Metric	Mitochondria	PSD	Vesicles	Axon	Membranes	Mit. Boundaries
MIX NAT 1	0.920	-	-	-	-	-
MIX NAT 5	0.924	**0.851**	0.708	0.527	0.876	-
MIX NAT 6	0.928	0.842	0.716	0.534	**0.877**	0.802
MIX DIFF 1	0.927	-	-	-	-	-
MIX DIFF 5	**0.944**	0.833	**0.734**	0.017	0.867	-
MIX DIFF 6	0.939	0.841	0.732	0.000	0.869	0.805
MIX GEOM 1	0.926	-	-	-	-	-
MIX GEOM 5	0.936	0.836	0.725	**0.789**	0.871	-
MIX GEOM 6	0.933	0.845	0.721	0.722	0.873	**0.807**
SYN NAT 1	0.883	-	-	-	-	-
SYN NAT 5	0.885	0.701	0.510	0.565	0.810	-
SYN NAT 6	0.839	0.652	0.512	0.542	0.793	0.638
SYN DIFF 1	0.918	-	-	-	-	-
SYN DIFF 5	**0.942**	0.781	**0.736**	0.072	***0.839***	-
SYN DIFF 6	**0.942**	**0.808**	0.730	0.025	*0.831*	**0.775**
SYN GEOM 1	0.891	-	-	-	-	-
SYN GEOM 5	0.905	0.704	0.623	0.882	0.792	-
SYN GEOM 6	0.905	0.708	0.609	**0.898**	0.790	0.704
ORIGINAL 1	0.913	-	-	-	-	-
ORIGINAL 5	**0.928**	**0.824**	**0.732**	**0.133**	0.872	-
ORIGINAL 6	**0.928**	0.814	0.724	0.070	**0.873**	0.799

We constructed histograms to visualize the statistics of the datasets, as presented in Figure 12. The histograms of real and synthetic datasets are indeed very similar, but the range of values in the real image is slightly wider. Based on the histogram comparison, it can be concluded that the diffusion model predominantly generates images similar to the original dataset. By visual analysis it was found that a small part of the generation contains errors. Examples of successful and unsuccessful generations are shown in Figures 5 and 6. However, this part containing errors was not excluded from the dataset.

Table 5. Comparison with existing mitochondrial segmentation methods.

Method	Labeling	Dice
HIVE-net [43]	Lucchi++	0.948
tiny-U-Net [2]	Lucchi++ plus DIFF 1 (165), 84	0.946
tiny-U-Net [2]	Lucchi++	0.934
tiny-U-Net [2]	Lucchi++, 100 (out of 165)	0.928
tiny-U-Net [2]	DIFF 1 (165), 84	0.927
tiny-U-Net [2]	DIFF 6 (42), 84	0.917
tiny-U-Net [2]	Lucchi++, 42 (out of 165)	0.913
3D Casser et al. [41] [1]	Lucchi++	0.942
Cheng et al. (3D) [44] [1]	Lucchi++	0.941
3D U-Net [11] [1]	Lucchi++	0.935
Cheng et al. (2D) [44] [1]	Lucchi++	0.928
U-Net [7] [1]	Lucchi++	0.915
Peng et al. [45] [1]	Lucchi++	0.909
3D Xiao et al. [10] [1]	Lucchi++	0.900
Cetina et al. [46] [1]	Lucchi++	0.864
Lucchi et al. [5] [1]	Lucchi++	0.860

[1] The Dice coefficient was taken from article [43]. [2] This article.



(**a**) (**b**)

Figure 12. Histograms of the dataset DIFF 6-class 42 layers (histogram of the original dataset in green, histogram of the synthetic dataset in red). (**a**) histograms of all dataset; (**b**) histograms of mitochondrion.

3.2.1. Training a Segmentation Neural Network on a Dataset Using Synthetic Images Generated by a Diffusion Model

In our previous work [35], we used a simplified U-Net model for segmentation (named tiny-U-Net). For mitochondria segmentation, tiny-U-Net demonstrated very close assessments like classical U-Net. The lightweight model contains 15.7 times fewer parameters and occupies 24 MB of RAM instead of 364 MB when using the original model.

The model was trained with the following hyperparameters:

- Input and output image size: 256 × 256 pixels;
- Batch size for the training set: 7;
- Number of epochs: 200;
- Adam optimizer with a variable learning rate from 1×10^{-4} to 1×10^{-6}.

The results of the multi-class segmentation of the electron microscopy data for the tiny-U-Net model for the EPFL dataset and the diffusion dataset are presented in Table 3. The numbers 5 and 6 indicate how many classes the diffusion model was trained for and how many classes the segmentation model was trained for. Our training dataset is EPFL6 with (5, 10, 15, 20, 30) layers. The test dataset is EPFL6. The best value of the two is in italics, the best value in the line is in bold, and the best value in the class is marked in light gray.

3.2.2. Experiments on Geometric Model for Dataset Synthesis

In order to determine the effectiveness of using the geometric expansion of the dataset, we trained the neural network on datasets:

GEOM is a training dataset that includes only geometric synthetic data. To obtain a geometric synthetic (GEOM) training dataset, we generated 2000 synthesized fragments of size 256 × 256.

MIX (GEOM) is a mixed training dataset. It includes 4914 fragments of EPFL6 (42 layers of the EPFL6 training dataset cut into 256 × 256 fragments, with an offset of 64 pixels) data and 2000 geometric synthesized fragments; thus, we have 6914 fragments in total.

To additionally increase the training datasets, we made random rotations of images, random shifts, and random scale changes in a small range (5%). We selected 20% of images from the training sample into a validation sample. The batch size was equal to seven.

We used Adam's optimizer with a dynamic learning rate from 1×10^{-4} to 1×10^{-6}. The learning rate after the 100th epoch decreased by 5 times, and this was repeated every 25 epochs.

We trained models for one, five, and six classes. The number of epochs in all experiments was 200. The Dice coefficients are presented in Table 4. The mitochondrial boundaries class is a subclass of the class of mitochondria with their boundaries, and the additional edge enhancement improves the segmentation results of the unifying class.

In commercial applications based on deep learning, in addition to quality metrics, the performance characteristics of algorithms also play a large role. Based on the values of the Dice quality metric given in Table 4 (the results for U-Net can be found in [34]), we see that with a tenfold decrease in the number of model weights (and, hence, the execution time), the results of the quality of work remain comparable.

3.2.3. Naturalization Results

Based on the testing results, it can be said that the diffusion model improves the texture of the inner region of the mitochondria well and improves the appearance of the PSD and vesicles. However, this improvement does not cope with those classes that are poorly represented in the training dataset. For example, in the original training dataset, the axon is represented by a single instance of a small size. Following this, the trained diffusion model tries to reduce by several times the area-size of the axon generated by the geometric model. For example, $\alpha = 4/20$ or $\alpha = 11/20$ and 14/20 and 14/20 (see Figure 9).

We can see the results in the NAT columns of Table 4 (augmentation due to the fusion of geometric and diffusion synthetics).

3.3. Results' Comparison

Table 4 presents the results of a comparative experiment of segmentation models for different types of synthetics. The ORIGINAL-prefixed rows indicate 42 layers of the EPFL6 train dataset. The rows entitled MIX indicate a mixed dataset: 42 original layers + synthetics. The type of synthetics is specified by the second word: GEOM—geometric synthetics; DIFF—diffusion synthetics, NAT—naturalization of geometric synthetics by diffusion model with $\alpha = 0.5$. The best value of the type MIX, SYN, or ORIGINAL is in bold font, and the best value in the class is marked in a light gray background. Test dataset of EPFL6 has five test layers.

In this section, we also visualize segmentation masks to provide a clearer understanding of the comparative results previously analyzed. By overlaying the masks on the original images, we highlight the differences between the ground truth and model predictions: the ground truth boundaries are shown in green, while the predicted boundaries are displayed in red. For visualization, we choose three models corresponding to the MIX NAT 5, MIX DIFF 5, and MIX GEOM 5 rows in Table 4.

Figure 13 shows that the MIX GEOM 5 model copes with axon segmentation significantly better than the others. The model MIX NAT 5 highlights the axon boundaries, while MIX DIFF 5 highlights almost nothing.

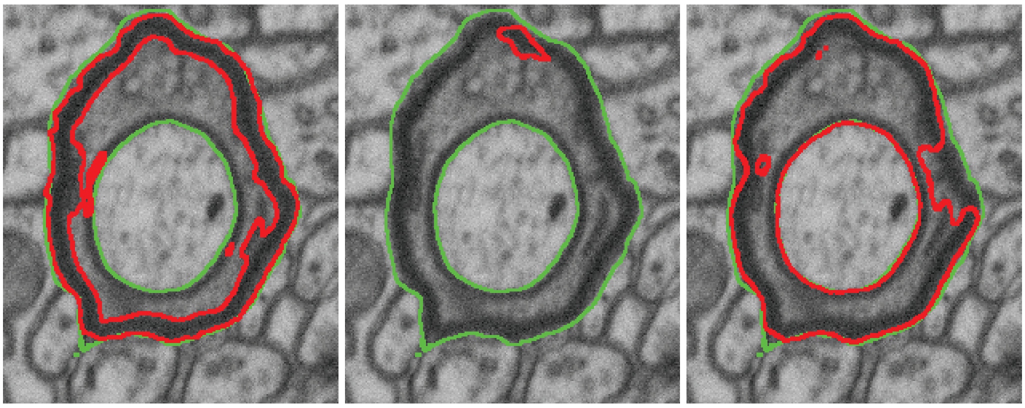


Figure 13. Axon prediction from left to right: MIX NAT 5 model prediction, MIX DIFF 5 model prediction, MIX GEOM 5 prediction.

The segmentation results for the vesicles and mitochondria classes are comparable, as evidenced by their similar Dice scores in Table 4. However, the diffusion model demonstrates superior performance in accurately identifying the boundaries of vesicles, highlighting its ability to capture finer details in these structures (see Figures 14 and 15).

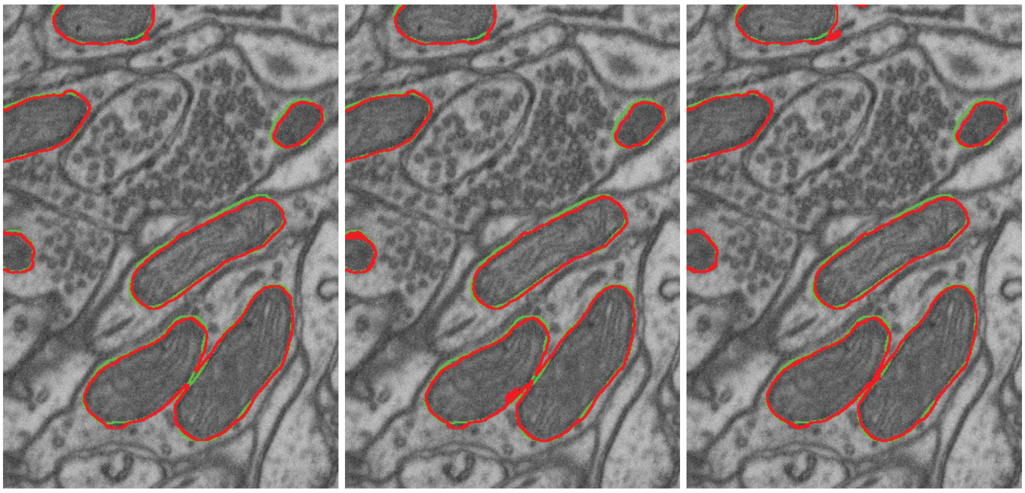


Figure 14. Mitochondria prediction from left to right: MIX NAT 5 model prediction, MIX DIFF 5 model prediction, MIX GEOM 5 prediction.

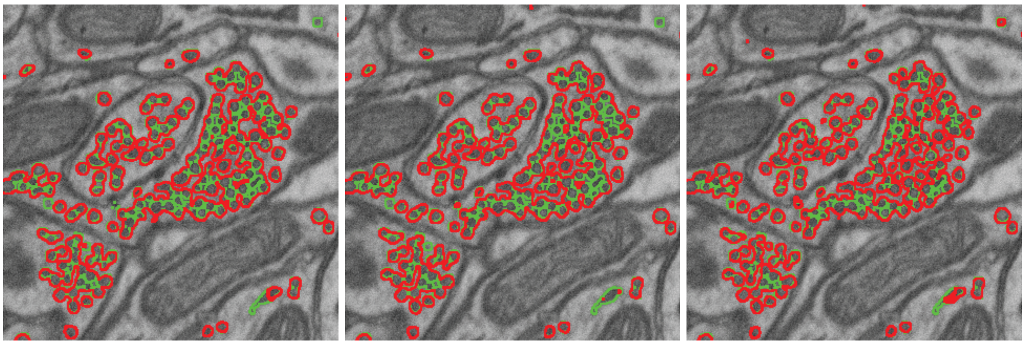


Figure 15. Vesicle prediction from left to right: MIX NAT 5 model prediction, MIX DIFF 5 model prediction, MIX GEOM 5 prediction.

4. Discussion

Table 3 contains the results of segmentation quality when training on synthetic datasets generated by the diffusion model. For testing, we used a comparison of class segmentation accuracies by the tiny-U-Net model trained on (1) the original EPFL6 dataset (ORIGn), (2) diffusion synthetic datasets (DIFFn), (3) original dataset augmented by the synthetic one (MIXn).

The results are shown for all classification objects and for all three labels: for one class (mitochondria), five and six classes. The data are averaged over five training runs of the segmentation neural network. The accuracy dependence on the training dataset size is analyzed for a small number of layers compared to Lucchi++. The results for mitochondria are given in the upper data strip: for five and six classes in the left half of the strip, the single-class labeling is moved, for the sake of table compactness, to the left columns of the right half of the strip, and the results for the sixth class of mitochondrial boundary are in the right columns. The dependencies on the number of layers in the training dataset behave similarly. When moving from 5 layers to 10 layers, a noticeable increase in accuracy is visible. Further, on the original data for all three markups, the accuracy remains almost constant, but the accuracy of mitochondria segmentation for classes 5 and 6 is noticeably higher than for the single-class markup. On the diffusion dataset, the average, by the number of layers, accuracy of the single-class markup is slightly worse than for ORIG 1. For classes 5 and 6, the average, by the number of layers, values are close to ORIG 1, but the maximums are noticeably higher and the minimums are noticeably lower. The values for MIX always exceed the values from ORIG and DIFF, and they give record values for the class. The mitochondria class gives the maximum segmentation accuracy for organelles. Synthetic datasets confidently increase the accuracy of mitochondria segmentation by augmentation, and, even with a small volume of the training dataset, they can, on average, replace the original dataset, only with a slightly larger standard deviation.

The Class 6 mitochondrial boundary shows weak but stable growth with an increasing number of layers in the training dataset for all three types of datasets. The DIFF column values are 1–4 hundredths lower than in ORIG, and MIX is always higher than ORIG and DIFF.

For the PSD class in the ORIG dataset, there is an initial jump, then fluctuations around the average level with maxima in the 20 s for five classes and 30 s for six classes. The DIFF dataset is generally lower than ORIG but repeats the position of the minimums and maximums. The MIX dataset results are consistently higher than the ORIG and DIFF datasets. That is, DIFF can be used for augmentation regardless of the size of the training dataset.

The membrane class in the original dataset (ORIG) experiences a jump when moving from 5 layers to 10, and then a plateau for five and six classes; the differences between the labeling options are 1–2 thousandths. The values for DIFF are noticeably lower (by a few hundredths) than ORIG. The values in MIX are mostly lower than in ORIG. This means that with a small number of training layers, synthetic membrane datasets cannot be used for augmentation. The table shows that for augmentation, a synthetic dataset can only be used for 5 layers or more than 30 layers.

The vesicle class is close to the typical behavior in the ORIG dataset, but in the DIFF dataset, on average, it exceeds ORIG, especially for five classes. However, with a training dataset volume of 30 layers or less, it is impossible to consistently use the vesicle synthetic datasets for augmentation. The possibility of using DIFF datasets instead of ORIG requires additional research because of this. It also requires research into the size of the DIFF training dataset at which it will stably model the ORIG dataset.

The axon class is a striking example of highly underrepresented classes and requires first compensating for the underrepresentation by other methods of creating synthetic datasets.

Table 4 shows the results of the experiment on the 42-layer (4914 tiles) EPFL6 training set, which, judging by Table 3, gives hope for greater stability than the previous experiment limited to 30 EPFL6 layers, which, however, revealed behavioral features of a number of classes that we would not have learned about otherwise.

The experiment was mainly aimed at solving the problem of significant underrepresentation, but we also had the opportunity to see how the properties of the synthetic datasets changed with a noticeable increase in the original dataset.

In the table, in addition to the three clear ORIGINAL lines, two main parts are highlighted: (1) the part in which the line names begin with SYN, and synthetic datasets of three types are considered—geometric (GEOM), diffusion (DIFF), and naturalization of geometry by the diffusion model (NAT). The number at the end of the identifier indicates the number of classes in the segmentation. (2) The part in which the names of the lines begin with MIX and the results of augmentation of the original dataset of each of the synthetics of the SYN part are considered.

Geometric parameterized synthetics were proposed and quite successfully implemented by us earlier [35] (see Section 2.5 and Figure 8). The volume of the synthetic dataset, built on the basis of parameterized geometric models, was 2000 images of size 256 × 256. For all classes, except axon, the quality of training on geometric synthetics is slightly worse than on the original dataset, but, nevertheless, the augmentation of the original dataset with it gives a stable positive result for three classes (mitochondria, mitochondria boundaries, PSD). For two, it is slightly worse or at the level of the ORIG dataset (vesicles, membranes). For axon, purely geometric synthetics are better than augmentation since the ORIG dataset cannot provide training and, therefore, augmentation.

The SIN NAT and MIX NAT lines explore the possibility of naturalizing geometric synthetic datasets with a diffusion model and then using them to augment the original dataset. We found that naturalized datasets give a lower accuracy when trained on them than on purely geometric ones. However, when used as an augmentation for two organelles, we obtained a positive effect: naturalized membranes became suitable for augmentation, and in the case of PSD, the result exceeded the capabilities of all other models.

Diffusion synthetic datasets with a volume of 1008 frames (images) of size 256 × 256, generated by the diffusion model trained on 42 layers of the original EPFL6 dataset, gave good results, especially for mitochondria, corresponding to the statistics of the original dataset.

In general, geometric synthesis turned out to be useful for augmenting the original dataset, and the naturalization of geometric synthetics by the diffusion model turned out to be effective on objects with special properties: geometric PSDs lack noise; geometric membranes provide continuity, but are also not sufficiently blurred; axons are large smoothed shapes with stochastic filling.

Also, according to Table 4, one can choose the most effective training policy for each class.

As a result, we can conclude that for the practical application of the technology, 42 layers were sufficient only for the mitochondria class. This means that the statistics of the training dataset were reproduced on synthetic datasets. It is important that the statistics of the training dataset ensure a high segmentation accuracy. It is useful to continue multi-class labeling of the open EPFL6 dataset to cover all EPFL layers.

Table 5 contains the results of experiments that demonstrate the suitability of synthetic datasets generated by the diffusion model for both stand-alone use instead of the original datasets and for augmenting the original dataset. The results are averaged over 20 implementations.

The first row of the table shows the results that have remained record-breaking for the Lucchi++ dataset since 2021. This is the result of a special HIVE-net model that, working on 2D layers, maintained the stability of the position of the mitochondrion center (axis) in 3D space.

The second through seventh rows show the dynamics of segmentation accuracy for a model trained on the original EPFL dataset in Lucchi++ markup and diffusion synthetic datasets trained on the original EPFL data. The results obtained on different numbers of layers of the original datasets are compared. Synthetic datasets are used both instead of the original ones (lines 5–6) and as an augmentation of the original dataset (line 2). Accuracy in lines 2–7 of the right column forms a monotonous explainably decreasing sequence, unbroken by transitions from the original data to synthetics, to augmentation, and back. It is shown that the synthetic dataset is able to augment the full volume of the original with a significant increase in accuracy. This indicates the absence of overlap between the datasets, as it should be as a result of using the diffusion model, and also that the record result of 0.948 can be improved by further augmentation. The maximum result obtained in five model training runs is 0.949 (averaged 0.946).

The result of 0.946 is better, including the results achieved by 3D segmentation models (lines 8–10 and 14).

5. Conclusions

After the discussion and owing to it, we can state that the technology of automatic multi-class labeling of brain electron microscopy (EM) objects based on the generative diffusion model has been built and can be used together with the open dataset EPFL6. The technology was built for the tasks of semantic segmentation of brain cell organelles because the open multi-class original datasets and software for their synthetic dataset generation are practically absent. Meanwhile the volume of EM data awaiting the multi-class and complete representation of brain cell organelles remains large. This research showed the following:

1. The quality of multi-class dataset synthesis by the diffusion model, which was trained on the original dataset (EPFL6), can be measured as the accuracy of the synthesized labeling and by the accuracy of class segmentation on the test part of the original dataset, which is achieved by the U-Net-like segmentation model trained on multi-class synthetics.
2. The quality (accuracy) of the labeling of the diffusion synthetic multi-class dataset generated via the technology corresponds to the accuracy of the original dataset (EPFL6).
3. The synthetic dataset does not replicate the original dataset but closely resembles it. Therefore, the synthetics it suitable for original dataset augmentation or even for use instead of the original data.
4. The augmentation of the dataset with adequate geometric synthetics is able to solve the problem of underrepresented classes.
5. The naturalization of geometric synthetics by the diffusion model is able to increase the accuracy of synthetic labeling and multi-class segmentation, which is trained on the synthetic dataset.
6. The size of the synthetic dataset in tiles (in this case of size 256 × 256) is practically unlimited; the number of classes is limited by the amount of memory and the reasonableness of other necessary computational resources.

The article contains an example of the augmentation of 165 layers of the original EPFL dataset in Lucchi++ markup (Table 5) with 84 layers of diffusion synthetics. The segmentation accuracy (average accuracy over 20 implementations) owing to synthetics increased from 0.934 to 0.946, and the maximum to 0.949, which corresponds to and exceeds the record accuracy of Dice = 0.948 achieved using 3D evaluation in Hive-net [43].

The listed properties of the technology are among its advantages. But it also has one disadvantage: synthetic images of size 256 × 256, which are elements of the synthetic dataset, were generated as independent and, therefore, cannot be stitched into an ordinary original layer. However, this fact does not interfere with training in any way and testing should be performed only on the original dataset.

We plan to continue working towards solving the problem of underrepresented classes by automating the construction of the minimal structural basis of brain EM images for its subsequent naturalization by the diffusion model to the level of EM images.

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

Funding: This work was funded by the Analytical Center for the Government of the Russian Federation grant No. 70-2023-001320 dated 27 December 2023.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available at https://github.com/GraphLabEMproj/unet, accessed on 2 December 2024.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

EM	Electron microscopy
DSC	Dice–Sorencen coefficient

References

1. Chen, M.; Mei, S.; Fan, J.; Wang, M. An Overview of Diffusion Models: Applications, Guided Generation, Statistical Rates and Optimization. *arXiv* **2024**, arXiv:2404.07771.
2. Bommasani, R.; Hudson, D.A.; Adeli, E.; Altman, R.; Arora, S.; von Arx, S.; Bernstein, M.S.; Bohg, J.; Bosselut, A.; Brunskill, E.; et al. On the Opportunities and Risks of Foundation Models. *arXiv* **2022**, arXiv:2108.07258.
3. Deerinck, T.; Bushong, E.; Lev-Ram, V.; Shu, X.; Tsien, R.; Ellisman, M. Enhancing Serial Block-Face Scanning Electron Microscopy to Enable High Resolution 3-D Nanohistology of Cells and Tissues. *Microsc. Microanal.* **2010**, *16*, 1138–1139. [CrossRef]
4. Ciresan, D.C.; Gambardella, L.M.; Giusti, A.; Schmidhuber, J. Deep neural networks segment neuronal membranes in electron microscopy images. In Proceedings of the NIPS, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 2852–2860.
5. Lucchi, A.; Smith, K.; Achanta, R.; Knott, G.; Fua, P. Supervoxel-Based Segmentation of Mitochondria in EM Image Stacks With Learned Shape Features. *IEEE Trans. Med Imaging* **2012**, *31*, 474–486. [CrossRef] [PubMed]
6. Helmstaedter, M.; Mitra, P.P. Computational methods and challenges for large-scale circuit mapping. *Curr. Opin. Neurobiol.* **2012**, *22*, 162–169. [CrossRef]
7. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. *arXiv* **2015**, arXiv:1505.04597.
8. Drozdzal, M.; Vorontsov, E.; Chartrand, G.; Kadoury, S.; Pal, C. The Importance of Skip Connections in Biomedical Image Segmentation. *arXiv* **2016**, arXiv:1608.04117.
9. Fakhry, A.E.; Zeng, T.; Ji, S. Residual Deconvolutional Networks for Brain Electron Microscopy Image Segmentation. *IEEE Trans. Med Imaging* **2017**, *36*, 447–456.
10. Xiao, C.; Liu, J.; Chen, X.; Han, H.; Shu, C.; Xie, Q. Deep contextual residual network for electron microscopy image segmentation in connectomics. In Proceedings of the 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018), Washington, DC, USA, 4–7 April 2018; pp. 378–381. [CrossRef]
11. Özgün Çiçek.; Abdulkadir, A.; Lienkamp, S.S.; Brox, T.; Ronneberger, O. 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. *arXiv* **2016**, arXiv:1606.06650.
12. Milletari, F.; Navab, N.; Ahmadi, S.A. V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation. In Proceedings of the 2016 Fourth International Conference on 3D Vision (3DV), Stanford, CA, USA, 25–28 October 2016; pp. 565–571.
13. Kamnitsas, K.; Ledig, C.; Newcombe, V.; Simpson, J.P.; Kane, A.D.; Menon, D.; Rueckert, D.; Glocker, B. Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. *Med. Image Anal.* **2017**, *36*, 61–78. [CrossRef]

14. Li, W.; Wang, G.; Fidon, L.; Ourselin, S.; Cardoso, M.J.; Vercauteren, T. On the Compactness, Efficiency, and Representation of 3D Convolutional Networks: Brain Parcellation as a Pretext Task. In *Proceedings of the Information Processing in Medical Imaging*; Niethammer, M., Styner, M., Aylward, S., Zhu, H., Oguz, I., Yap, P.T., Shen, D., Eds.; Springer: Cham, Switzerland, 2017; pp. 348–360.
15. Zhang, Z.; Wu, C.; Coleman, S.; Kerr, D. DENSE-INception U-net for medical image segmentation. *Comput. Methods Programs Biomed.* **2020**, *192*, 105395. [CrossRef] [PubMed]
16. Mubashar, M.; Ali, H.; Grönlund, C.; Azmat, S. R2U++: A multiscale recurrent residual U-Net with dense skip connections for medical image segmentation. *Neural Comput. Appl.* **2022**, *34*, 17723–17739. [CrossRef]
17. Chapelle, O.; Weston, J.; Bottou, L.; Vapnik, V. Vicinal Risk Minimization. In *Proceedings of the Advances in Neural Information Processing Systems*; Leen, T., Dietterich, T., Tresp, V., Eds.; MIT Press: Cambridge, MA, USA, 2000; Volume 13.
18. Simard, P.Y.; LeCun, Y.A.; Denker, J.S.; Victorri, B., Transformation Invariance in Pattern Recognition—Tangent Distance and Tangent Propagation. In *Neural Networks: Tricks of the Trade*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 235–269. [CrossRef]
19. Gong, X.; Chen, S.; Zhang, B.; Doermann, D. Style Consistent Image Generation for Nuclei Instance Segmentation. In Proceedings of the 2021 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 3–8 January 2021; pp. 3993–4002. [CrossRef]
20. Hou, L.; Agarwal, A.; Samaras, D.; Kurc, T.M.; Gupta, R.R.; Saltz, J.H. Robust Histopathology Image Analysis: To Label or to Synthesize? In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 8525–8534. [CrossRef]
21. Lin, Y.; Wang, Z.; Cheng, K.T.; Chen, H. InsMix: Towards Realistic Generative Data Augmentation for Nuclei Instance Segmentation. *arXiv* **2022**, arXiv:2206.15134. [CrossRef]
22. Wang, H.; Xian, M.; Vakanski, A.; Shareef, B. SIAN: Style-Guided Instance-Adaptive Normalization for Multi-Organ Histopathology Image Synthesis. *arXiv* **2022**, arXiv:2209.02412. [CrossRef]
23. Goodfellow, I.J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Networks. *arXiv* **2014**, arXiv:1406.2661. [CrossRef]
24. Dhariwal, P.; Nichol, A. Diffusion Models Beat GANs on Image Synthesis. *arXiv* **2021**, arXiv:2105.05233. [CrossRef]
25. Shijie, J.; Ping, W.; Peiyi, J.; Siping, H. Research on data augmentation for image classification based on convolution neural networks. In Proceedings of the 2017 Chinese Automation Congress (CAC), Jinan, China, 20–22 October 2017; pp. 4165–4170. [CrossRef]
26. Kingma, D.P.; Welling, M. Auto-Encoding Variational Bayes. *arXiv* **2013**, arXiv:1312.6114. [CrossRef]
27. Song, Y.; Sohl-Dickstein, J.; Kingma, D.P.; Kumar, A.; Ermon, S.; Poole, B. Score-Based Generative Modeling through Stochastic Differential Equations. *arXiv* **2020**, arXiv:2011.13456. [CrossRef]
28. Ho, J.; Jain, A.; Abbeel, P. Denoising Diffusion Probabilistic Models. *arXiv* **2020**, arXiv:2006.11239. [CrossRef]
29. Nichol, A.; Dhariwal, P.; Ramesh, A.; Shyam, P.; Mishkin, P.; McGrew, B.; Sutskever, I.; Chen, M. GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models. *arXiv* **2021**, arXiv:2112.10741. [CrossRef]
30. Wolleb, J.; Bieder, F.; Sandkühler, R.; Cattin, P.C. Diffusion Models for Medical Anomaly Detection. *arXiv* **2022**, arXiv:2203.04306. [CrossRef]
31. Trabucco, B.; Doherty, K.; Gurinas, M.; Salakhutdinov, R. Effective Data Augmentation With Diffusion Models. *arXiv* **2023**, arXiv:2302.07944. [CrossRef]
32. Yu, X.; Li, G.; Lou, W.; Liu, S.; Wan, X.; Chen, Y.; Li, H. Diffusion-based Data Augmentation for Nuclei Image Segmentation. *arXiv* **2024**, arXiv:2310.14197.
33. Kazerouni, A.; Aghdam, E.K.; Heidari, M.; Azad, R.; Fayyaz, M.; Hacihaliloglu, I.; Merhof, D. Diffusion models in medical imaging: A comprehensive survey. *Med Image Anal.* **2023**, *88*, 102846. [CrossRef] [PubMed]
34. Getmanskaya, A.; Sokolov, N.; Turlapov, V. Multiclass U-Net Segmentation of Brain Electron Microscopy Data Using Original and Semi-Synthetic Training Datasets. *Program Comput. Soft* **2022**, *48*, 164–171. [CrossRef]
35. Sokolov, N.; Vasiliev, E.; Getmanskaya, A. Generation and study of the synthetic brain electron microscopy dataset for segmentation purpose. *Comput. Opt.* **2023**, *47*, 778–787. [CrossRef]
36. Sohl-Dickstein, J.; Weiss, E.A.; Maheswaranathan, N.; Ganguli, S. Deep Unsupervised Learning using Nonequilibrium Thermodynamics. *arXiv* **2015**, arXiv:1503.03585. [CrossRef]
37. Lucchi, A.; Li, Y.; Fua, P. Learning for Structured Prediction Using Approximate Subgradient Descent with Working Sets. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 1987–1994. [CrossRef]

38. Arganda-Carreras, I.; Turaga, S.C.; Berger, D.R.; Cireşan, D.; Giusti, A.; Gambardella, L.M.; Schmidhuber, J.; Laptev, D.; Dwivedi, S.; Buhmann, J.M.; et al. Crowdsourcing the creation of image segmentation algorithms for connectomics. *Front. Neuroanat.* **2015**, *9*, 152591. [CrossRef]
39. Kasthuri, N.; Hayworth, K.; Berger, D.R.; Schalek, R.; Conchello, J.; Knowles-Barley, S.; Lee, D.; Vázquez-Reina, A.; Kaynig, V.; Jones, T.; et al. Saturated Reconstruction of a Volume of Neocortex. *Cell* **2015**, *162*, 648–661. [CrossRef]
40. Žerovnik Mekuč, M.; Bohak, C.; Hudoklin, S.; Kim, B.H.; Romih, R.; Kim, M.Y.; Marolt, M. Automatic segmentation of mitochondria and endolysosomes in volumetric electron microscopy data. *Comput. Biol. Med.* **2020**, *119*, 103693. [CrossRef]
41. Casser, V.; Kang, K.; Pfister, H.; Haehn, D. Fast Mitochondria Detection for Connectomics. *arXiv* **2020**, arXiv:1812.06024.
42. ITMM. 6-Class Labels for EPFL EM Dataset. 2023. Available online: https://github.com/GraphLabEMproj/unet/tree/master/data (accessed on 2 December 2024).
43. Yuan, Z.; Ma, X.; Yi, J.; Luo, Z.; Peng, J. HIVE-Net: Centerline-Aware HIerarchical View-Ensemble Convolutional Network for Mitochondria Segmentation in EM Images. *Comput. Methods Programs Biomed.* **2021**, *200*, 105925.
44. Cheng, H.C.; Varshney, A. Volume segmentation using convolutional neural networks with limited training data. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 590–594. [CrossRef]
45. Peng, J.; Yuan, Z. Mitochondria Segmentation From EM Images via Hierarchical Structured Contextual Forest. *IEEE J. Biomed. Health Inform.* **2020**, *24*, 2251–2259. [CrossRef]
46. Cetina, K.; Buenaposada, J.M.; Baumela, L. Multi-class segmentation of neuronal structures in electron microscopy images. *BMC Bioinform.* **2018**, *19*, 298.

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.

technologies

Article

Real-Time Deployment of Ultrasound Image Interpretation AI Models for Emergency Medicine Triage Using a Swine Model

Sofia I. Hernandez Torres [†], Lawrence Holland [†], Theodore Winter, Ryan Ortiz, Krysta-Lynn Amezcua, Austin Ruiz, Catherine R. Thorpe and Eric J. Snider *

Organ Support and Automation Technologies Group, U.S. Army Institute of Surgical Research, Joint Base San Antonio, Fort Sam Houston, San Antonio, TX 78234, USA

* Correspondence: eric.j.snider3.civ@health.mil; Tel.: +1-210-539-8721

[†] These authors contributed equally to this work.

Abstract: Ultrasound imaging is commonly used for medical triage in both civilian and military emergency medicine sectors. One specific application is the eFAST, or the extended focused assessment with sonography in trauma exam, where pneumothorax, hemothorax, or abdominal hemorrhage injuries are identified. However, the diagnostic accuracy of an eFAST exam depends on obtaining proper scans and making quick interpretation decisions to evacuate casualties or administer necessary interventions. To improve ultrasound interpretation, we developed AI models to identify key anatomical structures at eFAST scan sites, simplifying image acquisition by assisting with proper probe placement. These models plus image interpretation diagnostic models were paired with two real-time eFAST implementations. The first implementation was a manual AI-driven ultrasound eFAST tool that used guidance models to select correct frames prior to making any diagnostic predictions. The second implementation was a robotic imaging platform capable of providing semi-autonomous image acquisition combined with diagnostic image interpretation. We highlight the use of both real-time approaches in a swine injury model and compare their performance of this emergency medicine application. In conclusion, AI can be deployed in real time to provide rapid triage decisions, lowering the skill threshold for ultrasound imaging at or near the point of injury.

Keywords: artificial intelligence; emergency medicine; image interpretation; robotics; triage; ultrasound imaging

1. Introduction

Medical imaging has remained a central function for injury assessment in healthcare for decades and has become more widespread in recent years due to technology improvements [1], especially in emergency situations where triaging and the quick treatment of injuries can determine whether a life is saved or lost [2]. Ultrasound (US), in particular, is effective in modern military and emergency medicine [3]. In addition to being relatively low in cost and portable, it is useful for its ability in detecting free fluid, which is synonymous with injury in the thoracic and abdominal cavities. This is effective because assessments can be made while patients are being transported, or when they need to be examined swiftly in the field [4]. For triage, having tools outside of a definitive healthcare setting is crucial for administering different imaging procedures. This helps mitigate the devastating effect of emergency situations, which are prone to high fatality rates when there is no immediate access to definitive hospital care [5].

https://doi.org/10.3390/technologies13010029

One common and useful triage procedure is the extended focused assessment with sonography for trauma, or eFAST exam [6]. The eFAST exam is a point-of-care method of examination that non-invasively evaluates the thoracic and abdominal cavities for the presence of free fluid or air in order to identify abdominal hemorrhage (AH), hemothorax (HTX), and pneumothorax (PTX). This can allow for identifying the type of care needed to treat a trauma patient and the urgency needed for the intervention. However, there are several considerations that come with administering an eFAST exam. First, being able to properly use US equipment can be technically challenging for less-experienced personnel, as proper angles and positionings of the US transducer are required to identify the regions where fluid and air are most often pooled in the abdominal and thoracic cavities. Second, correctly identifying injury at the scan site is technically challenging, requiring an interpretation of anatomical landmarks and the identification of variable volumes of free fluid or air. Unfortunately, there is a projected shortage of medical providers that can properly perform and interpret injury from a US exam, which will be especially detrimental in mass casualty situations [7]. Therefore, despite the importance of an eFAST exam during triage and its ability to reduce the amount of time it takes for patients to be delivered to definitive care, there are assumptions and drawbacks to consider for effective eFAST exam utilization.

The development of artificial intelligence (AI) has accelerated in several fields of technology, including the healthcare industry. In the medical imaging field, AI has been proven to improve efforts in patient care and medical diagnoses of disease and abnormalities [8–10]. AI not only reduces the time it takes to diagnose these problems, but also gives supplemental insight to medical providers by finding and interpreting abnormalities that could have otherwise been missed by a human eye unfamiliar with discerning nuanced features [8]. In addition, technological advancements have allowed for improved care administration for trauma patients on the battlefield [11]. One example is the use of internet-based video communication to receive real-time advice from medical professionals to properly treat or address casualty patients. Closed-loop systems for fluid or drug administration utilize fully automated medical administration approaches to stabilize patients that are being transported to more definitive care [12–14]. Robotics have been pursued as well to improve the treatment administration of surgical interventions through telerobotic platforms [15].

Considering the history of AI in healthcare and medical imaging, we propose that the diagnostic capabilities used to detect and treat illnesses can be applied to the injury interpretation function in the eFAST exam. Previous studies have developed AI models with limited datasets for a FAST exam only, excluding thoracic image interpretation [16,17], while others have utilized fully supervised feature creation approaches for detecting pleural effusion in eFAST scan sites, a much more cumbersome automation approach [17]. There are also studies that summarize the progress in ultrasound applications with AI, including utilizing convolutional neural networks for diagnostic applications and a robotic arm for assistance in casualty classification in pre-hospital settings [18]. We previously explored the use of deep learning AI through the exploration and evaluation of a wide range of trained binary classification diagnostic models to detect injury at eFAST scan sites in swine subjects [19]. Having diagnostic models to interpret medical images only addresses part of the challenge with performing eFAST exams. The other issue is adequate medical image acquisition for discernable image capture so that AI models can interpret the presence of injury. For this, AI and robotics can be applied to the eFAST exam, utilizing computer vision AI to guide a robotic platform to the relevant scan points of the eFAST exam. We have previously shown that a robotic imaging platform can traverse a wide range of eFAST scan points, and assessed different US probe holder designs for this application [20].

In this study, we explored the integrations and capabilities of automated eFAST image acquisition and interpretation that our trained deep learning models allow for in a real-time setting, such as model inferencing in live and euthanized swine. Two image acquisition methods were evaluated. First, we evaluated a handheld AI-driven US application that guides the user to the correct scan site using AI guidance models and then runs AI diagnostic models. Second, we evaluated a robotic imaging platform equipped with computer vision AI to detect scan sites, as well as AI guidance and diagnostics to confirm proper image capture and make scan site diagnostic predictions. Each of these were tested in real time in live or euthanized swine to highlight the potential for AI automated eFAST examination. If eFAST US procedures can be fully automated, this life-saving triage exam can be more widely deployed in pre-hospital and emergency medicine situations for both civilian and military medicine.

2. Materials and Methods

2.1. Animal Procedures and Manual Ultrasound Image Capture

US scans were captured at eFAST scan sites using a swine model from three approved animal research protocols. Research was conducted in compliance with the Animal Welfare Act, the implementing Animal Welfare regulations, and the principles of the Guide for the Care and Use for Laboratory Animals. The Institutional Animal Care and Use Committee at the United States Army Institute of Surgical Research approved all research conducted in this study. The facility where this research was conducted is fully accredited by the AAALAC International. Live animal subjects were maintained under a surgical plane of anesthesia and analgesia throughout the studies. For all studies, images were captured immediately after instrumentation procedures and before laparotomy to remove the spleen (Scan #1, Figure 1). Each animal study was focused on different shock-related injuries, and splenectomies were performed to minimize the variability due to splenic contraction and autotransfusion [21,22]. Since the spleen was removed in all protocols, no US scans were captured in the left upper quadrant, or LUQ, scan site. After the subjects were euthanized, two imaging rounds took place: before (Scan #2) and after inducing abdominal hemorrhage (AH), pneumothorax (PTX), and hemothorax (HTX) injuries at the respective scan sites (Scan #3, Figure 1).

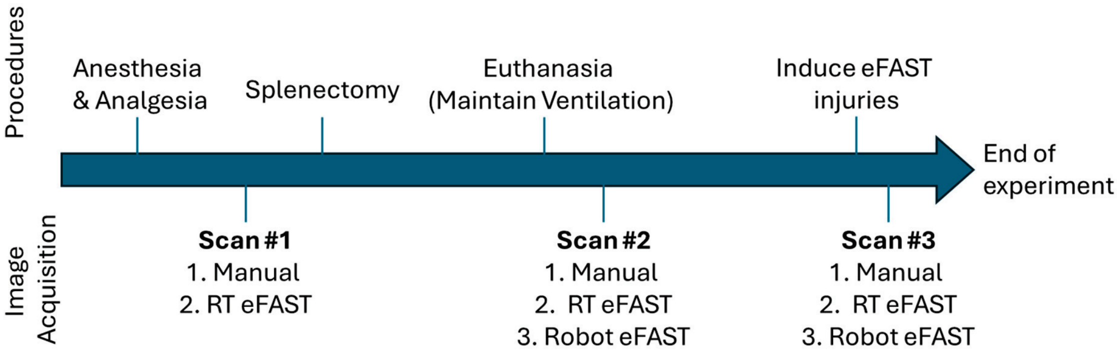


Figure 1. Overview of animal study procedures and image capture timepoints. Ultrasound images were captured prior to splenectomy in live swine, as well as at two time points in euthanized swine (before and after eFAST injury induction). Each scan landmark in the diagram lists how US images were captured. The three approaches were manual US image capture, image capture using the RT eFAST handheld application, and image capture using the robotic imaging platform.

For manual image capture, images in the thoracic region were captured using a linear array probe (L15, Sonosite, Fujifilm, Bothwell, WA, USA), and at the abdominal scan sites a curvilinear array probe was used (C5, Sonosite, Fujifilm, Bothwell, WA, USA), using a Sonosite PX (Fujifilm, Bothwell, WA, USA) US System. Images were captured for two different AI training applications: guidance and diagnostic AI models. For the diagnostic training dataset, thoracic US scans were captured as 10 s B-mode (brightness mode) clips or as 5 M-mode (motion mode) images, captured at multiple intercostal spaces. For guidance, 10 s B-mode clips were captured as a single swipe along all intercostal spaces of the thorax bilaterally. The abdominal scans were obtained at two locations: the right upper quadrant (RUQ), focusing on the kidney–liver interface, and the pelvic region (BLD), focusing on the areas around the bladder. For guidance image capture, 10 s region scans were captured in two motions: along the sagittal plane and along the medial plane. For diagnostic image capture, additional 10 s scans were captured while rocking the probe with the region of interest in view. All of these images were captured at the three experimental timepoints previously stated, and the injuries were created following a previously described methodology [19].

2.2. Data Processing

Ultrasound data from 36 pigs were exported from the US machine and sorted by experimental phase, subject ID, and scan point for both major scan types: guidance (scans along anatomical planes) and diagnostic (scans focused on organs, fluid accumulation sites), as diagrammed in Figure 2. All ultrasound videos were split into frames, and individual images were cropped and resized to 512 × 512 pixels using the Image Processing Toolbox extension from MATLAB version R2023b (MathWorks, Natick, MA, USA). Images were cropped to remove words and other artifacts on the US scans that the AI model may have focused on during training. The US scans were reshaped to a 512 × 512 pixel size to create a symmetrical image geometry at a high resolution to detect small injury features. We have developed successful US AI models for similar applications using this image input size [23]. For guidance frames, datastore file types were created containing random samples of the data, with major anatomical features labeled with bounding boxes around them: ribs for thoracic scans, the kidney for RUQ, and the bladder for BLD. Once the labels were generated, images in which the feature was not obviously visible were removed from the dataset. The bounding box labels were exported from MATLAB as four coordinates: x, y of the top left corner, and x-length, y-length of the bounding box.

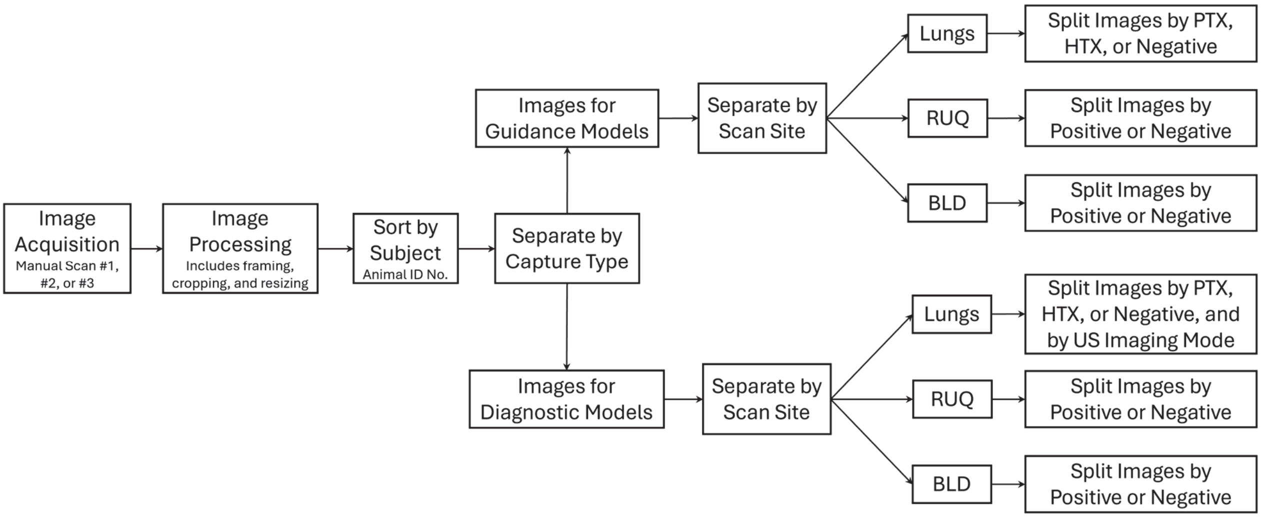
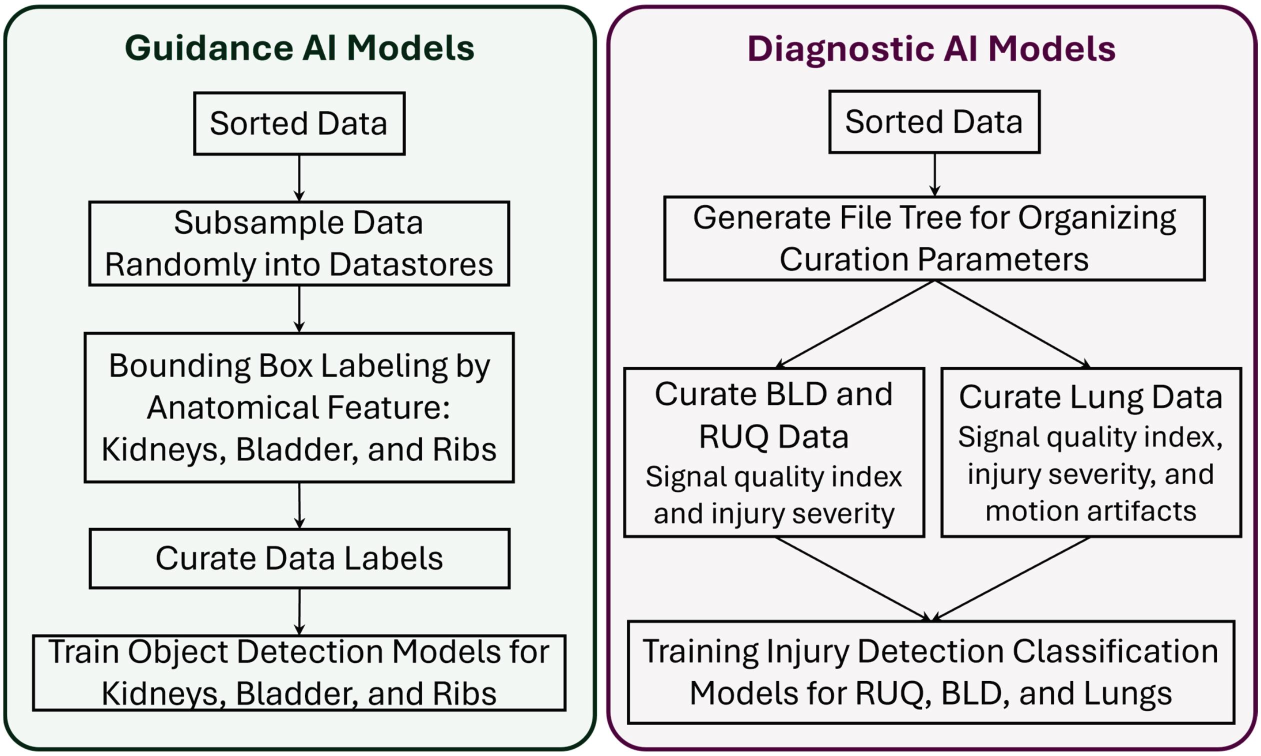


Image Acquisition
Manual Scan #1, #2, or #3
Image Processing
Includes framing, cropping, and resizing
Sort by Subject
Animal ID No.
Separate by Capture Type
Images for Guidance Models
Separate by Scan Site
Lungs
Split Images by PTX, HTX, or Negative
RUQ
Split Images by Positive or Negative
BLD
Split Images by Positive or Negative
Images for Diagnostic Models
Separate by Scan Site
Lungs
Split Images by PTX, HTX, or Negative, and by US Imaging Mode
RUQ
Split Images by Positive or Negative
BLD
Split Images by Positive or Negative

Figure 2. Overview of ultrasound image dataset structure and processing for images captured in swine for eFAST AI model training.

For diagnostic scans, images captured during the pre-splenectomy and pre-injury phases were preliminarily classified as negative for injury and the post-injury captures as positive for injury. Then, a file tree of all items was generated, which allowed the review of every entry. As part of data curation prior to training the AI models, all US scans were reviewed for the presence of injury and assessed for overall image quality score, injury severity (none, slight, positive), and the presence of motion artifacts (only applied to thoracic scans). Image quality evaluated whether the US scans could be used to diagnose an injury. A score of 1 corresponded to a poor image quality, with most frames captured at an incorrect location; a score of 5 corresponded to a high image quality captured at a proper eFAST scan point, where diagnostic status could be properly assessed. This was performed by two scorers who agreed on image quality scores for the initial frames to help standardize scoring and conferred to finalize data curation if disagreement occurred for any image. When selecting data for training the AI models, those with a signal quality score below 3 and thoracic scans with large motion artifacts were not included in the training datasets. Scans labeled as "slight" injury were maintained in the dataset as positive for injury. An overview of the AI model types used in this effort is shown in Figure 3.


Guidance AI Models

Sorted Data

Subsample Data Randomly into Datastores

Bounding Box Labeling by Anatomical Feature: Kidneys, Bladder, and Ribs

Curate Data Labels

Train Object Detection Models for Kidneys, Bladder, and Ribs

Diagnostic AI Models

Sorted Data

Generate File Tree for Organizing Curation Parameters

Curate BLD and RUQ Data
Signal quality index and injury severity

Curate Lung Data
Signal quality index, injury severity, and motion artifacts

Training Injury Detection Classification Models for RUQ, BLD, and Lungs

Figure 3. Summary of data flow for eFAST AI model training. For guidance models (diagram on the left), data were subsampled, labeled, and then curated. For diagnostic models (right diagram), the sorted data were curated and then used for classification model training.

2.3. Guidance AI Models

Once the data were labeled, the guidance AI models were trained using the YOLOv8 [24] object detection architecture, with separate models tailored specifically for the detection of the kidneys (9449 labelled US images), bladder (7039 labelled US images), or ribs (44,736 labelled US images). The training process utilized the YOLOv8-S pre-trained model weights, default training parameters, and 100 epochs to provide ample opportunity for the models to learn and refine their predictions. To ensure robust model validation, a distinct dataset from subjects not used in training was reserved for the holdout testing of model performance. YOLOv8 was selected as the model architecture due to a variety of advantages when compared with other state-of-the-art object detection models. Primarily, this effort focused on the real-time application of object detection models with an eFAST-focused purpose. This meant that speed of prediction time was of high importance, even at the expense of slightly reduced accuracy. This narrowed the scope of possible models to be used to 'single-stage' architectures, where the single-stage model undertakes a single pass through of the image through the layers to determine the object location and class. Models like Faster R-CNN, which can be more accurate, have a slower prediction time due to the image being processed into proposed regions of interest before being classified for objects. Moreover, when looking at single-stage models, YOLOv8 was amongst the fastest in frames per second, even beating out the single-shot detector (SSD) model and having only a slightly worse detection accuracy [25,26]. Ease of use was also a driving factor for the use of YOLOv8 in this environment. The Python library ultralytics [27] provides an API to allow for the seamless integration of YOLO models into existing software.

For each guidance model trained, predictions were compared against the ground truth labels for the respective image, and Intersection-Over-Union (IOU) scores were calculated for each image. IOU is a common metric for evaluating object detection models, calculated by dividing the area in which the predicted mask and ground truth mask overlap (intersection) by the total area covered by both masks (union). An IOU threshold of 0.5 is widely accepted in object detection applications as a standard for evaluating model performance, with scores at or above this threshold being acceptable [28]. For kidney and bladder predictions, one object was expected for each frame, whereas for the thoracic image, two objects were expected. Regardless, for all predictions, the IOU score was calculated as an average across the entire image.

2.4. Diagnostic AI Models

For the development of diagnostic AI models, different approaches were used for the thoracic and abdominal regions. Each approach utilized the same YOLOv8 model architecture, except configured for classification for this use case. Diagnosis of injury in the abdominal region is regularly made from B-mode scans; as such, AI models were only trained using this type of imaging. In the thoracic region, due to the nature of lung sliding and how injuries present in ultrasound, M-mode images are a common means of distinguishing between injured and non-injured states. Diagnostic models trained for the thoracic region used two approaches: predictions from US-system-generated M-mode scans, or custom-generated M-mode images from a static hold in B-mode imaging mode. The latter approach is described below, followed by overall AI training procedures for the other scan points.

2.4.1. Creating Custom Motion Mode Images from US Scans

For the development of diagnostic models focused on the thoracic region, we first generated M-mode images from the original B-mode US scans. This approach used a sequence of consecutive frames to create custom M-mode images. Each frame was processed through the guidance model for rib detection and, based on the predicted rib locations, the central point between the ribs was calculated. At this central point, a vertical slice was extracted from each frame (Figure 4). These slices were then concatenated to generate an image that closely resembled a genuine M-mode image. To ensure that a generated M-mode image was indicative of its diagnosis, the rib detection guidance model was used to filter out images without only two ribs visible. If a frame did not have exactly two ribs detected, that set of subsequent frames was not used for the M-mode creation process.

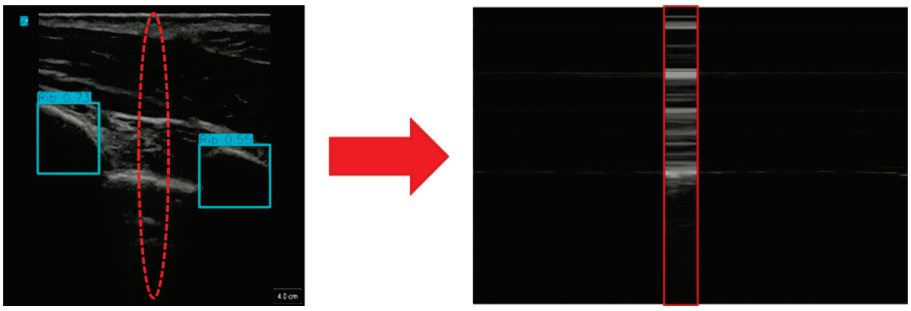


Figure 4. Overview of how M-mode images were generated from B-mode frames using rib guidance AI models. Shown first is a traditional B-mode ultrasound frame from which the guidance AI determined the location of the ribs (blue bounding boxes). A 3-pixel-wide region at the midpoint between the bound boxes (red dotted region) is selected across each frame to create a custom M-mode image (shown on the right).

An optimization process was conducted to determine the ideal set of parameters used to generate the images. These parameters included the number of frames per image, the width of the slice taken from each frame, the window stride between images, and the number of slices taken from each frame. The first two optimization parameters were concerned with the makeup of the generated M-mode images. For frames per image, we tested 30, 90, and 150 frames per image. Images were captured from a video running at 30 frames per second, so these represent 1, 3, and 5 s capture windows. Three slice widths were also tested, these being 1-, 3-, and 5-pixel widths.

The remaining optimization parameters were focused on the generation of our training image dataset. The window stride parameter refers to the number of frames the model moves forward between images. For example, if using 30 images per generated M-mode and a stride of 15, one generated image will use frames 1–30, and the next will use images 15–45. The stride options used during the optimization were either 6 or 15 images. The final optimization parameter was the number of slices taken from each image, with either 1 or 3 slices being taken from each image. These parameters would affect both the number and makeup of images present in the training dataset.

These options produced 36 unique combinations of training parameters to be validated in the grid search using a YOLOv8 classification model trained for 100 epochs. After optimization, the resulting best parameters were as follows: 150-frame window size, 5-pixel slice width, 15-frame stride, and 1 slice taken per frame.

2.4.2. Training AI Models for Injury Identification

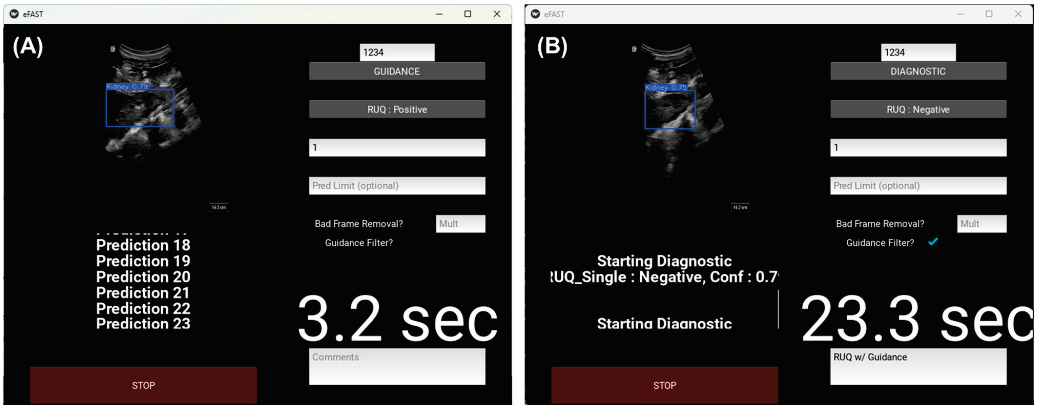
The diagnostic models were trained for injury detection at each eFAST scan site. For the abdomen, the AI models to identify AH injury were trained independently for the RUQ and BLD scan sites. For the thorax, two separate models were trained to predict if there was HTX, PTX, or no injury present, using either US-system-generated M-mode images or the custom generated ones as the input data. The dataset was split into 3 groups of 13 swine each to be able to perform the leave-one-subject-out (LOSO) cross-validation methodology. Each unique LOSO group was randomly generated from three research protocols and designated as a training, validation, or test set. We previously compared several AI model architectures to develop AI models for each eFAST scan site [19]. With the larger image dataset used in this study, these models needed to be retrained, and, for simplicity, they utilized the same YOLOv8 architecture for image classification that was used for the AI guidance model development. We applied the default training parameters over a span of 100 epochs to allow for sufficient learning and refinement. Predictions were then tested on a holdout set of images from subject data not in the training data to test model performance. The best performing model from each scan site was then selected to be used in real-time testing.

2.5. Real-Time Validation of AI Models

Real-time (RT) image capture was performed in three swine subjects completely separate from the dataset used to develop and test the underlying AI models. Each animal underwent imaging at the experimental timepoints shown in Figure 1. Three real-time approaches were used: (i) RT eFAST application, which allowed for selection of a single scan site and capture of images while AI predictions for guidance and diagnostics occurred in RT; (ii) full handheld, manual eFAST examination, driven by AI guidance and diagnostic models; (iii) automated eFAST image capture using a robotic imaging platform equipped with computer vision, guidance, and diagnostic AI models. Each of these approaches is described in more details below.

2.5.1. Real-Time eFAST Application

To enable the RT testing of models, a dedicated graphical user interface (GUI) was developed in Python using the Kivy library and designed to run on a laptop connected to the US machine via a Magewell USB Capture HDMI Gen 2 capture card (Magewell Electronics Co., Reading, PA, USA). The RT eFAST application allows users to input various experimental parameters, including subject identifier, scan mode (guidance or diagnostic), scan site (BLD, RUQ, M-mode, or RibsAI to generate M-Mode images), injury status, and number or duration of predictions (Figure 5). Additionally, the interface provides a comment section, with all inputs saved as a text file in addition to the prediction results from each individual scan. The best performing model for each scan site and method that received the best blind test accuracy score was selected to be used in the real-time experiments. The trained model weights were packaged along with the GUI code to allow for the quick deployment of models and switching between models in real time. Users also have the option to select filtering methods that can be applied during the scan, as shown in Figure 5B; these are further described in the next section.



1234

1

Prediction 18
Prediction 19
Prediction 20
Prediction 21
Prediction 22
Prediction 23

3.2 sec

1234

1

Starting Diagnostic
RUQ_Single : Negative, Conf : 0.7

Startina Diagnostic

23.3 sec

RUQ w/ Guidance

Figure 5. Overview of the real-time eFAST application. Developed graphical user interface for (**A**) guidance AI model use and (**B**) diagnostic AI model use, with guidance filtering active. Representative screen shots shown for a RUQ scan site. The time refers to how long the application took to make predictions.

The RT eFAST application can be used for testing AI models in real time, as well as for data collection while performing the eFAST exam. The GUI allows the user to select relevant parameters for the operation and to start image capture. This in turn initializes the video stream and activates a thirty-second timer, which is displayed on the application. US imaging and RT predictions run for thirty seconds or until the specified number of predictions is reached, whichever comes first. While the scanning mode is active, the predictions and corresponding images are shown in real time, along with the prediction confidence scores. To ensure smooth operation, process threading was employed to make predictions concurrently, preventing any interruption to the RT eFAST application's functionality. The system processed one frame at a time, waiting for each prediction to finish before loading the next frame.

As part of the data collection feature, the program can save all frames captured between predictions. A results folder was generated for every scan, containing subfolders for the saved intermediate frames and one for the frames used for the predictions, a CSV file listing model predictions with confidence scores, and a TXT file with user-input comments. For guidance scans, predicted images were stored with overlaid object detection boxes.

Ultrasound Image Filtering Features

Several filtering options are available to the user while scanning: bad frame removal, guidance filtering, and the option to turn both of these on at the same time. The bad frame removal filtering option performs an analysis of each image to quantify the quality of the image based on intensity-based and texture-based features before predictions are made. To attain this functionality, a sample of 2000 images was taken from each scan site in the dataset and then analyzed using noise and pattern analysis to find some correlation between the ultrasound images labeled "bad" and quantifiable characteristics, such as average pixel intensity, the standard deviation of pixel intensity, entropy, or the signal-to-noise ratio. Images were labeled "bad" by two US operators based on the quality of the image and the ability to make a diagnostic prediction from the image. The metrics that indicated the strongest correlation to image quality were the average and standard deviation of pixel intensity, corresponding to the brightness and contrast of the images, respectively. Using this analysis, the most ideal values for brightness, contrast, and the signal-to-noise ratio were selected as the parametric floor to classify an image as a bad frame. The user also has the option to adjust the aggressiveness of bad frame removal from the GUI by entering a multiplier value to be applied to the bad frame parameters. Bad frame removal was only used for the RUQ and BLD sites, as the M-mode capture process required multiple seconds of undisturbed data capture, making bad frame removal not possible during this capture process.

In addition to bad frame removal, we developed a guidance filter as a second filtering option. For this process, streamed frames were passed through the guidance model for the designated scan site before any predictions were made. The guidance AI models evaluated each image for the identification of relevant anatomical features, such as two ribs, a bladder, or a kidney. If these features were not detected, the GUI bypassed the frame and moved on to the next available one without making a diagnostic AI prediction. For the rib models, guidance occurred at the start of the scan. Once two ribs were identified, the GUI prompted the user to hold still for M-mode capture until the scan was complete, whether it was real or generated. For the RUQ and BLD models, guidance was applied before each prediction, with the model only proceeding if the appropriate anatomical features were detected in the image. When both filters were active, images were passed through bad frame removal first, followed by guidance filtering.

2.5.2. Manual eFAST Exam with AI Model Guidance

A python script was developed to test the guidance and diagnostic AI models during a full eFAST exam, recording the time taken to complete each scan point. The script prompted the operator to follow a scan order of upper-left thorax, lower-left thorax, upper-right thorax, lower-right thorax, RUQ, and BLD. For each scan point, the user prompts, model predictions, and the times taken to complete each scan were displayed in the command terminal. At the lung scan sites, the guidance model for lungs ran until it detected two ribs, and then prompted the user to stay in that location while it made three predictions using generated M-mode images, before telling the user to move to the next scan point. For RUQ and BLD, the user had to swap to the curvilinear transducer and then the guidance model ran continuously, only making a diagnostic prediction when the kidney or bladder was detected, until it reached 30 predictions. This imaging application was run in two modes: one in which the operator viewed the ultrasound screen during the exam, and a second "blind" scan where the user was unable to see the display. The manual eFAST exam with RT AI predictions was performed at the timepoints specified in Figure 1.

2.5.3. Automated Robotic US eFAST Exam

A UR5e robotic platform (Universal Robots, Odense, Denmark) was configured for semi-autonomous eFAST examination (Figure 6). The UR5e was programmed to navigate to eFAST scan sites using computer vision and stereo vision technology. Once at the scan site, the robotic arm was programmed to capture ultrasound images using a custom-made ultrasound probe holder to position the ultrasound probe and using integrated force feedback to apply the probe to the subject. Robotic navigation and image acquisition were further assisted by ultrasound-based guidance feedback that allowed the robot to search a scan site at several positions until relevant anatomical features were in view of the image. Finally, the ultrasound images captured by the UR5e were evaluated for injury interpretation using the diagnostic AI models.

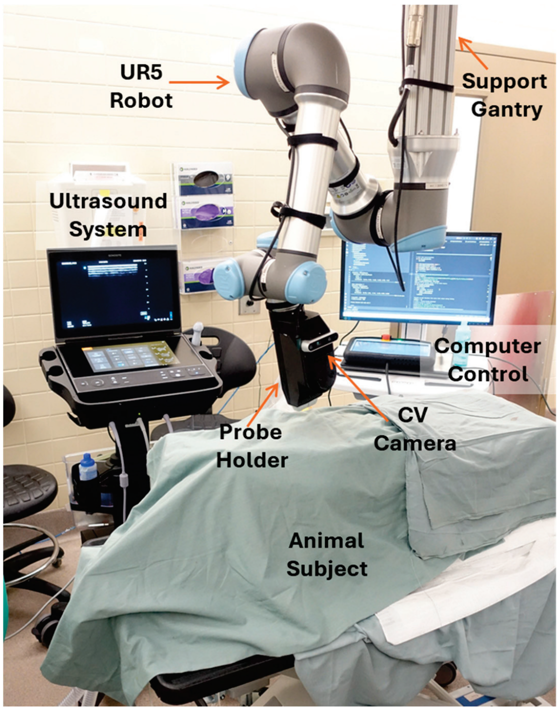


Figure 6. Overview of robotic configuration for automated eFAST in swine. Relevant features of the setup are labeled to better explain the experimental setup.

Robotic Platform Configuration

The computer vision AI model was developed to detect the location of relevant scan sites on the subject's body using external image features. Ultrasound images were used to confirm the location of the relevant anatomical features for each scan site, and a fiducial target in the form of a circular color-coded sticker was placed on the body of the subject at this location. The UR5e was programmed to travel around the body of the subject, capturing images using an Intel RealSense 435i camera (Intel, Santa Clara, CA, USA). Images were captured with and without the targets placed on the subject. eFAST scan sites were then labeled in MATLAB using the images that included targets. This process was repeated so that the image training dataset comprised images captured for two subjects. A computer vision model was then trained using YOLOv8s to accurately identify the color-coded stickers. Images of swine were also captured without stickers present to determine if the AI models could accurately identify scan sites without stickers present. Unfortunately, not enough data were captured for training models for this application and the computer vision models for detecting stickers at each scan site were used. IOU scores were calculated for model predictions during the testing performed on the three swine subjects reported in this study based on agreement between ground truth labeled sites and AI model prediction.

During testing, the UR5e was positioned over the subject at mid-torso using a hoist–lift structure (Figure 6). The UR5e was programmed to capture four images of the top, left side, and right side of the pig using an Intel RealSense camera fixed to the end of the robotic arm. For each image, the computer vision model was used to detect the location of each scan site, providing the UR5e with real-world scan site coordinates for computer-vision enabled navigation. The model returned the pixel value of the center of the color-coded targets that were detected in each image. Next, with the inherent depth reading capabilities of the Intel RealSense camera due to stereo vision technology, the real-world 3-dimensional location of the target relative to the lens of the camera was determined. The 3-dimensional location of the target was then transformed to the robot's coordinate system, allowing the robot to navigate to the scan site and apply the probe for image acquisition.

The quality of image acquisition was improved by using ultrasound image-based guidance feedback to scan a site, capturing multiple US images until an US image was acquired that could be used for proper diagnostic interpretation. For the abdominal sites, eight additional scan locations positioned in a circle equidistant apart at a 2.54 cm radial offset from the location of the original scan site were available for image capture. For the thoracic sites, the robot was programmed to scan linearly in intervals of 1.2 cm in the caudal direction before scanning another set of sites, following a line slightly offset in the same direction. This resulted in a total of 7 potential scan site positions for evaluation.

In addition to finding all the scan sites, radial positions, and linear positions on the subject, it was necessary to ensure that the probe was oriented orthogonally and applied sufficient contact force to the surface to receive a clear ultrasound image. To do so, depths were measured at the detected scan point, so that the slopes of the measured surface could be used to calculate the correct roll, pitch, and yaw coordinates that would allow the robot arm to position the probe normal to the surface at each scan site. By accounting for the local curvature of the anatomy of the subject, adequate contact was sought between the surface of the ultrasound probe and the surface of the subject at each scan position. For the abdominal scan sites, a rocking B-mode scan was performed, where upon reaching an adequate position, the robot rotated to four different angles at a 5-degree offset relative to the scan site and collected a set of ultrasound frames at each different angle to pass to the diagnostic model. The set of ultrasound frames was acquired over a period of a tenth of a second for both the guidance and diagnostic scans, yielding between 5 and 7 frames.

Robotic eFAST (RoboFAST) Exam with AI Model Guidance

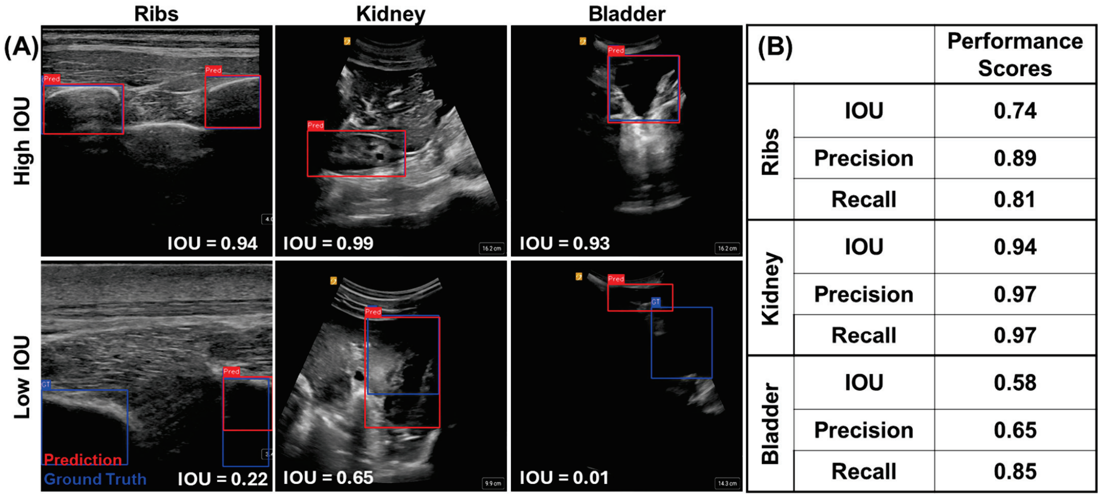
A set of three RoboFAST exams, each with a different set of criteria, were run on each of three experimental swine subjects at the two post-euthanasia timepoints (Figure 1). All trained diagnostic and guidance AI models were integrated into the RoboFAST algorithm to assess the robotic platform's capabilities and compare its performance to the manual eFAST exam performance. Upon detecting all scan sites and converting the pixel coordinates to coordinates relative to the origin of the robot, the robot started the respective experimental run.

The first run, referred to as "Radar", conducted a general eFAST exam where the robot scanned both the original scan site and additional radial and linear positions until the guidance AI model returned that the proper organ or anatomy was present, indicating that a suitable location to run the diagnostic model was found. If no such detections occurred, the robot moved on to the next site without conducting a diagnostic prediction. However, when the guidance AI returned that the relevant object was detected, the diagnostic AI provided an injury prediction result for five consecutive frames. For the second run, referred to as "No Radar", the robot performed a single image capture at the location where the colored sticker was detected. For the third experimental run, referred to as "All Radar", the robot performed image capture at each scan site and all of the corresponding additional positions, running the diagnostic AI multiple times depending on how many positions at a site contained suitable locations. The plurality of what the diagnostic model returned then determined the prediction of the RoboFAST algorithm.

3. Results

3.1. Guidance AI Performance

For each guidance model trained, model performance was evaluated against a test dataset comprising images from subjects not included in the training data. Examples of high and low IOU scores are shown for each guidance model in Figure 7A. The resulting average IOU scores varied across each model, with kidneys having the highest score at 0.94, followed by the ribs and bladder at 0.74 and 0.58, respectively (Figure 7B). The precision and recall metrics were also strong for each guidance model, apart from precision for the bladder model, which was only 0.65 (Figure 7B). A higher false-positive rate due to the pixels being identified as bladder in the model's prediction but not in the ground truth image resulted in this lower score for the bladder model. Overall, each model was trained at variable performance levels and was able to correctly identify anatomical features to aid with proper eFAST US image acquisition.



(B)		Performance Scores
Ribs	IOU	0.74
	Precision	0.89
	Recall	0.81
Kidney	IOU	0.94
	Precision	0.97
	Recall	0.97
Bladder	IOU	0.58
	Precision	0.65
	Recall	0.85

Figure 7. Guidance AI performance for each anatomical location. (**A**) Representative images are shown for high and low IOU scores for rib, kidney, and bladder predictions. (**B**) Testing performance scores for each anatomical guidance model for IOU, precision, and recall metrics.

3.2. Diagnostic AI Performance

For thoracic diagnostic models, models were trained for both M-mode and generated M-mode diagnostic models (as described in Section 2.4.1). The M-mode diagnostic model predictions had a higher accuracy compared to the generated M-mode diagnostic models, at 0.94 vs. 0.78 accuracy, respectively. From the confusion matrix analysis, the generative M-mode models had a higher accuracy for the ground truth PTX predictions but identified 27% of the ground truth HTX images and 22% of the negative images as PTX (Figure 8A,B). Conversely, M-mode models had a slight bias toward HTX predictions, with 7.6% and 6.5% of the PTX and negative ground truth images being incorrectly identified as HTX-positive. We further developed RUQ and BLD diagnostic prediction models, which were binary in nature: positive or negative for abdominal hemorrhage. The RUQ models reached 0.77 accuracy but had a lower specificity metric of 0.68 compared to a higher recall of 0.80, hinting at slight bias toward positive predictions across the testing dataset (Figure 8C). As for the BLD models, overall performance remained lower at 0.59 accuracy, with a much larger bias toward negative predictions in the testing dataset, as indicated by the confusion matrix and 0.49 recall metric (Figure 8D).

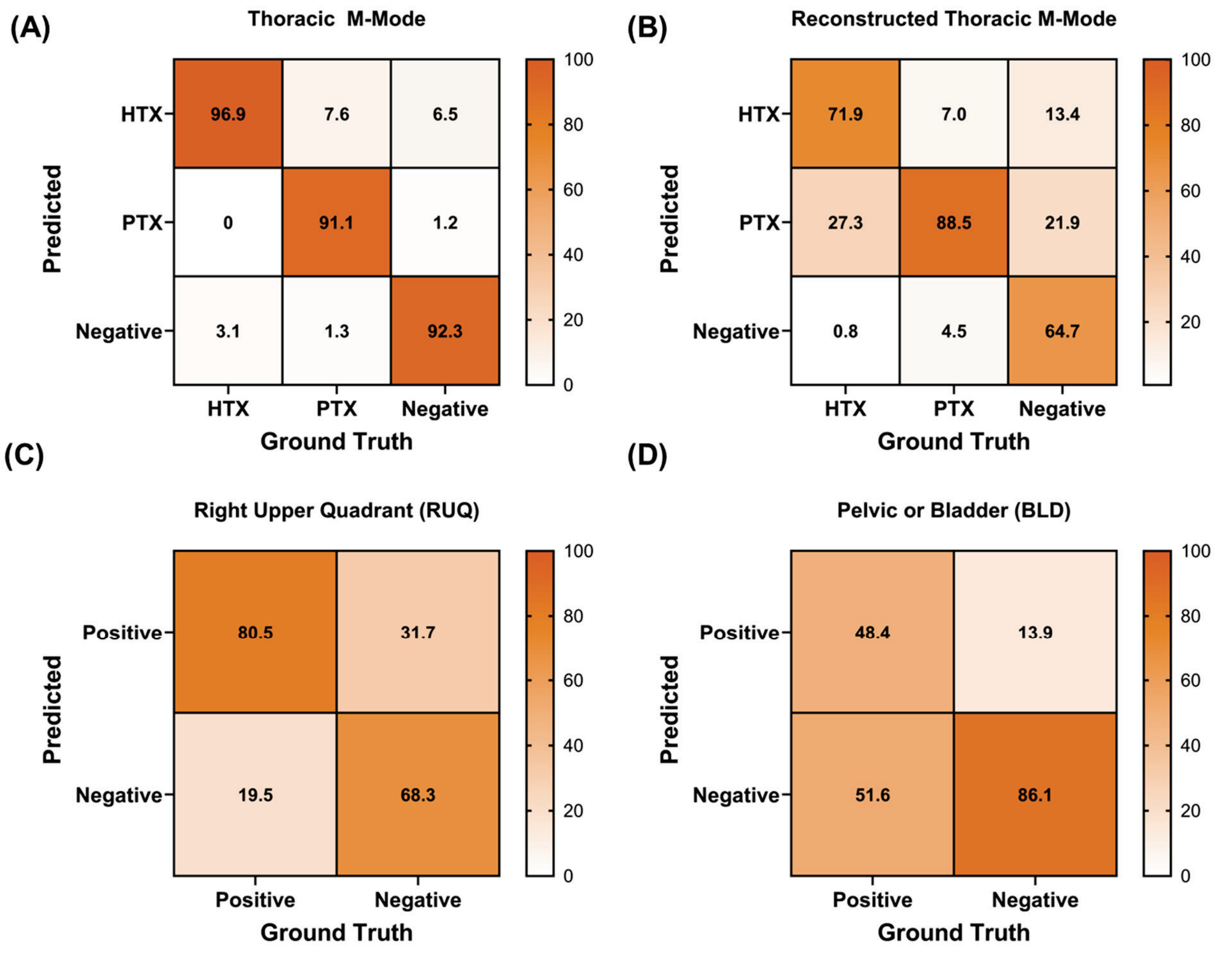


Figure 8. Diagnostic AI confusion matrices for each diagnostic model. (**A**) Three-class thoracic model using M-mode images; (**B**) three-class thoracic model using M-mode reconstructed from B-mode frames; (**C**) RUQ B-mode binary classification model; and (**D**) BLD B-mode binary classification model.

3.3. Real-Time Model Performance

We conducted real-time testing in three different ways. The first used the RT eFAST application and was primarily used to evaluate the AI guidance and diagnostic model performance at each scan site, along with the utility of different filtering approaches. The other two approaches were the manual, handheld eFAST exam with AI model feedback and RoboFAST. Both of these approaches allowed for a full eFAST exam to measure the timing of the procedures and how the AI models synergized with various image acquisition approaches.

3.3.1. Evaluation of the Real-Time eFAST Application

Starting with the RT eFAST application, the different filtering methods impacted the number of images that were captured at each scan site during a 30 s data capture window (Figure 9A). For ribs, on average, six less images were captured when using the guidance filter (approximately 37 vs. 31 images). Bad frame filtering was not applicable at this scan point due to M-mode capture needing to be continuous and not interrupted by frame removal procedures. The effects were more noticeable with RUQ and BLD, where bad frame filtering reduced the number of images by 12 and 3 images, respectively, while guidance filtering reduced the number of images by 30 and 16 images, respectively. Compounding these approaches reduced the number of images sent to the diagnostic models by 32 and approximately 18 images, respectively.

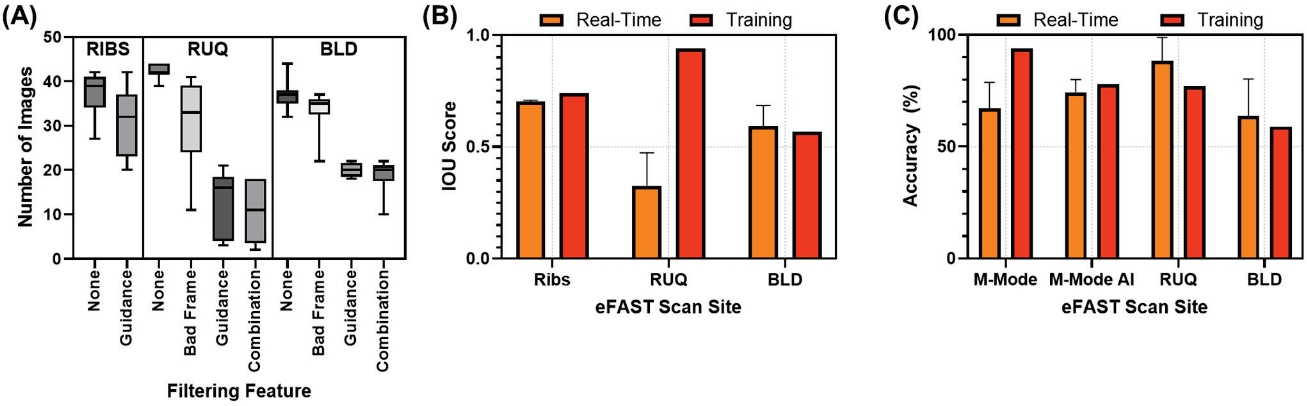


Figure 9. Evaluation of the real-time eFAST application. (**A**) Total number of images captured at each scan location for a set 30 s capture window for various pre-processing filter methods. Averages are shown along with the size of the box highlighting the 25th and 75th quartiles, while error bars denote minimum and maximum values. (**B**) Performance IOU results for AI-guided manual US image capture compared to test performance results during model training. (**C**) Diagnostic accuracy of real-time image capture compared to test accuracies during model training for each scan location. Mean values are shown with error bars denoting standard deviation.

Next, we evaluated how the guidance models performed using the RT eFAST application. This was undertaken without any filtering methods applied to obtain an overall IOU performance metric for each scan site (Figure 9B). In real time, performance decreased for ribs (0.70 real time vs. 0.74 training) and more substantially for the RUQ (0.33 real-time vs. 0.94 training), while BLD performance slightly increased (0.59 real time vs. 0.57 training). In terms of diagnostics, the effects of these filters on overall diagnostic accuracy were minimal, so the averaged diagnostic accuracy results comparing training performance are shown in Figure 9C. Performance was comparable to training data, with the exception of the M-mode thoracic model, which had a reduced accuracy of 0.67 compared to 0.94 during model training.

3.3.2. RoboFAST Evaluation

The robotic imaging platform relied on a computer vision model to identify each eFAST scan site automatically. The IOU scores for these predictions across scan sites were as follows: 0.51, 0.52, and 0.56 for ribs, RUQ, and BLD, respectively (Figure 10C). For US image capture, three approaches were used to capture images, as described in the Section 2, using the Robotic eFAST (RoboFAST) exam with AI model guidance: Radar, No Radar, and All Radar modalities. We first evaluated the effects of the various methods on the total number of images captured (Figure 10A). As anticipated, the All Radar approach captured the most images for each scan site, while No Radar and Radar had similar numbers of images for the RUQ and BLD scan sites. We next quantified the overall success of each scan site across the three swine subjects, where success is defined as at least one image being captured that could be used for diagnosis (Figure 10B). All approaches had high performance here, except for the RUQ/No Radar approach at 67% success. Factoring this in, Radar and All Radar had similar performance levels for this evaluation criterion. The guidance model IOU performance scores were similar for each RoboFAST imaging modality, with BLD having the highest IOU scores and RUQ performing the worst and having the highest subject variability (Figure 10C).

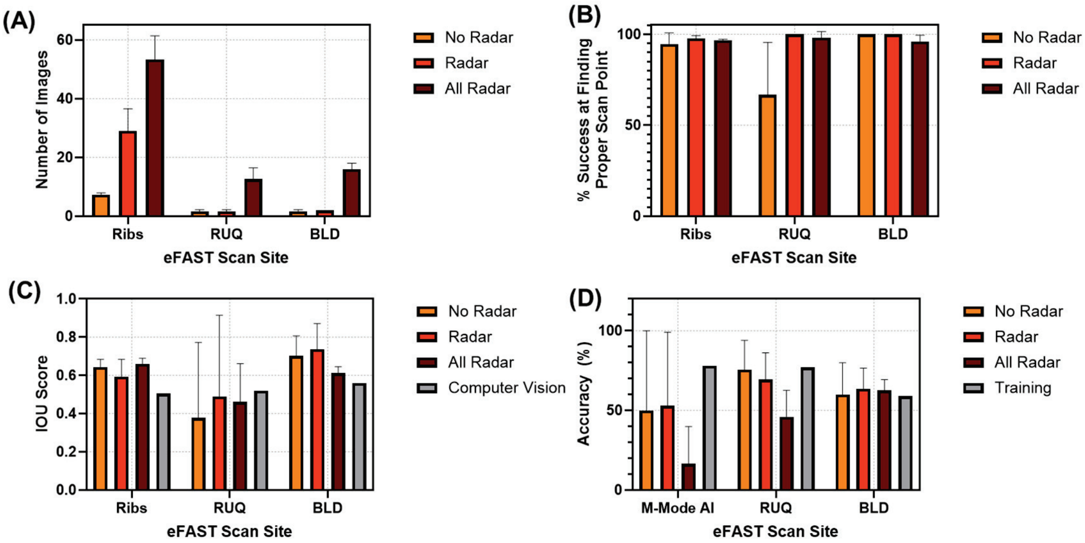


Figure 10. RoboFAST performance evaluation in swine. (**A**) Number of images captured with each imaging modality with the robotic imaging platform. (**B**) Overall success of RoboFAST in finding an US image to send to diagnostic AI models for each scan point and imaging modality. (**C**) IOU performance results for guidance AI models using No Radar, Radar, and All Radar modalities; computer vision IOU scores for identifying scan sites are also shown for ribs, RUQ, and BLD positioning. (**D**) Diagnostic accuracies for each scan modality compared to diagnostic model blind test accuracies during training. Averages are shown and error bars denote standard deviation across triplicate swine subjects throughout.

Lastly, we evaluated the diagnostic model performance. The All Radar modality resulted in the lowest accuracy for the M-mode thoracic AI (16.5%) and RUQ (46%) models (Figure 10D). Radar and No Radar performed similarly at each scan site. Compared to the test results obtained during model training, BLD and RUQ were comparable to the RoboFAST captured accuracies, while RoboFAST severely underperformed for the thoracic scan sites. This was likely a result of the robotic imaging platform experiencing difficulty reaching the proper thoracic scan site where pleural space was present, as shown in the representative US images captured during RoboFAST (Figure 11).

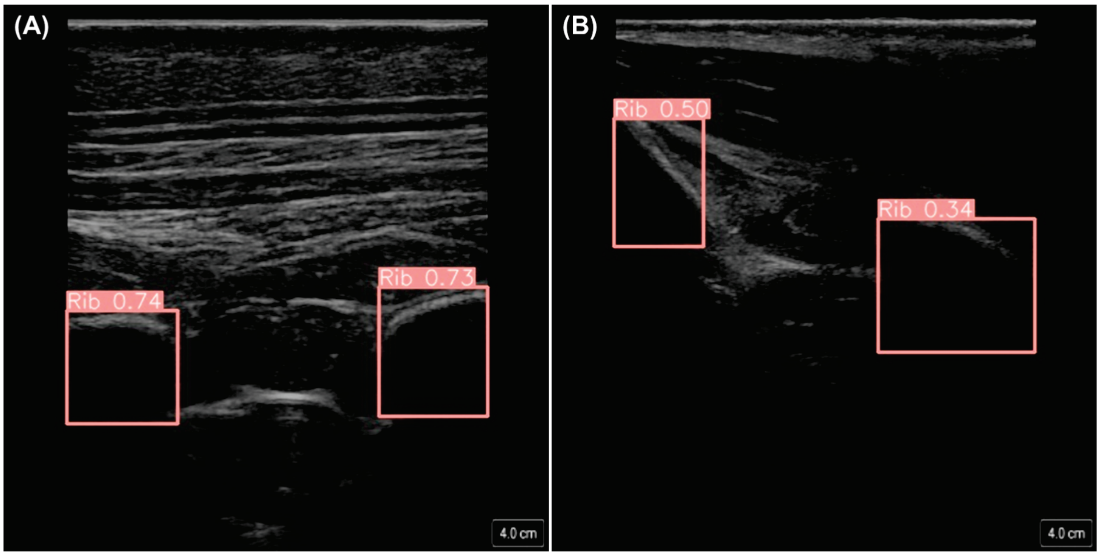


Figure 11. RoboFAST thoracic US images. Representative US images captured by the robotic platform with pleural space (**A**) in view and (**B**) not in view.

3.3.3. Timing Comparison Between Handheld eFAST Application and RoboFAST

Ultimately, we compared the overall time required to complete two RT eFAST imaging methodologies (Figure 12). Instead of the RT eFAST application, we configured the AI models for use in sequence across six total scan locations to mirror how the images were captured with the robotic imaging platform: (i) right thoracic top and (ii) bottom, (iii) left thoracic top and (iv) bottom, (v) RUQ, and (vi) BLD (described in Section 2.5.2). This matches the number of scan sites used during RoboFAST. We evaluated the timing of image capture by the end user having or not having the US screen visible (only relying on AI predictions and instructions to move to the next scan site), which resulted in a slightly longer time on average with no screen visible compared to when the screen was present (138 s manual, screen vs. 183 s manual, no screen). The RUQ scan site was most impacted by not looking at the US screen, as most captured images were excluded by the guidance filter. For the robotic imaging platform, the No Radar modality was the quickest (87 s), with rapid thoracic image capture compared to the slower Radar image capture (170 s), and the overall slowest All Radar modality (580 s).

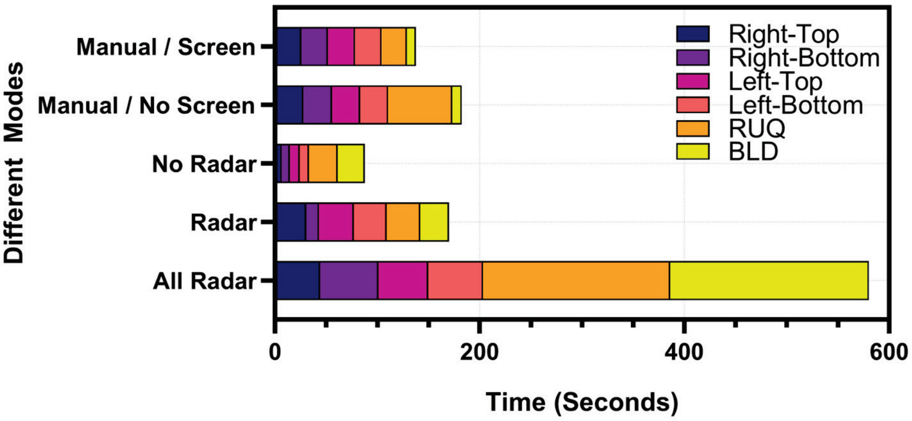


Figure 12. Summary of eFAST image capture times. Results are shown for all scan sites evaluated for each configuration of the manual AI-guided and automated robotic image platform. Average results are shown for each scan site across triplicate animal experiments.

4. Discussion

As ultrasound technology becomes smaller and more portable, its potential utility in emergency medicine widens. Pre-hospital triage by US imaging may be possible if the challenges of imaging can be reduced so that less-skilled personnel can perform initial triage assessments. This is especially true for military medicine, where triage decisions in the battlefield must prioritize limited evacuation opportunities in scenarios where air evacuation is not readily available, as has been the case in the ongoing conflict in Ukraine [29]. The AI-driven tools showcased in this research demonstrate how US imaging can be simplified to lower the skill threshold for triage on future battlefields or in other civilian emergency situations.

We have previously developed AI models for the diagnostic evaluation of eFAST scan sites, so this research effort was predominately focused on the automation of image acquisition techniques. Guidance object detection AI models were built using a YOLO model architecture, which was further tuned for use with swine datasets. Performance was mixed in the real-time implementation of these models, with BLD and RUQ underperforming compared to rib detection models. However, this still highlights how guidance models could assist with real-time scanning. These models can be used as a filter during manual scanning to exclude all frames in which key anatomical features are not present. Additionally, they may be used to provide autonomous feedback to robotic image acquisition platforms to acquire images with evident anatomical features that are required for proper diagnostic interpretation. However, these models need further refinement to ensure that not only anatomical features are present in the image, but also that the ideal anatomical features for diagnostic determination are identified. For instance, our models confirmed the presence of two ribs in each image so that the pleural space between the ribs can be evaluated for diagnosis. However, if the probe is not oriented correctly, the pleural space cannot be seen, making injury identification impossible. Additionally, our model confirms the presence of a kidney in each image to evaluate RUQ scan sites. However, since fluid often pools around the edges of the kidney, guidance models could be improved by confirming that the edges of the kidney are in view so that images used for diagnostic interpretation capture the area most likely to demonstrate evidence of injury. These additional improvements would further enhance their utility in providing ultrasound-based guidance feedback for image acquisition during an eFAST examination.

In addition, diagnostic AI models were further refined prior to further developing the models for real-time application. US image sets were expanded to more than 35 swine subjects to ideally allow for more robust model training performance. For simplicity in this study, all models were developed using a YOLOv8 image classification model. However, the guidance models were consistently more accurate compared to the new diagnostic models. A likely reason for the difference in model performance is that the guidance models were required to identify anatomical landmarks, while the diagnostic models were tasked with the more difficult task of interpreting nuanced changes in variable injury sizes. In the real-time testing, the BLD diagnostic models performed at low accuracy levels of 50–60%, similar to the initial model training performance. The overall low BLD performance could be due to three primary challenges: one, the additional image variability due to the size of the bladder being more variable than anatomical features at other scan sites; two, on US scans, the bladder presents as a dark fluid-filled feature, similar to abdominal hemorrhage fluid, possibly making the AI training task more complex; three, the urinary catheter balloon is often in view in the US scan images, which could be adding an additional artifact to the BLD training process. Additional image curation, robust model architecture, and rigorous model fine-tuning will be needed to improve AI training performance and the use of these models for real-time image interpretation. As for the methods of exploring model architectures, deep learning models used for segmentation can be applied to localize features of injury to help the models attribute the presence of fluid around the bladder, resulting in positive classifications. Long short-term memory (LSTM) networks used in video analysis can be explored to give the models more context on the appearance of variable injury sizes when making predictions on sequential images. Lastly, adding filters or pre-processing techniques with the purpose of amplifying relevant areas of the bladder can be tested for model training to help differentiate features between classifications.

AI models were evaluated in real time, with and without a robotic imaging platform, highlighting the different end-user applications of this technology. The handheld manual AI-guided application had faster performance, but still requires a user to position the probe in the right location. Filtering approaches were used to exclude images that were not suitable for diagnostic evaluation, which resulted in the exclusion of a large number of images from the diagnostic pipeline. Image filtering is critically needed for automated image acquisition in a handheld format, as less-experienced users may place the ultrasound probe at incorrect positions that may not have been included in diagnostic AI training datasets, resulting in a higher likelihood of incorrect diagnostic predictions. Rather than try and make diagnostic models more robust to handle these irregular images, filtering applications can prevent these images from impacting diagnostic predictions. Unfortunately, due to the lower performance of some of the diagnostic models during testing, it is hard to evaluate the effects of some of these filtering methods on overall eFAST performance metrics. Larger datasets paired with modified diagnostic models are needed to finalize the development of these filters and manual AI-guided eFAST image capture techniques.

However, there are some limitations with these filtering approaches when used in real time. For instance, over-filtering can result in removing viable images for diagnostic evaluation, leading to reduced model performance. Both bad frame removal and guidance filtering approaches could contribute to the over-filtering issue. The parameters for the bad frame removal filter were generated from a subset of 2000 images per scan site; as a result, the image subset could be not representative of the entire dataset or real-time testing data, leading to performance issues. Similarly, the guidance filter could impede real-time performance based on the guidance models' own performance biases. Further, the identification of anatomical features is not always indicative of where fluid pools around organs or in the pleural space. Another challenge with real-time implementation is the loss of image resolution and introduction of artifacts due to streaming the US signal. To account for this effect, the inclusion of streamed frames at different resolutions in AI model training data should be considered for improving performance in future implementations of this technology.

For the robot image capture platform, different configurations had a wide impact on the speed of performing an eFAST examination. However, this is mostly tied to the number of images that were being captured and the carefulness being applied to ensure that a proper eFAST viewpoint was captured at each scan site. The robot's limited range of motion was challenged by the deeper angles required to image the RUQ or the lower thorax, where HTX injuries are often identified. This was due to the bulkiness of the platform and poor clearance with the table on which the subject was placed. Guidance models performed as expected; however, diagnostic accuracies for the thoracic scan sites were low. This was not exclusively due to issues with the model, but also challenges with the robot's ability to correctly position and angle the probe on the chest to properly image two ribs in the area that was searched. More gradual movement and better tracking of the proper direction to move across the thoracic cavity could improve performance in future iterations of RoboFAST. Conversely, the RUQ and BLD had similar accuracy to the testing results of the diagnostic models. This provides evidence of the utility of robotic mechanisms to automate image capture, but more work is needed to further ready this platform for rapid and proper eFAST image acquisition.

The utility of the handheld and robotic eFAST imaging platforms differ greatly in their potential applications. Obviously, a large robotic system is not feasible in all pre-hospital settings but could be envisioned at a site for processing mass casualty scenarios, for automated triage assessment in a hospital, or later military echelons of care. Less human support is needed once the technology is further refined, so a more automated design can potentially streamline casualty in-processing. In direct contrast, the handheld tool, if paired with small, portable US devices, could be deployed in ambulatory civilian care or military care near the point of injury. While the technology will still require the user to manipulate the technology to proper positions, additional guidance measures in the software application can further lower the skill threshold during real-time deployment.

The next steps for this research effort will expand this application in several directions. First, the underlying AI models for detecting injuries need to reach higher performance metrics to be ready for deployment. This will require a more varied imaging dataset, as well as improvements to the underlying AI. Model performance may be improved if the AI is trained with temporal context from several frames, rather than relying on predictions from a single frame. Further translation of this work will require transfer learning AI models to use with human anatomy and injury states. To accomplish this, a large, curated dataset would need to be acquired through collaboration with emergency medicine departments, where these US images are routinely captured. Second, the real-time handheld application needs an improved end-user interface so that the end user can make smaller adjustments during thoracic scanning to ensure proper M-mode images are captured with varied angles at each scan site. One solution to this challenge is to further refine guidance functionality beyond the simple identification of anatomical features toward a determination of optimal scan placement. For example, if the kidney is imaged with optimal ultrasound probe placement and orientation, the edges of the kidney where fluid is more commonly seen would be in view. In addition, ensuring the pleural space is visible in the thoracic site for proper diagnostic interpretation is necessary. Lastly, the robotic platform will be further automated to overcome some of its limitations. Improved computer vision algorithms for anatomical landmark detection, automated ultrasound gel deployment, and automated probe swapping between linear and curvilinear probe types are just some of the modifications planned to improve this real-time application.

5. Conclusions

Ultrasound imaging can revolutionize medical triage in trauma cases, if it can be pushed further forward to the point where the first medical decisions are made in both civilian and military medicine. Towards this mindset, the real-time AI-driven triage tools showcased here have the potential to lower the skill threshold of image-based triage decisions. The handheld application has a small footprint optimal for ease of deployment if the end user can position the ultrasound probe correctly and make proper image interpretation decisions. The robotic-driven image capture application further automates the procedure; however, it does so at the expense of its larger size, which will not be suitable in the earliest phase of trauma medical care. In conclusion, both applications provide evidence of the promise AI can provide to simplify medical imaging and improve medical triage decisions on the future battlefield and in pre-hospital settings.

6. Patents

Eric J. Snider and Sofia I. Hernandez Torres are co-inventors on a provisional patent filed on the eFAST AI concept and usage (63/686,836; August 2024). Eric J. Snider, Sofia I. Hernandez Torres, and Krysta-Lynn Amezcua are co-inventors on a provisional patent filed on the robotic eFAST imaging concept (63/686,839; August 2024).

Author Contributions: Conceptualization, S.I.H.T., K.-L.A. and E.J.S.; Data curation, S.I.H.T., C.R.T. and E.J.S.; Formal analysis, L.H., K.-L.A. and E.J.S.; Funding acquisition, E.J.S.; Investigation, E.J.S.; Methodology, S.I.H.T., L.H. and E.J.S.; Project administration, E.J.S.; Resources, E.J.S.; Software, L.H., T.W., R.O., K.-L.A. and A.R.; Supervision, E.J.S.; Validation, S.I.H.T., K.-L.A. and E.J.S.; Visualization, S.I.H.T. and E.J.S.; Writing—original draft, S.I.H.T., L.H., T.W., R.O., K.-L.A., A.R. and E.J.S.; Writing—review and editing, S.I.H.T., L.H., T.W., R.O., K.-L.A., A.R., C.R.T. and E.J.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the U.S. Army Medical Research and Development Command (IS220007). This project was supported in part by an appointment to the Science Education Programs at National Institutes of Health (NIH), administered by ORAU through the U.S. Department of Energy Oak Ridge Institute for Science and Education (A.R., T.W., R.O., C.T.).

Institutional Review Board Statement: Research was conducted in compliance with the Animal Welfare Act, the implementing Animal Welfare regulations, and the principles of the Guide for the Care and Use of Laboratory Animals. The Institutional Animal Care and Use Committee approved all research conducted in this study. The facility where this research was conducted is fully accredited by AAALAC International.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets presented in this article are not readily available because they have been collected and maintained in a government-controlled database that is located at the U.S. Army Institute of Surgical Research. As such, data can be made available through the development of a business agreement with the corresponding author. Requests for the datasets should be directed to Dr. Eric J. Snider, eric.j.snider3.civ@health.mil.

Acknowledgments: The authors acknowledge Evan Ross, Jose Gonzalez, David Owen, Carlos Bedolla, and Benjamin Alexander for their assistance in eFAST image capture throughout the evaluation of the real-time image capture procedures.

Conflicts of Interest: Eric J. Snider and Sofia I. Hernandez Torres are co-inventors on a provisional patent filed on the eFAST AI concept and usage (63/686,836; August 2024). Eric J. Snider, Sofia I. Hernandez Torres, and Krysta-Lynn Amezcua are co-inventors on a provisional patent filed on the robotic eFAST imaging concept (63/686,839; August 2024). No other authors report any conflicts of interest.

DOD Disclaimer: The views expressed in this article are those of the authors and do not reflect the official policy or position of the U.S. Army Medical Department, Department of the Army, DoD, or the U.S. Government.

References

1. Nabrawi, E.; Alanazi, A.T. Imaging in Healthcare: A Glance at the Present and a Glimpse Into the Future. *Cureus* **2023**, *15*, e36111. [CrossRef]
2. Rigal, S.; Pons, F. Triage of Mass Casualties in War Conditions: Realities and Lessons Learned. *Int. Orthop.* **2013**, *37*, 1433–1438. [CrossRef] [PubMed]
3. Dubecq, C.; Dubourg, O.; Morand, G.; Montagnon, R.; Travers, S.; Mahe, P. Point-of-Care Ultrasound for Treatment and Triage in Austere Military Environments. *J. Trauma. Acute Care Surg.* **2021**, *91*, S124–S129. [CrossRef] [PubMed]

4. Stamilio, D.M.; McReynolds, T.; Endrizzi, J.; Lyons, R.C. Diagnosis and Treatment of a Ruptured Ectopic Pregnancy in a Combat Support Hospital during Operation Iraqi Freedom: Case Report and Critique of a Field-Ready Sonographic Device. *Mil. Med.* **2004**, *169*, 681–683. [CrossRef]
5. Remondelli, M.H.; Remick, K.N.; Shackelford, S.A.; Gurney, J.M.; Pamplin, J.C.; Polk, T.M.; Potter, B.K.; Holt, D.B. Casualty Care Implications of Large-Scale Combat Operations. *J. Trauma. Acute Care Surg.* **2023**, *95*, S180–S184. [CrossRef] [PubMed]
6. Bloom, B.A.; Gibbons, R.C. Focused Assessment with Sonography for Trauma. In *StatPearls*; StatPearls Publishing: Treasure Island, FL, USA, 2021.
7. Letter from the President: A Shortage of Preventive Medicine Physicians in the Military and Across the Country. Available online: https://www.acpm.org/news/2024/letter-from-the-president-a-shortage-of-preventive (accessed on 29 November 2024).
8. Pinto-Coelho, L. How Artificial Intelligence Is Shaping Medical Imaging Technology: A Survey of Innovations and Applications. *Bioengineering* **2023**, *10*, 1435. [CrossRef] [PubMed]
9. Liu, X.; Faes, L.; Kale, A.U.; Wagner, S.K.; Fu, D.J.; Bruynseels, A.; Mahendiran, T.; Moraes, G.; Shamdas, M.; Kern, C.; et al. A Comparison of Deep Learning Performance against Health-Care Professionals in Detecting Diseases from Medical Imaging: A Systematic Review and Meta-Analysis. *Lancet Digit. Health* **2019**, *1*, e271–e297. [CrossRef] [PubMed]
10. Lotter, W.; Diab, A.R.; Haslam, B.; Kim, J.G.; Grisot, G.; Wu, E.; Wu, K.; Onieva, J.O.; Boyer, Y.; Boxerman, J.L.; et al. Robust Breast Cancer Detection in Mammography and Digital Breast Tomosynthesis Using an Annotation-Efficient Deep Learning Approach. *Nat. Med.* **2021**, *27*, 244–249. [CrossRef]
11. Garcia, P. Telemedicine for the Battlefield: Present and Future Technologies. In *Surgical Robotics: Systems Applications and Visions*; Rosen, J., Hannaford, B., Satava, R.M., Eds.; Springer: Boston, MA, USA, 2011; pp. 33–68. ISBN 978-1-4419-1126-1.
12. Rinehart, J.; Lilot, M.; Lee, C.; Joosten, A.; Huynh, T.; Canales, C.; Imagawa, D.; Demirjian, A.; Cannesson, M. Closed-Loop Assisted versus Manual Goal-Directed Fluid Therapy during High-Risk Abdominal Surgery: A Case-Control Study with Propensity Matching. *Crit. Care* **2015**, *19*, 94. [CrossRef] [PubMed]
13. Kramer, G.C.; Kinsky, M.P.; Prough, D.S.; Salinas, J.; Sondeen, J.L.; Hazel-Scerbo, M.L.; Mitchell, C.E. Closed-Loop Control of Fluid Therapy for Treatment of Hypovolemia. *J. Trauma* **2008**, *64*, S333–S341. [CrossRef] [PubMed]
14. Vega, S.J.; Berard, D.; Avital, G.; Ross, E.; Snider, E.J. Adaptive Closed-Loop Resuscitation Controllers for Hemorrhagic Shock Resuscitation. *Transfusion* **2023**, *63*, S230–S240. [CrossRef]
15. Mohan, A.; Wara, U.U.; Arshad Shaikh, M.T.; Rahman, R.M.; Zaidi, Z.A. Telesurgery and Robotics: An Improved and Efficient Era. *Cureus* **2021**, *13*, e14124. [CrossRef] [PubMed]
16. Levy, B.E.; Castle, J.T.; Virodov, A.; Wilt, W.S.; Bumgardner, C.; Brim, T.; McAtee, E.; Schellenberg, M.; Inaba, K.; Warriner, Z.D. Artificial Intelligence Evaluation of Focused Assessment with Sonography in Trauma. *J. Trauma Acute Care Surg.* **2023**, *95*, 706–712. [CrossRef] [PubMed]
17. Huang, L.; Lin, Y.; Cao, P.; Zou, X.; Qin, Q.; Lin, Z.; Liang, F.; Li, Z. Automated Detection and Segmentation of Pleural Effusion on Ultrasound Images Using an Attention U-Net. *J. Appl. Clin. Med. Phys.* **2024**, *25*, e14231. [CrossRef] [PubMed]
18. Gao, X.; Lv, Q.; Hou, S. Progress in the Application of Portable Ultrasound Combined with Artificial Intelligence in Pre-Hospital Emergency and Disaster Sites. *Diagnostics* **2023**, *13*, 3388. [CrossRef] [PubMed]
19. Hernandez Torres, S.I.; Ruiz, A.; Holland, L.; Ortiz, R.; Snider, E.J. Evaluation of Deep Learning Model Architectures for Point-of-Care Ultrasound Diagnostics. *Bioengineering* **2024**, *11*, 392. [CrossRef] [PubMed]
20. Amezcua, K.-L.; Collier, J.; Lopez, M.; Hernandez Torres, S.I.; Ruiz, A.; Gathright, R.; Snider, E.J. Design and Testing of Ultrasound Probe Adapters for a Robotic Imaging Platform. *Sci. Rep.* **2024**, *14*, 5102. [CrossRef] [PubMed]
21. Boysen, S.R.; Caulkett, N.A.; Brookfield, C.E.; Warren, A.; Pang, J.M. Splenectomy Versus Sham Splenectomy in a Swine Model of Controlled Hemorrhagic Shock. *Shock* **2016**, *46*, 439. [CrossRef] [PubMed]
22. Watts, S.; Nordmann, G.; Brohi, K.; Midwinter, M.; Woolley, T.; Gwyther, R.; Wilson, C.; Poon, H.; Kirkman, E. Evaluation of Prehospital Blood Products to Attenuate Acute Coagulopathy of Trauma in a Model of Severe Injury and Shock in Anesthetized Pigs. *Shock* **2015**, *44*, 138. [CrossRef]
23. Snider, E.J.; Hernandez-Torres, S.I.; Boice, E.N. An Image Classification Deep-Learning Algorithm for Shrapnel Detection from Ultrasound Images. *Sci. Rep.* **2022**, *12*, 8427. [CrossRef] [PubMed]
24. Yaseen, M. What Is YOLOv8: An In-Depth Exploration of the Internal Features of the Next-Generation Object Detector 2024. *arXiv* **2024**, arXiv:2408.15857.
25. Bilous, N.; Malko, V.; Frohme, M.; Nechyporenko, A. Comparison of CNN-Based Architectures for Detection of Different Object Classes. *AI* **2024**, *5*, 2300–2320. [CrossRef]
26. Le, V.-H.; Pham, T.-L. Ovarian Tumors Detection and Classification on Ultrasound Images Using One-Stage Convolutional Neural Networks. *J. Robot. Control. (JRC)* **2024**, *5*, 21.

27. Jocher, G.; Chaurasia, A.; Qiu, J. YOLO by Ultralytics. 2023. Available online: https://github.com/ultralytics/ultralytics (accessed on 8 January 2025).
28. Cai, Z.; Vasconcelos, N. Cascade R-Cnn: Delving into High Quality Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; IEEE: New York City, NY, USA, 2018; pp. 6154–6162.
29. Epstein, A.; Lim, R.; Johannigman, J.; Fox, C.J.; Inaba, K.; Vercruysse, G.A.; Thomas, R.W.; Martin, M.J.; Konstantyn, G.; Schwaitzberg, S.D.; et al. Putting Medical Boots on the Ground: Lessons from the War in Ukraine and Applications for Future Conflict with Near-Peer Adversaries. *J. Am. Coll. Surg.* **2023**, *237*, 364–373. [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.

technologies

MDPI

Article

Towards Transparent AI in Medicine: ECG-Based Arrhythmia Detection with Explainable Deep Learning

Oleksii Kovalchuk [1], Oleksandr Barmak [1], Pavlo Radiuk [1,*], Liliana Klymenko [2] and Iurii Krak [3,4]

[1] Department of Computer Science, Khmelnytskyi National University, 11 Instytuts'ka Str., 29016 Khmelnytskyi, Ukraine; kovalchukov@khmnu.edu.ua (O.K.); barmako@khmnu.edu.ua (O.B.)
[2] Department of Family Medicine and Outpatient Care, Shupyk National Healthcare University of Ukraine, 9 Dorohozhytska Str., 04112 Kyiv, Ukraine; dr-liliana-ua@ukr.net
[3] Department of Theoretical Cybernetics, Taras Shevchenko National University of Kyiv, 4d Akademika Glushkova Ave, 03680 Kyiv, Ukraine; iurii.krak@knu.ua
[4] Laboratory of Communicative Information Technologies, V.M. Glushkov Institute of Cybernetics, 40 Akademika Glushkova Ave, 03187 Kyiv, Ukraine
* Correspondence: radiukp@khmnu.edu.ua; Tel.: +38-(097)-854-9148

Abstract: Cardiovascular diseases are the leading cause of death globally, highlighting the need for accurate diagnostic tools. To address this issue, we introduce a novel approach for arrhythmia detection based on electrocardiogram (ECG) that incorporates explainable artificial intelligence through three key methods. First, we developed an enhanced R peak detection method that integrates domain-specific knowledge into the ECG, improving peak identification accuracy by accounting for the characteristic features of R peaks. Second, we proposed an arrhythmia classification method utilizing a modified convolutional neural network (CNN) architecture with additional convolutional and batch normalization layers. This model processes a triad of cardio cycles—the preceding, current, and following cycles—to capture temporal dependencies and hidden features related to arrhythmias. Third, we implemented an interpretation method that explains CNN's decisions using clinically relevant features, making the results understandable to clinicians. Using the MIT-BIH database, our approach achieved an accuracy of 99.43%, with F1-scores approaching 100% for major arrhythmia classes. The integration of these methods enhances both the performance and transparency of arrhythmia detection systems.

Keywords: electrocardiography (ECG); arrhythmia detection; ECG classification; ECG interpretation; explainable artificial intelligence (XAI); transparent artificial intelligence; deep learning

1. Introduction

According to statistics from the World Health Organization, cardiovascular diseases are the leading cause of death worldwide [1,2]. Currently, there are many tools and means available to help clinicians prevent or detect heart problems. One of the most common methods is electrocardiography (ECG). ECG allows for the graphical recording of electrical phenomena from the human body that occurs in the heart muscle during its activity. The curve obtained from recording such activity is called an electrocardiogram or ECG. Thus, an ECG is a recording of fluctuations in the potential difference that occur in the heart during its excitation [3]. A standard ECG recording consists of 12 leads obtained from 10 electrodes [4].

The nature of the ECG is pseudoperiodic. The ECG consists of cardio cycles called QRST complexes (Figure 1). The appearance of the cardio cycle allows clinicians to determine the presence of potential heart pathologies from the ECG. It is important to note that the cardio cycle is visually identified by a clinician based on the R peak of the signal. It is also worth mentioning that existing datasets, in which specific heart pathologies are annotated, are also tied to the cardio cycle. Figure 1 shows the leading indicators (peaks and segments) clinicians use to analyze the cardio cycle for potential pathology.

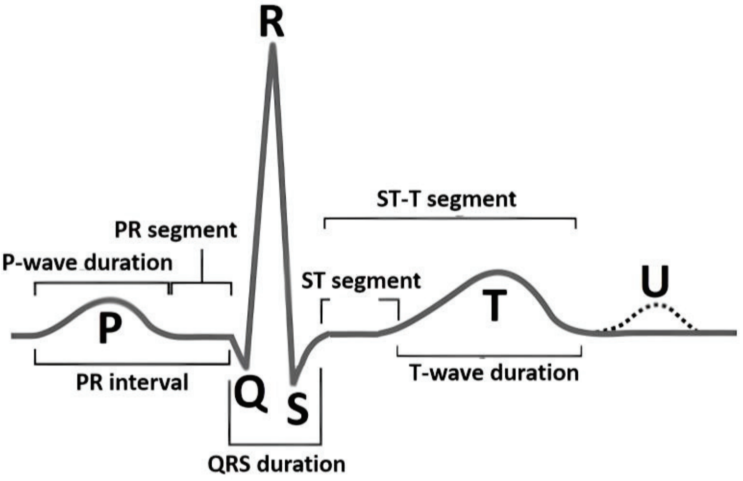


Figure 1. A schematic of a typical ECG waveform, illustrating the sequential components of an ideal cardiac cycle known as the QRST complex [4]. The diagram highlights the primary waveforms: P, Q, R, S, T, and U. Each segment and interval, including the PR interval, QRS duration, ST segment, and T-wave duration, is labeled to show phases of electrical activity in the heart.

Since there are currently many possible types of abnormalities in ECG and large volumes of ECG recordings (such as Holter monitoring), the analysis process can be time-consuming and prone to numerous errors. Therefore, information technology methods and approaches are used to address these issues. Due to the rapid development of artificial intelligence (AI) systems, tools like machine learning (ML) and deep learning (DL) have gained widespread use for classifying ECG pathologies.

The use of convolutional neural networks (CNNs) [5], as a DL method [6], for classifying ECGs has already demonstrated significant effectiveness in detecting various heart pathologies, such as arrhythmias, myocarditis, ischemic heart disease, and more [7,8]. Despite the significant advances in this field, the functioning of DL models remains a "black box" for end users [9], which is highly critical in a sensitive field like medicine [10,11].

Despite numerous existing solutions for ECG classification, several open problems remain unaddressed. In this work, we tackle unresolved issues in ECG analysis using DL that have been neglected or only partially considered. These issues include (i) low accuracy in R peak detection when R peaks exhibit atypical features, (ii) the inability of current classifiers to identify all possible arrhythmias, especially those underrepresented in available datasets due to class imbalance, and (iii) a lack of explainability and transparency in DL model decisions for end users (doctors).

Considering these challenges, the main contributions of this article are as follows:

- A method for identifying R peaks in ECGs: we integrate domain-specific knowledge to enhance R peak detection accuracy, allowing for more precise identification compared to existing methods.

- A method for classifying heart arrhythmias from ECGs: by presenting the input signal as three consecutive cardio cycles and using a modified CNN architecture, we improve classification quality over traditional approaches.
- A method for interpreting DL model classification results: we use features that are understandable to doctors, making the classification decisions transparent and enhancing explainability for end users.

The article's structure is as follows: Section 2 presents a review of current approaches for detecting heart rhythm, sequence, and contraction force disorders (arrhythmias) from ECGs using explainable AI (XAI) methods. Section 3 describes the proposed approach to solving the problem, which consists of three methods: R peak identification in ECGs, arrhythmia classification in ECGs, and classification result interpretation. Section 4 presents the experimental results of the proposed approach and a discussion.

2. Related Works

Preparing ECG data for use in DL models involves the mandatory segmentation of cardio cycles based on R peaks in the ECG. This approach is driven by the pseudoperiodic nature of ECGs, where the unit of heart activity analysis is the cardio cycle (i.e., QRST complex), and the accuracy of its detection depends on identifying the R peak. At the same time, the precision of R peak identification is crucial, as it affects the effectiveness of applying DL methods to solve the problem of classifying heart arrhythmias.

Currently, there are many approaches to detecting R peaks in ECGs, most of which show an efficiency rate exceeding 99%. However, many studies either indicate a significant margin of error or do not mention it at all when calculating statistical metrics. For example, in studies [12,13], high accuracy in peak detection was achieved, but the allowable error was ±75 ms, creating a total error window of 150 ms, which exceeds the normal duration of the QRST complex.

Since the medical field demands extremely high precision, such errors in studies can be critical. Therefore, researchers like B. Porr and P. W. Macfarlane [14] conducted an analysis of various methods for R peak detection, including Pan and Tompkins by Fariha et al. [15], Hamilton and Tompkins by Ahmad et al. [16], and Christov by Rahul et al. [17]. B. Porr and P. W. Macfarlane established that almost every study reported very high accuracy, above 98%, which can be explained using a large permissible error when calculating the accuracy of the proposed methods. The authors concluded that most experimental studies rely on an error of 100 ms or more, which is too high for real clinical cases involving ECGs. As a result, the task of identifying R peaks with minimal error remains relevant and requires further research.

The next step in ECG analysis, following R peak detection, is the classification of QRST complexes with R peaks according to pathology classes. There are numerous studies on the application of DL models for ECG classification.

For instance, Hassan et al. [18] trained a CNN-BiLSTM to classify five types of heart arrhythmias using the MIT-BIH dataset. They demonstrated that their DL model could classify heart arrhythmias with 98% accuracy, 91% sensitivity, and 91% specificity. Liu et al. [19] proposed an ensemble of LSTM and CNN, which achieved 99.1% accuracy, 99.3% sensitivity, and 98.5% specificity in classifying ECGs. Notably, the authors obtained these results by classifying ECGs into only four classes, excluding the "everything else" class. Xu et al. [20] developed a CNN classifier that categorizes ECGs into five classes, including "normal" and "others". However, their proposed method only allows the classification of three pathologies, covering a limited set of potential conditions.

In studies by Degirmenci et al. [21] and Rohmantri et al. [22], high classification accuracy was achieved using 2D images of ECGs with a size of 64 × 64 as input data for classifying arrhythmic heartbeats. There are also several works that transform one-dimensional signals into two-dimensional representations like spectrograms or scalograms, including [23]. Despite achieving high classification accuracy, these approaches result in significant computational cost, making them inefficient for real-time applications and devices with limited computing power. Additionally, the cited works do not utilize an "all other" class, which could potentially worsen classification outcomes.

In their study, Abdelhafid et al. [24] focused on classifying ECG arrhythmias using five classes without the "others" class. This likely contributed to the high classification metrics, but excluding the "others" class may not reflect real clinical cases, as such an approach ignores signals that do not fit predefined categories. Furthermore, their model takes input data for only one cardio cycle. Since the "Premature Ventricular Contraction" (PVC) class is included, this amount of data may be insufficient for classification, as this pathology is characterized by a "compensatory pause", which requires neighboring cardio cycles for detection. Thus, to effectively and accurately detect heart arrhythmias, it is crucial to develop a DL model that balances accuracy, computational complexity, and the ability to classify a greater number of pathologies, particularly a class that represents all other underrepresented conditions.

Singh and Sharma [25] introduced a deep CNN for arrhythmia interpretation and classification, which demonstrated high accuracy and efficiency. However, like other studies, they faced high computational requirements when using the proposed model in real-world applications. In addition, their study did not address the classification of signals that do not fall into the previous classes, which is crucial for practical applications.

In a recent paper, Ayano et al. [26] suggested an interpretive DL model for 12-lead ECG diagnosis. Their work stands out due to its interpretation and careful analysis of multiplexes, offering a detailed understanding of the diagnostic process. However, the complexity of their model may hinder its use in the absence of significant computing resources, as high interpretation often comes at the cost of computational complexity.

From the analysis above, we see that current approaches do not provide a full interpretation of the classification results that can be transparent and understandable to doctors in practical conditions. Specifically, we point out several issues that warrant further investigation:

- High error rates in R peak identification.
- A limited number of classified pathology classes.
- Classification based on a single cardio cycle without considering preceding or subsequent cycles, thereby ignoring hidden features from adjacent cycles.
- High computational complexity in pathology classification tasks.
- A lack of explainability in DL model decisions using features understandable to healthcare professionals.

Therefore, the aim of this study is to improve the quality and accuracy of detecting heart activity disorders (i.e., arrhythmia) from ECG analysis using DL, while also making the results interpretable to doctors. To achieve this goal, we propose a new approach for arrhythmia detection in ECGs using XAI, which comprises three methods: (i) identifying R peaks in ECGs, (ii) classifying arrhythmia in ECGs, and (iii) interpreting classification results using features that are understandable to doctors.

3. Methods and Materials

In this study, we propose an approach for detecting heart activity disorders related to rhythm, sequence, and contraction strength of the heart muscle (arrhythmias) using ECG with XAI. The overall scheme of the proposed approach is shown in Figure 2.

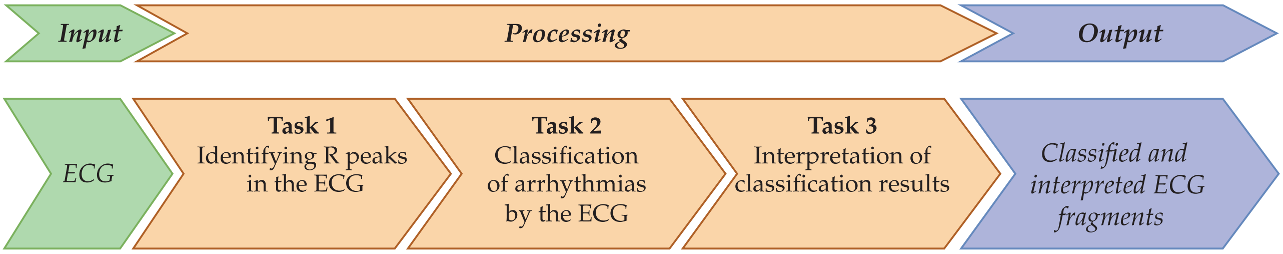

Input
Processing
Output
ECG
Task 1
Identifying R peaks in the ECG
Task 2
Classification of arrhythmias by the ECG
Task 3
Interpretation of classification results
Classified and interpreted ECG fragments

Figure 2. This figure outlines a three-task ECG arrhythmia classification approach using XAI. It starts with ECG input and proceeds through R peak identification, arrhythmia classification, and result interpretation, resulting in classified ECG fragments. This approach integrates domain knowledge to improve diagnostic accuracy and interpretability for clinicians.

The proposed approach has several assumptions, which, according to the authors, improves the quality and accuracy of detecting heart activity disorders (arrhythmias) by analyzing the ECG using DL, followed by interpreting the results in terms understandable to the end user (i.e., doctor). Thus, the approach includes the following:

- Integrating domain knowledge into the ECG to enhance R peak identification.
- Representing the input signal as a triad of cardio cycles to improve the model's ability to detect hidden dependencies related to pathologies in the input ECG. Each cardio cycle is supplemented with its predecessor and successor, as considering only one cardio cycle is insufficient for making the right decision from a doctor's perspective. Information about what happened before and after the current cardio cycle is also required.
- Presenting DL model decisions as a combination of features relevant to medical practice, which either confirm or refute the DL decision.

The proposed approach is implemented by breaking down the study's goal into smaller and interrelated tasks.

The input data of the approach is an ECG obtained from recording devices. The ECG is represented as a one-dimensional array s, which reflects the course of the time signal recorded at a specific moment for a particular lead. The data are digitized at a sampling rate of 360 samples per second with 11-bit resolution over a range of 10 mV. It is worth mentioning that information from recording devices in other structures is converted into the required format using simple algorithmic transformations.

Task 1 is intended to identify R peaks in the input ECG (Figure 1). This task is necessary because the CNN requires comparable signal segments as input. One way to achieve this comparability is to segment the signal so that each segment centers around an R peak.

Task 2 is for classifying the pathologies indicative of arrhythmias. The classification applies to ECG segments identified based on the R peaks.

Task 3 is for interpreting the obtained classification results, meaning that the decision made by the DL model is explained in terms understandable to the doctor.

The output of the proposed approach is a classified ECG along with explanations for the classification decision regarding each cardio cycle.

To solve the tasks given, corresponding methods are proposed. Each method is discussed in detail below.

3.1. Method for Identifying R Peaks in ECG

Based on the review presented in Section 2, it is evident that, despite the impressive results of R peak detection methods, there are certain shortcomings that need to be addressed. To improve the current approaches for R peak detection in ECGs, we proposed a method illustrated in Figure 3.

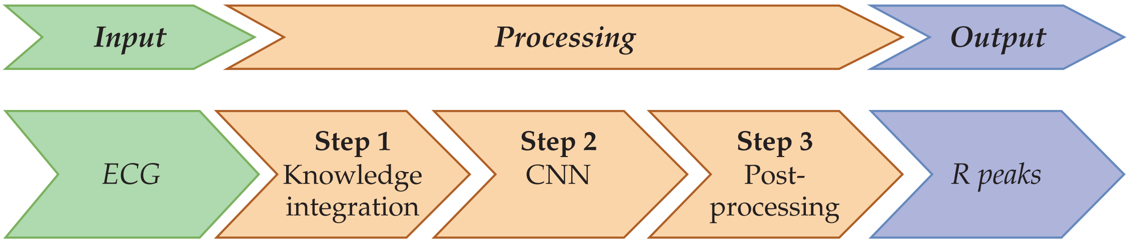


Figure 3. This figure illustrates the R peak detection method in ECG analysis, consisting of three key steps: knowledge integration, CNN processing, and post-processing. This method leverages domain knowledge of the reference heart cycle to enhance R peak detection accuracy, producing precise R peak locations in the output.

The main feature of the proposed method is the integration of knowledge about the reference heart cycle into the input ECG. The hypothesis of this study echoes our previous work [13] in that such an integrated signal is more effective in detecting the necessary information (R peaks) and is more resistant to signal artifacts.

It should be noted that integrating knowledge about the reference heart cycle into the input ECG is not a new approach. A similar idea of integrating knowledge into the ECG was proposed in [27].

In our case, for the implementation of knowledge integration, a characteristic feature of the R peak is used, namely that the R peak has the maximum positive deviation within the cardio cycle.

The method involves the following transformation steps:

Input data: (1) ECG S as a one-dimensional data array, and (2) a corresponding array K, initialized with zero values across its entire length. This array is used to store knowledge about the ideal peaks of the reference heart rhythm at the specified positions during the subsequent steps.

Step 1: Integrate knowledge about the reference cardio cycle into the ECG.

Step 2: Process the integrated signal using the CNN model.

Step 3: Post-process the results of the DL model to identify R peaks.

Output data: A filled array K.

Details of each step are provided below.

It is known that in leads I–II and V1–V6 of an ECG, the R peak is characterized by the highest positive deviation of the signal in a specific region. To integrate knowledge of this into each ECG segment, the following steps are applied as depicted in Figure 4:

Step 1.1: Extract a segment of the ECG containing 260 elements. This number of elements was determined experimentally and is sufficient to cover a cardio cycle.

Step 1.2: Conduct preliminary R peak identification, i.e., determine the maximum positive deviation, p, in the extracted segment of the signal. The detected maximum deviation is then checked to decide whether it represents a peak. If the deviation does not increase on the left and does not decrease on the right, the identified deviation is not at the peak, and the process returns to the previous step to process the next segment of the signal. If the check is successful, the process moves to the next step.

Step 1.3: Populate the array K with knowledge. Based on the identified peak, its global index i in the ECG S is determined. The array K is then filled with a value of 1 over the range $[i - 20; i + 20]$. This range usually encompasses the QRS complex, which includes the R peak.

Step 1.4: Skip the search for a new maximum deviation immediately after finding the deviation p at Step 1.2, as R peaks occur at regular intervals.

A visualization of knowledge integration and the ECG is shown in Figure 5.

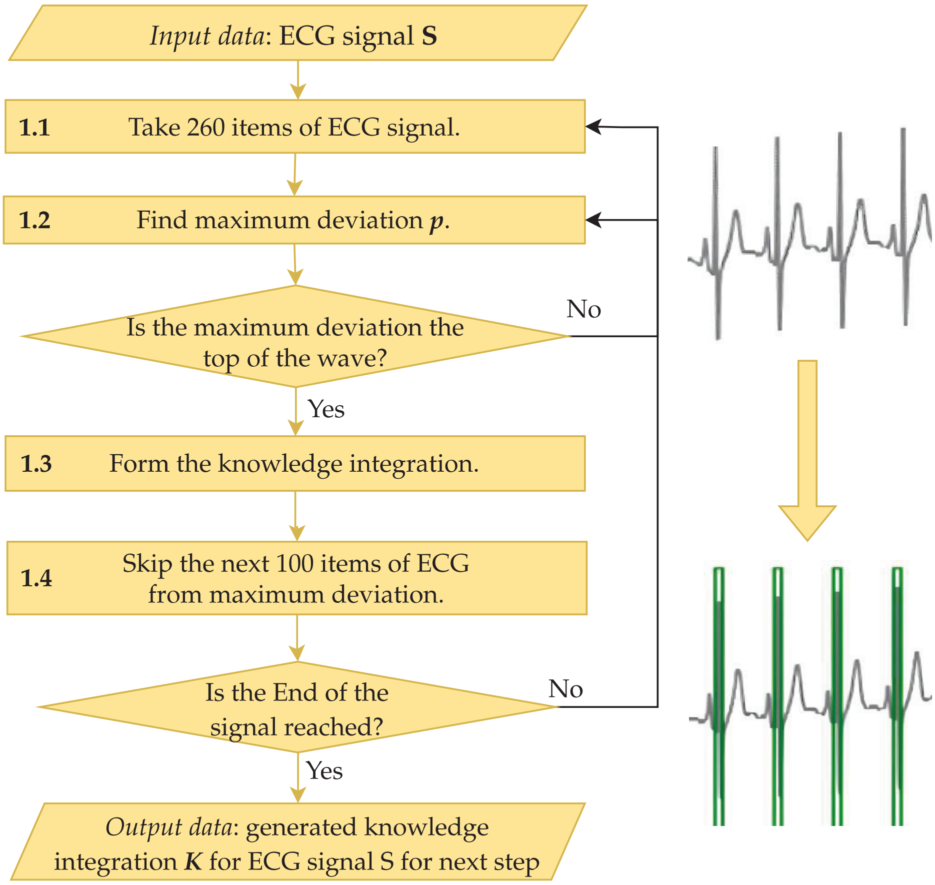

Input data: ECG signal **S**
1.1 Take 260 items of ECG signal.
1.2 Find maximum deviation ***p***.
Is the maximum deviation the top of the wave?
No
Yes
1.3 Form the knowledge integration.
1.4 Skip the next 100 items of ECG from maximum deviation.
Is the End of the signal reached?
No
Yes
Output data: generated knowledge integration ***K*** for ECG signal S for next step

Figure 4. This figure details Step 1 of the R peak detection method, focusing on integrating reference ECG knowledge. The process begins by analyzing 260 data points of the ECG to find the maximum deviation, representing the wave peak. If confirmed, knowledge integration is applied. The process skips 100 items after each peak to avoid redundancy until the end of the signal, generating a knowledge-integrated array K for further processing.

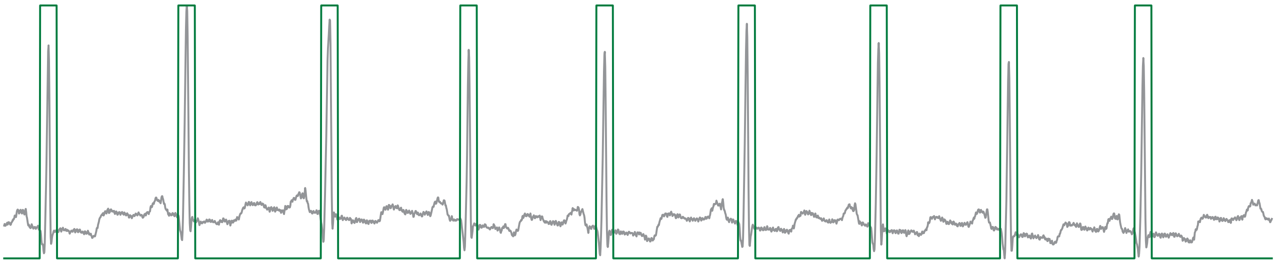


Figure 5. A schematic representation illustrating the integration of reference ECG knowledge into the current ECG signal. The grey curve represents the raw ECG waveform, highlighting the natural fluctuations in cardiac electrical activity. The green overlay marks the regions where knowledge integration is applied, focusing on the R-peaks.

Signals S and K are then fed into the DL model for R peak detection (Step 2). Based on the analysis conducted in Section 2, we propose using a CNN with an architecture described in Table A1 in Appendix A.

For loss measurement during network training, the Binary Cross Entropy Loss (BCELoss) function [28] is used. It should be noted that this function was chosen due to its resistance to class imbalance in the training dataset, which is relevant to our task. Moreover, BCELoss might be also essential for the task at hand, as the imbalance is substantial—signals with R peaks are far less common than those without them.

The output from the DL model needs to be represented as a data array of the same size as the input array, where the necessary labels for R peaks are placed in the corresponding positions of the input array.

Step 3 is designed to process the CNN output P, transforming it into indices corresponding to R peaks in the input signal. The scheme of Step 3 is shown in Figure 6.

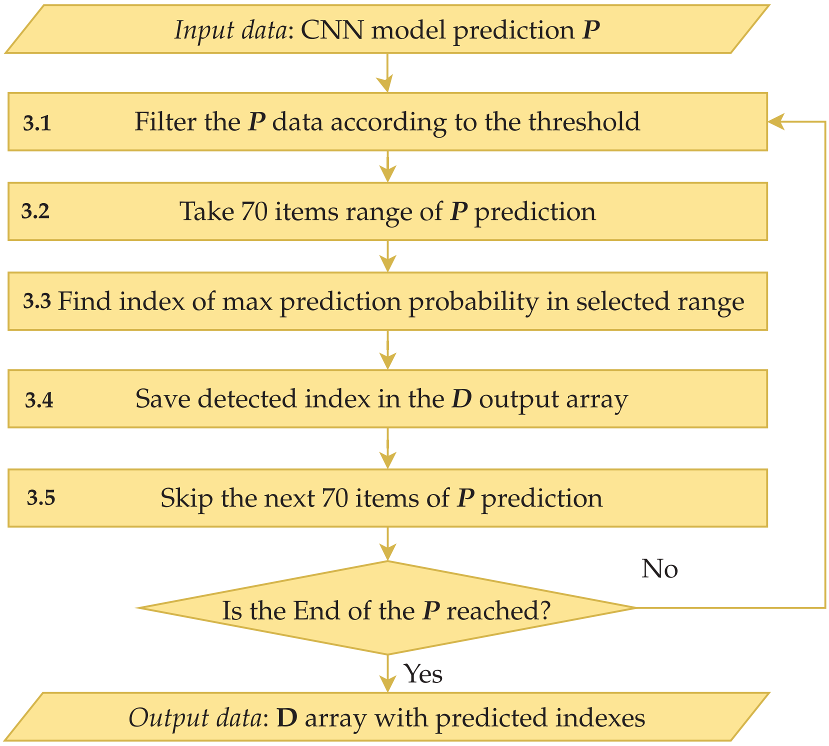

Input data: CNN model prediction ***P***
3.1 Filter the ***P*** data according to the threshold
3.2 Take 70 items range of ***P*** prediction
3.3 Find index of max prediction probability in selected range
3.4 Save detected index in the ***D*** output array
3.5 Skip the next 70 items of ***P*** prediction
Is the End of the ***P*** reached?
No
Yes
Output data: **D** array with predicted indexes

Figure 6. This figure illustrates the post-processing steps for CNN-predicted R peak identification. Starting with CNN predictions, the process filters data, identifies the maximum prediction in each range, and saves it in an output array D. It iterates through the signal to create a comprehensive index array of R peak positions.

The encoder-decoder CNN output from Step 2 is an array of the same size as the input ECG array.

Step 3.1 involves filtering the input data P based on a pre-determined threshold to ensure only relevant data points are considered. This threshold helps exclude less significant predictions and focus on deviations that are likely to indicate R peaks. Experimentally, this threshold was set to 0.1.

After filtering the data at Step 3.2, the algorithm searches for the next deviation, considered a possible R peak position. To accurately determine the R peak index, Steps 3.3–3.4 analyze a range of 70 consecutive prediction elements (equivalent to 175 ms), starting from the identified deviation. The element with the highest predicted probability in the chosen range is determined, and its index is stored in the output array D. This array accumulates the indices of all significant points detected throughout the process.

Step 3.5 skips the elements processed in previous steps to avoid re-processing these values. The algorithm continues until the end of the input data array P is reached.

Upon completion, the output array D contains a full list of indices corresponding to the R peaks of the ECG as determined by the CNN model.

3.2. Method for Classifying Arrhythmia Based on ECG

To improve existing approaches for ECG classification, particularly for arrhythmia pathologies, we propose a method represented schematically in Figure 7.

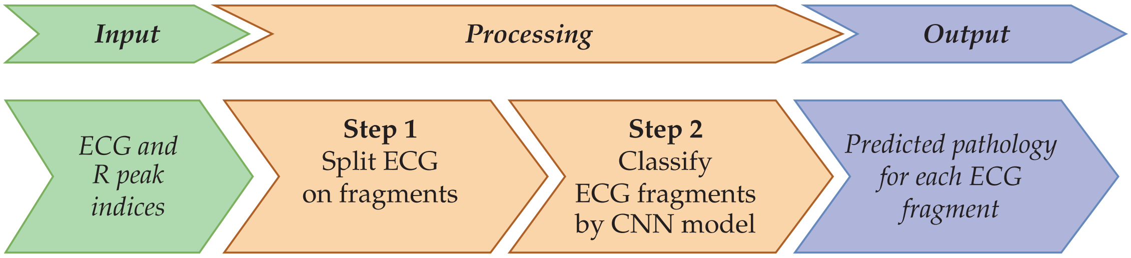


Figure 7. This figure presents an ECG classification method for arrhythmia detection, beginning with ECG and R peak indices. The process involves splitting the ECG into fragments and using a CNN model to classify each fragment, resulting in predicted pathology labels for individual ECG segments to support clinical diagnosis.

The results in our previous work [29] suggest that CNN models typically use a single cardio cycle as input, but this approach lacks sufficient context for accurate pathology detection. To address this limitation, we propose augmenting the input with neighboring cardio cycles, allowing the DL model to uncover additional hidden dependencies in the ECG data and enhance pathology identification.

Method overview is as follows:

Input Data: ECG signals and indices of R peaks identified previously.

Step 1: Prepare ECG input samples.

Step 2: Classify using an enhanced CNN model.

Output Data: ECG classified according to detected pathologies.

Below, we provide implementation details of the proposed method.

In Step 1, we preprocess the ECG by segmenting them into fragments of 700 samples. This length, determined empirically, includes three cardio cycles—the previous, current (central), and next R peaks—providing a broader temporal context for analysis.

In Step 2, we classify these samples using an improved CNN architecture. We modify the CNN presented in [20], which achieved an accuracy of 99.43% but did not classify all pathologies in the dataset. Our enhanced CNN, detailed in Table A2 of Appendix A, accommodates the new input format and an expanded set of pathologies.

To enable the CNN to identify more distinctive features and handle a larger number of classes, we add an extra convolutional layer. Recognizing that this increases computational complexity, we also incorporate Batch Normalization layers [30] after each convolutional layer and the first linear layer. Batch Normalization stabilizes the training process by normalizing activations within each batch, preventing sudden spikes or drops in activation levels.

We also include a Dropout layer after the first linear layer to improve generalization and prevent overfitting. This layer randomly deactivates a fraction of neurons during training, enhancing the model's robustness.

Given these architectural changes, we perform hyperparameter optimization on the CNN layers, adjusting parameters such as kernel size, stride, padding, and dropout probability. The optimized hyperparameters are listed in Table A2 of Appendix A.

Applying our proposed CNN to the ECG fragments results in an array containing the classification of each fragment's pathology, providing a more accurate and comprehensive analysis of the ECG data.

3.3. Method for Interpreting Classification Results

Considering the sensitivity of the subject, i.e., medicine, and given that the proposed DL-based solutions are inherently nontransparent (i.e., a "black box" in terms of the mechanism and parameters used to make decisions), there is a need for interpreting the decisions in a form understandable to the end user (doctor). The method is described in detail below.

3.3.1. General Idea of the Method

Our approach aims to present the decisions from the previous method using features that doctors rely on when diagnosing ECG pathologies. These are specific, observable features in the cardio cycles that help doctors agree or disagree with the DL model's decision.

While traditional ECG classification methods using ML involved feature vectors based on these clinical features, their results were less impressive than those of DL models. In this study, we identify these features not for classification but to help doctors understand the DL decisions. By visualizing these features, we make the model's decisions more accessible to clinicians.

Doctors consider various indicators when diagnosing ECG pathologies; each pathology has predefined features that may or may not all be present, leaving the final judgment to the doctor. To illustrate our method, we focus on one specific feature associated with a particular pathology, for example, the "presence or absence of the P-wave (P peak)" in a cardio cycle, which relates to premature ventricular contractions (PVC).

Figure 8 shows a schematic diagram for interpreting one feature of the proposed method; similar diagrams apply to other features.

The main steps of the proposed method for interpreting classification results are presented below.

Input Data: The cardio cycle signal as presented to the CNN classifier and the pathology class determined by the classifier.

Step 1: Empirically determine the zone of interest in the signal where the pathology feature may be present. Use this signal segment as a feature vector to explain the selected pathological feature.

Step 2: Choose a method to inform the doctor about the presence or absence of the feature in the signal fragment. This involves sequentially analyzing information through the following steps until the classification result can be interpreted:

Step 2.1: Formulate the interpretation using formulas or statistical indicators understandable to the doctor.

Step 2.2: Generate an interpretation by visually comparing the signal fragment with similar pieces from the training set, annotated as either the pathology in question or normal/other pathologies.

Step 2.3: Use visual analytics tools like principal component analysis (PCA) [31], multidimensional scaling (MDS) [32], or t-distributed stochastic neighbor embedding (t-SNE) [33] to form the interpretation.

Step 2.4: Employ ML models to aid interpretation.

Step 2.5: Utilize DL models for interpretation.

Step 2.6: Apply other methods that may complement the proposed approach.

Output Data: A conclusion regarding the presence or absence of the considered feature in the classified ECG.

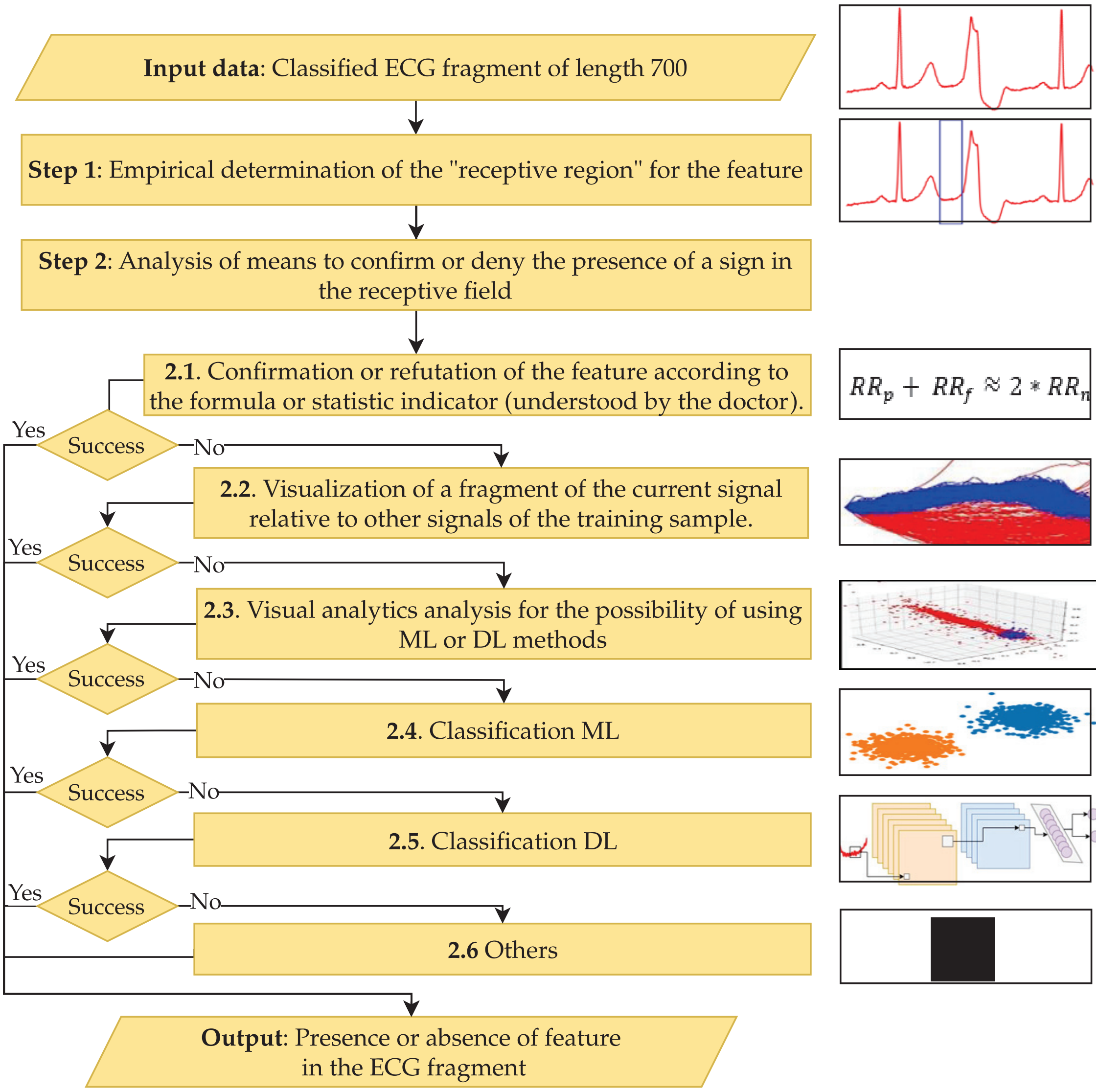
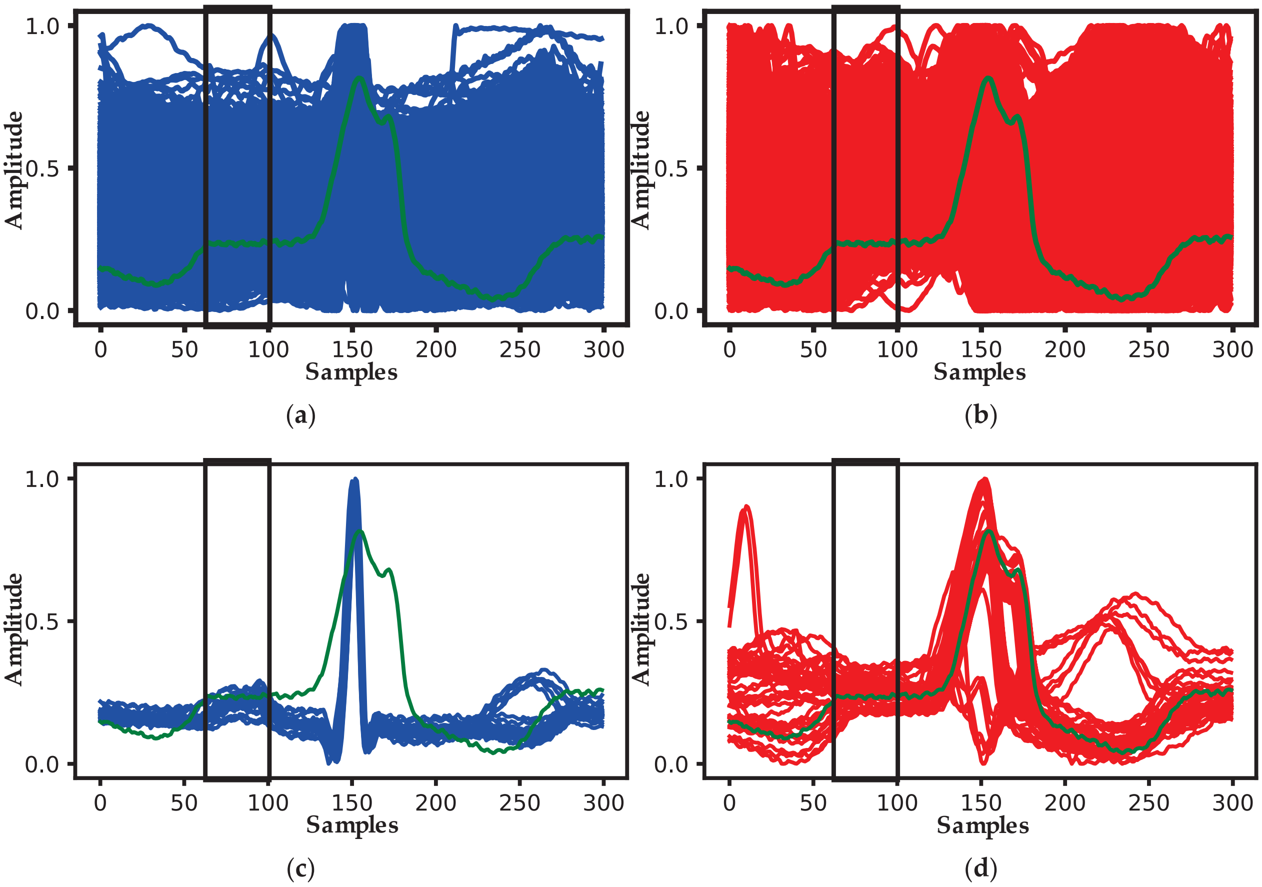

Input data: Classified ECG fragment of length 700
Step 1: Empirical determination of the "receptive region" for the feature
Step 2: Analysis of means to confirm or deny the presence of a sign in the receptive field
2.1. Confirmation or refutation of the feature according to the formula or statistic indicator (understood by the doctor).
$RR_p + RR_f \approx 2 * RR_n$
Yes
Success
No
2.2. Visualization of a fragment of the current signal relative to other signals of the training sample.
Yes
Success
No
2.3. Visual analytics analysis for the possibility of using ML or DL methods
Yes
Success
No
2.4. Classification ML
Yes
Success
No
2.5. Classification DL
Yes
Success
No
2.6 Others
Output: Presence or absence of feature in the ECG fragment

Figure 8. This figure illustrates a step-by-step method for detecting specific features in a classified ECG fragment. Starting with a 700-length ECG segment, it empirically identifies the receptive region, analyzes the presence of features, and applies methods like formula-based verification, visualization, and ML or DL classification. Each step is designed to confirm or deny feature presence, with outcomes supporting clinical interpretation of ECG data.

3.3.2. Mechanisms for Detecting Features That Aid Doctors in Decision-Making

We outline mechanisms used to detect ECG features, according to Steps 2.1–2.5 of our proposed interpretation method, which are employed in Step 2 to identify features associated with various pathologies (see Figure 9).

To detect features visible within a cardio cycle (see Figure 1), it is essential to identify its main elements: peaks, intervals, and periods. Since the R peak is already identified from the initial ECG processing stage, we use Neurokit2 v0.2.7 package (free and open-source software under MIT License) [34] to locate other elements of the cardio cycle. Some features are derived using statistical indicators or formulas (Step 2.1). For example, the "Presence of a Compensatory Pause" can be calculated using specific formulas, and the presence or absence of the P peak can be verified using Neurokit2.



(**a**)

(**b**)

(**c**)

(**d**)

Figure 9. Training samples for two ECG classes, illustrating separation clarity within a designated zone of interest. Subfigures (**a**,**b**) show cases with unclear separation, while (**c**,**d**) display distinct separation patterns in the zone. The cardiac cycle, highlighted in green, is overlaid with a black rectangle to emphasize the zone of interest and represent the relevant signal fragment.

Another mechanism involves visualizing cardio cycles from the training set divided into two class groups (Step 2.2) alongside the classified cardio cycle under interpretation. In Figure 9a, one group represents "Normal" ECGs, while the other includes ECGs with P peak abnormalities (e.g., PVC).

In Figure 9a, the red and blue areas overlap completely, making simple visual comparison ineffective. Therefore, we proceed to the next steps to find a resolving mechanism. In contrast, Figure 9b shows no overlap, allowing us to confirm that the feature in the analyzed cardio cycle (green graph) corresponds to the identified class.

If visual analysis reveals significant overlap of features in the zone of interest, we advance to Step 2.3. Here, we represent signal fragments from the zone of interest (highlighted in Figure 9) as vectors and input them into visual analytics tools like PCA, MDS, or t-SNE. The resulting representation is shown in Figure 10.

In Figure 10, we observe specific groupings of ECG data from the zone of interest, but clear separation between groups is lacking. If we do observe groupings with separation—possibly among more than two groups—we can apply ML methods to these vectors to build a classifier (Step 2.4).

When visual analytics methods produce overlapping groups or ML methods fail to provide a solution, we turn to Step 2.5, applying DL methods to interpret the presence of the feature in the ECG. Similar to ML, the input for DL methods is the ECG segment from the defined zone of interest.

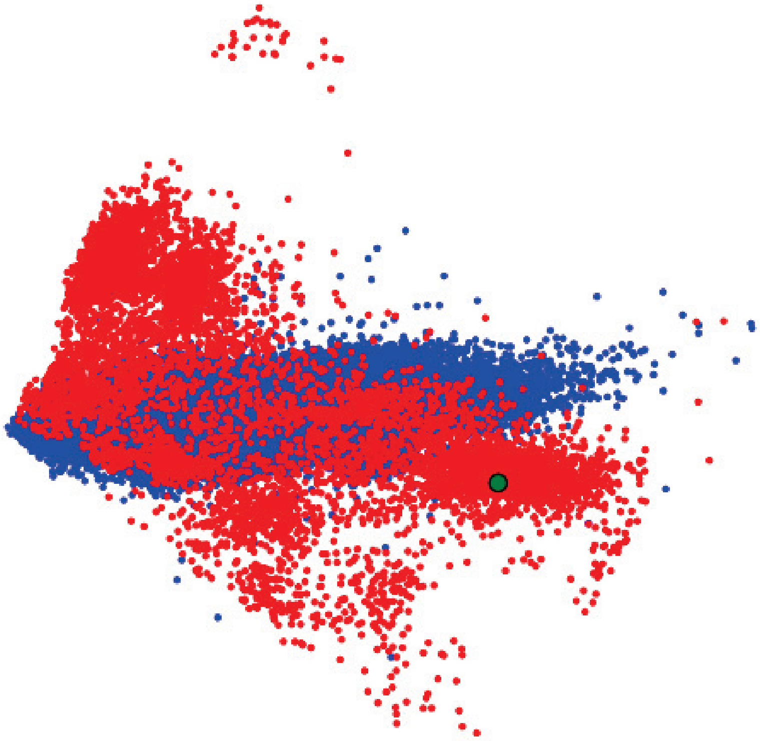


Figure 10. The application of PCA on ECG fragments within the zone of interest. Red and blue clusters represent two distinct classes, highlighting areas of overlap and separation; the green dot represents the target ECG.

For classification, we prepared a CNN model for a binary classification task. The fine-tuned parameters of this model are detailed in Table A3 in Appendix A. Finally, Step 2.6 provides an opportunity to expand the proposed approach with other methods for interpreting ECG features.

3.3.3. List of Features Used by Doctors for Decision-Making

The proposed mechanism for feature detection in ECG is recommended for the following list of features defined by doctors.

1. For a normal ECG (i.e., cardiocyte), the following features are considered:
   - Presence of all cardio cycle elements (peaks and intervals).
   - QRS complex is not widened or deformed.
   - P peak precedes each QRST complex.
   - Presence of a normal PQ interval.
2. For PVC or “Ventricular Extrasystole”, the following features are indicative:
   - Absence of P peak.

- Widened QRST complex.
- Deformed QRST complex deformation refers to a change in the shape of the QRST complex. Right ventricular extrasystole appears as left bundle branch block (LBBB) in lead V1, while left ventricular extrasystole appears as "Right Bundle Branch Block" (RBBB).
- Presence of a complete compensatory pause. This is the interval between two consecutive ventricular complexes of the sinus rhythm, between which an extrasystole occurs, equal to double the RR interval of the sinus rhythm.

3. RBBB is characterized by:
   - Deep, wide S-waves in standard and left chest (V5–V6) leads.
   - Widened and deformed QRS complex with an rSR′ pattern or in chest leads (V1–V2) resembling the letter "M".
   - Depression of the ST segment. ST segment depression is a decrease of this segment below the isoelectric line.
   - Inverted T-wave in right chest leads. An inversion is opposite to the normal polarization of the wave.
   - Prolonged intra-QRS deflection time (IQRDT) in right chest leads. IQRDT reflects the time from the beginning of the QRS complex (Q or R wave) to the maximum deviation of the QRS complex (usually the R peak). Normally, IQRDT $\leq$ 0.04 s.
4. LBBB is characterized by the following:
   - Deformed and widened QRS complex with a duration exceeding 120 ms.
   - Deep, wide S-waves in the right chest lead.
   - Discordant changes in the ST-T complex relative to the QRS complex. Discordant changes in ST-T include depression or elevation of the ST segment in the direction opposite to the main vector (R or S wave).
   - Prolonged IQRDT in left chest leads.
5. Fusion of Ventricular Beats is characterized by the following:
   - Cardio cycle with characteristic features of a ventricular extrasystole.
   - Absence of a compensatory pause.

The proposed mechanisms for detecting features and the list of features allow for presenting the obtained classification result (assignment to a specific pathology class) in a form understandable to the doctor.

3.4. Evaluation Metrics for DL Models in Medical Systems

In this study, we utilized several essential metrics to assess the performance of our DL models in medical applications, covering both binary and multiclass classification tasks. Our findings are consistent with existing research on model evaluation in medical AI, as highlighted in Rainio et al. [35].

We employed confusion matrices to identify classification errors and compute metrics such as accuracy, precision, recall, and F1-score. This approach provided a nuanced view of the models' capabilities, highlighting areas of correct classifications (true positives and true negatives) and misclassifications (false positives and false negatives). These insights were crucial for understanding the models' overall performance.

While accuracy was measured, it offered limited insight into dataset imbalances. To overcome this, we calculated precision to assess the proportion of correctly predicted positives and recall (sensitivity) to evaluate the models' ability to detect actual positive cases. The F1-score, which balances precision and recall, proved particularly valuable in addressing uneven class distributions, delivering a comprehensive evaluation of the models' classification performance.

Additionally, we applied advanced metrics such as Cohen's Kappa coefficient and Area Under the Curve (AUC) with Receiver Operating Characteristic (ROC) curves. Cohen's Kappa measured the agreement between models beyond chance, while AUC-ROC illustrated the models' proficiency in distinguishing between positive and negative cases across various thresholds. These metrics provided deeper insights into the reliability and discriminative power of our DL models.

3.5. Datasets

For training the CNN model, the following datasets were employed:

- MIT-BIH Arrhythmia Database (MIT-BIH) [36]: The most used dataset for arrhythmia classification in ECG using ML and DL methods. It was created through the collaboration between Beth Israel Hospital and MIT and became the first publicly available set of test materials for evaluating arrhythmia detectors. It contains 48 ECG recordings, each approximately 30 min long, collected during clinical studies. The signal frequency is 360 Hz, and each ECG recording includes annotations indicating the occurrence of specific pathologies related to arrhythmia.
- QT Database (QT) [37]: Developed for evaluating algorithms that detect ECG segment boundaries. It includes 105 two-channel Holter ECG recordings of 15 min each. Annotations mark peaks and boundaries of the QRS complex, P, T, and U waves (if present).
- China Physiological Signal Challenge-2020 (CPSC-2020) [38]: Created for the 3rd China Physiological Signal Challenge 2020, aimed at designing algorithms to detect premature ventricular and supraventricular contractions. Signals were collected with a portable ECG device at a sampling rate of 400 Hz. The dataset contains 10 single-lead ECG recordings collected from patients with heart arrhythmia. Each recording lasts approximately 24 h.
- University of Glasgow Database (UoG) [39]: A high-precision database from the University of Glasgow that includes ECGs annotated with R peaks, recorded under realistic conditions from 25 participants. ECG recordings were performed for over two minutes while participants performed five different activities: sitting a math test on a tablet, walking on a treadmill, running on a treadmill, and using a hand-cycle. The sampling rate was 250 Hz.

4. Results and Discussion

4.1. R Peak Identification

The above datasets were preprocessed to address task-specific requirements, primarily removing samples with poorly annotated R peaks. Such samples would hinder accurate training and testing of the neural network. From the MIT-BIH dataset, signals with inaccurate annotations (e.g., with identifiers 108 and 207) were excluded.

Due to the varying sampling rates of the signals in the datasets mentioned above, further transformations were performed to ensure all signals had a uniform sampling rate of 400 Hz. The signals were segmented into fragments of 8000 samples to be used as input for the neural network. Fragments obtained from datasets 1–3 were split into training and test sets in an 80/20 ratio.

The UoG dataset was used to create an independent test set. This set included signal fragments recorded in lead II during activities such as sitting, doing math, and walking. Table 1 provides the distribution of ECG fragments in the training and test sets.

Table 1. The distribution of ECG fragments across different datasets used in the study. Training data includes signals from MIT-BIH, QT, and CPSC-2020 databases, while testing data also includes MIT-BIH, QT, CPSC-2020, and a unique test set from the UoG ECG database.

Sample Title	Database Title	The Number of Fragments
Training data	MIT-BIH	3312
	QT	2484
	CPSC 2020	33,195
Testing data	MIT-BIH	828
	QT	712
	CPSC 2020	8299
Unique test data	ECG Database—UoG	438

We used 80% of the MIT-BIH Arrhythmia dataset, along with the QT and CPSC-2020 databases for training. From the training data, 10% was reserved for validation. Training was conducted in two stages using the Adam optimizer [40]. The first stage ran for 45 epochs with a learning rate of 0.001, achieving a loss of 0.000821. The second stage continued for 15 epochs with a reduced learning rate of 0.0001, resulting in a loss of 0.000580. The total training time was 82 min.

To evaluate classification quality (see Section 3.4), we used seven different random splits of the data into training and testing sets, derived from the MIT-BIH Arrhythmia, QT, and CPSC-2020 databases. Statistical metrics for each dataset are presented in Tables A4–A6 in Appendix B. We also tested an independent test set from the UoG database, with results shown in Table A7.

Accuracy across all random splits was consistently high for both training and test sets, averaging around 99.9%. Minimal standard deviations indicate stable performance regardless of data splits, suggesting good generalization and accurate classification. With accuracy near 100%, the model is highly effective for this task.

Precision remained high on both training and test sets, averaging 99.8–99.9% across all datasets. Low standard deviations confirm the model's reliability in accurately identifying positives with few false alarms, which is essential when false positives are costly. This indicates the model is both accurate and selective in its positive classifications.

Recall for test sets consistently exceeded 98%, averaging 99.1–99.2%, showing the model effectively captures nearly all relevant cases. Small standard deviations highlight stability and consistent true positive identification across different splits.

F1-scores remained high, averaging 98.8–98.9% on test sets. Low standard deviations indicate a stable balance between precision and recall across random splits. This consistency suggests the model maintains performance with diverse data, making it a reliable tool for the task.

Table 2 compares our model's statistical metrics with those known approaches discussed in Section 2 using the same test datasets.

Table 2. A comparison of the proposed model's performance on accuracy, precision, recall, and F1-score across multiple ECG databases (MIT, QT, CPSC-2020, UoG) against established approaches. Bold values indicate the highest scores.

Database	Approach	Accuracy	Precision	Recall	F1-Score
MIT	Zahid et al. [12]	**0.9999**	0.9905	0.9858	**0.9893**
	NeuroKit2 [34]	0.9997	0.9644	0.9340	0.9490
	Rodrigues et al. [41]	0.9992	0.8322	0.9491	0.8868
	Koka et al. [42]	0.9992	0.8938	0.8699	0.8817
	Our	**0.9999**	**0.9921**	**0.9872**	0.9890
QT	Zahid et al. [12]	**0.9999**	0.9789	0.9778	0.9783
	NeuroKit2 [34]	0.9997	0.9655	0.9410	0.9531
	Rodrigues et al. [41]	0.9991	0.7824	0.9427	0.8551
	Koka et al. [42]	0.9993	0.8866	0.8767	0.8816
	Our	**0.9999**	**0.9803**	**0.9818**	**0.9819**
CPSC-2020	Zahid et al. [12]	**0.9999**	0.9855	0.9927	0.9891
	NeuroKit2 [34]	0.9997	0.9514	0.9514	0.9514
	Rodrigues et al. [41]	0.9989	0.7763	0.9212	0.8426
	Koka et al. [42]	0.9995	0.9232	0.8972	0.9100
	Our	**0.9999**	**0.9862**	**0.9943**	**0.9897**
UoG	Zahid et al. [12]	0.9995	0.9838	0.8666	0.9215
	NeuroKit2 [34]	**0.9998**	**0.9932**	0.9596	**0.9761**
	Rodrigues et al. [41]	0.9996	0.9083	**0.9990**	0.9515
	Koka et al. [42]	0.9994	0.9194	0.8968	0.9080
	Our	0.9996	0.9831	0.9010	0.9239

Overall, the statistical metrics indicate that the proposed CNN model provides reliable and accurate classification. Minimal deviations across different splits suggest the model's performance does not depend on specific data, highlighting its stability and generalizability.

Comparing our data to the previous results, we conclude the following:

- Accuracy remains consistently high, indicating good generalization.
- Precision decreased slightly on the independent test set, suggesting the model encounters more challenging classification tasks.
- Recall decreased, indicating reduced sensitivity to true positives in this dataset, possibly due to ECG characteristics not present or rare in the training set.
- F1-score decreased, reflecting the impact of lower recall on overall model performance.

4.2. Pathology Classification

The same 80% of the MIT-BIH Arrhythmia database as in the previous step was used to train the network. As in the previous step, 10% of the training dataset was set aside for validation to monitor the model for overfitting during training.

Based on the annotations in the MIT-BIH Arrhythmia database, the following classes/pathologies were selected for classification:

- Normal beat.
- PVC.
- Paced beat.
- RBB beat.
- LBBB beat.
- Atrial premature beat.
- Fusion of ventricular and normal beat.
- Fusion of paced and normal beat.

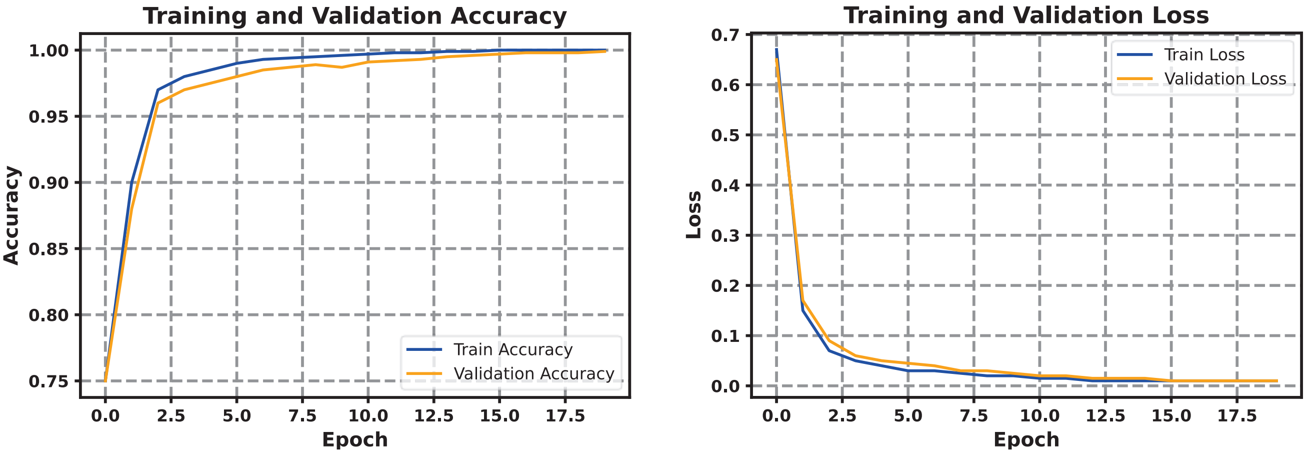
- Others.

CNN model training was conducted in two stages using the Adam optimizer, following the idea of parallel neural network computations, proposed in [43]. In the first stage, training was performed with a learning rate of 0.001, resulting in a loss value of 0.024269–0.019391. In the second stage, training continued at a learning rate of 0.0001, resulting in a loss value of 0.00746–0.004003. A total of 18 epochs were conducted.

Figure 11 presents examples of the loss curves (Figure 11a) and the accuracy curves (Figure 11b) for both training and validation sets.

Figure 11a illustrates the loss function plot, with epochs on the *x*-axis and loss values on the *y*-axis. The blue curve represents training loss, while the orange curve depicts validation loss. Both curves decrease rapidly during the initial epochs, indicating effective training. They eventually stabilize at low loss values and nearly overlap, suggesting consistent performance and successful avoidance of overfitting.

In Figure 11b, the accuracy over epochs is shown, with the blue curve for training accuracy and the orange curve for validation accuracy. Accuracy increases swiftly in the early epochs, surpassing 95% within the first few iterations. Both curves then level off near 100%, reflecting high classification quality and strong generalization.



(**a**) (**b**)

Figure 11. Training and validation curves for accuracy (**a**) and loss (**b**) over 18 epochs. The rapid convergence of accuracy and reduction in loss indicate effective training with minimal overfitting, demonstrating the CNN model's stability and generalizability for ECG classification tasks.

Similar to the R peak detection evaluation, we assessed the pathology classification method using seven randomly generated training and testing datasets. Table A8 in Appendix B presents the average statistical metrics and their deviations for both sets. Training accuracy ranged from 99.90% to 99.92%, while test accuracy ranged from 99.08% to 99.44%.

The model exhibited excellent classification performance on the training set, with nearly perfect Precision, Recall, and F1-score across all classes. On the test set, while performance remained strong, there was a slight decline in these metrics. The most significant drops occurred in classes 7 and 9, indicating these classes are more challenging to classify in new data. This may be due to their smaller representation in the dataset, limiting the CNN's ability to fully learn their characteristics. Despite this, the Recall for the test set stayed above 80%, confirming the classifier's effectiveness.

Low standard deviations (under 5%) in the training set indicate that the model's predictions are consistent and stable. In the test set, standard deviations increased slightly, as expected when encountering unfamiliar data, but the mean deviation remained below 5%. The higher standard deviations for classes 7 and 9 suggest some inconsistency in the model's performance on these less familiar classes.

Figure 12 displays ROC curves to evaluate the quality for each class. Most ROC curves are near the top-left corner, confirming the model's high efficiency. From Figure 12, high AUC values for all classes—mostly equal to 1.00—demonstrate excellent discrimination between positive and negative examples. Even for classes where the AUC is slightly lower (e.g., class 9), performance remains strong.

We further evaluated the model using the "One-vs-One" approach for ROC curves, as shown in Figure 13.

In most cases, the ROC curves and AUC values are nearly ideal. Most combinations have ROC curves close to the top-left corner, indicating high efficiency. AUC values ranging from 0.99 to 1.00 confirm the model's strong ability to distinguish between classes. Notably, the greatest deviation from the top-left corner, with an AUC between 0.99 and 1.00, occurs in combinations involving classes 1 and 9 (Figure 13a). This suggests the classifier is slightly better at identifying class 1 as positive compared to class 9, reflecting the slight difference in AUC. This discrepancy may stem from differing representations of these classes in the training and test datasets.

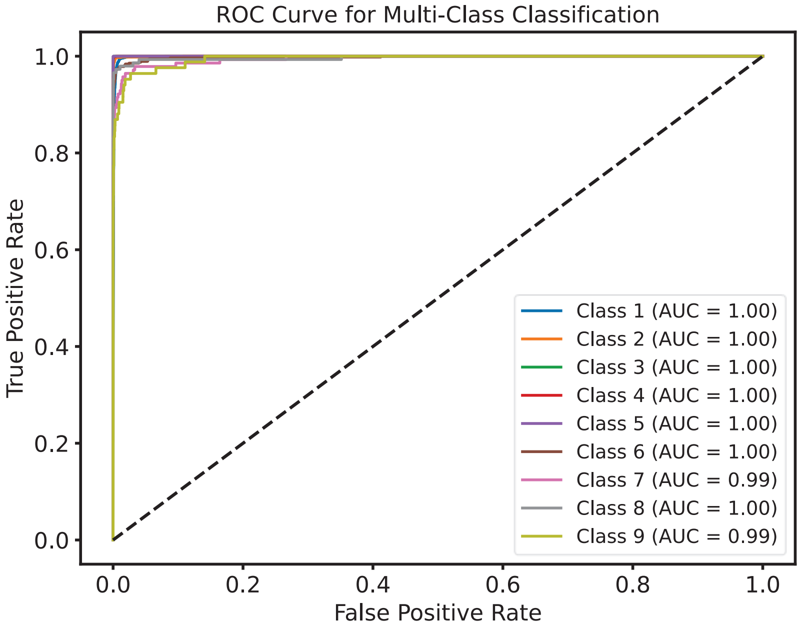


Figure 12. ROC curves for a multi-class classification of ECG data, showing near-perfect AUC values (mostly 1.00), indicating high model performance. Minor deviations in classes 7 and 9 suggest slight inconsistencies in distinguishing these classes, reflecting model robustness overall.

We compared our method with state-of-the-art approaches and summarized the statistical metrics. The proposed method achieved an average test accuracy of 99.26%. Table 3 presents the macro and weighted average statistical metrics.

Table 3. Average metrics for nine-class ECG classification.

Metric	Precision	Recall	F1-Score
Macro	0.97	0.95	0.96
Weighted	0.99	0.99	0.99

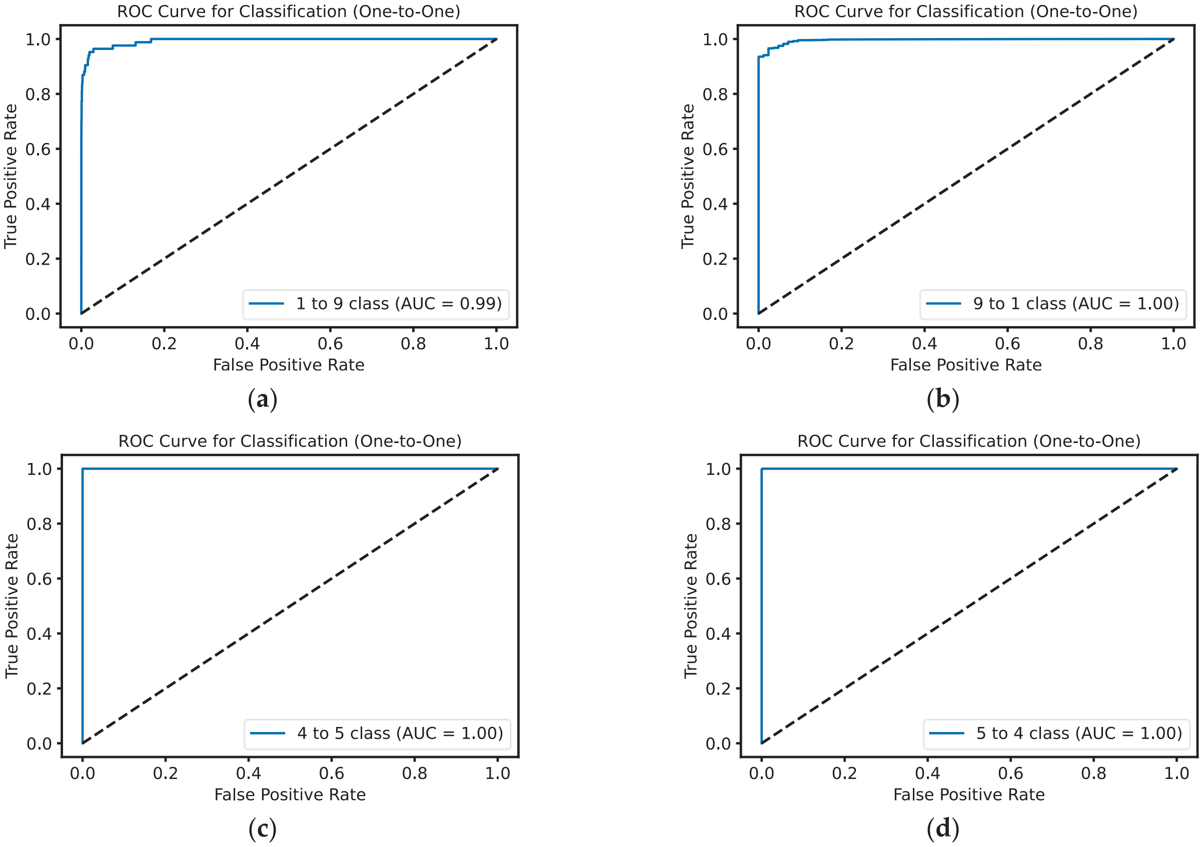


Figure 13. This figure presents One-vs-One ROC curves for ECG classification, demonstrating high accuracy in class differentiation: (**a**) shows normal vs. others (AUC 0.99), (**b**) class 9 vs. 1 (AUC 1.00), (**c**) LBBB vs. RBBB (AUC 1.00), and (**d**) classes 5 vs. 4 (AUC 1.00).

Among all the classes supported during classification, the "Others" class is the least stable. This could be explained by the fact that cardio cycles of this class were significantly underrepresented in the dataset.

Additionally, the "Others" class is characterized by greater variability, which further emphasizes the issue of the small number of signals for this class. This affects the Macro metric since all classes are equally weighted regardless of their representation in the dataset. If the "Others" class is excluded, the statistical metrics take on the values presented in Table 4.

Table 4. Average metrics for ECG classification when excluding the "Others" class due to its underrepresentation and variability.

Metric	Precision	Recall	F1-Score
Macro	0.97	0.95	0.96
Weighted	0.99	0.99	0.99

When comparing the statistical results with modern approaches, it is worth noting that all approaches can be divided into two types:

- Approaches that group cardio cycles into categories and classify them based on their group membership.
- Approaches that classify each cardio cycle into a specific atomic pathology class.

The first type is based on recommendations from the Association for the Advancement of Medical Instrumentation (AAMI) [44], which involves grouping cardio cycle classes. Examples of such groups include non-ectopic beat, supraventricular ectopic beat, ventricular ectopic beat, fusion, and unknown. Grouping provides an advantage during network training, as it helps to avoid data scarcity in individual classes, leading to better classification. However, this classification approach may not always meet a clinician's needs, as knowing the specific pathology of a cardio cycle, not just the group it belongs to, is essential for proper diagnosis.

Since the proposed method is not based on AAMI, it is impossible to provide a completely equivalent comparison of the statistical metrics of AAMI-based methods. Table 5 presents a comparison of the statistical metrics of ECG classification methods based on AAMI with the method proposed in this work.

Despite focusing on classifying nine ECG classes, the proposed method generally demonstrates better results compared to methods that classify cardio cycle signals into group memberships.

Table 6 presents a comparison of the statistical results of the proposed method with the results of methods belonging to the second category, which classify cardio cycles into specific atomic pathology classes.

Table 5. A comparison of the proposed ECG classification method with AAMI-based classification approaches, highlighting superior performance across all metrics. Bold color indicates the highest value of each metric.

Approach	Number of Groups	Accuracy	Precision	Recall	F1-Score
Hassan et al. [18] Our	5	0.980 **0.993**	0.920 **0.970**	0.910 **0.940**	0.915 **0.965**
Xu et al. [20] Our	5	**0.998** 0.993	**0.980** 0.960	**0.940** **0.940**	**0.961** 0.950
Ahmed et al. [45] Our	4	0.990 **0.993**	0.930 **0.960**	**0.940** **0.940**	0.935 **0.950**
Kumar et al. [46] Our	5	0.987 **0.993**	**0.989** 0.960	0.939 **0.940**	**0.963** 0.950
Mahmud et al. [47] (Signal) Our	6	0.940 **0.993**	0.950 **0.960**	0.900 **0.940**	0.920 **0.950**
Mahmud et al. [47] (Image) Our	6	0.930 **0.993**	0.930 **0.960**	0.930 **0.940**	0.930 **0.950**

Since each method may support classification for a different set of cardio cycle classes, a comparison of the average classification metrics is not equivalent. Therefore, in Table 6, along with the overall average metrics for each method, the metrics for classifying only the common classes between the studied and proposed methods are also provided.

Overall, we may conclude that the proposed improvements to the ECG classification method allow for the highly accurate classification of nine ECG classes.

Table 6. A comparison of the proposed ECG classification method with existing approaches for atomic pathology classification, showing consistently high metrics. Bold color indicates the highest value of each metric.

Approach	Number of Classes	Mutual Classes	Precision	Recall	F1-Score
Liu et al. [19]	4	4	–	0.993	–
Our			**0.993**	**0.995**	**0.994**
Degirmenci et al. [21]	5	5	**0.995**	**0.997**	0.992
Our			0.993	**0.997**	**0.995**
Rohmantri et al. [22]	8	7	**0.973**	0.893	0.927
Our			0.970	**0.950**	**0.960**
Ullah et al. [23]	8	6	0.985	0.977	0.981
Our			**0.988**	**0.984**	**0.986**
Yang et al. [48]	6	6	**0.991**	–	0.966
Our			0.988	**0.984**	**0.986**

4.3. Clinical Trials

For the clinical studies, we obtained 10 ECG fragments from real patient medical histories. These ECGs were provided anonymously, with all metadata excluded to ensure patient confidentiality. The ECGs were presented in raster format, and an example of such a signal is shown in Figure 14.

Along with the ECG images, an annotation indicating the presence of pathologies in the cardio cycles was provided. According to the annotation, the ECGs obtained contain 59 normal cardio cycles and 17 cardio cycles with PVC.

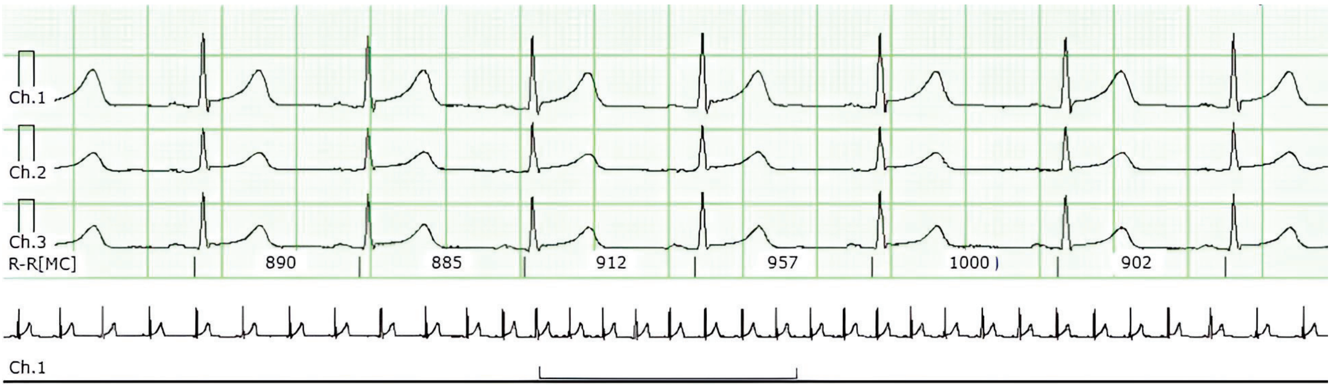


Figure 14. A multi-channel ECG recording used in clinical trials, with annotations indicating normal cycles and those with PVC.

To extract the ECG from the images, preprocessing was conducted, including the following:

- Removal of textual information from the input image.
- Splitting the input image into separate fragments representing the ECG for each channel individually (Figure 15).

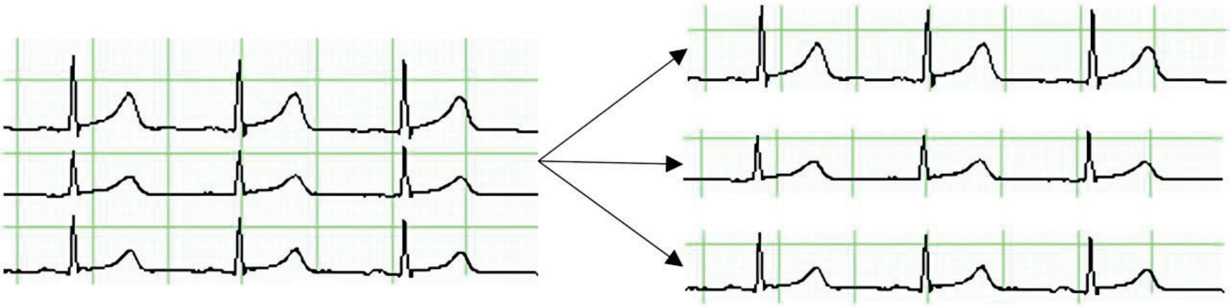


Figure 15. This figure illustrates the process of splitting a multi-channel ECG image into separate channel-specific images. Each channel is isolated to enable focused analysis, facilitating the detection of specific patterns and anomalies within each individual ECG trace, such as PVC.

The process of extracting ECG from an image began with converting the image to a grayscale. Then, using OpenCV v4.9.0 (free and open-source software under Apache 2 License) [49], the image was converted to a binary format, where all pixels were either black or white. This simplified detecting the ECG line, as it was now represented by black pixels on a white background. The vertical coordinates of the black pixels were collected to reconstruct the signal line. These coordinates were transformed into a 1D array representing the ECG amplitudes. Thus, the ECG image was converted to a digital format for further processing and classification using the proposed methods.

Initially, clinical trials were conducted for the R peak detection method, and Cohen's Kappa coefficient was calculated. The obtained value was 0.940, which falls within the range of 0.81–0.99, confirming almost perfect agreement between the results of the proposed method and the expert who annotated the signals.

Next, clinical trials were conducted for the pathology classification method in ECG. The clinical dataset comprised ECGs with three possible pathologies (three classes): "Normal", RBBB, and PVC. The classification model tested on the clinical dataset achieved a Cohen's Kappa coefficient of 0.8905, placing it within the 0.81–0.99 range that indicates almost perfect agreement between the model's predictions and the expert's annotations. Based on these results, a confusion matrix was generated to illustrate the distribution of classifications obtained during the trials, as shown in Figure 16.

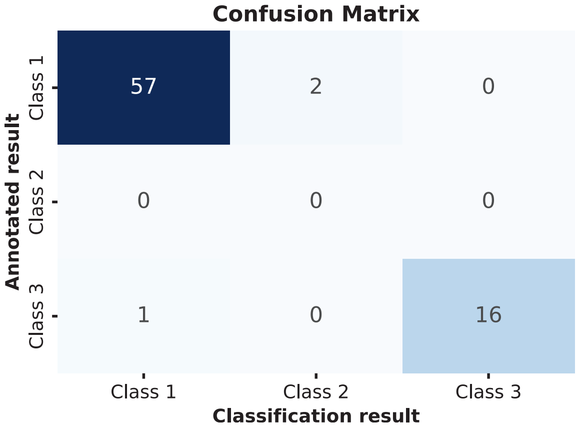


Figure 16. This confusion matrix summarizes the performance of the pathology classification model in clinical trials. Two Class 1 ("Normal") cycles were misclassified as Class 3 (PVC), and one Class 3 cycle was misclassified as Class 1, indicating high but not flawless consistency. No cases were predicted as Class 2 (RBBB) because the clinical dataset contained no instances meeting its diagnostic criteria, leaving the model without examples to learn from or identify for that category.

In Figure 16, one cardio cycle was misclassified as "Normal" and two cardio cycles were misclassified. It should also be noted that the dataset used in these clinical trials contained no signal samples for RBBB, so the model's misclassification of any cycle as RBBB was unsurprising. Apart from the above exceptions, all other cycles were classified correctly.

4.4. Interpretation of Classification Results by Medical Features

The interpretation of the results obtained by the proposed method is performed for each cardio cycle.

Below are examples of the interpretation of the decisions made.

Figure 17a shows an example of an input ECG classified as "Normal ECG".

Each provided feature, according to its criteria, is visually confirmed by highlighting the corresponding peak or signal fragment. In Figure 17b, the presence of key peaks in the ECG cardio cycle is confirmed. In particular, Figure 17c shows the confirmation of the presence of PQ and ST segments in the cardio cycle, and Figure 17d highlights a signal fragment confirming the feature "QRS non-extended and undeformed".

In Figure 17b, a signal fragment confirming the presence of ST segment depression is highlighted. Specifically, in Figure 17c, a signal fragment confirming the feature "QRS complex extended and deformed" is highlighted.

Despite the presence of at least five features that a doctor uses to detect the pathology RBBB, Figure 18 shows the interpretation of only two features. This is due to the absence of certain ECG leads in the MIT-BIH database. If the relevant ECG leads were available, the pathology features could be interpreted similarly to the supported features.

According to the formed interpretation result, Figure 19b highlights a signal fragment that confirms the feature of an extended QRS complex. Figure 19c shows a signal fragment confirming the presence of discordant changes in the ST-T segment. The prolonged intraventricular delay time is confirmed in the highlighted signal fragment shown in Figure 19d.

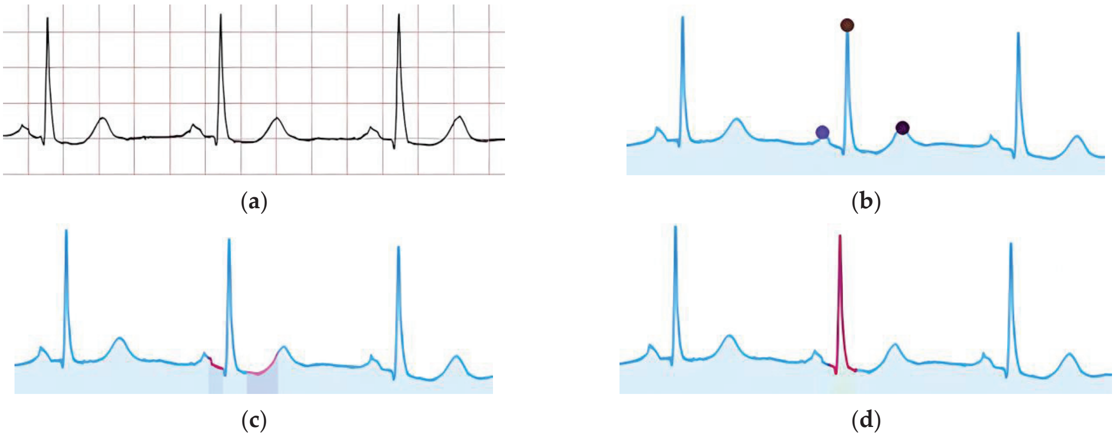


Figure 17. Visual confirmation of key features and abnormalities in ECG cycles, including (**a**) a normal ECG cycle, (**b**) key peak markers represented by colored dots: purple for the P wave, brown for the R peak, and dark purple for the T wave, (**c**) PQ and ST segments decline highlighted in red to indicate deviations from the baseline, and (**d**) a deformed QRS complex with red markings to emphasize morphological distortions.

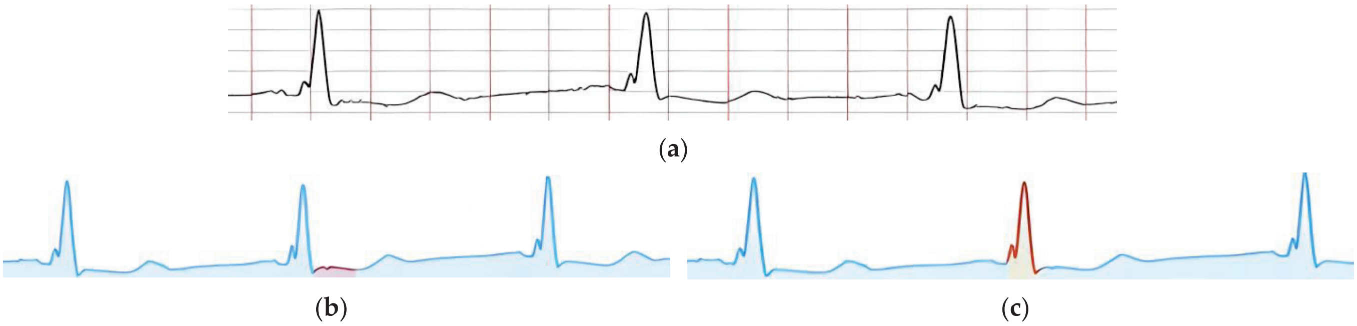


Figure 18. Visual confirmation of ECG pathology features, illustrating (**a**) a normal ECG cycle with standard waveforms, (**b**) an extended QRS complex highlighted in red to emphasize its abnormal duration, and (**c**) discordant changes in the ST-T segment, where red markings indicate deviations from the baseline.

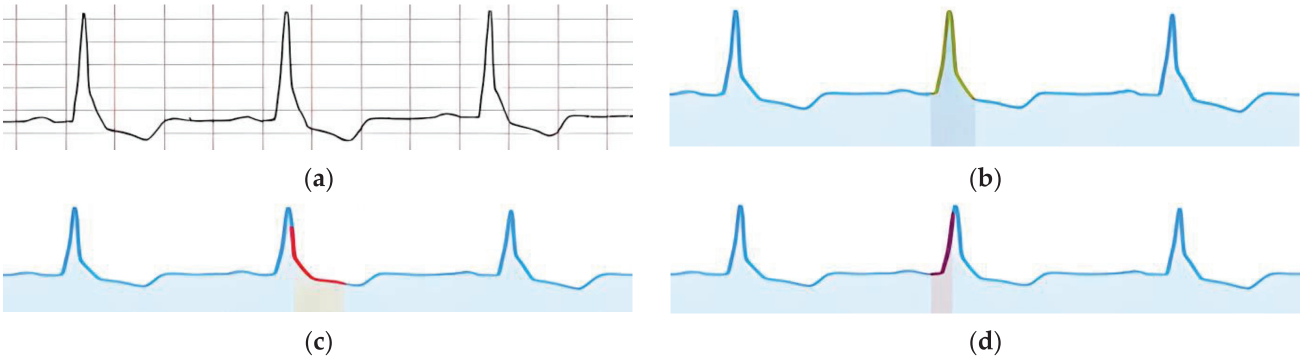


Figure 19. Visual confirmation of ECG features, associated with LBBB, including (**a**) a baseline ECG, (**b**) a signal fragment marked with a yellow highlight that confirms the feature of an extended QRS complex, (**c**) ST-segment elevation emphasized in red to signify abnormal changes, and (**d**) a widened QRS complex highlighted in red to depict the prolonged intraventricular delay time.

The formed interpretation result for the pathology LBBB does not include the feature "Deep, wide S waves in the right chest leads", as the right chest leads are not present in the MIT-BIH database.

Figure 20a shows an example of an input ECG classified as "Ventricular Extrasystole". The visual confirmation of the feature "Extended and deformed QRS" is shown in Figure 20b. In Figure 20c, a signal fragment is highlighted, within which the absence of the P peak is confirmed, and in Figure 20d, the part of the signal where the compensatory pause is present is highlighted.

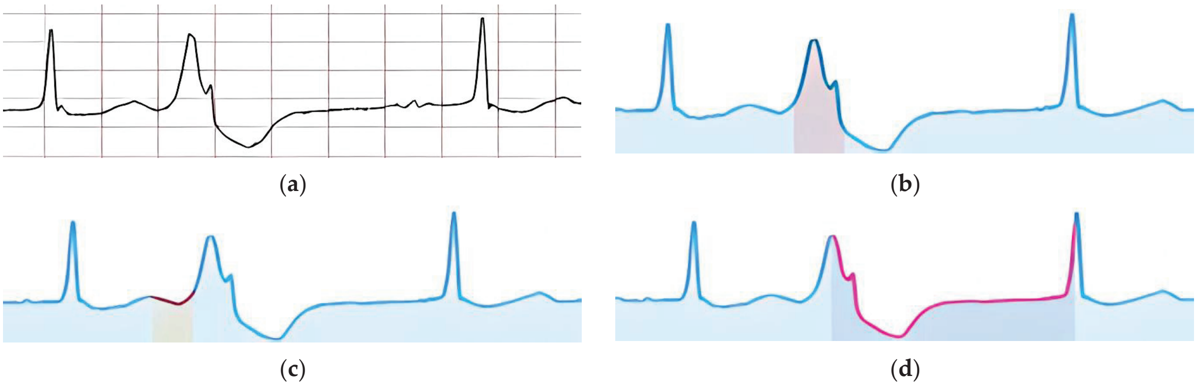


Figure 20. Visual confirmation of feature "Ventricular Extrasystole" in an ECG, including (**a**) a baseline ECG, (**b**) an extended and deformed QRS complex highlighted in light red to signify its abnormal morphology, (**c**) a segment marked in red indicating the absence of the P wave, and (**d**) a compensatory pause emphasized in pink to highlight the recovery period following the ectopic beat.

Figure 21 shows an example of an input ECG classified as "Fusion of Ventricular Extrasystole".

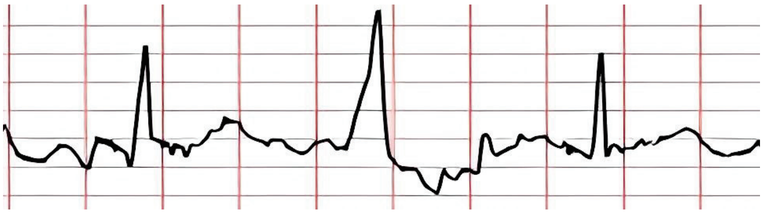


Figure 21. This figure displays an ECG classified as "Fusion of Ventricular Extrasystole". The waveform demonstrates characteristics of both normal and ectopic ventricular beats, indicative of fusion, where premature ventricular and normal impulses overlap, producing a unique hybrid beat.

According to the specified features, their visual confirmation was formed (Figure 22).

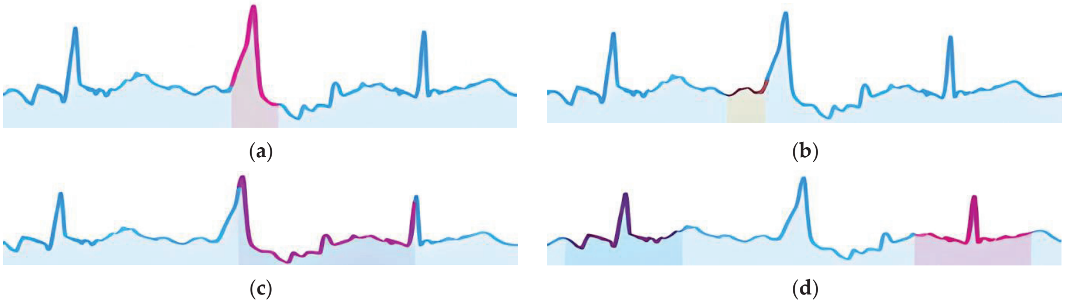


Figure 22. Visual confirmation of feature "Fusion of Ventricular Extrasystole" in an ECG, illustrating (**a**) an extended and deformed QRS complex highlighted in pink to indicate abnormal morphology, (**b**) the absence of the P peak marked in light brown to show missing atrial depolarization, (**c**) the lack of a compensatory pause emphasized with magenta to highlight the abnormal rhythm recovery, and (**d**) ventricular extrasystole highlighted in purple and pink that occurs between normal cardiac cycles.

Figure 22a shows the highlighted signal fragment used to confirm the feature of the extended and deformed QRS complex. The zone where the P peak is expected to be ab-sent is highlighted in Figure 22b. Figure 22c highlights the part of the signal where the lack of a compensatory pause is identified. Figure 22d shows visual confirmation of the ventricular extrasystole occurring between two normal cardio cycles.

Finally, to highlight the real-world feasibility and interpretability of our approach, we provide a demonstration of the proposed methods in action in addition to a Supplementary Materials file. This additional content comprises detailed clinical case examples derived from actual patient ECG data. It offers a comprehensive illustration of how the proposed methods are applied to genuine clinical signals, from preprocessing to final classification and interpretation.

4.5. Limitations of the Proposed Approach

Despite the significant scientific contribution of the proposed approach, it has several limitations in clinical use.

First, the main limitation of the proposed approach is its dependence on the accuracy of R peak detection in ECG. Although the method of integrating knowledge of the reference cardio cycle improves the accuracy of this process, it still faces challenges due to artifacts or noise in the data. Inaccuracies in detecting R peaks may lead to errors in the subsequent classification of arrhythmias, as cardio cycles are segmented for analysis based on these peaks. One possible way to address this issue involves including historical data for RR intervals, which could reflect physiological changes under different stress levels, although in certain cases heart rate variability may dominate. As a result, signal artifacts or unusual cardio cycle shapes can still reduce the model's effectiveness, particularly in clinical practice.

The second limitation concerns the number of pathology classes that the model can classify. Despite its high accuracy in detecting arrhythmias, there is a risk that the model may not account for all possible pathologies, especially if such anomalies occur infrequently in the training data. This situation may lead to weaker generalization for rare or atypical cases. A practical countermeasure involves integrating or generating an additional dataset specifically focused on these seldom-seen pathologies, thereby broadening coverage during training. Furthermore, class imbalance in the dataset may cause a bias toward more common pathologies, reducing accuracy for less common ones, which is a critical factor in clinical practice.

Finally, adding additional convolutional layers and using a triad of cardio cycles for analysis increases computational complexity, which may impact system performance when used in real time or on devices with limited resources.

5. Conclusions

This study presents a comprehensive approach for ECG-based arrhythmia detection, emphasizing accuracy and explainability through three integrated methods. First, the enhanced R peak detection method incorporates domain-specific knowledge into the ECG, leading to improved detection accuracy even in the presence of noise and artifacts. Second, the proposed arrhythmia classification method employs modified CNN architecture with additional convolutional and batch normalization layers. By analyzing a triad of cardio cycles—the previous, current, and next cycles—the model captures hidden dependencies and temporal features essential for accurate classification. This approach achieved an overall accuracy of 99.43% on the MIT-BIH database, with F1-scores nearing 100% for classes such as normal beats, RBBB and LBBB. Finally, the interpretation method translates CNN's decisions into clinically understandable features, enhancing transparency and aiding clinicians in decision-making.

Despite the advancements, certain limitations should also be considered. Foremost, the model exhibits sensitivity to signal artifacts and class imbalance, which may affect the detection of rare arrhythmias. Additionally, the increased computational complexity due to the enhanced CNN architecture and triad-cycle input may impede real-time applications or deployment on resource-constrained devices.

Future work will focus on optimizing the model's architecture to reduce computational costs and improve efficiency. This includes implementing advanced data augmentation techniques to address class imbalance and enhance the detection of rare pathologies. We also aim to increase the model's robustness against noise and artifacts and expand its classification capabilities to include a broader range of arrhythmia.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/technologies13010034/s1. Figure S1. Example of ECG images obtained from real patient case histories for use in clinical trials. This figure illustrates raw ECG recordings before any preprocessing steps, visually representing actual patient data. Figure S2. Example of dividing the ECG image into isolated channel-specific images. This figure highlights how each channel's waveform is separated for individual analysis, aiding in more precise signal processing. Figure S3. Visualization of the resulting 1D ECG. This figure demonstrates how the pixel-based ECG waveform, extracted from the original image, is converted into a one-dimensional array suitable for subsequent analysis. Figure S4. Visualization of the knowledge overlay applied to the input ECG. This figure shows how domain knowledge of the reference ECG (zones highlighted in green) is merged with the real ECG recording to facilitate more accurate R peak detection. Figure S5. Visualization of the detected R peaks. Here, red circles indicate the positions of R peaks identified by the encoder-decoder neural network on the processed ECG. Figure S6. Visualization of segmented ECG fragments for classification. Each fragment, centered around an R peak, is prepared for input into the CNN model to identify potential heart arrhythmia. Each colored rectangle (orange, green, and purple) represents one of the cardio cycles derived from the input ECG by splitting it into three cardio cycles. Figure S7. Visualization of each cardio cycle's class within the input ECG signal. In this figure, "N" indicates a normal cycle, and "V" indicates a ventricular extrasystole. The classification allows clinicians to identify arrhythmic events with a quick visual reference. Figure S8. Illustration of (a) a cardio cycle with a "Ventricular Extrasystole," (b) the attention zone for the "Absent P wave" feature marked in black, and (c) the attention zone for the "Extended and deformed QRS complex" feature highlighted in black. These annotated views help explain the characteristic regions of interest for diagnosing a "Ventricular Extrasystole". Figure S9. Visual comparison of the ECG signal with (a) "Normal" class signals and (b) signals of the "Ventricular Extrasystole" class. The cardiac cycle, highlighted in green, is overlaid with a black rectangle to emphasize the zone of interest and represent the relevant signal fragment. Figure S10. Application of visual analytics to the ECG signal. Red and blue clusters represent two distinct classes, highlighting areas of overlap and separation; the green dot represents the target ECG. This figure demonstrates how advanced visualization techniques (such as dimensionality reduction) help identify potential clusters or overlaps within the chosen signal segments. Figure S11. Illustration of a cardio cycle with the attention zone for the "Presence of a Compen-satory Pause" feature (marked in red rectangle). This scheme highlights the segment immediately following a "Ventricular Extrasystole," crucial for detecting whether a pause matches clinical definitions.

Author Contributions: Conceptualization, O.B. and I.K.; methodology, O.K. and O.B.; software, O.K.; validation, O.K., O.B. and L.K.; formal analysis, O.K., O.B. and P.R.; investigation, O.K.; resources, P.R. and L.K.; data curation, P.R. and L.K.; writing—original draft preparation, O.K. and O.B.; writing—review and editing, P.R.; visualization, O.K. and P.R.; supervision, I.K. and O.B.; project administration, I.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: Publicly available datasets were analyzed in this study. This data can be found here: https://github.com/okovalchuk98/ExplainableEcgClassification (accessed on 8 November 2024).

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

Abbreviations

The following abbreviations included in the text are reported alphabetically:

AAMI	Association for the Advancement of Medical Instrumentation
AI	Artificial Intelligence
AUC	Area Under the Curve
BCELoss	Binary Cross Entropy Loss
CNN	Convolutional Neural Network
CPSC	China Physiological Signal Challenge
DL	Deep Learning
ECG	Electrocardiogram
IQRDT	Intra-QRS Deflection Time
LBBB	Left Bundle Branch Block
ML	Machine Learning
MIT-BIH	Massachusetts Institute of Technology—Beth Israel Hospital
MDS	Multidimensional Scaling
PCA	Principal Component Analysis
PVC	Premature Ventricular Contraction
QRST	QRS Complex on a Typical Electrocardiogram
QT	QT Interval Dataset
RBBB	Right Bundle Branch Block
ROC	Receiver Operating Characteristic
ST	ST Segment
t-SNE	T-Distributed Stochastic Neighbor Embedding
UoG	University of Glasgow Database
XAI	Explainable Artificial Intelligence

Appendix A

Table A1. This table outlines the encoder-decoder architecture utilized for R peak identification within the proposed method. The architecture comprises convolutional, ReLU, max-pooling, and upsampling layers, configured to optimize ECG feature extraction and R peak detection.

Layer Name	Input	Output	Kernel Size	Stride	Padding	Scale Factor
			Encoder			
Conv1d	2	32	3	1	1	
ReLU						
MaxPool1d			2	2		
Conv1d	32	64	3	1	1	
ReLU						
MaxPool1d			2	2		
Conv1d	64	128	3	1	1	
ReLU						
MaxPool1d			2	2		
Conv1d	128	256	3	1	1	
ReLU						
MaxPool1d			2	2		

Table A1. *Cont.*

Layer Name	Input	Output	Kernel Size	Stride	Padding	Scale Factor
			Decoder			
Upsample						2
Conv1d	256	128	3	1	1	
ReLU						
Upsample						2
Conv1d	128	64	3	1	1	
ReLU						
Upsample						2
Conv1d	64	32	3	1	1	
ReLU						
Upsample						2
Conv1d	32	1	3	1	1	
ReLU						
Sigmoid						

Table A2. This table describes the improved CNN architecture within the proposed ECG classification method. The architecture combines convolutional, batch normalization, max-pooling, linear, and dropout layers to enhance feature extraction and classification accuracy for the ECGs. Bold color represents the layers that were included by the authors.

Layer Name	Input	Output	Kernel Size	Stride	Padding	Probability
			Encoder			
Conv1d	1	64	5	3	1	
ReLU						
BatchNorm1d	**64**					
Conv1d	64	64	5	2	1	
ReLU						
BatchNorm1d	**64**					
MaxPool1d			1	2		
Conv1d	64	128	3	1	1	
ReLU						
BatchNorm1d	**128**		**1**	**2**		
Conv1d	128	128	3	2	1	
ReLU						
BatchNorm1d	**128**					
Conv1d	**128**	**256**	**3**	**1**	**0**	
ReLU						
BatchNorm1d	**256**					
MaxPool1d			1	2		
			Decoder			
Linear	3584	1024				
ReLU						
BatchNorm1d	**1024**					
Dropout						**0.68**
Linear	1024	128				
ReLU						
Linear	128	9				

Table A3. This table outlines fine-tuned parameters of the improved CNN architecture for ECG classification, detailing layer types, input-output dimensions, kernel size, stride, and padding.

Layer Name	Input	Output	Kernel Size	Stride	Padding
Encoder					
Conv1d	1	32	5	1	2
ReLU					
MaxPool1d			2	2	0
Conv1d	32	64	3	1	2
ReLU					
Conv1d	64	128	3	1	2
ReLU					
MaxPool1d			2	2	0
Decoder					
Linear	3584	64			
ReLU					
Linear	64	1			
Sigmoid					

Appendix B

Table A4. Statistical metrics on the MIT-BIH dataset.

Metrics	Sample	Random Breakdowns							Avg	Std
		1	2	3	4	5	6	7		
Accuracy	train	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.000
	test	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.000
Precision	train	0.994	0.994	0.994	0.995	0.994	0.992	0.994	0.994	0.001
	test	0.993	0.992	0.991	0.992	0.993	0.992	0.992	0.992	0.001
Recall	train	0.991	0.991	0.991	0.992	0.991	0.987	0.991	0.991	0.002
	test	0.987	0.986	0.985	0.988	0.987	0.986	0.987	0.987	0.001
F1-score	train	0.992	0.992	0.993	0.994	0.992	0.989	0.992	0.992	0.002
	test	0.989	0.989	0.988	0.990	0.990	0.989	0.989	0.989	0.001

Table A5. Statistical metrics on the QT dataset.

Metrics	Sample	Random Breakdowns							Avg	Std
		1	2	3	4	5	6	7		
Accuracy	train	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.000
	test	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.000
Precision	train	0.988	0.988	0.987	0.987	0.987	0.985	0.988	0.987	0.001
	test	0.979	0.981	0.980	0.984	0.978	0.979	0.977	0.980	0.002
Recall	train	0.989	0.988	0.988	0.988	0.987	0.985	0.988	0.988	0.001
	test	0.978	0.981	0.978	0.994	0.976	0.979	0.978	0.981	0.006
F1-score	train	0.988	0.988	0.987	0.987	0.987	0.985	0.988	0.987	0.001
	test	0.978	0.98	0.983	0.985	0.977	0.98	0.977	0.980	0.003

Table A6. Statistical metrics on the CPSC-2020 dataset.

Metrics	Sample	Random Breakdowns							Avg	Std
		1	2	3	4	5	6	7		
Accuracy	train	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.000
	test	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.000
Precision	train	0.987	0.988	0.988	0.990	0.987	0.986	0.987	0.988	0.001
	test	0.986	0.986	0.986	0.988	0.985	0.985	0.985	0.986	0.001
Recall	train	0.996	0.996	0.996	0.996	0.996	0.995	0.996	0.996	0.000
	test	0.994	0.994	0.994	0.994	0.994	0.995	0.994	0.994	0.000
F1-score	train	0.991	0.992	0.992	0.993	0.991	0.990	0.991	0.991	0.001
	test	0.990	0.989	0.990	0.990	0.989	0.989	0.989	0.989	0.001

Table A7. Statistical metrics on the UoG dataset (independent test set).

Metrics	Random Breakdowns							Avg	Std
	1	2	3	4	5	6	7		
Accuracy	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.000
Precision	0.981	0.982	0.990	0.990	0.977	0.974	0.989	0.983	0.007
Recall	0.885	0.900	0.912	0.922	0.904	0.874	0.910	0.901	0.017
F1-score	0.910	0.923	0.933	0.942	0.926	0.900	0.930	0.923	0.014

Table A8. Statistical metrics for pathology classification on training and testing subsets.

Class	Training Samples						Testing Samples					
	Precision		Recall		F1-Score		Precision		Recall		F1-Score	
	Avg	Std	Avg	Std	Avg	Std	Avg	Std	Avg	Std	Avg	Std
1	1	0	1	0	1	0	0.994	0.005	1	0	1	0
2	0.999	0.004	0.999	0.004	1	0	0.990	0.005	0.990	0.005	0.990	0.005
3	1	0	1	0	1	0	0.993	0.005	1	0	1	0.005
4	1	0	1	0	1	0	0.999	0.004	1	0	0.999	0
5	1	0	1	0	1	0	1	0	1	0	1	0
6	0.999	0.004	1	0	1	0	0.971	0.012	0.954	0.008	0.962	0.007
7	0.949	0.013	0.963	0.016	0.957	0.005	0.924	0.024	0.860	0.037	0.883	0.016
8	1	0	0.997	0.005	1	0	0.994	0.022	0.952	0.038	0.975	0.018
9	0.993	0.005	0.997	0.005	0.994	0.005	0.864	0.042	0.800	0.124	0.831	0.08

References

1. Patwardhan, V.; Gil, G.F.; Arrieta, A.; Cagney, J.; DeGraw, E.; Herbert, M.E.; Khalil, M.; Mullany, E.C.; O'Connell, E.M.; Spencer, C.N.; et al. Differences across the lifespan between females and males in the top 20 causes of disease burden globally: A systematic analysis of the Global Burden of Disease Study 2021. *Lancet Public Health* **2024**, *9*, e282–e294. [CrossRef] [PubMed]
2. Death Statistics. Available online: https://deadorkicking.com/death-statistics/ (accessed on 20 October 2024).
3. Kaplan Berkaya, S.; Uysal, A.K.; Sora Gunal, E.; Ergin, S.; Gunal, S.; Gulmezoglu, M.B. A survey on ECG analysis. *Biomed. Signal Process. Control* **2018**, *43*, 216–235. [CrossRef]
4. Nasim, A.; Kim, Y.S. DE-PNN: Differential evolution-based feature optimization with probabilistic neural network for imbalanced arrhythmia classification. *Sensors* **2022**, *22*, 4450. [CrossRef] [PubMed]
5. Berezsky, O.; Liashchynskyi, P.; Pitsun, O.; Izonin, I. Synthesis of convolutional neural network architectures for biomedical image classification. *Biomed. Signal Process. Control* **2024**, *95*, 106325. [CrossRef]
6. Radiuk, P.; Barmak, O.; Krak, I. An approach to early diagnosis of pneumonia on individual radiographs based on the CNN information technology. *Open Bioinf. J.* **2021**, *14*, 93–107. [CrossRef]

7. Sane, R.K.S.; Choudhary, P.S.; Sharma, L.N.; Dandapat, S. Detection of myocardial infarction from 12 lead ECG images. In Proceedings of the 2021 National Conference on Communications (NCC-2021), Kanpur, India, 27–30 July 2021; IEEE Inc.: New York, NY, USA, 2021; pp. 1–6. [CrossRef]
8. Khater, H.M.; Suliman, A. Deep learning-based ECG analysis for myocardial infarction detection. In Proceedings of the 2023 15th International Conference on Innovations in Information Technology (IIT-2023), Al Ain, United Arab Emirates, 14–15 November 2023; IEEE Inc.: New York, NY, USA, 2023; pp. 61–66. [CrossRef]
9. Radiuk, P.; Barmak, O.; Manziuk, E.; Krak, I. Explainable deep learning: A visual analytics approach with transition matrices. *Mathematics* **2024**, *12*, 1024. [CrossRef]
10. Altameem, A.; Kovtun, V.; Al-Ma'aitah, M.; Altameem, T.; Fouad, H.; Youssef, A.E. Patient's data privacy protection in medical healthcare transmission services using back propagation learning. *Comput. Elect. Eng.* **2022**, *102*, 108087. [CrossRef]
11. Radiuk, P.; Kovalchuk, O.; Slobodzian, V.; Manziuk, E.; Krak, I. Human-in-the-loop approach based on MRI and ECG for healthcare diagnosis. In Proceedings of the 5th International Conference on Informatics & Data-Driven Medicine (IDDM-2022), Lyon, France, 18–20 November 2022; Shakhovska, N., Chretien, S., Izonin, I., Campos, J., Eds.; CEUR-WS: Aachen, Germany, 2022; Volume 3302, pp. 9–20.
12. Zahid, M.U.; Kiranyaz, S.; Ince, T.; Devecioglu, O.C.; Chowdhury, M.E.H.; Khandakar, A.; Tahir, A.; Gabbouj, M. Robust R-peak detection in low-quality Holter ECGs using 1D convolutional neural network. *IEEE Trans. Biomed. Eng.* **2022**, *69*, 119–128. [CrossRef]
13. Kovalchuk, O.; Radiuk, P.; Barmak, O.; Krak, I. Robust R-peak detection using deep learning based on integrating domain knowledge. In Proceedings of the 6th International Conference on Informatics & Data-Driven Medicine (IDDM-2023), Bratislava, Slovakia, 17–19 November 2023; Shakhovska, N., Kovac, M., Izonin, I., Chretien, S., Eds.; CEUR-WS: Aachen, Germany, 2024; Volume 3609, pp. 1–14.
14. Porr, B.; Macfarlane, P.W. A new QRS detector stress test combining temporal jitter and accuracy (JA) reveals significant performance differences amongst popular detectors. *bioRxiv* **2023**, 722397. [CrossRef]
15. Fariha, M.A.Z.; Ikeura, R.; Hayakawa, S.; Tsutsumi, S. Analysis of Pan-Tompkins algorithm performance with noisy ECG signals. *J. Phys. Conf. Ser.* **2020**, *1532*, 012022. [CrossRef]
16. Ahmad, I. QRS detection for heart rate monitoring. *Int. J. Elect. Eng. Technol.* **2020**, *11*, 360–367. [CrossRef]
17. Rahul, J.; Sora, M.; Sharma, L.D. A novel and lightweight P, QRS, and T peaks detector using adaptive thresholding and template waveform. *Comput. Biol. Med.* **2021**, *132*, 104307. [CrossRef] [PubMed]
18. Hassan, S.U.; Mohd Zahid, M.S.; Abdullah, T.A.; Husain, K. Classification of cardiac arrhythmia using a convolutional neural network and bi-directional long short-term memory. *Digit Health* **2022**, *8*, 205520762211027. [CrossRef] [PubMed]
19. Liu, F.; Zhou, X.; Cao, J.; Wang, Z.; Wang, H.; Zhang, Y. A LSTM and CNN based assemble neural network framework for arrhythmias classification. In Proceedings of the ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-2019), Brighton, UK, 12–17 May 2019; IEEE Inc.: New York, NY, USA, 2019; pp. 1303–1307. [CrossRef]
20. Xu, X.; Liu, H. ECG heartbeat classification using convolutional neural networks. *IEEE Access* **2020**, *8*, 8614–8619. [CrossRef]
21. Degirmenci, M.; Ozdemir, M.A.; Izci, E.; Akan, A. Arrhythmic heartbeat classification using 2D convolutional neural networks. *IRBM* **2022**, *43*, 422–433. [CrossRef]
22. Rohmantri, R.; Surantha, N. Arrhythmia classification using 2D convolutional neural network. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*, 201–208. [CrossRef]
23. Ullah, A.; Anwar, S.M.; Bilal, M.; Mehmood, R.M. Classification of arrhythmia by using deep learning with 2-D ECG spectral image representation. *Remote Sens.* **2020**, *12*, 1685. [CrossRef]
24. Abdelhafid, E.; Aymane, E.; Benayad, N.; Abdelalim, S.; My Hachem, E.Y.A.; Rachid, O.H.T.; Brahim, B. ECG arrhythmia classification using convolutional neural network. *IJETAE* **2022**, *12*, 186–195. [CrossRef]
25. Singh, P.; Sharma, A. Interpretation and classification of arrhythmia using deep convolutional network. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 2518512. [CrossRef]
26. Ayano, Y.M.; Schwenker, F.; Dufera, B.D.; Debelee, T.G.; Ejegu, Y.G. Interpretable hybrid multichannel deep learning model for heart disease classification using 12-lead ECG signal. *IEEE Access* **2024**, *12*, 94055–94080. [CrossRef]
27. Wang, J.; Li, R.; Li, R.; Fu, B. A knowledge-based deep learning method for ECG signal delineation. *Future Gener. Comput. Syst.* **2020**, *109*, 56–66. [CrossRef]
28. Mao, A.; Mohri, M.; Zhong, Y. Cross-entropy loss functions: Theoretical analysis and applications. In Proceedings of the 40th International Conference on Machine Learning (ICML-2023), Honolulu, HI, USA, 23–29 July 2023; Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., Scarlett, J., Eds.; Volume 202, pp. 23803–23828.

29. Kovalchuk, O.; Radiuk, P.; Barmak, O.; Krak, I. ECG arrhythmia classification and interpretation using convolutional networks for intelligent IoT healthcare system. In Proceedings of the 1st International Workshop on Intelligent & CyberPhysical Systems (ICyberPhyS-2024), Khmelnytskyi, Ukraine, 28 June 2024; Hovorushchenko, T., Savenko, O., Popov, P.T., Lysenko, S., Eds.; CEUR-WS: Aachen, Germany, 2024; Volume 3736, pp. 47–62.
30. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the 32nd International Conference on International Conference on Machine Learning (ICML-2015), Lille, France, 6–11 July 2015; Bach, F., Blei, D., Eds.; Volume 37, pp. 448–456.
31. Pearson, K. LIII. On lines and planes of closest fit to systems of points in space. *Lond. Edinb. Dublin Philos. Mag. J. Sci.* **1901**, *2*, 559–572. [CrossRef]
32. Kruskal, J.B. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika* **1964**, *29*, 1–27. [CrossRef]
33. Hinton, G.; Roweis, S. Stochastic neighbor embedding. In Proceedings of the 16th International Conference on Neural Information Processing Systems (NeurIPS-2002), Vancouver, BC, Canada, 9–14 December 2002; Becker, S., Thrun, S., Obermayer, K., Eds.; MIT Press: Cambridge, MA, USA, 2003; pp. 833–840.
34. Makowski, D.; Pham, T.; Lau, Z.J.; Brammer, J.C.; Lespinasse, F.; Pham, H.; Schölzel, C.; Chen, S.H.A. NeuroKit2: A python toolbox for neurophysiological signal processing. *Behav. Res.* **2021**, *53*, 1689–1696. [CrossRef]
35. Rainio, O.; Teuho, J.; Klén, R. Evaluation metrics and statistical tests for machine learning. *Sci. Rep.* **2024**, *14*, 6086. [CrossRef] [PubMed]
36. Moody, G.B.; Mark, R.G. The impact of the MIT-BIH arrhythmia database. *IEEE Eng. Medicine Biol. Mag.* **2001**, *20*, 45–50. [CrossRef]
37. Laguna, P.; Mark, R.G.; Goldberg, A.; Moody, G.B. A database for evaluation of algorithms for measurement of QT and other waveform intervals in the ECG. In Proceedings of the Computers in Cardiology 1997, Lund, Sweden, 7–10 September 1997; IEEE Inc.: New York, NY, USA, 2002; pp. 673–676. [CrossRef]
38. Alday, E.A.P.; Gu, A.; Shah, A.J.; Robichaux, C.; Wong, A.-K.I.; Liu, C.; Liu, F.; Rad, A.B.; Elola, A.; Seyedi, S.; et al. Classification of 12-Lead ECGs: The PhysioNet/Computing in Cardiology Challenge 2020. *Physiol. Meas.* **2020**, *41*, 124003. [CrossRef]
39. Howell, L.; Porr, B. *High Precision ECG Database with Annotated R Peaks, Recorded and Filmed Under Realistic Conditions*; University of Glasgow: Glasgow, UK, 2018. [CrossRef]
40. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2017**, arXiv:1412.6980. [CrossRef]
41. Rodrigues, R.; Couto, P. Semi-supervised learning for ECG classification. In Proceedings of the 2021 Computing in Cardiology (CinC-2021), Brno, Czech Republic, 13–15 September 2021; IEEE Inc.: New York, NY, USA, 2022; pp. 1–4. [CrossRef]
42. Koka, T.; Muma, M. Fast and sample accurate R-peak detection for noisy ECG using visibility graphs. In Proceedings of the 2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC-2022), Glasgow, UK, 11–15 July 2022; IEEE Inc.: New York, NY, USA, 2022; pp. 121–126. [CrossRef]
43. Bilski, J.; Smoląg, J.; Kowalczyk, B.; Grzanek, K.; Izonin, I. Fast computational approach to the Levenberg-Marquardt algorithm for training feedforward neural networks. *J. Artif. Intell. Soft Comput. Res.* **2023**, *13*, 45–61. [CrossRef]
44. Young, B.; Schmid, J.-J. Updates to IEC/AAMI ECG standards, a new hybrid standard. *J. Electrocardiol.* **2018**, *51*, 103–105. [CrossRef]
45. Ahmed, A.A.; Ali, W.; Abdullah, T.A.A.; Malebary, S.J. Classifying cardiac arrhythmia from ECG signal using 1D CNN deep learning model. *Mathematics* **2023**, *11*, 562. [CrossRef]
46. Kumar, S.; Mallik, A.; Kumar, A.; Ser, J.D.; Yang, G. Fuzz-ClustNet: Coupled fuzzy clustering and deep neural networks for arrhythmia detection from ECG signals. *Comput. Biol. Med.* **2023**, *153*, 106511. [CrossRef]
47. Mahmud, T.; Barua, A.; Islam, D.; Hossain, M.S.; Chakma, R.; Barua, K.; Monju, M.; Andersson, K. Ensemble deep learning approach for ECG-based cardiac disease detection: Signal and image analysis. In Proceedings of the 2023 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD-2023), Dhaka, Bangladesh, 21–23 September 2023; IEEE Inc.: New York, NY, USA, 2023; pp. 70–74. [CrossRef]
48. Yang, H.; Wei, Z. Arrhythmia recognition and classification using combined parametric and visual pattern features of ECG morphology. *IEEE Access* **2020**, *8*, 47103–47117. [CrossRef]
49. Itseez Open Source Computer Vision Library. Available online: https://opencv.org/ (accessed on 14 August 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.

technologies

MDPI

Article

Enhancing Thyroid Nodule Detection in Ultrasound Images: A Novel YOLOv8 Architecture with a C2fA Module and Optimized Loss Functions

Shidan Wang [1,†], Zi-An Zhao [2,†], Yuze Chen [3], Ye-Jiao Mao [2] and James Chung-Wai Cheung [2,4,*]

[1] School of Microelectronics and Communication Engineering, Chongqing University, Chongqing 400044, China; stacey.os.w@cqu.edu.cn
[2] Department of Biomedical Engineering, Faculty of Engineering, The Hong Kong Polytechnic University, Hong Kong 999077, China; 24057373r@connect.polyu.hk (Z.-A.Z.); yejiao.mao@connect.polyu.hk (Y.-J.M.)
[3] College of Computer Science, Chongqing University, Chongqing 400044, China; yuze.chen@cqu.edu.cn
[4] Research Institute of Smart Ageing, The Hong Kong Polytechnic University, Hong Kong 999077, China
* Correspondence: james.chungwai.cheung@polyu.edu.hk; Tel.: +852-2766-7673
† These authors contributed equally to this work.

Abstract: Thyroid-related diseases, particularly thyroid cancer, are rising globally, emphasizing the critical need for the early detection and accurate screening of thyroid nodules. Ultrasound imaging has inherent limitations—high noise, low contrast, and blurred boundaries—that make manual interpretation subjective and error-prone. To address these challenges, YOLO-Thyroid, an improved model for the automatic detection of thyroid nodules in ultrasound images, is presented herein. Building upon the YOLOv8 architecture, YOLO-Thyroid introduces the C2fA module—an extension of C2f that incorporates Coordinate Attention (CA)—to enhance feature extraction. Additionally, loss functions were incorporated, including class-weighted binary cross-entropy to alleviate class imbalance and SCYLLA-IoU (SIoU) to improve localization accuracy during boundary regression. A publicly available thyroid ultrasound image dataset was optimized using format conversion and data augmentation. The experimental results demonstrate that YOLO-Thyroid outperforms mainstream object detection models across multiple metrics, achieving a higher detection precision of 54%. The recall, calculated based on the detection of nodules containing at least one feature suspected of being malignant, reaches 58.2%, while the model maintains a lightweight structure. The proposed method significantly advances ultrasound nodule detection, providing an effective and practical solution for enhancing diagnostic accuracy in medical imaging.

Keywords: thyroid nodule detection; ultrasound imaging; YOLO; deep learning; medical image analysis

1. Introduction

The thyroid gland is a vital endocrine organ in the human body. The incidence of thyroid-related diseases, particularly thyroid cancer, has been rising rapidly recently and has become a significant global public health concern [1]. Early detection and accurate screening, especially for thyroid nodules—which are often considered early indicators of potential malignancies—play a critical role in mitigating this trend. Clinical studies have confirmed that timely and accurate diagnosis of thyroid nodules can significantly reduce the incidence and mortality rates of thyroid cancer [2]. Various diagnostic examinations

https://doi.org/10.3390/technologies13010028

are commonly used to evaluate the thyroid gland, including ultrasound (US), computed tomography (CT), magnetic resonance imaging (MRI), thyroid scans, and elastography [3].

While the majority of thyroid nodules are benign [4], a small percentage can be malignant, making accurate diagnosis essential for appropriate treatment. Accurate diagnosis is essential for appropriate treatment, but distinguishing malignant from benign nodules using non-invasive methods remains challenging. Ultrasound imaging is the preferred screening tool due to its convenience, low cost, and absence of radiation exposure. However, ultrasound images often suffer from high noise, low contrast, and blurred boundaries [5], which makes interpretation subjective and dependent on the radiologist's experience. This subjectivity can lead to inconsistent diagnoses and potentially result in unnecessary invasive procedures, such as fine-needle aspiration biopsy (FNAB) [6]. Traditional ultrasonography identifies features associated with malignancy, such as hypoechogenicity, irregular margins, microcalcifications, and a taller-than-wide shape [7]. Although these features aid in assessment, they cannot definitively indicate malignancy, and a biopsy is required for a conclusive diagnosis.

To improve diagnostic accuracy, researchers have explored artificial intelligence (AI) and deep learning techniques to provide a more objective assessment [8,9]. Machine learning models, particularly convolutional neural networks (CNNs), have shown great promise in classifying thyroid nodules and assisting in diagnosis [10]. For instance, Zheng et al. [11] proposed an improved U-Net architecture called DSRU-Net, which enhances the automatic segmentation of thyroid glands and nodules in ultrasound images by incorporating ResNeSt blocks, atrous spatial pyramid pooling, and deformable convolution v3. Similarly, Zhou et al. [12] introduced a thyroid nodule detection model named Thyroid-DETR, utilizing the Transformer architecture along with deformable convolution, multi-head self-attention, and a dual-stream training structure to improve detection accuracy in ultrasound images. Additionally, Chen et al. [13] developed a multi-view learning model called MLMSeg, which integrates CNNs, Transformers, and Graph Convolutional Networks to enhance thyroid nodule segmentation by capturing local, global, and spatial structural features. Consequently, computer-aided detection (CAD) methods for thyroid nodules have become a research hotspot, holding significant potential for future advancements.

Despite advances, existing CAD methods still face challenges when applied to thyroid nodules. One major issue is that datasets often suffer from class imbalance, causing models to be biased toward the majority class during training, which adversely affects performance in detecting and classifying malignant nodules [4,14]. Additionally, many studies employ semantic segmentation techniques for thyroid nodule detection due to their precise pixelwise delineation. However, semantic segmentation networks are often computationally intensive and slower, making them less suitable for real-time clinical applications and large-scale screenings. To address these issues, an improved model named YOLO-Thyroid is proposed for the automatic detection of nodules in thyroid ultrasound images, building upon YOLOv8. YOLO (You Only Look Once) models, with their efficient single-stage detection architecture, have demonstrated strong performance in detecting medical anomalies in pulmonary nodules (chest X-rays), breast masses (mammograms), and brain tumors (MRI scans) [15–19]. YOLO offers a high detection speed suitable for prompt decision-making, which is crucial in clinical settings.

Although newer versions of the YOLO model are available, YOLOv8 [19] was selected due to its optimal balance between speed and accuracy, as well as its optimized performance, flexibility, and extensibility, which make it particularly suitable for the current application scenario. The Coordinate Attention (CA) mechanism is introduced to enhance the extraction of important features. CA embeds positional information into channel attention, allowing the model to focus on the most informative regions of the feature maps, which is crucial for accurately detecting thyroid nodules with variable sizes and blurred boundaries. CA is integrated into the existing C2f module of YOLOv8, resulting in the modified C2fA module. This integration allows YOLO-Thyroid to emphasize important features related to target objects, thereby improving detection performance. Additionally, specialized loss functions are employed: the class-weighted binary cross-entropy (CW-BCE) loss function, used to alleviate the problem of class imbalance, and the SCYLLA-IoU (SIoU) loss function, which comprehensively considers factors such as the target's position, size, and shape during boundary regression to improve localization accuracy. Through these improvements, the YOLO-Thyroid model achieves better detection performance while maintaining a lightweight structure.

1.1. Main Contributions

The main contributions of this paper are as follows:

1. The proposal of an improved model, YOLO-Thyroid, with the introduction of the C2fA module and the CW-BCE and SIoU loss functions, strengthening the model's ability to extract and fuse important features, and improving performance under class imbalance conditions.
2. Extensive experiments and comparisons demonstrating that the YOLO-Thyroid model outperforms current mainstream object detection models across multiple performance metrics, validating its effectiveness.

1.2. Objective

The primary objective of this study is to address the critical challenges in thyroid nodule detection in ultrasound imaging, particularly the limitations of existing methods in terms of feature extraction, class imbalance, and localization accuracy. Specifically, the objectives are:

1. To design an improved YOLOv8-based detection model, YOLO-Thyroid, incorporating a novel C2fA module and optimized loss functions to enhance feature extraction and localization performance.
2. To alleviate the impact of class imbalance in the dataset by introducing a class-weighted binary cross-entropy (CW-BCE) loss function, ensuring the robust detection of both benign and malignant nodules.
3. To integrate advanced attention mechanisms and a lightweight architecture, enabling the model to achieve improved detection accuracy and balanced recall while maintaining computational efficiency for real-time clinical applications.
4. To assess the model's performance in terms of detecting nodules and identifying at least one feature suspected of being malignant (e.g., TIRADS categories 4a, 4b, 4c, and 5), with an emphasis on determining which category exhibits the highest detection sensitivity. Additionally, this objective seeks to evaluate the model's ability to differentiate varying levels of risk and establish a threshold for the number of detected features required to warrant further clinical investigation for potential malignancies.
5. To validate the proposed model on a publicly available thyroid ultrasound dataset and demonstrate its superiority compared to state-of-the-art object detection methods.

This study aims to provide a practical and effective solution for automatic thyroid nodule detection, contributing to improved diagnostic accuracy and efficiency in medical imaging, while offering insights into the clinical implications of the model's ability to differentiate risk levels across classes and establish decision-making thresholds.

The structure of this paper is organized as follows: Section 2 provides a detailed description of the dataset preprocessing methods, as well as the structure and improvements of the YOLO-Thyroid model. Section 3 presents the experimental results and analyses, comparing them with other advanced models. Section 4 discusses the advantages of the model and possible directions for improvement. Section 5 summarizes the findings and suggests future research directions.

2. Materials and Methods

An overview of the methodological workflow is presented in Figure 1. To adapt the dataset for object detection tasks, a comprehensive method was developed to convert ultrasound image labels into the YOLO format. Next, the thyroid ultrasound images were preprocessed to remove irrelevant regions and interfering markers, completing the data preparation for model input. To address the issue of a small sample size, data augmentation techniques were employed. Subsequently, the ultrasound data were trained using the proposed YOLO-Thyroid model, as shown in Figure 1. To improve the model's detection accuracy across different categories of nodules, the C2fA module was proposed, enabling the model to better capture spatial information and important features. Additionally, to enhance the model's robustness against data imbalance and convergence issues, the CW-BCE and SIoU loss functions were incorporated into the object detection loss function, thereby further improving model performance.

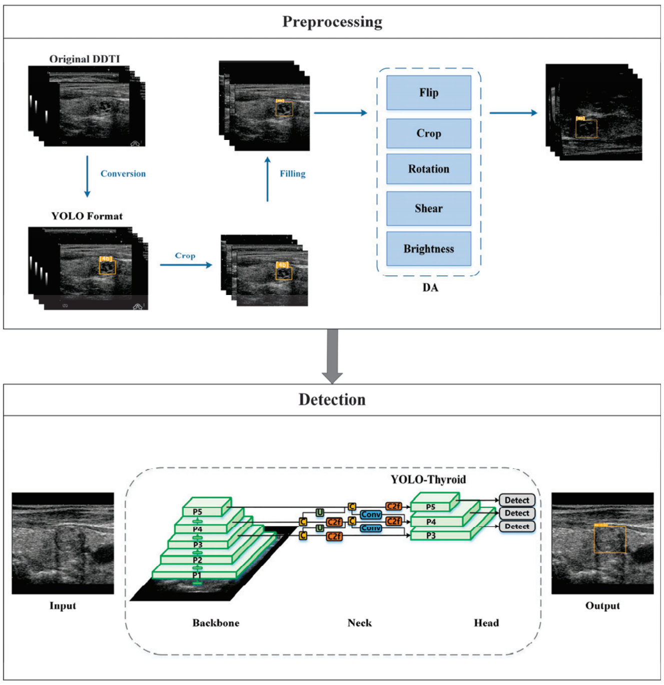


Figure 1. Overall workflow of the proposed method. C and U are concat and upsample layers respectively. The green frame and the number in Preprocessing is the ground truth of nodule location and class. The yellow frame and the number in Detection indicates the prediction of location and class.

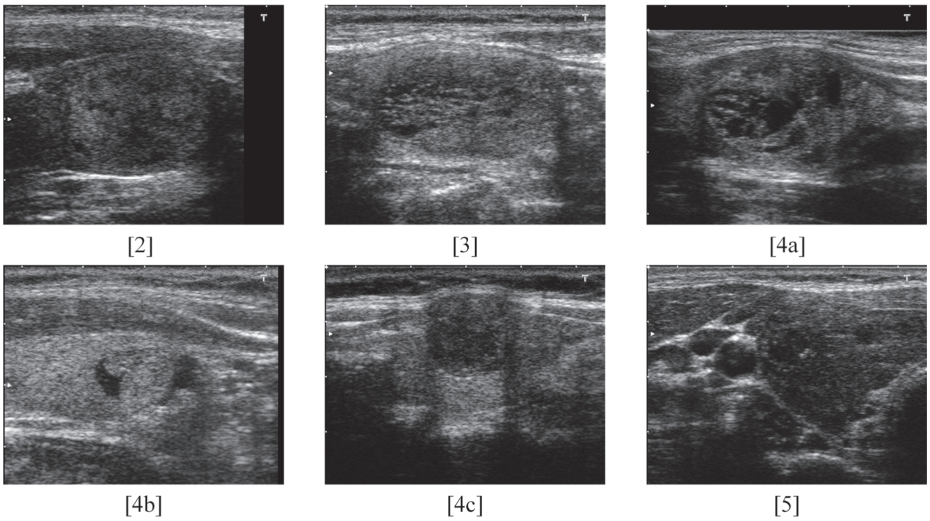
2.1. Dataset

2.1.1. Data Description

In this study, the Digital Database of Thyroid Images (DDTI) [20] was utilized. This publicly available dataset provides a comprehensive collection of B-mode ultrasound images, along with detailed descriptions and annotations of suspicious thyroid lesions. The dataset contains 480 images from 400 medical cases, saved in JPG format with a resolution of 560 × 360 pixels. Label information is stored in XML files. The nodule information includes composition, size, echogenicity, edge characteristics, presence or absence of calcification, and Thyroid Imaging Reporting and Data System (TIRADS) scores. Nodule annotations are manually segmented by radiologists and recorded in the form of coordinates. The TIRADS [21] is used to assess the malignancy risk of thyroid nodules by standardizing the evaluation of ultrasound features and classifying nodules into different levels. The TIRADS scores in the dataset include:

- [2] Benign (0% risk of malignancy);
- [3] No suspicious US feature (<5% malignancy);
- [4a] One suspicious US feature (5–10% malignancy);
- [4b] Two suspicious US features (10–80% malignancy);
- [4c] Three or four suspicious US features (10–80% malignancy);
- [5] Five suspicious features (>80% malignancy).

These classifications are illustrated in Figure 2. The dataset was cleaned by removing images with incomplete coordinate annotations and missing TIRADS score labels, ultimately obtaining 339 images. Figure 3 presents a statistical overview of the dataset. According to established guidelines, classes 2 and 3 are categorized as benign, whereas classes 4a, 4b, 4c, and 5 are classified as malignant [20].


[2] [3] [4a]

[4b] [4c] [5]

Figure 2. TIRADS classification for assessing malignancy risk in thyroid nodules. [2], [3], [4a], [4b], [4c], [5] are the TIRADS scores to classify nodules. The details are described above.

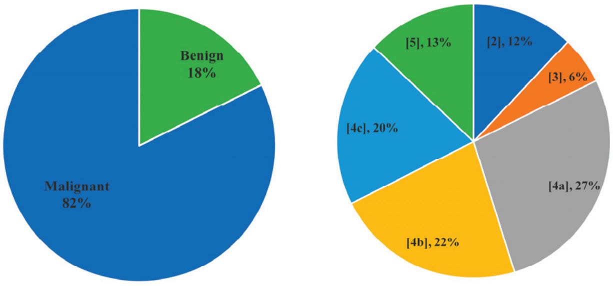
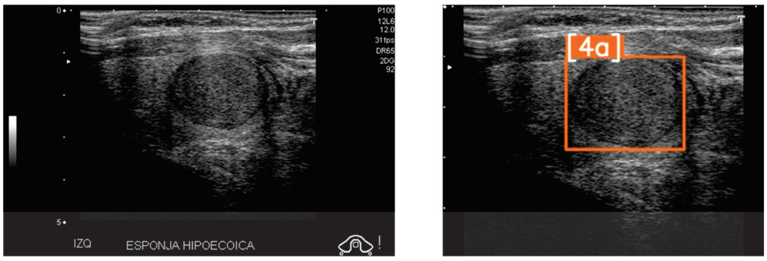


Figure 3. Statistical overview of the DDTI dataset. [2], [3], [4a], [4b], [4c], [5] are the TIRADS scores to classify nodules. The details are described above.

2.1.2. Data Preprocessing

In this paper, the image preprocessing process includes four key steps to convert the dataset with original labels into YOLO data format. Non-essential information had to be removed to ensure that the model focuses only on important features. Next, the images were standardized in size to enhance data consistency and improve model performance.

Then, the dataset was divided and augmented to increase data diversity and enhance the model's generalization ability. Examples of the preprocessed images are shown in Figure 4. Detailed descriptions of each step are as follows.



(a)

(b)

Figure 4. Examples of preprocessed thyroid ultrasound images. (**a**) Original ultrasound image. (**b**) Preprocessed image. The rectangle in (**b**) represents the detection label in YOLO format, indicating the location of the thyroid nodule. The '4a' label corresponds to the TIRADS score.

Data Format Conversion: The segmentation labels were transformed into the format required by the YOLO model to prepare the data for subsequent object detection model training. For each image, the minimum bounding rectangle of each target nodule was extracted where the nodule is present in the segmentation label. Then, the center coordinates of the bounding box and its width and height were calculated. These values were normalized by dividing by the image's width and height, ensuring they ranged between 0 and 1, as required by the YOLO format. The formatted labels included the class identifier along with the normalized center coordinates, width, and height. This conversion enabled the YOLO model to accurately interpret the bounding boxes for training and detection purposes. The detailed pseudocode is provided in Algorithm A1 in Appendix A.

Removal of Non-essential Information: The original ultrasound images contain additional information besides the target area, such as grayscale scales, parameter information, text labels, and probe icons. This information might interfere with the model's training and detection accuracy. Therefore, during preprocessing, the images were cropped to remove these non-essential elements, retaining only the ultrasound images containing the target area [22,23]. The purpose of this step is to eliminate noise and irrelevant features, enhancing the model's focus on the target.

Image Adjustment: In the cropped images, the edges were padded with black pixels to unify the image size to 640 × 640 pixels. The purpose of this step is to prevent the target shape from being distorted when resizing the images, thus preserving the true characteristics of the target.

Dataset Partitioning and Augmentation: The dataset was divided into three independent subsets: the training set, validation set, and test set, in a ratio of 7:1.5:1.5. Table 1 lists the number of samples in each subset of the DDTI dataset. In the training set, a series of data augmentation techniques were applied to enhance the model's generalization ability and prevent overfitting. Specific augmentation methods included horizontal and vertical flipping to increase sample diversity—acceptable in thyroid imaging due to the gland's bilateral symmetry. Cropping operations were used to simulate different imaging distances and perspective changes, reflecting variations that occur in clinical practice. Rotation within a limited angle range was applied to enhance the model's robustness to varying probe orientations during ultrasound examinations. Shear transformations were employed to mimic slight geometric distortions that may result from probe pressure or angle variations during imaging. Brightness adjustments enabled the model to adapt to variations in image intensity caused by different machine settings or patient characteristics. These augmentation techniques are commonly used in medical image analysis to improve model performance while maintaining clinical validity [24,25].

Table 1. The distribution of samples in the training, validation, and test sets of the DDTI dataset.

Property	Class	Malignancy Risk	Training Set	Validation Set	Testing Set	Total
Recognized Benign	2	0%	30	4	7	41
	3	<5%	12	4	3	19
Suspicious of Malignancy	4a	5–10%	62	14	17	93
	4b	10–80%	58	9	9	76
	4c	10–80%	43	13	10	66
	5	>80%	32	7	5	44
Total			237	51	51	339

In the augmented dataset, the augmentation methods are applied randomly and can be combined, resulting in composite augmentations where multiple techniques are applied together. Figure 5 illustrates the original image alongside the augmented images. Collectively, these augmentation techniques enhanced the diversity of the training data, thereby improving the model's performance and generalization to unseen data. Through these augmentation methods, the number of samples in the training set was expanded to three times the original, and the augmented dataset is shown in Table 2.

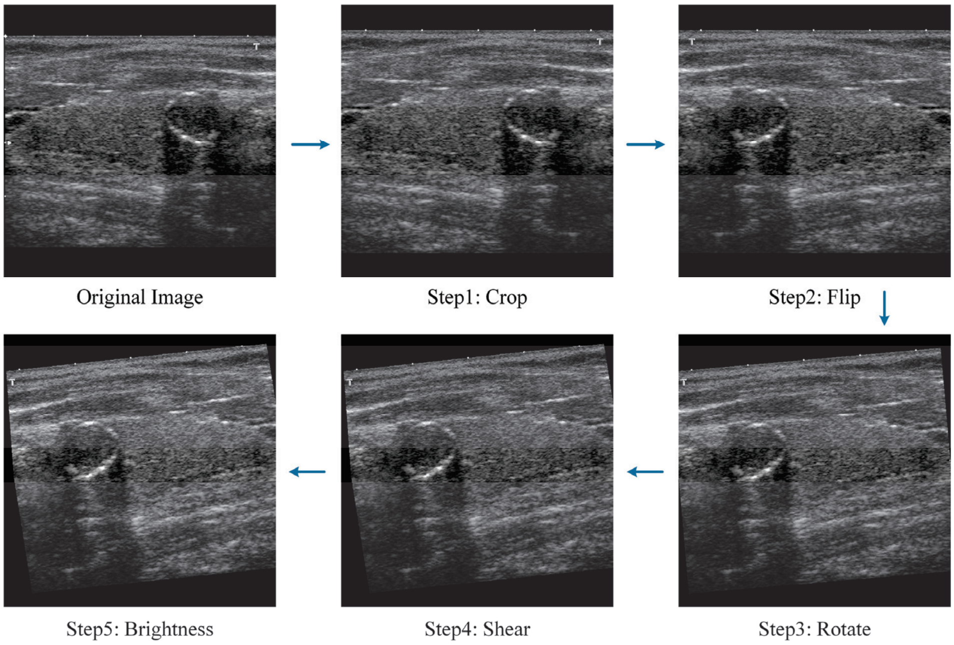


Figure 5. Examples of data augmentation techniques applied to a single image. Step 1: cropping to retain 95% of the image centered at coordinates (7%, 46%). Step 2: horizontal flipping. Step 3: rotation by 5 degrees. Step 4: shear transformations with 9 degrees along the X-axis and −2 degrees along the Y-axis. Step 5: a brightness adjustment of 9%. The final image is the result of the combined effect of these augmentation methods.

Table 2. Expanded sample sizes in the augmented training set.

Property	Class	Malignancy Risk	Training Set	Validation Set	Testing Set	Total
Recognized Benign	2	0%	90	4	7	101
	3	<5%	36	4	3	43
Suspicious of Malignancy	4a	5–10%	186	14	17	217
	4b	10–80%	174	9	9	192
	4c	10–80%	129	13	10	152
	5	>80%	96	7	5	108
Total			711	51	51	813

2.2. Methods

In this study, a detection model based on the YOLOv8 architecture, named YOLO-Thyroid, is proposed. YOLOv8 [19] is an advanced single-stage object detection model with efficient feature extraction capabilities and fast inference speed, as shown in Figure 6. To further enhance performance in thyroid ultrasound nodule detection tasks, two key modules were introduced. First, the feature extraction of the model was optimized by designing a C2fA module to enhance its perception of thyroid nodules, which enhanced detection performance in complex ultrasound image backgrounds. Second, considering the characteristics of ultrasound datasets, its loss function was improved to enhance the model's performance under class imbalance conditions. These two improvements allow YOLO-Thyroid to enhance nodule detection performance while maintaining the original model's efficiency. The design concepts and implementation details will be detailed in the following sections.

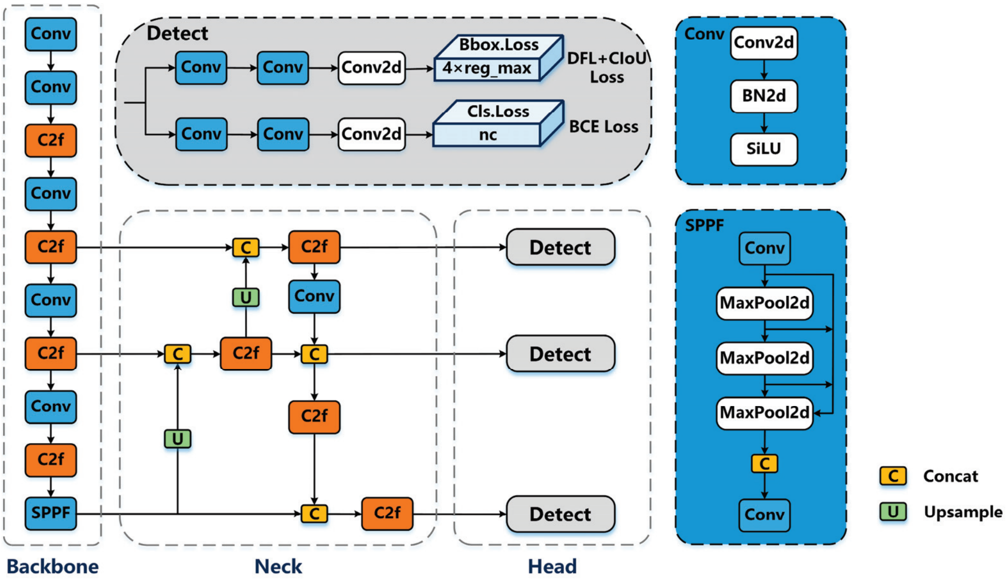


Figure 6. Architecture of YOLOv8 [19].

2.2.1. C2fA Module

When a neural network extends to multiple convolutional layers, its ability to enhance feature representation learning becomes significant. However, increasing deep convolutional layers consumes large memory and computational resources, which is a primary challenge in constructing deep CNNs. To improve model performance without escalating computational complexity, attention mechanisms have emerged as an effective alternative [26]. These mechanisms strengthen the learning of discriminative features and are easily integrated into neural networks due to their flexible structure. The YOLO-Thyroid model incorporates the CA mechanism [27], which embeds positional information into channel attention. By employing one-dimensional pooling operations to capture feature encodings along horizontal and vertical directions, CA effectively integrates spatial coordinate information into attention maps. This enhancement improves the model's ability to perceive target positions and increases the accuracy of feature extraction.

Drawing inspiration from the CA mechanism, the C2fA module was designed and introduced. As illustrated in Figure 7, the C2fA module combines the efficient feature aggregation of the C2f module with the CA mechanism. The architecture of the C2fA module is detailed as follows.

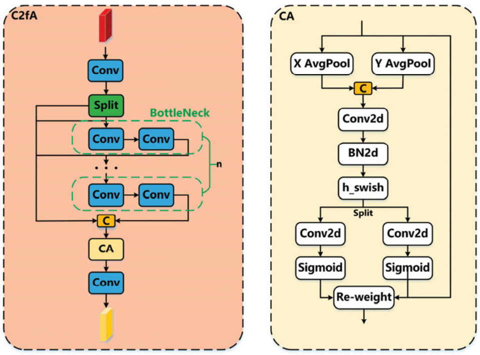


Figure 7. C2fA module design.

The module takes an input feature map $\mathbf{X}$ (a matrix) from the previous layer. First, $\mathbf{X}$ is split into two parts, $\mathbf{X}_1$ and $\mathbf{X}_2$:

$$\mathbf{X}_1, \mathbf{X}_2 = \mathrm{Split}(\mathbf{X}) \tag{1}$$

This splitting operation allows the network to process different portions of the features separately, enabling the diversification of the feature extraction process. The second part $\mathbf{X}_2$ is then passed sequentially through n bottleneck blocks, producing a series of intermediate outputs denoted as $\mathbf{X}_2^{(i)}$ for $i = 1, \ldots, n$. This process extracts higher-level features while reducing computational load. Each bottleneck operation can be described as:

$$\mathbf{X}_2^{(i)} = \mathit{Bottleneck}\left(\mathbf{X}_2^{(i-1)}\right), \ \mathit{where}\ \mathbf{X}_2^{(0)} = \mathbf{X}_2 \tag{2}$$

This bottleneck structure effectively reduces dimensionality and focuses on essential features, thereby enhancing computational efficiency. After the bottleneck transformation, the original and processed feature maps are concatenated to form the output:

$$\mathbf{Y} = \mathrm{Concat}\left(\mathbf{X}_1\,, \mathbf{X}_2\,, \mathbf{X}_2^{(1)}, \ldots, \mathbf{X}_2^{(n)}\right) \tag{3}$$

The concatenation allows the network to combine unprocessed and processed features, providing a richer and more diverse set of features for subsequent layers. It ensures that both the original information and the enhanced features contribute to the learning process.

To enhance the spatial and channel-wise feature representation, a CA mechanism is applied to $\mathbf{Y}$. The output is then passed through another 1×1 convolution to adjust the channel size, producing the final output:

$$\mathrm{Output} = \mathrm{Conv}1 \times 1(CA(\mathbf{Y})) \tag{4}$$

By integrating the CA mechanism into the C2f layer, the C2fA module enables the model to identify important features more precisely, thereby improving the detection accuracy of nodules in different categories. Specifically, the last three C2f layers in the backbone network were replaced with C2fA modules. This strategic placement ensures that attention mechanisms are applied to higher-level feature maps. By integrating attention into feature processing, the model can better capture spatial information and important features while maintaining computational efficiency. The experimental results demonstrate that the model incorporating the C2fA module surpasses the original model in both accuracy and efficiency, verifying the effectiveness of the proposed method.

2.2.2. Loss Function

In nodule detection tasks, there is often a significant class imbalance among different categories: some categories have a large number of samples, while others are relatively scarce. This imbalance can cause the model to be biased toward predicting categories with more samples, thereby affecting the overall detection performance. To address this issue, a class-weighted binary cross-entropy (CW-BCE) loss function was introduced. By assigning appropriate weights to each category inversely proportional to its sample frequency, the model focuses more on underrepresented categories during training, thus improving detection performance for these categories.

The weight w_i for each category i is calculated based on its sample frequency:

$$w_i = \frac{N}{C \times n_i} \tag{5}$$

where N is the total number of samples in the dataset, C is the total number of categories, and n_i is the number of samples in category i. This formula ensures that categories with fewer samples receive larger weights, effectively balancing their influence during training. The CW-BCE loss for each sample is then defined as:

$$L_{\text{CW}-\text{BCE}} = -w_c[y\ log(p) + (1-y)\ log(1-p)] \tag{6}$$

Here, y is the true label of the sample, set to 1 if the sample belongs to category c and 0 otherwise. p is the predicted probability that the sample belongs to category c, obtained after applying the Sigmoid function to the model's output, and w_c is the weight corresponding to category c.

Consider three categories $[c_1,\ c_2,\ c_3]$ with the following number of samples: [1000, 500, 100] The total number of samples is $N = 1600$. The weights for each category $[w_1,\ w_2,\ w_3]$ are calculated as [0.53, 1.07, 5.33]. Category c_3, which has the fewest samples, receives the highest weight. This higher weight increases the contribution of category c_3 samples to the loss function, encouraging the model to focus more on accurately classifying these underrepresented samples during training. Class weights are incorporated into the loss function to adjust the loss contribution of each sample based on its category weight. This approach ensures that during training, the model pays more attention to underrepresented categories, reducing misclassification rates for these categories and effectively mitigating the impact of class imbalance on overall model performance.

2.2.3. Outcome Measure

To further enhance the localization accuracy of bounding boxes, the SIoU loss function [28] was introduced into the model. SIoU is an improved IoU loss that comprehensively considers the geometric relationship between the predicted box and the ground truth box, including overlap area, center point distance, area ratio, and shape differences. By simultaneously considering these discrepancies, the SIoU loss guides the model to learn more accurately, improving localization precision and regression accuracy. The SIoU loss function is expressed as:

$$L_{box} = 1 - \text{IoU} + \frac{\Delta + \Omega}{2} \tag{7}$$

$$\text{IoU} = \frac{intersection}{union} \tag{8}$$

where Δ represents the distance loss, quantifying the distance between the center points of the predicted box and the ground truth box, and incorporating an angle cost to make the penalty of the distance loss positively correlated with the angle difference; Ω represents the shape loss, penalizing the differences in width and height between the predicted box and the ground truth box; IoU is the Intersection over Union that calculates the ratio of the intersection area over the union area between the predicted box and the ground truth box. mAP (mean average precision) is a metric used to evaluate the performance of object detection models, representing the average precision across all classes at different threshold levels, providing a comprehensive assessment of the model's detection capabilities.

3. Results

3.1. Experimental Setting

To validate the effectiveness of the proposed YOLO-Thyroid model, comprehensive evaluations were conducted. The DDTI dataset [20], professionally labeled and preprocessed, was used in evaluation to ensure the reliability and validity of both the training and testing phases. Nodule diagnoses adhere to the TIRADS scoring system [21], encompassing six nodule categories (2, 3, 4a, 4b, 4c, and 5), including both benign (2 and 3) and malignant (4a, 4b, 4c, and 5) nodules. Table 2 summarizes the sample counts in the training, validation, and test sets of the augmented DDTI dataset. To enhance the dataset, data augmentation techniques were applied to the original dataset, including flipping [29], rotation [29], cropping [29], shearing [30], and brightness [31] adjustment. As a result, the number of samples in the augmented dataset increased from 339 to 813.

In the experiments, all methods were implemented using Ultralytics [19] on an NVIDIA GeForce RTX 3080 GPU (manufactured by NVIDIA Corporation, Santa Clara, California, USA) with 8704 CUDA cores and 10 GB of memory. For thyroid nodule detection, macro-average precision (P), macro-average recall (R), mAP0.5, and mAP0.5:0.95 were utilized to thoroughly evaluate model performance. Precision measures the accuracy of the model's positive predictions, while recall assesses the model's ability to identify all positive samples. mAP0.5 represents the mAP at an IoU threshold of 0.5, and mAP0.5:0.95 represents the mAP across IoU thresholds ranging from 0.5 to 0.95. Additionally, to assess the training and inference efficiency of the model, the training time (Tr, minutes per epoch), testing time (Te, milliseconds per image), parameters (Params), and the number of floating-point operations (FLOPs) were recorded. These metrics provide insights into the computational complexity and resource requirements, contributing to a comprehensive assessment of each model's efficiency and scalability. During model training, the batch size was set to 16, and training was conducted over 300 epochs. Dropout techniques and an early stopping strategy were employed to prevent overfitting.

3.2. Ablation Studies

To evaluate the impact of each module in the proposed model on ultrasound nodule detection performance, ablation experiments were conducted, with the results presented in

Table 3. Starting from the baseline model YOLOv8-N, the C2fA module, the CW-BCE loss function, and the SIoU loss function were sequentially added.

As detailed in Table 3, the base YOLOv8-N model achieved a precision of 53.7%, recall of 37.4%, mAP0.5 of 37.4%, and mAP0.5:0.95 of 25.7%. Introducing the C2fA module increased precision to 67.0% but reduced recall to 28.9%, indicating a trade-off between accuracy and nodule detection. Employing the CW-BCE loss function resulted in an improvement of recall to 39.5% and an increase in the mAP metrics. These findings demonstrate that, although CW-BCE is a simple and widely used technique, its application in this manner effectively enhances the model's sensitivity to the minority class. This validates its effectiveness in the specific application of detecting thyroid nodules, where class imbalance poses a significant challenge. The SIoU loss increased recall significantly to 56.6% and improved mAP metrics, though precision decreased to 25.4%, suggesting more target detections but higher false positives due to boundary optimization. The integration of all components yielded the best overall performance, with mAP0.5 reaching 43.6%, mAP0.5:0.95 increasing to 28.7%, average detection precision being 54%, and the detection of nodules containing at least one suspicious feature recall of 58.2%, respectively. This synergy enhances feature extraction, addresses class imbalance, and optimizes boundary regression. In summary, the C2fA module improves feature representation, the weighted binary cross-entropy loss addresses class imbalance, and the SIoU loss enhances boundary localization. Together, they achieve a balance between precision and recall, significantly advancing ultrasound nodule detection performance and highlighting the effectiveness of the proposed method.

Table 3. Results of ablation experiments evaluating module impact.

Base		Components		P	R	mAP0.5	mAP0.5:0.95	Tr	Te
Model	C2fA	Lcwbce	Lsiou	(%)↑	(%)↑	(%)↑	(%)↑	(min/epoch) ↓	(ms/image) ↓
YOLOv8-N				53.7	37.4	37.4	25.7	0.057	9.5
	√			**67.0**	28.9	38.9	25.3	0.058	7.7
		√		35.3	39.5	40.5	28.2	**0.054**	8.6
			√	25.4	**56.6**	39.9	27.4	0.055	8.2
	√		√	55.8	33.5	36.4	25.5	0.059	8.2
	√	√		60.5	36.5	39.2	27.0	0.058	7.8
		√	√	62.1	34.2	39.1	27.3	0.055	**7.2**
	√	√	√	54.0	41.6	**43.6**	**28.7**	0.058	8.1

↑ indicates superior performance with a higher value, while ↓ indicates superior performance with a lower value. √ indicates the components existing. The best performance in the column was bolded.

Figure 8 illustrates the training and validation dynamics of the YOLO-Thyroid model over 300 epochs. The plots depict the evolution of various loss metrics, including box loss, class loss, and distribution-focused loss (DFL loss), alongside performance metrics such as macro-average precision, macro-average recall, mAP0.5, and mAP0.5:0.95. The training losses (top row) consistently decline, indicating effective learning and convergence, while the validation losses (bottom row) also decrease, suggesting good generalization capabilities in this specific dataset. Notably, precision and recall metrics progressively improve, reflecting the model's enhanced ability to accurately detect and classify nodules. The mAP measures show significant growth, underscoring the model's robust performance across varying IoU thresholds. These overall trends confirm the efficacy of the proposed model's improvements in enhancing detection accuracy and reliability.

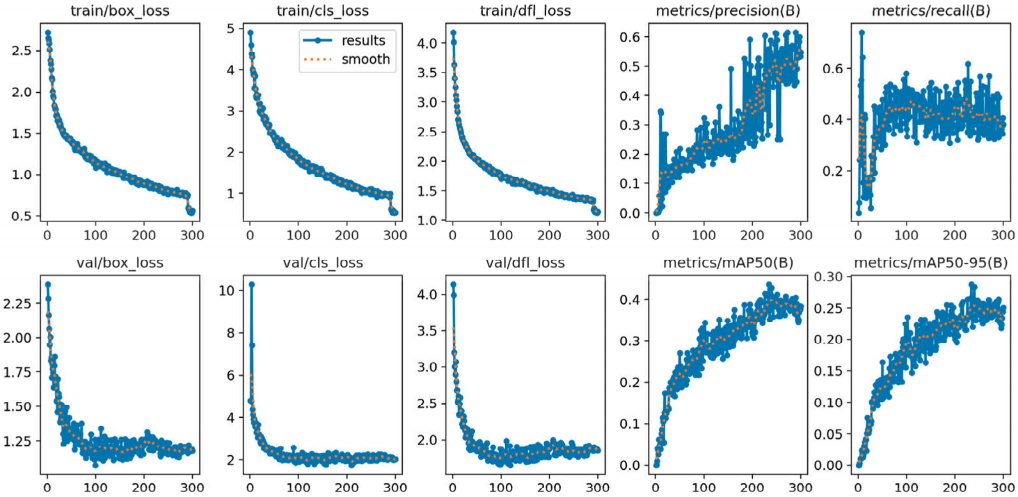


Figure 8. The training and validation dynamics of the YOLO-Thyroid model over 300 epochs. The orange dots indicate the smoothed curves.

3.3. Comparison with State-of-the-Art Methods

In this section, the proposed model is compared with state-of-the-art YOLO series models (YOLOv5 [32], YOLOv6 [33], YOLOv8 [19], YOLOv9 [34], YOLOv10 [35]), and DETR series models (RT-DETR-R50 [36], RT-DETR-L [36]). All models were trained and evaluated on the same dataset and under identical experimental conditions to ensure fairness and reliability in the comparison. The experimental results are presented in Table 4.

Table 4. The comparative performance of the proposed model with state-of-the-art YOLO and DETR series models.

Model Type	Model	mAvg-P (%) ↑	mAvg-R (%) ↑	mAP0.5 (%) ↑	mAP0.5:0.95 (%) ↑	Tr (min/epoch) ↓	Te (ms/image) ↓	Params (M) ↓	FLOPs (G) ↓
DETR	RT-DETR-R50	37.7	24.7	23.8	14.5	0.304	15.4	40.00	125.6
	RT-DETR-L	26.8	27.7	24.5	16.0	0.277	15.2	30.51	103.5
YOLO	YOLOv5-N	**59.9**	33.1	32.4	21.8	**0.055**	9.0	2.39	**7.1**
	YOLOv5-L	59.0	33.6	35.3	22.8	0.257	13.3	50.67	134.7
	YOLOv6-N	57.9	30.5	38.9	26.6	**0.055**	**8.0**	4.04	11.8
	YOLOv6-L	13.3	21.3	17.3	10.3	0.453	19.4	105.73	391.2
	YOLOv8-N	53.7	37.4	37.4	25.7	0.057	9.5	2.87	8.1
	YOLOv8-L	40.4	39.6	31.5	22.3	0.245	14.1	41.59	164.8
	YOLOv9-T	22.1	41.5	34.9	24.1	0.096	9.5	**1.88**	7.6
	YOLOv9-C	40.6	35.5	39.9	23.7	0.211	14.1	24.15	102.3
	YOLOv10-N	51.1	27.9	31.1	20.7	0.07	8.6	2.57	8.2
	YOLO-Thyroid	54.0	41.6	**43.6**	**28.7**	0.058	8.1	2.89	8.1

↑ indicates superior performance with a higher value, while ↓ indicates superior performance with a lower value. The best performance in the column was bolded.

Table 5 presents the performance of precision (P) and recall (R) for different TIRADS classes (4a, 4b, 4c, and 5), along with their weight average values (Avg). Precision indicates the proportion of correctly detected nodules containing at least one relevant feature among all detected nodules, while recall reflects the proportion of true nodules that were successfully detected with at least one suspicious feature.

YOLO-Thyroid demonstrated excellent performance across all metrics. Specifically, it achieved a precision of 54.0%, a recall of 58.2%, an mAP0.5 of 43.6%, and an mAP0.5:0.95 of 28.7%, outperforming the other models overall. This indicated that the proposed model can detect more true positives while reducing false detections, achieving a favorable balance between precision and recall. Additionally, YOLO-Thyroid had a parameter count of 2.89 M, FLOPs of 8.1 G, and an inference time of 8.1 milliseconds per image. It reduced the complexity and computational load while maintaining high accuracy, making it suitable for applications in resource-constrained environments.

Table 5. Performance of Precision (P) and Recall (R) Based on Nodules Containing at Least One Relevant Feature Across Different TIRADS Classes.

	4a	**4b**	**4c**	**5**	**Weight Avg**
P (%)	61.2	36.9	38.2	58	49.9
R (%)	68.5	57.1	42.9	55.9	58.2

In contrast, other models such as YOLOv5-N, YOLOv6-N, YOLOv8-N, YOLOv9-T, and YOLOv10-N, although they also had smaller model sizes, did not match YOLO-Thyroid in detection performance. Larger models like YOLOv9-C and YOLOv10-L, while showing improvements in some metrics, had significantly increased parameter counts and FLOPs, and their inference speeds decreased noticeably. Moreover, the DETR series models had larger scales and higher computational demands but did not exhibit corresponding advantages in detection performance and inference speed, rendering them inferior to YOLO-Thyroid. This further confirms the advantages of the YOLO-Thyroid model in structural design and optimization, achieving higher detection performance with a smaller model size. These results validate the effectiveness of YOLO-Thyroid for specific medical imaging applications, providing important technical support for diagnosis.

To visually illustrate the superior performance of the proposed method in nodule detection tasks, the actual detection results are presented in Figure 9. This figure compares YOLO-Thyroid with the detection results of other models. As shown in the figure, YOLO-Thyroid can more accurately locate and identify nodules, maintaining high detection accuracy even in complex scenarios and significantly reducing missed detections and false positives. This demonstrates the effectiveness and reliability of YOLO-Thyroid in practical applications.

The generalization ability of the model was assessed by evaluating its performance on both the original and augmented datasets. As shown in Table 6, the YOLO-Thyroid model demonstrated significant improvement with data augmentation: a macro-average recall from 34.5% to 41.6%, and an mAP0.5 from 33.0% to 43.6%. This improvement suggests that data augmentation enhanced the model's ability to learn robust features and generalize to unseen data.

Table 6. A performance comparison of the YOLO-Thyroid model on original and augmented datasets.

Dataset	**mAvg-P (%) ↑**	**mAvg-R (%) ↑**	**mAP0.5 (%) ↑**	**mAP0.5:0.95 (%) ↑**
Original	35.8	34.5	33.0	18.5
Augmented	54.0	41.6	43.6	28.7

↑ indicates superior performance with a higher value.

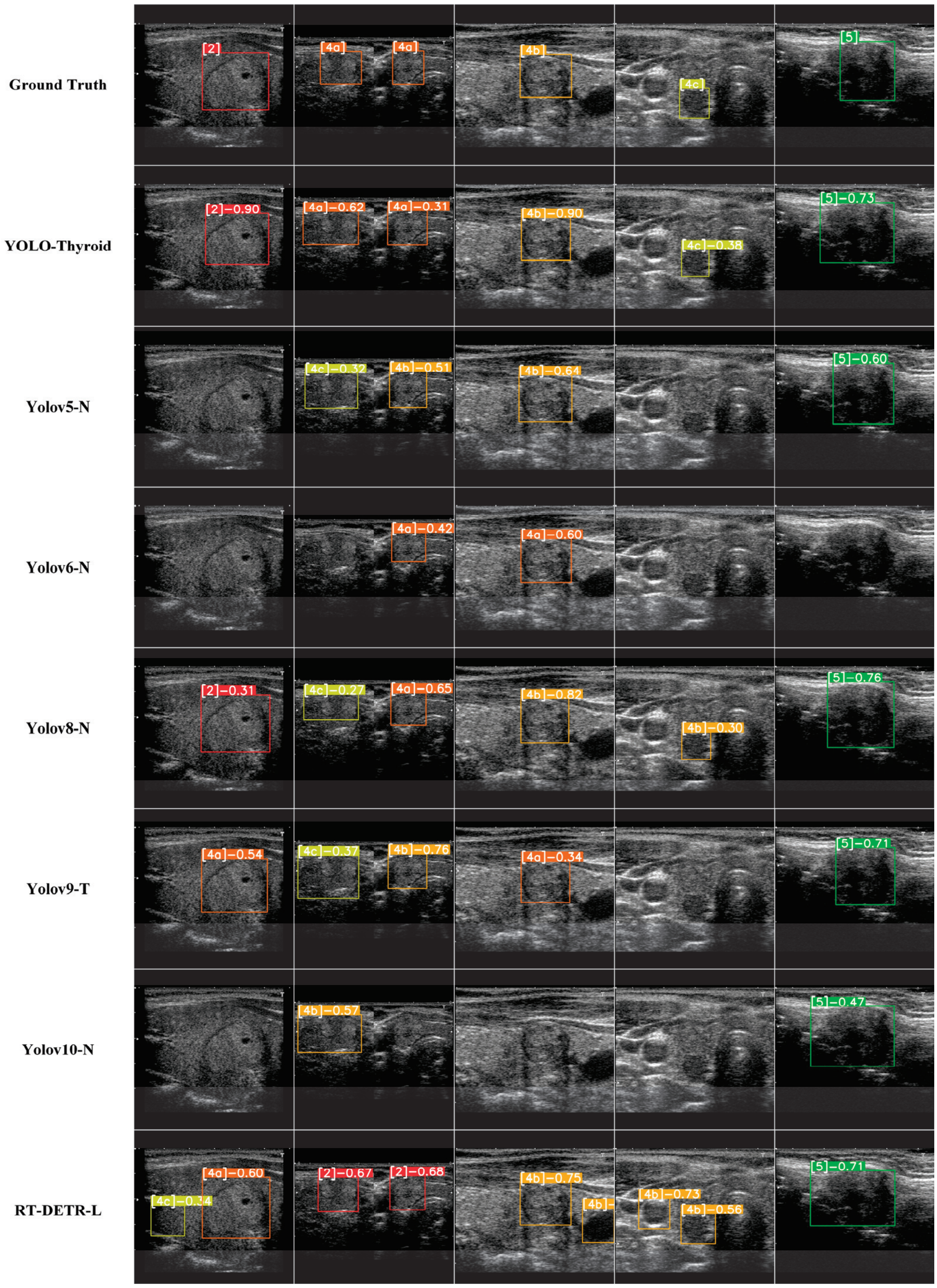


Figure 9. A visual comparison of the nodule detection results across different models. Each detected nodule is highlighted with a bounding box in different colors and labeled with a number in brackets, indicating the class, followed by the confidence score of the prediction.

4. Discussion

This article presents the YOLO-Thyroid model, designed for the task of ultrasound nodule detection. YOLO-Thyroid effectively enhances detection performance through the introduction of the C2fA module and improved loss functions. The C2fA module combines spatial and feature information, enabling the model to better capture the features of nodules, particularly improving detection accuracy for small and complex nodules.

This enhancement resulted in a precision increase by 18% and an improvement in mAP0.5 by 10.6% compared to the baseline model. The improved loss functions consider the class imbalance issue inherent in the dataset and multiple factors such as the target's position, size, and shape, leading to more accurate localization and reduced detection errors. This is evidenced by a detection macro-average recall increase by 7%. The experimental results demonstrate that this model outperforms current state-of-the-art object detection methods across various performance metrics. Specifically, YOLO-Thyroid achieved an increase in mAP0.5 to 43.6%, compared to 37.4% from the baseline model, indicating a substantial enhancement in overall detection accuracy. This outcome validates the effectiveness of YOLO-Thyroid for specific medical imaging tasks and provides significant technical support for clinical diagnosis.

In object detection tasks, such as thyroid nodule detection, mAP is the standard evaluation metric as it effectively captures both precision and recall across multiple classes and detection thresholds [37]. Precision reflects the model's ability to correctly identify relevant objects, expressed as the percentage of true positive predictions among all positive detections. Recall, on the other hand, measures the model's ability to detect all relevant cases, represented by the percentage of true positive predictions among all ground-truth bounding boxes [37,38]. mAP is widely adopted in major benchmarks and challenges, including PASCAL VOC [39] and COCO [40], because it provides a comprehensive assessment of a model's performance in both classification and localization. This dual capability is crucial in tasks where the accurate localization of thyroid nodules is essential.

Several studies have explored semantic segmentation models like SegNet [41], DeepLab [42], and PSPNet [43], which offer precise pixel-level delineation but come with significant drawbacks. DeepLab, for example, has high GPU utilization due to its complex structure, while models like PSPNet and SegNet are slower because of their intricate architectures, making them less efficient for real-time applications [44–46]. These models are computationally intensive, requiring more processing power and time, which limits their practicality in large-scale clinical screening [47], where speed is crucial. In contrast, YOLO, a one-stage framework, overcomes the shortcomings of two-stage methods like R-CNN [48] by simplifying the detection process. YOLO performs both object localization and classification in a single network, resulting in faster processing and reduced computational overhead [49]. This speed and efficiency make YOLO more suitable for real-time thyroid nodule detection, especially in resource-constrained environments, where rapid decision-making is essential [49,50].

Recent studies have integrated CA mechanisms with YOLOv8 in various applications [51,52]. However, their methods are not directly transferable to ultrasound nodule detection due to the unique challenges of medical ultrasound imaging. The proposed approach introduces a customized C2fA module specifically tailored for ultrasound nodule detection, involving modifications that differ from those in previous studies. Comparing YOLO-Thyroid to the two-stage detection method used by [53], which reports higher precision on the DDTI dataset, highlights important considerations for clinical applications. Two-stage detectors excel in precision due to their sequential proposal and refinement processes but are computationally intensive. Although the one-stage YOLO-Thyroid model has a lower precision, it offers significant advantages in inference speed and computational efficiency, which are critical for real-time ultrasound imaging and prompt clinical decision-making. Additionally, the datasets used in the two studies differ substantially. The DDTI dataset may contain higher-quality images or specific characteristics that favor higher detection precision, whereas the dataset used in this study encompasses a broader spectrum of ultrasound images with varying complexity and noise. These differences emphasize the importance of considering dataset characteristics when evaluating and comparing model performance.

This study further analyzes the model's ability to differentiate and detect nodules across different TIRADS classes to identify categories that require special attention for clinical decision-making. Sensitivity (measured through recall) was used as the primary indicator of the model's effectiveness in detecting nodules within each risk level, particularly for TIRADS 4a, 4b, 4c, and 5 classes.

For TIRADS classes 4a, 4b, and 4c:

- TIRADS 4a demonstrated the highest recall among these three categories, reflecting the model's ability to effectively detect nodules in this lower-risk class. However, nodules in TIRADS 4a generally have fewer malignant features, and their clinical urgency is relatively lower compared to TIRADS 4b and 4c. As a result, while maintaining high recall for 4a is important for ensuring comprehensive screening, it is not the most critical category for guiding clinical decision-making regarding further investigations.
- TIRADS 4b achieved a higher recall compared to TIRADS 4c, highlighting the model's stronger ability to identify nodules in this category. Given its maximum malignancy risk of 80%, TIRADS 4b represents a key decision-making threshold where detecting sufficient diagnostic features is critical for recommending further investigations, such as fine-needle aspiration biopsy (FNAB).
- TIRADS 4c, which shares the same maximum malignancy risk as 4b, exhibited a relatively lower recall. This suggests that further optimization is needed to improve detection for this category to ensure consistent performance across all high-risk classes.
- For TIRADS 5, while recall alone cannot fully evaluate its significance due to its minimum malignancy risk already exceeding 80%, this category remains clinically crucial. Nodules in TIRADS 5 often exhibit obvious malignant features, enabling quicker triage and more immediate clinical action. High detection performance in this category ensures that patients with overtly high-risk nodules are promptly referred for further investigation and treatment, which is vital for improving clinical efficiency.

Based on these findings, TIRADS 4b exhibits the highest detection sensitivity (i.e., two features suspected of being malignant, with a 10% to 80% malignant risk). Therefore, this risk class is recommended as the primary reference for supporting clinical decisions. Meanwhile, maintaining high recall for TIRADS 4a supports comprehensive screening efforts, and consistent performance in TIRADS 5 ensures the rapid identification and triage of overtly malignant nodules.

By establishing TIRADS 4b as the sensitivity benchmark and balancing detection performance across other categories, this study provides a quantitative framework for identifying high-risk nodules that require prompt follow-up. This approach ensures optimal utilization of clinical resources while minimizing missed diagnoses.

Despite the excellent results achieved by the YOLO-Thyroid model, there are still limitations and areas for improvement. First, the research is primarily based on a specific dataset, and the scale and diversity of this dataset may affect the model's generalization capability. In future work, the model's performance will be validated on larger and more diverse datasets to enhance the model's generalizability and robustness. Incorporating other advanced models or technologies, such as the MAMBA [54] and KAN [55] architectures, may further refine the model's performance. Additionally, exploring multimodal data integration, such as combining ultrasound images with patient demographic and clinical data, may improve the model's diagnostic capabilities. Furthermore, combining denoising and image enhancement techniques could reduce the impact of noise and variability in ultrasound images, thereby improving model efficacy. Potential applications of the YOLO-Thyroid model extend beyond thyroid nodule detection. By retraining the model on different medical imaging modalities, such as MRI or CT scans, it could detect other illnesses and abnormalities, thus expanding its diagnostic utility across the healthcare spectrum. Moreover, the model can be adapted for critical applications like fall detection in the elderly [56] by processing visual data from monitoring devices to provide real-time alerts and enhance safety. These adaptations demonstrate the model's versatility and its potential to significantly contribute to patient care and safety in diverse contexts.

5. Conclusions

In this paper, a YOLO-based model, YOLO-Thyroid, is proposed for ultrasound nodule detection. By introducing the C2fA module and improved loss functions, YOLO-Thyroid achieves an optimal balance between detection performance and model complexity. Through a series of ablation experiments, the effectiveness of the C2fA module and the new loss function in enhancing model performance has been verified. These improvements strengthen the model's ability to extract and represent nodule features, increasing detection accuracy for small and complex nodules. Simultaneously, the new loss function enables more precise boundary regression, reducing detection errors. Furthermore, comparative results with state-of-the-art models indicate that the YOLO-Thyroid model achieves superior performance across all evaluation metrics.

The mAP is a primary metric for evaluating detection performance, balancing precision and recall across classes and thresholds. Recall is particularly important in medical contexts to ensure all relevant cases are identified. In this study, the YOLO-Thyroid model was developed and optimized to achieve a high mean average precision (mAP) of 43.6% (mAP@0.5) and a recall of 58.2%. These results indicate superior performance compared to state-of-the-art models.

This research provides an efficient and reliable solution for automatic nodule detection, which is anticipated to play a significant role in clinical diagnosis. In future research, the dataset will be expanded by incorporating more ultrasound images from different devices and patient populations to enhance the model's applicability.

Author Contributions: Conceptualization, J.C.-W.C.; methodology, J.C.-W.C.; software, S.W. and Z.-A.Z.; validation, S.W. and Z.-A.Z.; formal analysis, Z.-A.Z. and Y.C.; investigation, Y.C.; resources, J.C.-W.C.; data curation, S.W. and Z.-A.Z.; writing—original draft preparation, S.W.; writing—review and editing, Y.-J.M. and J.C.-W.C.; supervision, J.C.-W.C.; project administration, J.C.-W.C.; funding acquisition, J.C.-W.C. 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 original data presented in the study are openly available from the Thyroid Digital Image Database at http://cimalab.unal.edu.co/?lang=en&mod=program&id=5 (accessed on 5 October 2024).

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

Abbreviations

The following abbreviations are used in this manuscript:

ATA	American Thyroid Association
C2	CSP Bottleneck with Two Convolutions
CA	Coordinate Attention
CNNs	Convolutional Neural Networks
CSP	Cross Stage Partial
CT	Computed Tomography
CW-BCE	Class-Weighted Binary Cross-Entropy
DDTI	Digital Database of Thyroid Images
DFL	Loss Distribution-Focused Loss
FLOPs	Floating-Point Operations
FNAB	Fine-Needle Aspiration Biopsy
IoU	Intersection Over Union
MRI	Magnetic Resonance Imaging
NMS	Non-Maximum Suppression
P	Precision
mAvg-P	Macro-Average Precision
Params	Parameters
R	Recall
mAvg-R	Macro-Average Recall
SIoU	SCYLLA-IoU
Te	Testing Time
TIRADS	Thyroid Imaging Reporting And Data System
Tr	Training Time
US	Ultrasound
YOLO	You Only Look Once

Appendix A

The algorithm outlines the process of converting the original DDTI dataset's segmentation labels into the YOLO format required for object detection. For each image and its corresponding segmentation label, the algorithm extracts the minimum bounding rectangle that encompasses the target nodule by identifying the smallest and largest x and y coordinates where the nodule is present. It then calculates the center coordinates of this bounding box by averaging the minimum and maximum x and y values and determines the width and height by computing the difference between the maximum and minimum coordinates. The normalized center coordinates, width, and height, along with the class identifier of the nodule, are combined to form the YOLO label for each image.

Algorithm A1 Conversion of Original DDTI to YOLO Format

```
Input: Original DDTI with images and their corresponding segmentation labels
Output: YOLO-formatted label dataset
for each image $I_i$ and its segmentation label $L_i$ in Original DDTI do
    Load $I_i$ and $L_i$
    Extract bounding box coordinates:
        $x_{min}, x_{max} = min/max(x \mid L_i(x, y) = 1)$
        $y_{min}, y_{max} = min/max(y \mid L_i(x, y) = 1)$
    Compute YOLO format parameters:
        $x_{center} = (x_{min} + x_{max})/2$
        $y_{center} = (y_{min} + y_{max})/2$
        width $= x_{max} - x_{min}$
        height $= y_{max} - y_{min}$
    Normalize coordinates:
        $x_{center_{norm}} = x_{center}/$image_width
        $y_{center_{norm}} = y_{center}/$image_width
      width$_{\text{norm}}$ = width/image_width
      height$_{\text{norm}}$ = height/image_height
    Create YOLO label:
        $L_{yolo} = [class\_id, x_{center_{norm}}, y_{center_{norm}},$ width$_{\text{norm}}$, height$_{\text{norm}}$
    Save $L_{yolo}$ to YOLO label file
end for
```

References

1. Siegel, R.L.; Miller, K.D.; Fuchs, H.E.; Jemal, A. Cancer Statistics, 2022. *CA Cancer J. Clin.* **2022**, *72*, 7–33. [CrossRef] [PubMed]
2. Cancer of the Thyroid—Cancer Stat Facts. Available online: https://seer.cancer.gov/statfacts/html/thyro.html (accessed on 29 September 2024).
3. Mao, Y.-J.; Zha, L.-W.; Tam, A.Y.-C.; Lim, H.-J.; Cheung, A.K.-Y.; Zhang, Y.-Q.; Ni, M.; Cheung, J.C.-W.; Wong, D.W.-C. Endocrine Tumor Classification Via Machine-Learning-Based Elastography: A Systematic Scoping Review. *Cancers* **2023**, *15*, 837. [CrossRef] [PubMed]
4. Zhang, X.-Y.; Wei, Q.; Wu, G.-G.; Tang, Q.; Pan, X.-F.; Chen, G.-Q.; Zhang, D.; Dietrich, C.F.; Cui, X.-W. Artificial Intelligence-Based Ultrasound Elastography for Disease Evaluation-a Narrative Review. *Front. Oncol.* **2023**, *13*, 1197447. [CrossRef] [PubMed]
5. Zheng, Z.; Su, T.; Wang, Y.; Weng, Z.; Chai, J.; Bu, W.; Xu, J.; Chen, J. A Novel Ultrasound Image Diagnostic Method for Thyroid Nodules. *Sci. Rep.* **2023**, *13*, 1654. [CrossRef]
6. Hairu, L.; Yulan, P.; Yan, W.; Hong, A.; Xiaodong, Z.; Lichun, Y.; Kun, Y.; Ying, X.; Lisha, L.; Baoming, L.; et al. Elastography for the Diagnosis of High-Suspicion Thyroid Nodules Based on the 2015 American Thyroid Association Guidelines: A Multicenter Study. *BMC Endocr. Disord.* **2020**, *20*, 43. [CrossRef]
7. Iannuccilli, J.D.; Cronan, J.J.; Monchik, J.M. Risk for Malignancy of Thyroid Nodules as Assessed by Sonographic Criteria: The Need for Biopsy. *J. Ultrasound Med.* **2004**, *23*, 1455–1464. [CrossRef]
8. Sarkar, O.; Islam, R.; Syfullah, K.; Islam, T.; Ahamed, F.; Ahsan, M.; Haider, J. Multi-Scale Cnn: An Explainable Ai-Integrated Unique Deep Learning Framework for Lung-Affected Disease Classification. *Technologies* **2023**, *11*, 134. [CrossRef]
9. Khonina, S.N.; Kazanskiy, N.L.; Oseledets, I.V.; Nikonorov, A.V.; Butt, M.A. Synergy between Artificial Intelligence and Hyperspectral Imagining—A Review. *Technologies* **2024**, *12*, 163. [CrossRef]
10. Kshatri, S.S.; Singh, D. Convolutional Neural Network in Medical Image Analysis: A Review. *Arch. Comput. Methods Eng.* **2023**, *30*, 2793–2810. [CrossRef]
11. Zheng, T.; Qin, H.; Cui, Y.; Wang, R.; Zhao, W.; Zhang, S.; Geng, S.; Zhao, L. Segmentation of Thyroid Glands and Nodules in Ultrasound Images Using the Improved U-Net Architecture. *BMC Med. Imaging* **2023**, *23*, 56. [CrossRef]
12. Zhou, Y.-T.; Yang, T.-Y.; Han, X.-H.; Piao, J.-C. Thyroid-Detr: Thyroid Nodule Detection Model with Transformer in Ultrasound Images. *Biomed. Signal Process. Control* **2024**, *98*, 106762. [CrossRef]
13. Chen, G.; Tan, G.; Duan, M.; Pu, B.; Luo, H.; Li, S.; Li, K. Mlmseg: A Multi-View Learning Model for Ultrasound Thyroid Nodule Segmentation. *Comput. Biol. Med.* **2024**, *169*, 107898. [CrossRef] [PubMed]
14. Ghosh, K.; Bellinger, C.; Corizzo, R.; Branco, P.; Krawczyk, B.; Japkowicz, N. The Class Imbalance Problem in Deep Learning. *Mach. Learn.* **2024**, *113*, 4845–4901. [CrossRef]
15. Montalbo, F.J.P. A Computer-Aided Diagnosis of Brain Tumors Using a Fine-Tuned Yolo-Based Model with Transfer Learning. *KSII Trans. Internet Inf. Syst.* **2020**, *14*, 4816–4834.
16. Al-Antari, M.A.; Han, S.-M.; Kim, T.-S. Evaluation of Deep Learning Detection and Classification Towards Computer-Aided Diagnosis of Breast Lesions in Digital X-Ray Mammograms. *Comput. Methods Programs Biomed.* **2020**, *196*, 105584. [CrossRef] [PubMed]
17. Su, Y.; Liu, Q.; Xie, W.; Hu, P. Yolo-Logo: A Transformer-Based Yolo Segmentation Model for Breast Mass Detection and Segmentation in Digital Mammograms. *Comput. Methods Programs Biomed.* **2022**, *221*, 106903. [CrossRef]

18. Rouzrokh, P.; Ramazanian, T.; Wyles, C.C.; Philbrick, K.A.; Cai, J.C.; Taunton, M.J.; Kremers, H.M.; Lewallen, D.G.; Erickson, B.J. Deep Learning Artificial Intelligence Model for Assessment of Hip Dislocation Risk Following Primary Total Hip Arthroplasty from Postoperative Radiographs. *J. Arthroplast.* **2021**, *36*, 2197–2203.e3. [CrossRef]
19. Jocher, G.; Chaurasia, A.; Qiu, J. Ultralytics YOLOv8. 2023. Available online: https://github.com/ultralytics/ultralytics (accessed on 29 September 2024).
20. Pedraza, L.; Vargas, C.; Narváez, F.; Durán, O.; Muñoz, E.; Romero, E. An Open Access Thyroid Ultrasound Image Database. In Proceedings of the 10th International Symposium on Medical Information Processing and Analysis, Cartagena de Indias, Colombia, 14–16 October 2014.
21. Kwak, J.Y.; Han, K.H.; Yoon, J.H.; Moon, H.J.; Son, E.J.; Park, S.H.; Jung, H.K.; Choi, J.S.; Kim, B.M.; Kim, E.K. Thyroid Imaging Reporting and Data System for Us Features of Nodules: A Step in Establishing Better Stratification of Cancer Risk. *Radiology* **2011**, *260*, 892–899. [CrossRef]
22. Zheng, H.; Dong, Z.; Liu, T.; Zheng, H.; Wan, X.; Bao, J. Enhancing Gastrointestinal Submucosal Tumor Recognition in Endoscopic Ultrasonography: A Novel Multi-Attribute Guided Contextual Attention Network. *Expert Syst. Appl.* **2024**, *242*, 122725. [CrossRef]
23. Ding, X.; Liu, Y.; Zhao, J.; Wang, R.; Li, C.; Luo, Q.; Shen, C. A Novel Wavelet-Transform-Based Convolution Classification Network for Cervical Lymph Node Metastasis of Papillary Thyroid Carcinoma in Ultrasound Images. *Comput. Med. Imaging Graph.* **2023**, *109*, 102298. [CrossRef]
24. Garcea, F.; Serra, A.; Lamberti, F.; Morra, L. Data Augmentation for Medical Imaging: A Systematic Literature Review. *Comput. Biol. Med.* **2023**, *152*, 106391. [CrossRef]
25. Goceri, E. Medical Image Data Augmentation: Techniques, Comparisons and Interpretations. *Artif. Intell. Rev.* **2023**, *56*, 12561–12605. [CrossRef] [PubMed]
26. Zhu, H.; Xie, C.; Fei, Y.; Tao, H. Attention Mechanisms in Cnn-Based Single Image Super-Resolution: A Brief Review and a New Perspective. *Electronics* **2021**, *10*, 1187. [CrossRef]
27. Hou, Q.; Zhou, D.; Feng, J. Coordinate Attention for Efficient Mobile Network Design. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2021, Nashville, TN, USA, 20–25 June 2021.
28. Gevorgyan, Z. Siou Loss: More Powerful Learning for Bounding Box Regression. *arXiv* **2022**, arXiv:2205.12740.
29. Maharana, K.; Mondal, S.; Nemade, B. A Review: Data Pre-Processing and Data Augmentation Techniques. *Glob. Transit. Proc.* **2022**, *3*, 91–99. [CrossRef]
30. Zhong, Z.; Zheng, L.; Kang, G.; Li, S.; Yang, Y. Random Erasing Data Augmentation. In Proceedings of the AAAI Conference on Artificial Intelligence 2020, New York, NY, USA, 7–12 February 2020.
31. Alomar, K.; Aysel, H.I.; Cai, X. Data Augmentation in Classification and Segmentation: A Survey and New Strategies. *J. Imaging* **2023**, *9*, 46. [CrossRef]
32. Jocher, G. Ultralytics YOLOv5. 2020. Available online: https://github.com/ultralytics/yolov5 (accessed on 29 September 2024). [CrossRef]
33. Li, C.; Li, L.; Geng, Y.; Jiang, H.; Cheng, M.; Zhang, B.; Ke, Z.; Xu, X.; Chu, X. Yolov6 V3. 0: A Full-Scale Reloading. *arXiv* **2023**, arXiv:2301.05586.
34. Wang, C.-Y.; Yeh, I.-H.; Liao, H.-Y.M. Yolov9: Learning What You Want to Learn Using Programmable Gradient Information. *arXiv* **2024**, arXiv:2402.13616.
35. Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; Ding, G. Yolov10: Real-Time End-to-End Object Detection. *arXiv* **2024**, arXiv:2405.14458.
36. Zhao, Y.; Lv, W.; Xu, S.; Wei, J.; Wang, G.; Dang, Q.; Liu, Y.; Chen, J. Detrs Beat Yolos on Real-Time Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2024, Seattle, WA, USA, 17–21 June 2024.
37. Padilla, R.; Netto, S.L.; Da Silva, E.A. A Survey on Performance Metrics for Object-Detection Algorithms. In Proceedings of the 2020 International Conference on Systems, Signals and Image Processing (IWSSIP) 2020, Niterói, Brazil, 1–3 July 2020.
38. Padilla, R.; Passos, W.L.; Dias, T.L.B.; Netto, S.L.; Da Silva, E.A.B. A Comparative Analysis of Object Detection Metrics with a Companion Open-Source Toolkit. *Electronics* **2021**, *10*, 279. [CrossRef]
39. Everingham, M.; Eslami, S.A.; Van Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes Challenge: A Retrospective. *Int. J. Comput. Vis.* **2015**, *111*, 98–136. [CrossRef]
40. Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft Coco: Common Objects in Context. In Proceedings of the Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014; Part V 13. pp. 740–755.
41. Saood, A.; Hatem, I. COVID-19 Lung Ct Image Segmentation Using Deep Learning Methods: U-Net Versus Segnet. *BMC Med. Imaging* **2021**, *21*, 19. [CrossRef] [PubMed]

42. 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]
43. Yuan, W.; Wang, J.; Xu, W. Shift Pooling Pspnet: Rethinking Pspnet for Building Extraction in Remote Sensing Images from Entire Local Feature Pooling. *Remote Sens.* **2022**, *14*, 4889. [CrossRef]
44. Yang, R.; Yu, Y. Artificial Convolutional Neural Network in Object Detection and Semantic Segmentation for Medical Imaging Analysis. *Front. Oncol.* **2021**, *11*, 638182. [CrossRef]
45. Cheng, L.; Xiong, R.; Wu, J.; Yan, X.; Yang, C.; Zhang, Y.; He, Y. Fast Segmentation Algorithm of Usv Accessible Area Based on Attention Fast Deeplabv3. *IEEE Sens. J.* **2024**, *24*, 24168–24177. [CrossRef]
46. Guo, Z.; Ma, D.; Luo, X. A Lightweight Semantic Segmentation Algorithm Integrating Ca and Eca-Net Modules. *Optoelectron. Lett.* **2024**, *20*, 568–576. [CrossRef]
47. Zeng, P.; Liu, S.; He, S.; Zheng, Q.; Wu, J.; Liu, Y.; Lyu, G.; Liu, P. Tuspm-Net: A Multi-Task Model for Thyroid Ultrasound Standard Plane Recognition and Detection of Key Anatomical Structures of the Thyroid. *Comput. Biol. Med.* **2023**, *163*, 107069. [CrossRef]
48. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2014, Columbus, OH, USA, 23–28 June 2014.
49. Aldughayfiq, B.; Ashfaq, F.; Jhanjhi, N.Z.; Humayun, M. Yolo-Based Deep Learning Model for Pressure Ulcer Detection and Classification. *Healthcare* **2023**, *11*, 1222. [CrossRef]
50. Ragab, M.G.; Abdulkadir, S.J.; Muneer, A.; Alqushaibi, A.; Sumiea, E.H.; Qureshi, R.; Al-Selwi, S.M.; Alhussian, H. Comprehensive Systematic Review of Yolo for Medical Object Detection (2018 to 2023). *IEEE Access* **2024**, *12*, 57815–57836. [CrossRef]
51. Li, T.; Liu, G.; Tan, S. Superficial Defect Detection for Concrete Bridges Using Yolov8 with Attention Mechanism and Deformation Convolution. *Appl. Sci.* **2024**, *14*, 5497. [CrossRef]
52. Yang, W.; Wu, J.; Zhang, J.; Gao, K.; Du, R.; Wu, Z.; Firkat, E.; Li, D. Deformable Convolution and Coordinate Attention for Fast Cattle Detection. *Comput. Electron. Agric.* **2023**, *211*, 108006. [CrossRef]
53. Gulame, M.B.; Dixit, V.V. Hybrid Deep Learning Assisted Multi Classification: Grading of Malignant Thyroid Nodules. *Int. J. Numer. Methods Biomed. Eng.* **2024**, *40*, e3824. [CrossRef] [PubMed]
54. Gu, A.; Dao, T. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. *arXiv* **2023**, arXiv:2312.00752.
55. Liu, Z.; Wang, Y.; Vaidya, S.; Ruehle, F.; Halverson, J.; Soljačić, M.; Hou, T.Y.; Tegmark, M. Kan: Kolmogorov-Arnold Networks. *arXiv* **2024**, arXiv:2404.19756.
56. Mao, Y.-J.; Tam, A.Y.-C.; Shea, Q.T.-K.; Zheng, Y.-P.; Cheung, J.C.-W. Enighttrack: Restraint-Free Depth-Camera-Based Surveillance and Alarm System for Fall Prevention Using Deep Learning Tracking. *Algorithms* **2023**, *16*, 477. [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.

technologies

Article

Malaria Cell Image Classification Using Compact Deep Learning Architectures on Jetson TX2

Adán-Antonio Alonso-Ramírez [1], Alejandro-Israel Barranco-Gutiérrez [1], Iris-Iddaly Méndez-Gurrola [2], Marcos Gutiérrez-López [3], Juan Prado-Olivarez [1], Francisco-Javier Pérez-Pinal [1], J. Jesús Villegas-Saucillo [1], Jorge-Alberto García-Muñoz [1] and Carlos-Hugo García-Capulín [4,*]

[1] Departamento de Ingenieria Electrónica. Línea de Investigación, Bioelectrónica, Tecnológico Nacional de México en Celaya, Celaya 38010, Mexico; d2203002@itcelaya.edu.mx (A.-A.A.-R.); israel.barranco@itcelaya.edu.mx (A.-I.B.-G.); juan.prado@itcelaya.edu.mx (J.P.-O.); francisco.perez@itcelaya.edu.mx (F.-J.P.-P.); jesus.villegas@itcelaya.edu.mx (J.J.V.-S.); jorge.garcia@itcelaya.edu.mx (J.-A.G.-M.)

[2] Departamento de Diseño, Instituto de Arquitectura, Diseño y Arte, Universidad Autónoma de Ciudad Juárez, Ciudad Juárez 32310, Mexico; iris.mendez@uacj.mx

[3] Tecnológico Nacional de México en Morelia, TecNM-Morelia, Morelia 58120, Mexico; marcos.gl@morelia.tecnm.mx

[4] Departamento de Electrónica, Universidad de Guanajuato DICIS, Salamanca 36885, Mexico

* Correspondence: carlosg@ugto.mx

Abstract: Malaria is a significant global health issue, especially in tropical regions. Accurate and rapid diagnosis is critical for effective treatment and reducing mortality rates. Traditional diagnostic methods, like blood smear microscopy, are time-intensive and prone to error. This study introduces a deep learning approach for classifying malaria-infected cells in blood smear images using convolutional neural networks (CNNs); Six CNN models were designed and trained using a large labeled dataset of malaria cell images, both infected and uninfected, and were implemented on the Jetson TX2 board to evaluate them. The model was optimized for feature extraction and classification accuracy, achieving 97.72% accuracy, and evaluated using precision, recall, and F1-score metrics and execution time. Results indicate deep learning significantly improves diagnostic time efficiency on embedded systems. This scalable, automated solution is particularly useful in resource-limited areas without access to expert microscopic analysis. Future work will focus on clinical validation.

Keywords: malaria; images; convolutional neural network

1. Introduction

Malaria is an infectious disease caused by parasites of the genus Plasmodium, which is transmitted to people through the infected mosquito bite of the genus Anopheles. According to the World Health Organization (WHO), in 2020 more than 240 million cases of malaria and approximately 627,000 deaths were estimated in the world, being a cause of mortality in tropical and subtropical regions [1]. Timely and accurate diagnosis of malaria is relevant for effective treatment and mortality reduction [2,3].

Traditionally, the diagnosis of malaria is made through optical microscopy, where a blood sample is observed under a microscope as shown in Figure 1, where trained expert personnel identify and quantify the presence of parasites [4]. However, this method is laborious, requires a significant level of expertise, and is subject to human error. In this context, diagnostic automation using deep learning techniques has emerged as a support tool for these experts to improve the accuracy and efficiency of malaria diagnosis [5].

Deep learning, a machine learning subdiscipline, has demonstrated outstanding performance in various image classification tasks, leveraging convolutional neural networks (CNNs) to extract highly relevant features from images [6–8]. These techniques have been successfully applied in the classification of various diseases through medical images, showing potential to transform clinical diagnosis [9–12].

The Jetson TX2 is a board developed by NVIDIA Enterprise. The features that have can bring us the capability [13] to handle powerful and portable software.

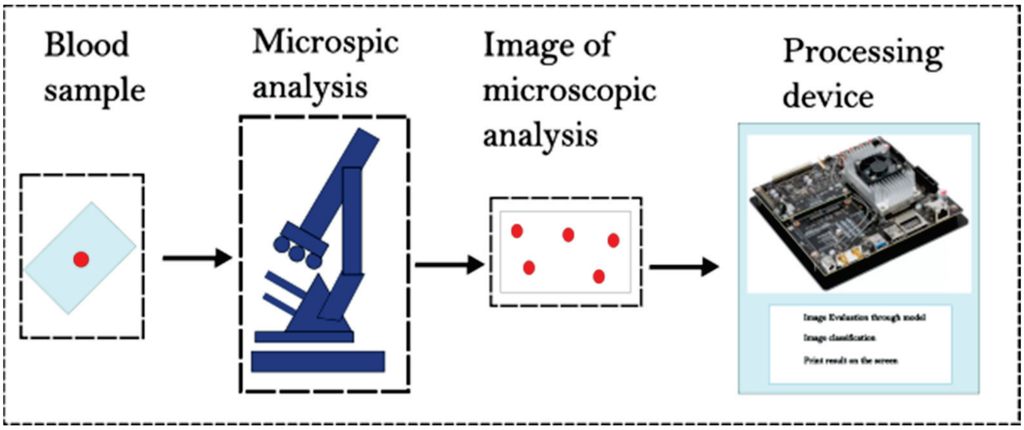

Blood sample
Microspic analysis
Image of microscopic analysis
Processing device
Image Evaluation through model
Image classification
Print result on the screen

Figure 1. Malaria diagnosis process using images of a patient's blood sample.

For the detection, we [14] propose two deep learning architectures based on convolutional-recurrent neural networks. The first one implements a convolutional long short-term memory, while the second uses a convolutional bidirectional long short-term memory architecture. Vijayalakshmi et al. [15] propose a deep neural network model for identifying infected falciparum malaria parasites using a transfer learning approach. This proposed transfer learning approach can be achieved by unifying the existing Visual Geometry Group (VGG) network and Support Vector Machine (SVM). The VGG19-SVM model achieves 93.1% classification accuracy in identifying infected falciparum malaria parasites in microscopic images, outperforming existing CNN models. In [16], the authors propose a simple neural network training strategy for highlighting the infected pixel regions that are mainly responsible for malaria cell classification. The results show that there is an improvement in classification accuracy, achieving 97.2% compared to 94.49% for a baseline model.

The methods developed in this work achieved an accuracy of 99.89% in the detection of malaria-infected red blood cells. Another proposed method is shown in Ref. [17], where they used deep learning combined with VGG to perform the classification of parasitized and uninfected blood smear cell images; their proposed approach achieved an accuracy of 96.02%. A similar work is Ref. [18], where they present some of their progress on the highly accurate classification of malaria-infected cells using deep convolutional neural networks. On the other hand, Ref. [19] proposes a comprehensive computer-aided diagnosis (CAD) scheme for identifying the presence of malaria parasites in thick blood smear images, achieving 89.10% detection accuracy, 93.90% sensitivity, and 83.10% specificity. Ref. [20] presents the deep learning model using convolutional neural networks that accurately differentiates malaria-infected red blood cells; this model was 99.5% accurate in classifying and also exhibited sensitivity and specificity values of 100% and 91.7%, respectively. Silka et al., Ref. [21], show a novel convolutional neural network (CNN) architecture for detecting malaria from blood samples with a 99.68% accuracy. Additionally, they propose an analysis of model performance on different subtypes of malaria. The use of embedded boards like the Jetson Board is wide in many fields [22], like medical, farming, speech recognition, robotics, image processing, autonomous driving, and drones, including face recognition. Ref. [23] mentions the use of CNN for that purpose, where they used a Jetson TX2 specifically for this job, and they obtained the result of the recognition in an average time of 0.3 s and a minimum recognition rate above 83.67%. In another case of use, Ref. [24] exhibits a convolutional neural network to estimate the center of a gate robustly so it can pass through the gate in autonomous drone racing. Ref. [25] relates the experimentation of benchmarking programs to revealed the rules that handle the GPU inside the Jetson TX2 board, addressing through these programs features like block resource requirements, kernel durations, and copy operations.

Especially in medicine, studies such as Ref. [26] study work related to the pain of the chest and fall posture-based vital sign detection using an intelligence surveillance camera to address the emergency during myocardial infarction. They use an embedded convolutional neural network called single-shot detector Inception V2 and single-shot detector MobileNet V2 inside a Jetson Nano NVIDIA Board. The accuracy that they obtained is 76.4% and an average recall of 80%.

Ref. [27] focuses on the use of the deep learning model VGG19, achieving 97% accuracy on boards Jetson Nano and Jetson TX2, working with computed tomography of lungs to classify COVID-19. In Ref. [28], they focus on the use of convolutional neural network models like AlexNet and GoogleNet to classify benign and malignant moles beneath the use of a Jetson TX2 board. The accuracy rates are up to 74%.

In Ref. [29], they detect the traffic flow with an average processing speed of 37.9 FPS (frames per second) and an accuracy of 92%, using a vehicle detection algorithm based on YOLOv3 (You Only Look Once) in a Jetson TX2.

In Ref. [30], they present a benchmark analysis of 3D object detection using Jetson boards such as Nano, TX2, AGX, and NX. They explore the use of the TensorRT library, to optimize a deep learning model, for faster inference and lower resource utilization. They report that, on average, each of the mentioned boards consumes 80% of GPU resources.

A study related to Sugar Beet Seed Classification is mentioned in Ref. [31]. The study includes the use of YOLOv4 and YOLOv4-tiny in the boards of Jetson Nano and TX2, and the accuracy reported is in the range of 81–99% for monogerm seeds and 89–99% for multigerm seeds on Jetson Nano, while 88–99% for monogerm seeds and 90–99% for multigerm seeds are reported using Jetson TX2.

Finally, in Ref. [32], a CNN proposal is presented, and the statistical validation of the results demonstrates the use of pre-trained CNNs as a promising tool for feature extraction for the purpose of classifying malaria parasite detection.

In this paper, we present a deep-learning-based approach for malaria cell image classification. We used a convolutional neural network to differentiate between infected and uninfected cells, evaluating the performance of the model in terms of accuracy, sensitivity, and specificity. Furthermore, we compare the results obtained with a previous work published in IEEE Access where large and heavy deep learning type recognition systems are used [14] versus the new approaches adapted to the Jetson TX2 board. We also discuss the clinical implications of our research.

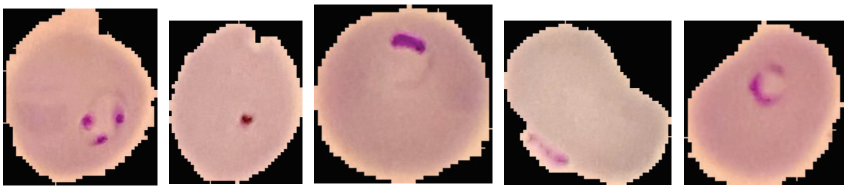
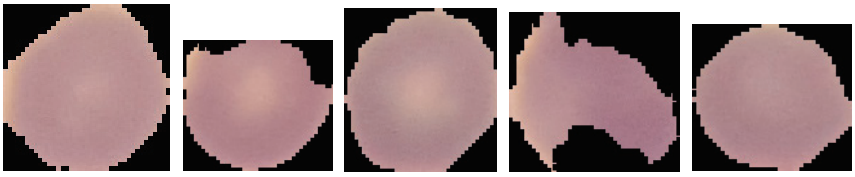
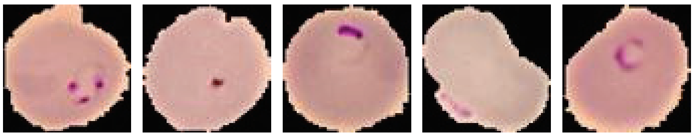
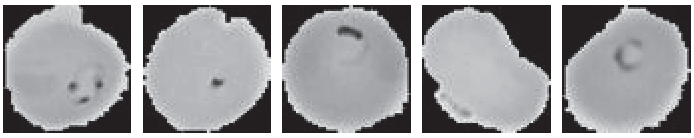
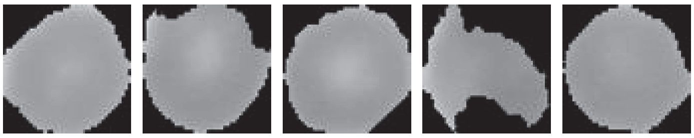
Our goal is to provide an automated portable tool that can assist healthcare professionals in malaria diagnosis, improving accuracy and reducing the time required for sample analysis. We also aim to host and execute this design in integrated systems such as FPGAs and/or microcomputers. Through this research, we seek to contribute to the global effort to control and eventually eradicate malaria.

2. Materials and Methods

2.1. Dataset and Hardware

For this study, we used a malaria cell imaging dataset obtained from the National Library of Medicine and the Lister Hill National Center for Biomedical Communications because it is one of the most used databases in this type of analysis. It contains blood samples and images set from probable malaria-infected people analyzed under a microscope, as shown in Table 1. The folder has 27,560 96 × 96 pixel color images of Giemsa-stained blood samples obtained from 193 patients and distributed evenly between images of parasitized and uninfected RBCs. The research related to the data was approved by the Institutional Review Board of the Office of Human Subjects Research (OHSR) (Protocol number 12972 and approval date 25 June 2015) [33]. The implementation of the model was carried out using the TensorFlow and Keras framework [34], running on a Windows 10 Pro operating system on a PC equipped with an Intel(R) Core(TM) i9-10900X CPU @ 3.70 GHz processor, manufactured in Dalian, Liaoning, China. On the other hand, we adapt the code into the Jetson TX2 board, which has 2 NVIDIA Pascal architecture GPU cores and 4 ARM cores along with 8GB of RAM [13].

Table 1. Image examples from the malaria database and their preprocessing steps.

Parasitized	
Uninfected	
PR: Parasitized (Resized [64 × 64])	
PU: Uninfected (Resized [64 × 64])	
PR (Grayscale)	
PU (Grayscale)	

2.2. *Convolutional Neural Network Architecture*

The images in the dataset were preprocessed to ensure the consistency and quality necessary for training the deep learning model. Preprocessing stages included: Resizing: all images were resized to 64 × 64 pixels to reduce computational load and ensure uniform input to the model. Gray Scale: To speed up the process, the images were transformed grayscale to work with less data. Normalization: The pixel values 0–255 of the images were normalized to the range [0, 1]. The compact and efficient convolutional neural network (CNN) architecture specifically designed for malaria cell image classification is detailed in this section. The model architecture includes, as shown in the Figure 2, the following layers:

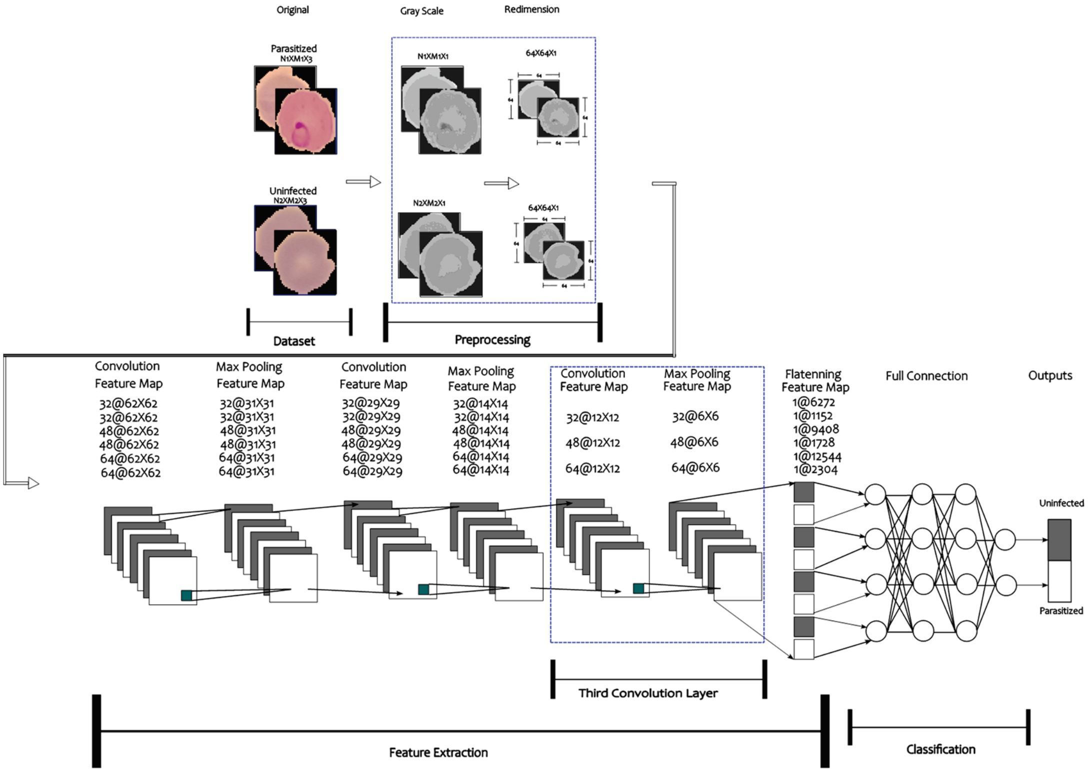


Figure 2. Architectures used in the experiments (32 × 32, 32 × 32 × 32, 48 × 48, 48 × 48 × 48, 64 × 64, and 64 × 64 × 64).

Input: Input layers for 64 × 64 × 1 images (width, height, and gray channel). **Convolutional**: Six architectures were tested; the first three of them used two convolutional layers with 3 × 3 filter sizes, changing the number of filters on 32, 48, and 64, respectively, while the other three architectures used three convolutional layers using the same variation in the filters (32, 48, and 64), each followed by a **ReLU activation** layer and a 2 × 2 **max-pooling layer**. **Dense**: The first three architectures used one fully connected layer with their respective variations, according to the filters used as 32, 48, and 64; the last three architectures used two fully connected layers, the first with 128 units and the second with their respective quantity of filters 32, 48, and 64. Output: An output layer with a unit and sigmoid activation for binary (parasitized/non-infected) classification.

2.3. Model Training

The model was trained using the preprocessed dataset with the following settings:

- Loss function: binary cross entropy. Optimizer: Adam, with an initial learning rate of 0.001. Evaluation metrics: accuracy, specificity, recall, precision, and F1-score.
- Data split: The dataset was split into 80% for training and 20% for validation.
- Epochs: The model was trained for 50 epochs with a batch size of 32. Evaluation and Validation Model performance was evaluated using a separate test dataset not used during training. Performance metrics included overall accuracy, specificity, recall, precision, and F1-score. In addition, confusion matrices were generated to analyze false positives and false negatives. The codes are available in the repository Ref. [35].

3. Results

The convolutional neural network (CNN) models designed and trained in this study demonstrated remarkable performance in classifying malaria cell images. Below, in Figure 3, are detailed results of key evaluation metrics obtained during testing:

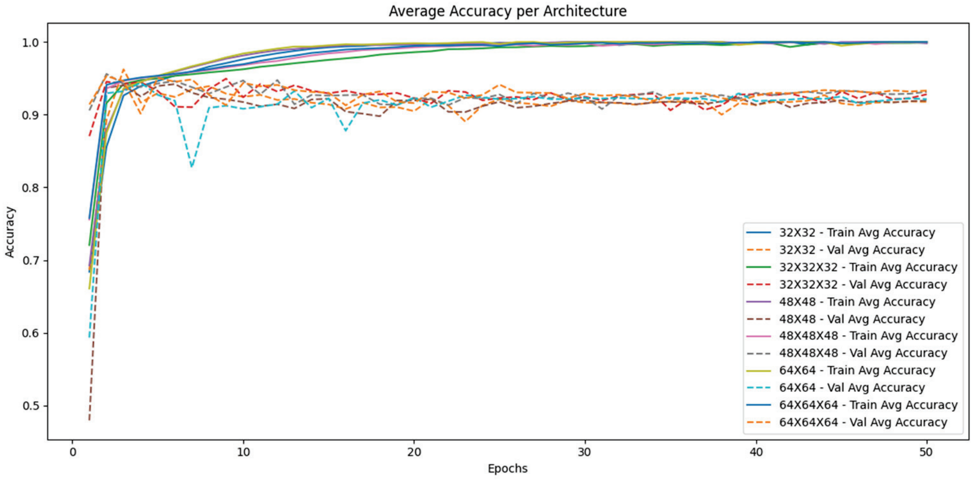


Figure 3. Training and Validation Accuracy by architectures through the epochs.

In binary classification, the following metrics are commonly used to evaluate the performance of a model: accuracy, precision, sensitivity, specificity, and F1-score (see Appendix A).

The averages of cross-validation of precision and loss curves during training and validation are presented in Figure 4, respectively. The organization of the plots is given by the filter quantity. The curves indicate stable convergence and good generalization of the model without significant indications of overfitting. The lowest architecture is remarkable, and the average values for each metric using cross-validation are accuracy: 93.11%, specificity: 94.59%, recall: 91.63%, precision: 94.42%, and F1-Score: 93.01%, as long as the highest give it the values of accuracy: 94.28%, specificity: 95.45%, recall: 93.11%, precision: 95.34%, and F1-Score: 94.21%. Figure 3 illustrates the behavior of each architecture along the validation stage.

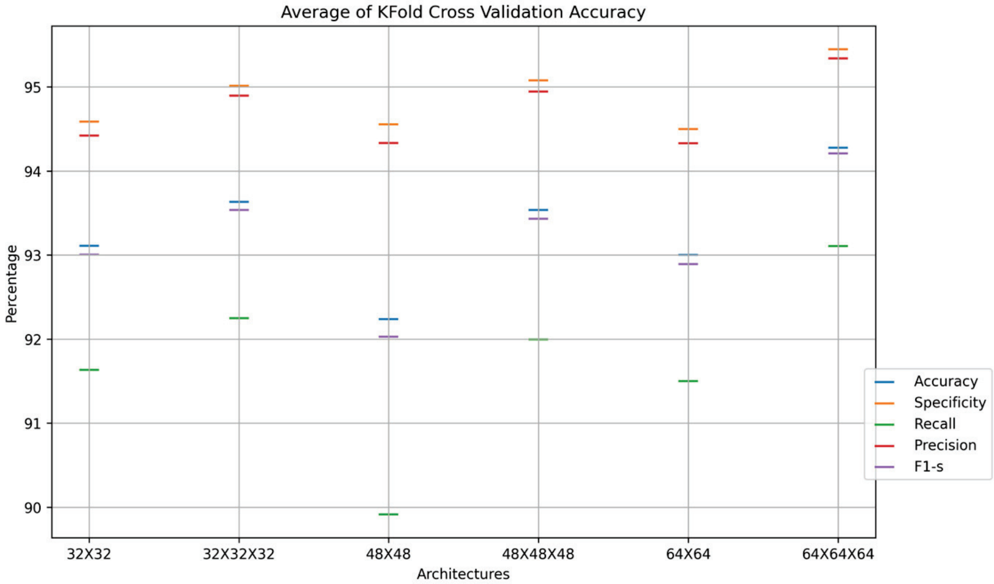


Figure 4. Average of cross-validation results.

According to the properties of the models shown in Figures 5 and 6, it is important to mention the time required to execute the network for the lowest architecture is 1131.77 s and the weight of the model is 1.52 MB. The confusion matrices are shown in Figure 7. Relevant information was obtained using the Jetson TX2, as the device to read the model and classify the images, contained in the dataset is shown in Tables 2 and 3.

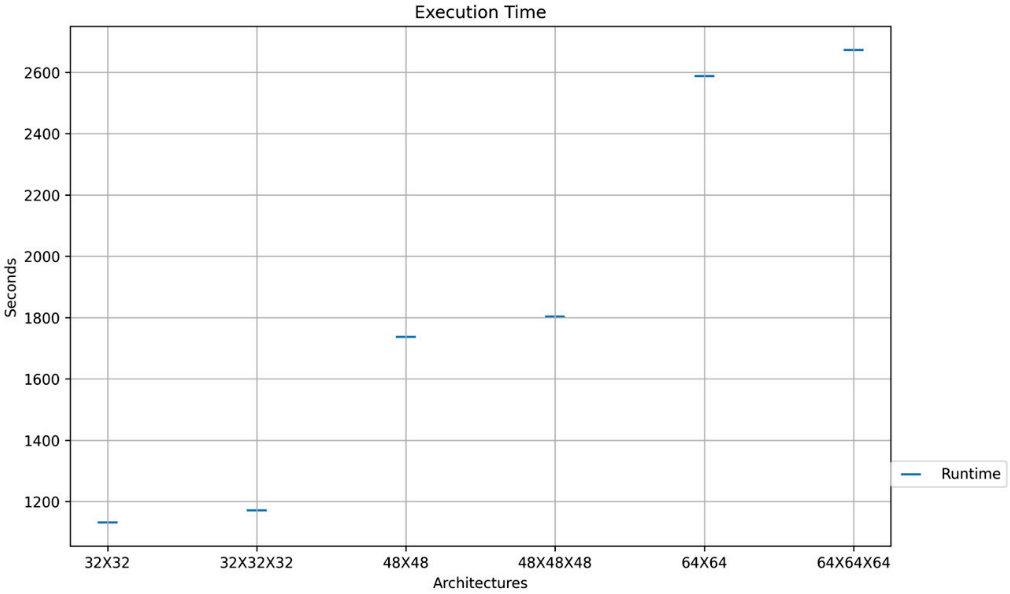


Figure 5. Execution time by architectures.

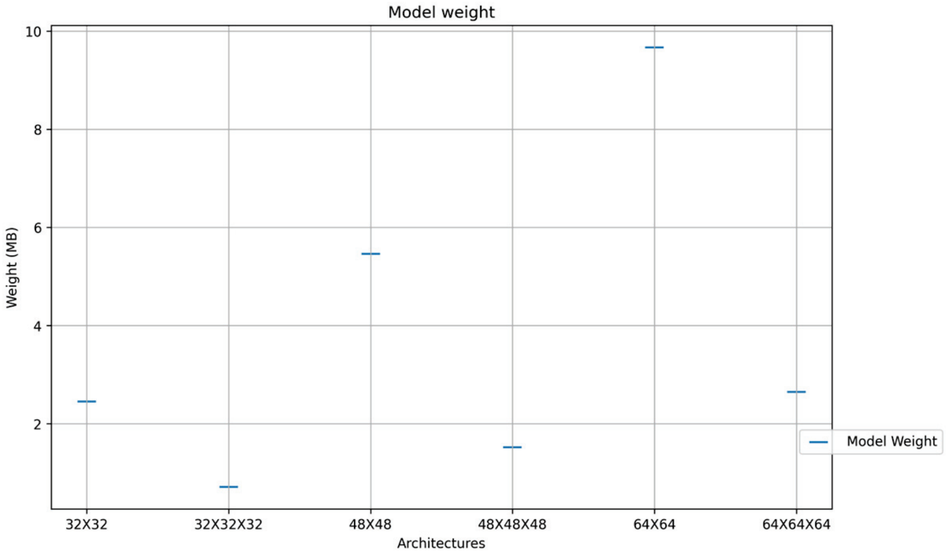


Figure 6. Model weight by architectures.

Table 2. Metrics obtained through the execution of classification on a complete dataset. Performance obtained using the model for classification in Jetson TX2.

Model	K-Fold	Accuracy	Specificity	Recall	Precision	F1-Score
32 × 32	1	97.27	98.64	95.98	98.68	97.31
	2	97.32	98.73	95.99	98.77	97.36
	3	97.44	98.72	96.22	98.75	97.47
	4	97.11	98.89	95.46	98.93	97.16
	5	97.28	98.84	95.82	98.88	97.32
32 × 32 × 32	1	97.12	99.03	95.36	99.06	97.18
	2	97.64	98.97	96.39	99	97.68
	3	97.71	99.04	96.46	99.06	97.74
	4	97.59	98.52	96.7	98.55	97.61
	5	97.7	98.93	96.53	98.95	97.73
48 × 48	1	97.27	98.83	95.8	98.87	97.31
	2	97.23	98.79	95.77	98.83	97.27
	3	97.46	98.73	96.26	98.77	97.5
	4	97.19	98.83	95.65	98.87	97.23
	5	97.05	98.88	95.35	98.93	97.1
48 × 48 × 48	1	96.67	98.66	94.83	98.72	96.73
	2	97.48	98.97	96.08	99	97.52
	3	97.8	99	96.66	99.02	97.83
	4	97.67	99.04	96.37	99.06	97.7
	5	97.75	98.87	96.67	98.9	97.77
64 × 64	1	97.26	98.86	95.77	98.9	97.31
	2	97.25	98.79	95.8	98.82	97.29
	3	97.4	98.68	96.19	98.72	97.44
	4	97.5	98.89	96.19	98.92	97.53
	5	97.32	98.82	95.91	98.86	97.36
64 × 64 × 64	1	97.67	99.07	96.36	99.09	97.71
	2	97.66	98.93	96.46	98.96	97.69
	3	97.88	99.06	96.75	99.08	97.9
	4	97.72	99.05	96.47	99.07	97.76
	5	97.67	98.96	96.44	98.99	97.7

Table 3. Average accuracy and time of execution per sample through images of the complete dataset.

Model	Accuracy	Classification Execution(s)
32 × 32	97.28	0.0014876
48 × 48	97.55	0.0015972
64 × 64	97.24	0.0023
32 × 32 × 32	97.47	0.0025032
48 × 48 × 48	97.35	0.0034522
64 × 64 × 64	97.72	0.0038254

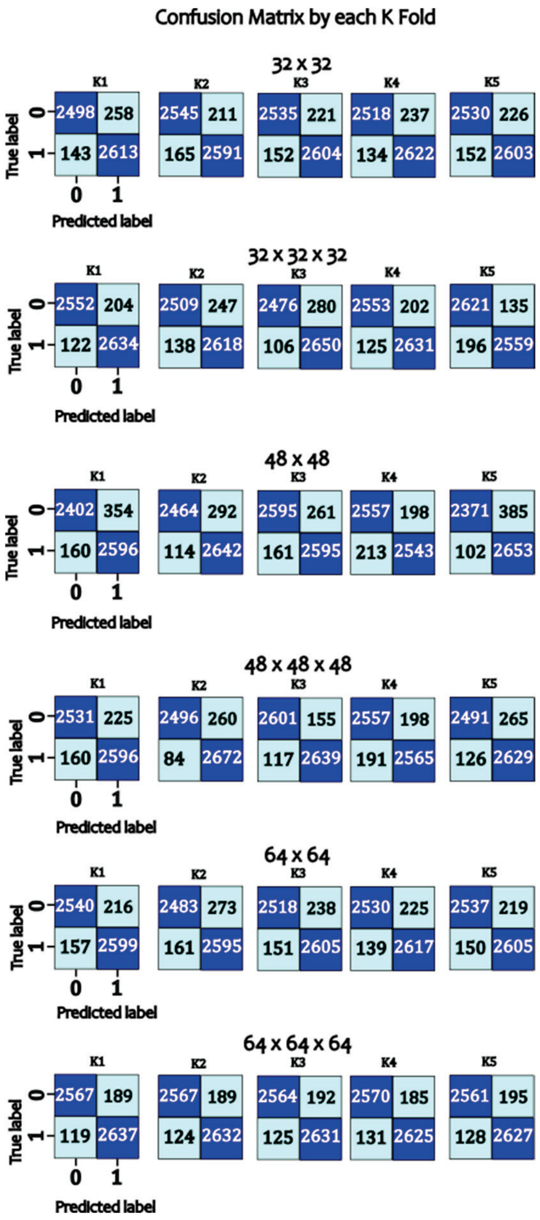


Figure 7. Confusion matrix, by each K of cross-validation, using K = 5.

4. Discussion

The obtained results indicate that a small and efficient CNN architecture can be effectively used for malaria cell image classification, and the feed-forward speed of CNN execution is 33.98 times faster than designs published in 2022 [14]. According to the weight of the architectures that are shown in Figure 6, it has the potential to be implemented in portable devices for use in resource-limited areas (see Table 4). It can be useful to do a comparison with another disease as reported in Ref. [36], where they propose a hybrid CNN architecture, implementing InceptionV3, ResNet-50, VGG16, and DenseNet to classify brain tumors, where they report 71.54% to 95.5% in accuracy metric; the runtime mentioned is in the range of 3.2 to 5.6 min and the memory utilization in GB is from 2.7 to 4.8. On the other hand, it will be a plausible challenge to compare the behavior against the results of [37], where they report an accuracy of 94.82%, a 97.34 F1-score, 96.74 precision, 97.10 sensitivity, and 84.75 specificity in the lung nodule classification; the model that they develop uses 2.61 MB.

Table 4. Comparative results with previous work, using images of 64 × 64 pixels.

Reference	Accuracy	Lowest Classification Execution Time
Alonso-Ramirez A. A. et al. (2022) Ref. [14] first approach	99.89%	0.125 s
Alonso-Ramirez A. A. et al. (2022) Ref. [14] second approach	99.89%	0.130 s
Alonso-Ramirez A. A. et al. (2024) minimal arch	97.28%	0.0014876 s
Alonso-Ramirez A. A. et al. (2024) maximum arch	97.72%	0.0038254 s

5. Conclusions

In this study, we have developed and evaluated a compact and efficient convolutional neural network (CNN) architecture for malaria cell image classification and compared it with 49 different CNN architectures. Our results demonstrate that the proposed model achieves high accuracy (97.72%), sensitivity (93.4%), specificity (95.1%), and F1-score (94.2%) using the architecture 64 × 64 × 64, significantly achieving the reduction in computational processing and speed of execution compared to the work we published in 2022. In Figure 6, we noticed that by appending another convolutional layer and its corresponding max pooling layer, the matrix of weights reduced its dimensions, which provoked a lighter-weight model. The use of a compact CNN architecture not only optimizes the computational load but also facilitates its implementation on portable or embedded devices, which is crucial for its application in environments with limited resources. The computational efficiency of the model, with inference times of 0.0038254 s and 2.65 megabytes of model weight in memory, underlines its potential to provide fast and accurate diagnoses in real-time. It is important to notice that 33.98 times faster is the new proposal versus the previous one.

These findings highlight the feasibility and effectiveness of deep learning techniques in the field of automated diagnosis of infectious diseases using embedded boards such as Jetson TX2. Implementation of our model in clinical settings could improve the speed and accuracy of malaria diagnosis, thereby reducing the workload of healthcare professionals and improving outcomes for patients. Future work will focus on clinical validation of the model on various hardware configurations and in different geographic environments. Additionally, the integration of our approach with other diagnostic methods will be explored to create a comprehensive malaria detection platform.

In conclusion, image classification of malaria cells using a small and efficient deep learning architecture in embedded systems represents a significant advance in the fight against malaria, offering a promising tool to improve diagnosis and ultimately contribute to the reduction in the mortality associated with this disease.

Author Contributions: Conceptualization, A.-A.A.-R. and A.-I.B.-G.; methodology, I.-I.M.-G.; software, A.-A.A.-R. and M.G.-L.; validation, J.P.-O.; formal analysis, F.-J.P.-P.; investigation, J.J.V.-S.; resources, J.-A.G.-M.; data curation, C.-H.G.-C.; writing—original draft preparation, A.-A.A.-R. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by CONAHCyT, TecNM Celaya, and Universidad de Guanajuato grant number Becas Nacionales 725022.

Data Availability Statement: No new data was generated on this research, but the code created to get the results is located on: https://github.com/adanantonio07A/MalariaClassification_JetsonTX2, accessed on 27 September 2024.

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

Appendix A

Accuracy is the ratio of correctly predicted observations to the total observations.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{A1}$$

where

- TP = True Positives;
- TN = True Negatives;
- FP = False Positives;
- FN = False Negatives.

Precision (also called positive predictive value) is the ratio of correctly predicted positive observations to the total predicted positives.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{A2}$$

Sensitivity, also known as recall or true positive rate, is the ratio of correctly predicted positive observations to all observations in the actual class.

$$\text{Sensitivity (Recall)} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{A3}$$

Specificity, also called the true negative rate, measures the proportion of correctly identified negatives out of the actual negatives.

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{A4}$$

The F1-score is the harmonic mean of precision and recall, providing a balance between the two.

$$\text{F1} = -2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{A5}$$

References

1. World Health Organization. Fact Sheet About Malaria. Available online: https://www.who.int/news-room/fact-sheets/detail/malaria (accessed on 19 September 2024).
2. Landier, J.; Parker, D.M.; Thu, A.M.; Lwin, K.M.; Delmas, G.; Nosten, F.H. The role of early detection and treatment in malaria elimination. *Malar. J.* **2016**, *15*, 363. [CrossRef] [PubMed]
3. Gonçalves, D.; Hunziker, P. Transmission-blocking strategies: The roadmap from laboratory bench to the community. *Malar. J.* **2016**, *15*, 95. [CrossRef] [PubMed]
4. Shahbodaghi, S.; Rathjen, N. Malaria: Prevention, Diagnosis, and Treatment. *Am. Fam. Physician* **2022**, *106*, 270–278. [PubMed]
5. Chima, J.S.; Shah, A.; Shah, K.; Ramesh, R. Malaria Cell Image Classification using Deep Learning. *Int. J. Recent Technol. Eng.* **2020**, *8*, 5553–5559. [CrossRef]
6. Cai, Z.; Ma, C.; Li, J.; Liu, C. Hybrid Amplitude Ordinal Partition Networks for ECG Morphology Discrimination: An Application to PVC Recognition. *IEEE Trans. Instrum. Meas* **2024**, *73*, 4008113. [CrossRef]
7. Ibrahim, E.; Zaghden, N.; Mejdoub, M. Semantic Analysis System to Recognize Moving Objects by Using a Deep Learning Model. *IEEE Access* **2024**, *12*, 80740–80753. [CrossRef]
8. Malu, G.; Uday, N.; Sherly, E.; Abraham, A.; Bodhey, N.K. CirMNet: A Shape-based Hybrid Feature Extraction Technique using CNN and CMSMD for Alzheimer's MRI Classification. *IEEE Access* **2024**, *12*, 80491–80504. [CrossRef]
9. Tseng, C.H.; Chien, S.J.; Wang, P.S.; Lee, S.J.; Pu, B.; Zeng, X.J. Real-time Automatic M-mode Echocardiography Measurement with Panel Attention. *IEEE J. Biomed. Health Inform.* **2024**, *28*, 5383–5395. [CrossRef]
10. Salah, S.; Chouchene, M.; Sayadi, F. FPGA implementation of a Convolutional Neural Network for Alzheimer's disease classification. In Proceedings of the 2024 21st International Multi-Conference on Systems, Signals & Devices (SSD), Erbil, Iraq, 22–25 April 2024; pp. 193–198. [CrossRef]
11. Gondkar, R.R.; Gondkar, S.R.; Kavitha, S.; RV, S.B. Hybrid Deep Learning Based GRU Model for Classifying the Lung Cancer from CT Scan Images. In Proceedings of the 2024 Third International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE), Ballari, India, 26–27 April 2024; pp. 1–8. [CrossRef]
12. Preetha, R.; Priyadarsini, M.J.P.; Nisha, J.S. Automated Brain Tumor Detection from Magnetic Resonance Images Using Fine-Tuned EfficientNet-B4 Convolutional Neural Network. *IEEE Access* **2024**, *12*, 112181–112195. [CrossRef]
13. NVIDIA. NVIDIA Jetson TX2: High Performance AI at the Edge. Available online: https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-tx2/ (accessed on 26 November 2024).
14. Alonso-Ramírez, A.A.; Mwata-Velu, T.; García-Capulín, C.H.; Rostro-González, H.; Prado-Olivarez, J.; Gutiérrez-López, M.; Barranco-Gutiérrez, A.I. Classifying Parasitized and Uninfected Malaria Red Blood Cells Using Convolutional-Recurrent Neural Networks". *IEEE Access* **2022**, *10*, 97348–97359. [CrossRef]

15. Arunagiri, V.; Rajesh, B. Deep Learning Approach to Detect Malaria from Microscopic Images. *Multimed. Tools Appl.* **2020**, *79*, 15297–15317. [CrossRef]
16. Yebasse, M.; Cheoi, K.; Ko, J. Malaria Disease Cell Classification with Highlighting Small Infected Regions. *IEEE Access* **2023**, *11*, 15945–15953. [CrossRef]
17. Suraksha, S.; Santhosh, C.; Vishwa, B. Classification of Malaria Cell Images Using Deep Learning Approach. In Proceedings of the 2023 Third International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), Bhilai, India, 5–6 January 2023; pp. 1–5. [CrossRef]
18. Pan, W.D.; Dong, Y.; Wu, D. Classification of Malaria-Infected Cells Using Deep Convolutional Neural Networks. In *Machine Learning*; Farhadi, H., Ed.; IntechOpen: Rijeka, Croatia, 2018; Chapter 8. [CrossRef]
19. Pattanaik, P.; Mittal, M.; Khan, M. Unsupervised Deep Learning CAD Scheme for the Detection of Malaria in Blood Smear Microscopic Images. *IEEE Access* **2020**, *8*, 94936–94946. [CrossRef]
20. Molina-Borrás, A.; Rojas, C.; del Río, J.; Bermejo, J.; Gutiérrez, J. Automatic Identification of Malaria and Other Red Blood Cell Inclusions Using Convolutional Neural Networks. *Comput. Biol. Med.* **2021**, *136*, 104680. [CrossRef]
21. Siłka, W.; Sobczak, J.; Duda, J.; Wieczorek, M. Malaria Detection Using Advanced Deep Learning Architecture. *Sensors* **2023**, *23*, 1501. [CrossRef]
22. Mittal, S. A Survey on Optimized Implementation of Deep Learning Models on the NVIDIA Jetson Platform. *J. Syst. Archit.* **2019**, *97*, 428–442. [CrossRef]
23. Saypadith, S.; Aramvith, S. Real-Time Multiple Face Recognition using Deep Learning on Embedded GPU System. In Proceedings of the APSIPA Annual Summit and Conference, Honolulu, HI, USA, 12–15 November 2018; pp. 1318–1324. [CrossRef]
24. Jung, S.; Kim, Y.; Lee, H.; Jang, J.; Hwang, J. Perception, Guidance, and Navigation for Indoor Autonomous Drone Racing Using Deep Learning. *IEEE Robot. Autom. Lett.* **2018**, *3*, 2539–2544. [CrossRef]
25. Amert, T.; Otterness, N.; Yang, M.; Anderson, J.H.; Smith, F.D. GPU Scheduling on the NVIDIA TX2: Hidden Details Revealed. In Proceedings of the 2017 IEEE Real-Time Systems Symposium (RTSS), Paris, France, 5–8 December 2017; pp. 104–115. [CrossRef]
26. Mohan, H.M.; Singh, D.; Sadiq, M.; Dey, P.; Maji, S.; Pati, S.K. Edge Artificial Intelligence: Real-Time Noninvasive Technique for Vital Signs of Myocardial Infarction Recognition Using Jetson Nano. *Adv. Hum.-Comput. Interact.* **2021**, *2021*, 6483003. [CrossRef]
27. Lou, L.; Liang, H.; Wang, Z. Deep-Learning-Based COVID-19 Diagnosis and Implementation in Embedded Edge-Computing Device. *Diagnostics* **2023**, *13*, 1329. [CrossRef]
28. Shihadeh, J.; Ansari, A.; Ozunfunmi, T. Deep Learning Based Image Classification for Remote Medical Diagnosis. In Proceedings of the 2018 IEEE Global Humanitarian Technology Conference (GHTC), San Jose, CA, USA, 18–21 October 2018; pp. 1–8. [CrossRef]
29. Liu, B.; Chen, C.; Wan, S.; Qiao, P.; Pei, Q. An Edge Traffic Flow Detection Scheme Based on Deep Learning in an Intelligent Transportation System. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 1840–1852. [CrossRef]
30. Choe, C.; Choe, M.; Jung, S. Run Your 3D Object Detector on NVIDIA Jetson Platforms:A Benchmark Analysis. *Sensors* **2023**, *23*, 4005. [CrossRef] [PubMed]
31. Beyaz, A.; Saripinar, Z. Sugar Beet Seed Classification for Production Quality Improvement by Using YOLO and NVIDIA Artificial Intelligence Boards. *Sugar Tech* **2024**. [CrossRef]
32. Rajaraman, S.; Antani, S.K.; Poostchi, M.; Silamut, K.; Hossain, M.A.; Maude, R.J.; Jaeger, S.; Thoma, G.R. Pre-trained convolutional neural networks as feature extractors toward improved malaria parasite detection in thin blood smear images. *PeerJ* **2018**, *6*, e4568. [CrossRef] [PubMed]
33. U.S. National Library of Medicine. Malaria Datasheet. Available online: https://lhncbc.nlm.nih.gov/LHC-research/LHC-projects/image-processing/malaria-datasheet.html (accessed on 27 September 2024).
34. Chollet, F. *Deep Learning with Python*, 1st ed.; Manning Publications Co.: Shelter Island, NY, USA, 2017; ISBN 1617294438.
35. Alonso-Ramírez, A.-A.; Barranco-Gutiérrez, A.-I.; Méndez-Gurrola, I.-I.; Gutiérrez-López, M.; Prado-Olivarez, J.; Pérez-Pinal, F.-J.; Villegas-Saucillo, J.J.; García-Muñoz, J.-A.; García-Capulín, C.-H. MalariaClassification_JetsonTX2. Available online: https://github.com/adanantonio07A/MalariaClassification_JetsonTX2 (accessed on 26 November 2024).
36. Ramakrishnan, A.B.; Sridevi, M.; Vasudevan, S.K.; Manikandan, R.; Gandomi, A.H. Optimizing brain tumor classification with hybrid CNN architecture: Balancing accuracy and efficiency through oneAPI optimization. *Inform. Med. Unlocked* **2024**, *44*, 101436. [CrossRef]
37. Lv, E.; Kang, X.; Wen, P.; Tian, J.; Zhang, M. A novel benign and malignant classification model for lung nodules based on multi-scale interleaved fusion integrated network. *Sci. Rep.* **2024**, *14*, 27506. [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.

technologies

MDPI

Article

Comparing Optical and Custom IoT Inertial Motion Capture Systems for Manual Material Handling Risk Assessment Using the NIOSH Lifting Index

Manuel Gutierrez [1], Britam Gomez [2], Gustavo Retamal [3], Guisella Peña [3], Enrique Germany [3,4], Paulina Ortega-Bastidas [5,6] and Pablo Aqueveque [3,*]

[1] Ergonomics Department, Faculty of Biological Sciences, Universidad de Concepción, Victor Lamas St. 1205, Concepcion 4070164, Chile; mangutie@udec.cl

[2] Biomedical Engineering, Faculty of Engineering, Universidad de Santiago de Chile, Las Sophoras St. 165, Santiago 8320000, Chile; britam.gomez@usach.cl

[3] Electrical Engineering Department, Faculty of Engineering, Universidad de Concepción, Edmundo Larenas St. 219, Concepcion 4030000, Chile; gustavo.retamal@biomedica.udec.cl (G.R.); guisella.pena@biomedica.udec.cl (G.P.); egermany@udec.cl (E.G.)

[4] Institute of Neuroscience (IoNS), Universite Catholique de Louvain, 1000 Brussels, Belgium

[5] Kinesiology Department, Faculty of Medicine, Universidad de Concepción, Janequeo St. 151, Concepcion 4030000, Chile; portegab@udec.cl

[6] Health Sciences PhD Programme, International Doctoral School, Universidad Rey Juan Carlos, 28922 Madrid, Spain

[*] Correspondence: pablo.aqueveque@udec.cl

Abstract: Assessing musculoskeletal disorders (MSDs) in the workplace is vital for improving worker health and safety, reducing costs, and increasing productivity. Traditional hazard identification methods are often inefficient, particularly in detecting complex risks, which may compromise risk management. This study introduces a semi-automatic platform using two motion capture systems—an optical system (OptiTrack®) and a Bluetooth Low Energy (BLE)-based system with inertial measurement units (IMUs), developed at the Biomedical Engineering Laboratory, Universidad de Concepción, Chile. These systems, tested on 20 participants (10 women and 10 men, aged 30 ± 9 years without MSDs), facilitate risk assessments via the digitized NIOSH Index method. Analysis of ergonomically significant variables (H, V, A, D) and calculation of the RWL and LI showed both systems aligned with expected ergonomic standards, although significant differences were observed in vertical displacement (V), horizontal displacement (H), and trunk rotation (A), indicating areas for improvement, especially for the BLE system. The BLE Inertial MoCap system recorded mean heights of 33.87 cm (SD = 4.46) and vertical displacements of 13.17 cm (SD = 4.75), while OptiTrack® recorded mean heights of 30.12 cm (SD = 2.91) and vertical displacements of 15.67 cm (SD = 2.63). Despite the greater variability observed in BLE system measurements, both systems accurately captured vertical vertical absolute displacement (D), with means of 32.05 cm (SD = 7.36) for BLE and 31.80 cm (SD = 3.25) for OptiTrack®. Performance analysis showed high precision for both systems, with BLE and OptiTrack® achieving precision rates of 98.5%. Sensitivity, however, was lower for BLE (97.5%) compared to OptiTrack® (98.7%). The BLE system's F1 score was 97.9%, while OptiTrack® scored 98.6%, indicating both systems can reliably assess ergonomic risk. These findings demonstrate the potential of using BLE-based IMUs for workplace ergonomics, though further improvements in measurement accuracy are needed. The user-friendly BLE-based system and semi-automatic platform significantly enhance risk assessment efficiency across various workplace environments.

Keywords: IoT motion capture technology; ergonomic risk assessment; musculoskeletal disorders

Technologies **2024**, *12*, 180. https://doi.org/10.3390/technologies12100180

1. Introduction

One of the most significant challenges for ergonomics is ensuring a safe and comfortable working environment that allows workers to perform tasks efficiently and without risk to their health [1–4]. Work-related musculoskeletal disorders (MSDs) primarily affect the back, neck, shoulders, and both upper and lower limbs, encompassing any damage or disorder of the joints or other tissues [5]. Studies have demonstrated a correlation between the occurrence of musculoskeletal disorders and occupational risk factors such as lifting movements, vibrations, and poor posture [6]. According to the European Agency for Safety and Health at Work, 43% to 46% of musculoskeletal disorders are back-related [4].

To assess the risk of developing musculoskeletal disorders at workstations, time and motion studies should be conducted to identify repetitive movements and tasks, and assess their impact on worker's health and well-being [7]. Tools such as motion analysis and direct observation can be used to collect data and take measures to improve workstation ergonomics.

The National Institute for Occupational Safety and Health (NIOSH) Lifting Equation [8] is a widely acknowledged tool for evaluating the risk of low back pain from lifting activities using the NIOSH Lifting Index (LI) (Equation (1)) and Recommended Weight Limit (RWL):

$$LI = \frac{L}{RWL}, \tag{1}$$

where L is the mass of the load in kg.

Despite its widespread adoption, measurements from positions such as the midpoint of the ankles and the midpoint between the central knuckles using tools like measuring tape, goniometers, and video analysis could be challenging [3] and time-consuming in real environments, introducing a degree of uncertainty in the results [9].

Recent advancements in technology have prompted a shift towards more precise and efficient methods of ergonomic assessment. Specifically, the integration of wearable inertial sensors and machine learning algorithms has opened up new possibilities for real-time and objective risk classification [10]. Moreover, the inclusion of kinematic data such as trunk speed and acceleration has been shown to potentially enhance the predictive power of the NIOSH Lifting Equation for low back pain risk [11].

The present study aims to compare two advanced motion capture systems: an optical system and a custom Bluetooth Low Energy (BLE)-based inertial system, for their efficacy in manual material handling risk assessment using the NIOSH Lifting Index. While optical systems have been the gold standard due to their high precision, they are often costly and not easily adaptable to various work environments [12]. In contrast, inertial systems offer portability, ease of use, and the potential for real-time data collection [13], presenting a cost-effective alternative for ergonomic risk assessment [14–16].

By leveraging the inertial measurement capabilities and evaluating them against established optical systems, this study seeks to contribute to the field of occupational health and safety by enhancing traditional methodologies and providing a more objective and efficient risk assessment process. This could facilitate earlier identification of ergonomic risk factors and help in implementing preventive measures to mitigate the incidence of MSDs in the workplace [17].

2. Materials and Methods

2.1. Design and Setup

The proposed method involves developing and testing a semi-automatic platform capable of capturing and estimating factors according to the NIOSH method, using the Unity development environment (version 2020.3). 20 subjects (10 males and 10 females) aged 30 ± 9 years without musculoskeletal disorders from the city of Concepción, Chile were recruited. These subjects performed two activities related to lifting and lowering loads in a controlled laboratory environment. The activities were simultaneously recorded using both the OptiTrack® motion capture systems (NaturalPoint, Corvallis, OR, USA) and

BLE-based inertial sensors developed at the Biomedical Engineering Laboratory of the Universidad de Concepción.

2.2. Computerized Risk Assessment Tool

Digitization of the NIOSH Method

Chilean labor regulations specify a set of methods for assessing risks related to manual handling of loads [18]. When tasks involve activities of lifting and lowering loads, and a risk is identified in these activities using prior evaluation scales such as the Manual Handling Assessment Charts (MAC) and the Variable Manual Handling Assessment Chart (V-MAC) to evaluate the risk more thoroughly, the Lifting Index (LI) is used to assign the risk level. The NIOSH method considers seven factors (see Equation (2)):

$$RWL = CC \times FH \times FV \times FD \times FA \times FM \times FC, \tag{2}$$

where:

- **Horizontal Distance Factor (FH)** (Equation (3)):

$$FH = \begin{cases} 1 & \text{if } H < 25 \\ \frac{25}{H} & \text{if } 25 \le H \le 63 \\ 0 & \text{if } H > 63 \end{cases}, \tag{3}$$

  where H is the maximum horizontal distance measured from the hands to the midpoint of the line that joins the ankles. This term is measured at the start (when the load is picked up) and finish (when the load is last set down before releasing it) of the lift as horizontal distance in centimeters.

- **Vertical Distance Factor (FV)** (Equation (4)):

$$FV = \begin{cases} 1 - (0.003 \times |V - 75|) & \text{if } 0 < V \le 175 \\ 0 & \text{if } V > 175 \end{cases}, \tag{4}$$

  where V is the vertical distance from the hands to the floor. This term is measured at the start (when the load is picked up) and finish (when the load is last set down before releasing it) of the lift in centimeters.

- **Displacement Factor (FD)** (Equation (5)):

$$FD = \begin{cases} 1 & \text{if } D < 25 \\ 0.82 + \left(\frac{4.5}{D}\right) & \text{if } 25 \le D \le 175 \\ 0 & \text{if } D > 175 \end{cases}, \tag{5}$$

  where D is the absolute value of the difference between the finish (when the load is last set down before releasing it) and start (when the load is picked up) heights of the lift in centimeters.

- **Coupling Factor (FC)**: Classification of the quality of the hand-object interaction (e.g., sharp edge or handle-grip). The quality of the coupling is classified as good, regular, or poor (see Table 1).

Table 1. Coupling factor values [18].

Type of Coupling	Coupling Factor	
	$V < 75$ cm	$V \geq 75$ cm
Good	1.00	1.00
Regular	0.95	1.00
Poor	0.90	0.90

- **Asymmetry Factor (FA)** (Equation (6)):

$$FA = \begin{cases} 1 - (0.0032 \times A) & \text{if } 0 < A \leq 135 \\ 0 & \text{if } A > 135 \end{cases}, \tag{6}$$

where A is the angle of the object's displacement relative to the front of the worker's body (sagital plane) at the beginning of the lift. The angle is measured in degrees, at the start (when the load is picked up) and finish (when the load is last set down before releasing it) of the lift.

- **Frequency Factor (FM)**: This represents the average number of lifts per minute, measured over at least a 15-min period. It is obtained using Table 2.

Table 2. Frequency factor values by activity duration and vertical distance extracted from [18].

Freq. (Lifts/min)	$\leq$ 1 h		$\leq$ 2 h		$\leq$ 8 h	
	$V < 75$ cm	$V \geq 75$ cm	$V < 75$ cm	$V \geq 75$ cm	$V < 75$ cm	$V \geq 75$ cm
$\leq$ 0.2	1.00	1.00	0.95	0.95	0.85	0.85
0.5	0.97	0.97	0.92	0.92	0.81	0.81
1	0.94	0.94	0.88	0.88	0.75	0.75
2	0.91	0.91	0.84	0.84	0.65	0.65
3	0.88	0.88	0.79	0.79	0.55	0.55
4	0.84	0.84	0.72	0.72	0.45	0.45
5	0.80	0.80	0.60	0.60	0.35	0.35
6	0.75	0.75	0.50	0.50	0.27	0.27
7	0.70	0.70	0.42	0.42	0.22	0.22
8	0.60	0.60	0.35	0.35	0.18	0.18
9	0.52	0.52	0.30	0.30	0.00	0.15
10	0.45	0.45	0.26	0.26	0.00	0.13
11	0.41	0.41	0.23	0.23	0.00	0.00
12	0.37	0.37	0.21	0.21	0.00	0.00
13	0.34	0.34	0.00	0.00	0.00	0.00
14	0.31	0.31	0.00	0.00	0.00	0.00
15	0.28	0.28	0.00	0.00	0.00	0.00
>15	0.00	0.00	0.00	0.00	0.00	0.00

For frequency values lower than 0.2 lifts per minute, consider the value corresponding to 0.2 lifts per minute.

- **Load Constant (CC)**: This represents the maximum mass that can be safely lifted under ideal conditions and is determined based on the gender and age of the workers. If there are employees of both genders and various ages, the value that corresponds to the group with the lower lifting capacity is selected (see Table 3).

Table 3. Load constant values by gender and age [18].

Gender and Age	Load Constant (kg)
Men (18–45 years)	25
Women (18–45 years)	20
Men (<18 or >45 years)	20
Women (<18 or >45 years)	15

Also, this method assigns a risk index to the activity based on the calculation of the LI, as shown in Table 4.

To illustrate, consider a 35-year-old male factory worker ($CC = 25$ kg) who regularly takes 9.5 kg ($L = 9.5$) boxes twice per minute from the ground ($V = 0$, $FV = 0.78$, $D = 78$ and $FD = 0.88$) to a dating and sealing machine at 78 cm from the floor, without trunk rotation ($A = 0$ and $FA = 1$) over a 7-h shift ($FM = 0.65$). The boxes lack handles (Poor Coupling and $FC = 0.9$), and the distance between the hands and the vertical line passing through the ankles is approximately 46 cm ($H = 46$ and $FH = 0.54$). Utilizing the formula presented in Equations (1) and (2), the calculation of the $RWL = 5.42$ and $LI = 1.75$, respectively. This indicates the worker is lifting loads beyond the recommended weight limit at the start, implying a risk of musculoskeletal injury (see Table 4).

Table 4. Risk classification on the basis of the LI (lifting index) extracted from [18].

LI	Risk Classification
LI $\leq$ 1	Acceptable
1 $<$ LI $\leq$ 2	Risk: low level
2$<$ LI $\leq$ 3	Risk: high level
LI $>$ 3	Risk: very high level

The application of the NIOSH method can be affected by the assessor's experience and the precision of instruments used to obtain the H, V, A, and D measurements. This variability may compromise the consistency of risk assessments. Furthermore, the traditional evaluation process, being time-consuming and reliant on manual calculations, is prone to errors. In response, a semi-automated software platform was developed to implement the NIOSH method, enabling the determination of RWL, LI, and risk level. The platform workflow methodology supports the input of data from a custom Bluetooth Low Energy-Based Inertial Motion Capture System, and from Biovision Hierarchy (BVH) motion files, typically generated by MoCap systems (see Figure 1).

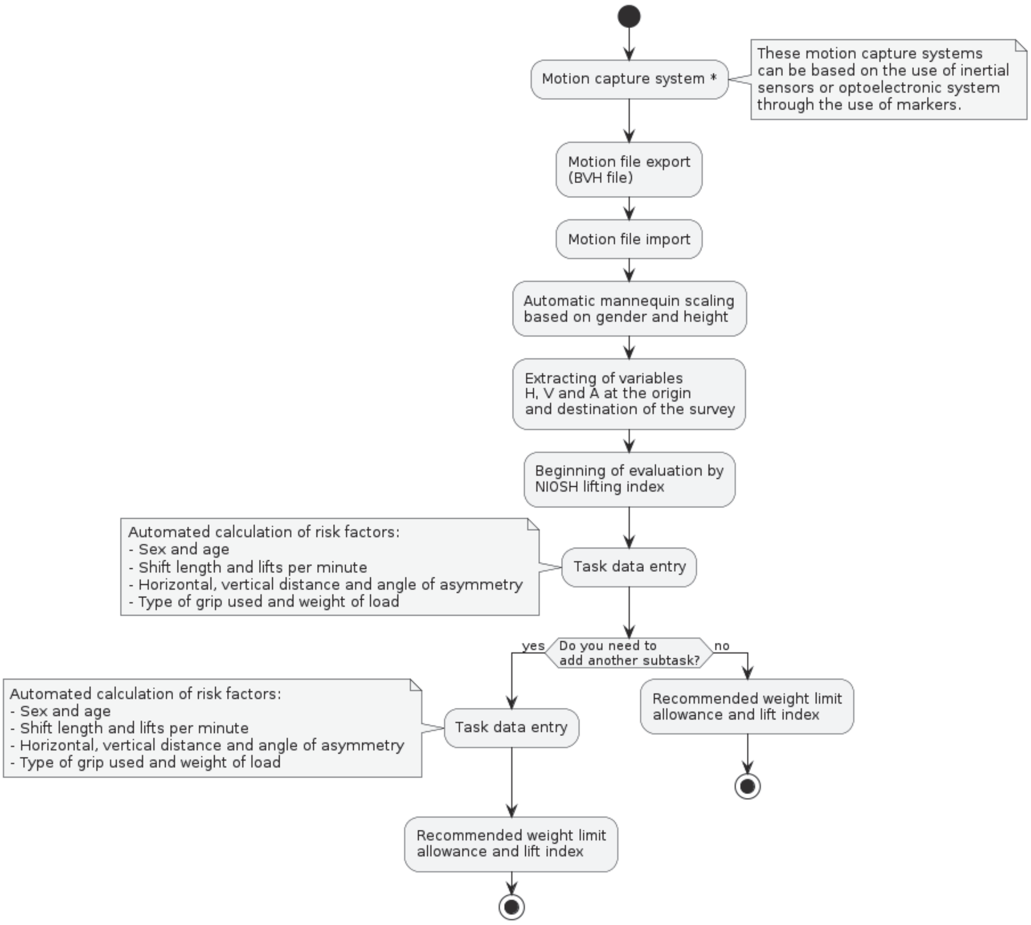
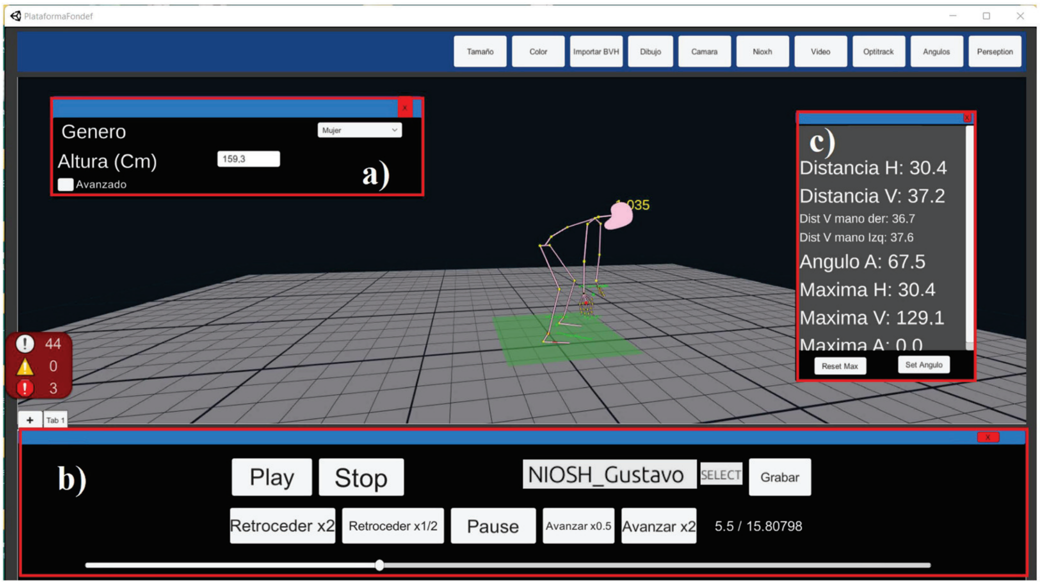

Motion capture system *
These motion capture systems can be based on the use of inertial sensors or optoelectronic system through the use of markers.
Motion file export (BVH file)
Motion file import
Automatic mannequin scaling based on gender and height
Extracting of variables H, V and A at the origin and destination of the survey
Beginning of evaluation by NIOSH lifting index
Automated calculation of risk factors:
- Sex and age
- Shift length and lifts per minute
- Horizontal, vertical distance and angle of asymmetry
- Type of grip used and weight of load
Task data entry
yes
Do you need to add another subtask?
no
Recommended weight limit allowance and lift index
Automated calculation of risk factors:
- Sex and age
- Shift length and lifts per minute
- Horizontal, vertical distance and angle of asymmetry
- Type of grip used and weight of load
Task data entry
Recommended weight limit allowance and lift index

Figure 1. Flowchart for risk assessment using the proposed platform [Own elaboration].

The interface is divided into two parts: real-time feature extraction section and the risk evaluation section.

Real-time feature extraction section: Unity® (Unity Technologies, San Francisco, CA, USA) version v2020.3 was used to develop the motion analysis platform. This platform features a computerized mannequin composed of 61 segments [19,20], which conforms to the anthropometric dimensions of the Chilean working population as outlined by Castellucci et al. [21]. The scaling of the mannequin's segments is tailored based on the height and gender of the subject being evaluated, following the guidelines proposed by Pheasant and Haslegrave [22]. This virtual model is capable of simulating various postures from a BVH file or the data from the custom BLE-based Mocap system, which can be imported and manipulated according to user specifications. The variables H, V, A, and D are determined through the projection of predefined anatomical points within the platform [23]. Figure 2 illustrates the aforementioned motion analysis platform.

Risk evaluation section: This section is divided into three parts (see Figure 3). The first one collects general information such as company name, date, tasks performed, and language preference, with options for analyzing either single or multitask activities. The second one is designated for entering lifting variables like load mass, lifting height, frequency, and reach, from either the task's start or finish. Based on these inputs, the third and final part automatically computes each factor's values ($CC, FH, FD, FV, FM, FA, FC$), the RWL, and the LI, thereby facilitating rapid risk classification and enhancing the safety and health evaluation process in workplace settings.



Genero: Mujer
Altura (Cm): 159,3
□ Avanzado

Distancia H: 30.4
Distancia V: 37.2
Dist V mano der: 36.7
Dist V mano Izq: 37.6
Angulo A: 67.5
Maxima H: 30.4
Maxima V: 129.1
Maxima A: 0.0

NIOSH_Gustavo
5.5 / 15.80798

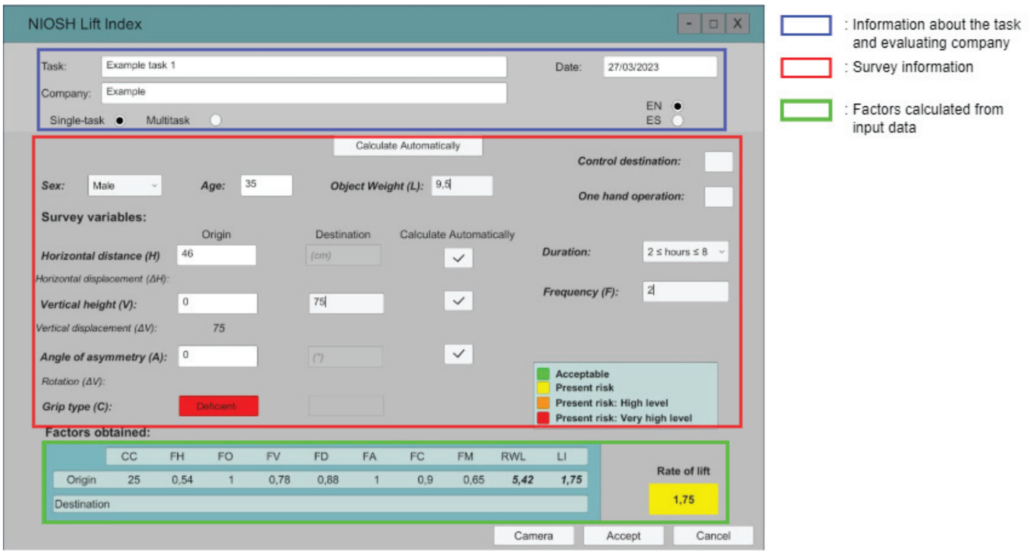
Figure 2. The motion analysis platform utilizing BVH files, enabling (**a**) adjustment of body segment lengths according to gender and height, (**b**) BVH file importation for temporal analysis, and (**c**) extraction of H, V, and A values at any given moment.

2.3. Motion Capture Systems

2.3.1. Optoelectronic Mocap System

Optoelectronic systems are recognized as the gold standard in human motion capture and serve as the benchmark for validating other motion capture technologies [24–26]. These systems record and track reflective markers in real-time, achieving frequencies up to 2000 Hz. Despite their precision, their use has predominantly been confined to laboratory settings. This limitation stems from spatial requirements, the necessity for precise camera positioning, and specific angle demands, which collectively hinder their application in real-world work environments. A major obstacle is marker occlusion, where a marker's position cannot be reconstructed because it is hidden by the shelf, body, or load in too many camera views [27–29].

The OptiTrack® motion analysis laboratory used in this study was conformed of eight Prime x22 cameras (NaturalPoint, Corvallis, OR, USA) and passive markers of 14 mm diameter located on the body using the Plug-in Gait guidelines. The cameras boast a resolution of 2.2 MP and frame rate of 240 frames per second. The Mocap system is located at the Ergonomics Building, Faculty of Biological Sciences of the Universidad de Concepción. The optoelectronic MoCap system setup is shown in Figure 4.


NIOSH Lift Index

Task: Example task 1

Date: 27/03/2023

Company: Example

Single-task ● Multitask ○

EN ● ES ○

Sex: Male

Age: 35

Object Weight (L): 9,5

Control destination: □

One hand operation: □

Survey variables:

	Origin	Destination	Calculate Automatically
Horizontal distance (H)	46	(cm)	✓
Horizontal displacement (ΔH):			
Vertical height (V):	0	75	✓
Vertical displacement (ΔV):	75		
Angle of asymmetry (A):	0	(°)	✓
Rotation (ΔV):			
Grip type (C):	Deficient		

Duration: 2 ≤ hours ≤ 8

Frequency (F): 2

Factors obtained:

	CC	FH	FO	FV	FD	FA	FC	FM	RWL	LI
Origin	25	0,54	1	0,78	0,88	1	0,9	0,65	5,42	1,75
Destination										

Rate of lift: 1,75

- □ ×
Calculate Automatically
🟩 Acceptable
🟨 Present risk
🟧 Present risk: High level
🟥 Present risk: Very high level
Camera
Accept
Cancel
: Information about the task and evaluating company
: Survey information
: Factors calculated from input data

Figure 3. Digitized NIOSH platform (data entered in example).

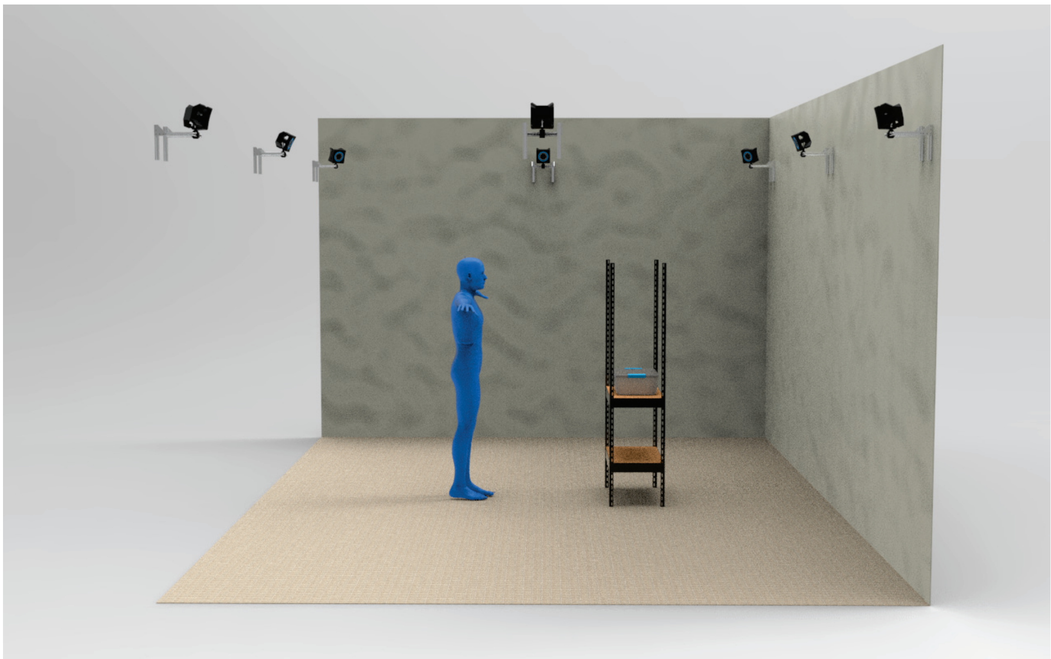


Figure 4. The motion analysis laboratory setup featuring an optoelectronic system alongside evaluation elements. The calibration volume dimensions were 5 m in length, 3 m in width, and 2 m in height.

2.3.2. BLE-Based Inertial Mocap System

IMU-based systems are frequently used in human motion studies [30]. IMUs comprises an accelerometer, a gyroscope, and a magnetometer. They can be worn directly on the body and do not suffer from occlusion, making them suitable for field studies in real working environments. They are also less expensive and more time-efficient to set up compared to optical motion capture systems, and hence, making them more versatile and quicker to deploy [31–33].

The inertial MoCap system used in this study was developed in the Biomedical Engineering Laboratory at the Universidad de Concepción, and comprises 18 measurement units. Its primary feature is the utilization of Bluetooth 5.0 for wireless communication to a central acquisition computer running a specialized program for ergonomic risk assessment. The organization of the sensor connections and communication is based on a "tree topology" network, where 5 of the 18 sensors function as central units and the remaining ones as peripheral units (see Figure 5).

The sensor organization is as follows:

- Head sensor (Central Unit 1) connects with three peripherals (thoracic spine, lumbar spine, and sacral spine).
- The right shoulder sensor (Central Unit 2) connects with three peripherals (right arm, right forearm, and right hand).
- The left shoulder sensor (Central Unit 3) connects with three peripherals (left arm, left forearm, and left hand).
- Right thigh sensor (Central Unit 4) connects with two peripherals (right leg and right foot).
- Left thigh sensor (Central Unit 5) connects with two peripherals (left leg and left foot).
- A Bluetooth USB serial dongle is a hub for the five central units and connects to the PC.

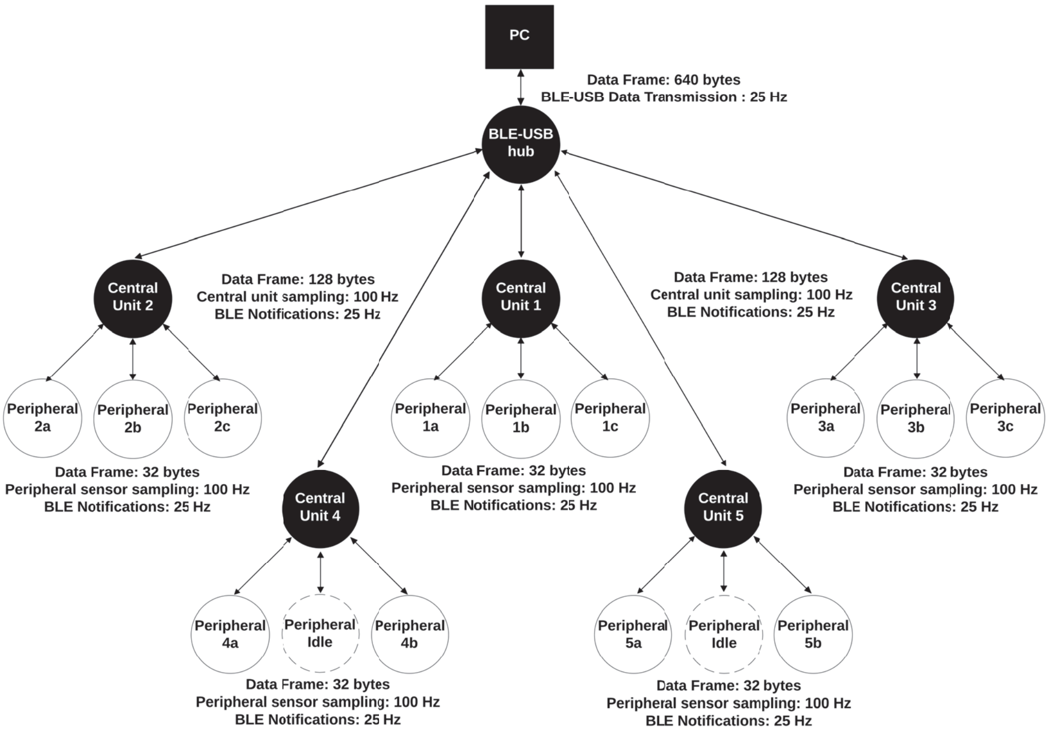


Figure 5. Bluetooth 5.0 tree topology implemented for the wireless communication of the custom inertial motion capture system, where: Central Units 1 to 5 represent the head, right shoulder, left shoulder, right thigh, and left thigh sensors, respectively. Peripheral 1a to 1c represent thoracic, lumbar, and sacral spine sensors. Peripheral 2a to 2c and 3a to 3c represent arm, forearm, and hand sensors for the right and left sides, respectively. Peripheral 4a to 4b and 5a to 5b represent leg and foot sensors for the right and left sides, respectively. The Peripheral Idle represents a virtual device that contains dummy data to maintain the frame format for each central node.

Each sensor unit (central and peripheral) integrates a 9-degree-of-freedom Inertial Measurement Unit (IMU) consisting of a 3-axis accelerometer, 3-axis gyroscope, and 3-axis magnetometer. The system uses a BNO055 from Bosch Sensortec, an STM32L4 series microcontroller from STMicroelectronics, and a Bluetooth Low Energy v5.0 (BLE v5.0) working as a Host Processor. The BLE's main characteristics ensure a maximum data throughput of 2 Mb/s at low energy consumption, allowing for approximately six hours of continuous measurements in our case.

Each sensor measures global quaternion rotations at a frequency of 100 Hz, organized into 8-byte frames. Unlike inertial sensors that connect directly to a central computer or using a router or Wi-Fi hub, to ensure low power consumption typically used in BLE applications, the stability of the topology, and communication via BLE notifications, a pipeline was used where each central and peripheral unit accumulates four samples. This approach results in each sensor sending notifications at a lower rate of 25 Hz to the central units. The central units organize their data, including the peripheral units' data, creating a 128-byte data frame, which is then sent via notifications to the BLE-USB hub at 25 Hz. This ensures that notifications are generated within the time constraints of the BLE protocol, maintaining communication stability and guaranteeing a sampling rate of 100 Hz. Additionally, this allows the limbs and the dorsal trunk to be measured independently from the other units, as they are composed of separate central units.

Upon powering the peripheral units, they send advertising packets scanned by the central units. When the central units scan their peripherals, they request connections. Once the central unit establishes connections with all its corresponding peripherals, it starts sending advertising packets for the BLE-USB hub to recognize and initiate connection requests. After the connection with the hub is established, the Service Discovery process starts, followed by subscription to the notifications from the connected central units and enabling inertial sensor sampling. This, in turn, causes the central units to subscribe to the notifications from their corresponding peripherals and enables sampling of their respective measurement units. All the connection and communication protocol mentioned is illustrated in Figure 6.

Figure 7 displays the positioning of each inertial sensor that comprises the designed suit. It also illustrates the method by which the data are transmitted from the sensors to the computer.

2.4. Test Procedure

Before the trials, each participant signed an informed consent approved by the Vice-rectory for Research and Development of the Universidad de Concepción, code CEBB 794-2020. Each subject was asked to perform two lifts (Task 1 and Task 2) of a load to a shelf located in front of them (Figure 8A,B). The load was a plastic box with dimensions $42 \times 15 \times 32$ cm (width × height × depth) and a fixed mass of 9 kg. The target shelf was 74 cm wide and 39 cm long, with fixed heights of 32 and 79 cm. The protocol consisted of lifting the box from the ground, using a "Good" grip ($FC = 1$), and placing it on the aforementioned shelf. This activity was designed to create variation in the measurements of H, V, A, and D. For the remaining factors related to the method, a 7-h workday was assumed, with a lifting frequency of two lifts per minute followed by a sound rhythm to control it ($FM = 0.65$).

The H, V, A, and D measurements at both the start and finish of the lift were determined using a manual goniometer and a tape measure. These tools were also used to mark positions on the ground and on the shelf, providing a traditional measurement to take the necessary measurements to calculate the reference values that are the Gold Standard in this study. These data allowed us to compute the lifting index traditionally for comparison with the proposed semi-automatic system.

The expected values for these metrics were based on the ergonomic standards outlined in the Technical Guide for Manual Handling of Loads from the Chilean Social Security.

These values provide a baseline from which deviations in sensor measurements can be assessed, enabling a direct comparison of the accuracy and reliability of the inertial and optoelectronic systems in capturing key ergonomic metrics.

At the end of the tests, Biovision Hierarchy (BVH) motion files were exported from an optoelectronic MoCap system for each subject and imported into the proposed platform.

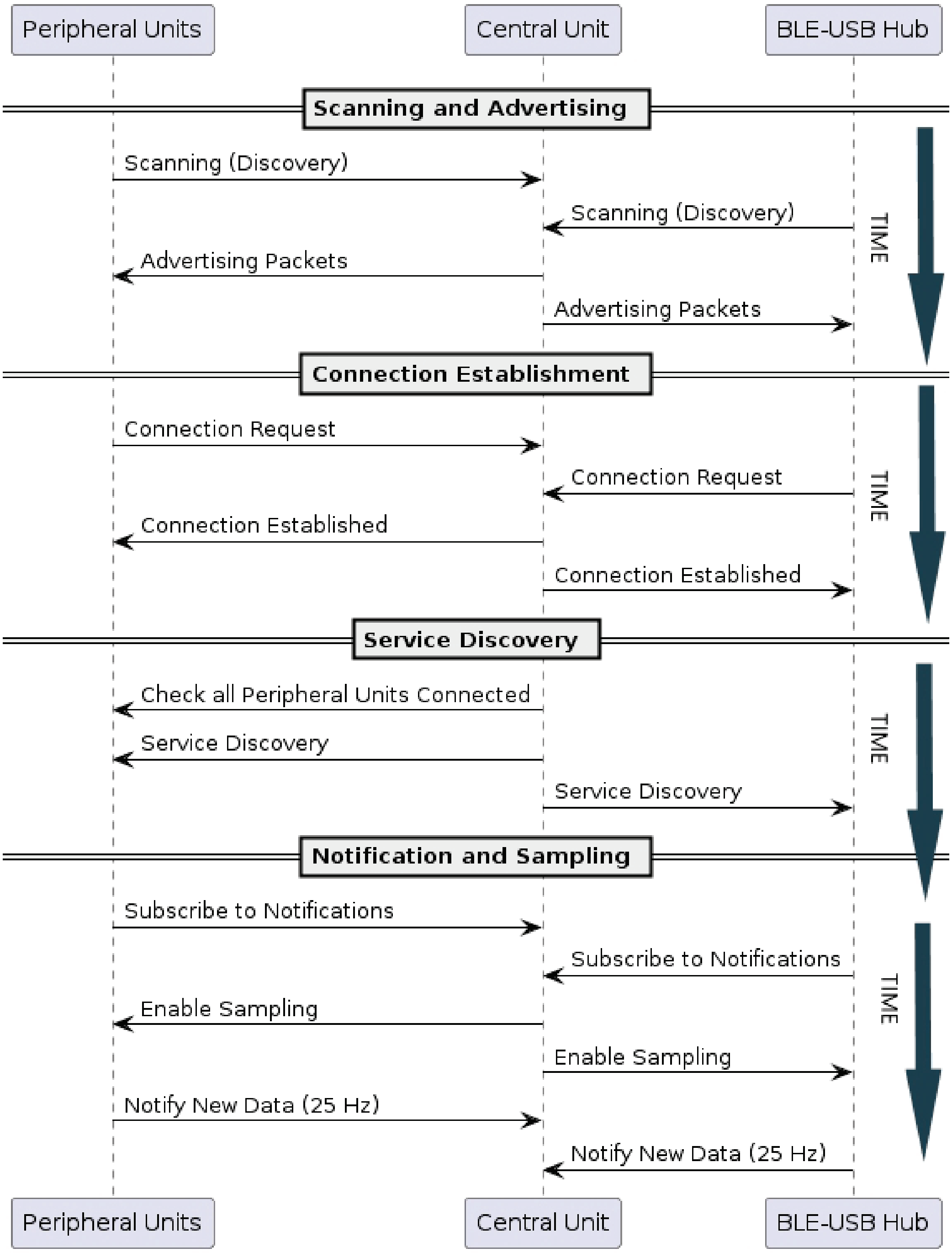


Figure 6. Bluetooth Low Energy communication sequence and connection establishment diagram.

Figure 8A,B show the experimental tasks conducted, illustrating the path from the initial position. Specifically, Figure 8A corresponds to Task 1, with a fixed trunk rotation of 45° without foot movement from the initial position. Conversely, Figure 8B represents Task 2, with a fixed trunk rotation of 90° without foot movement from the initial position.

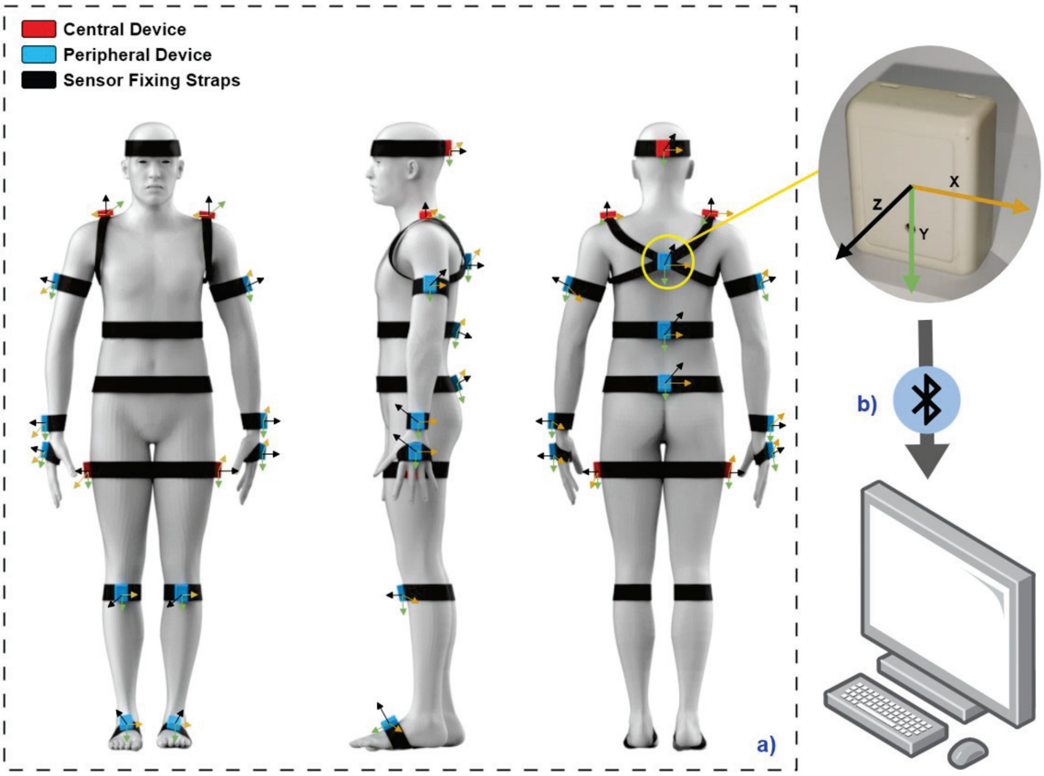
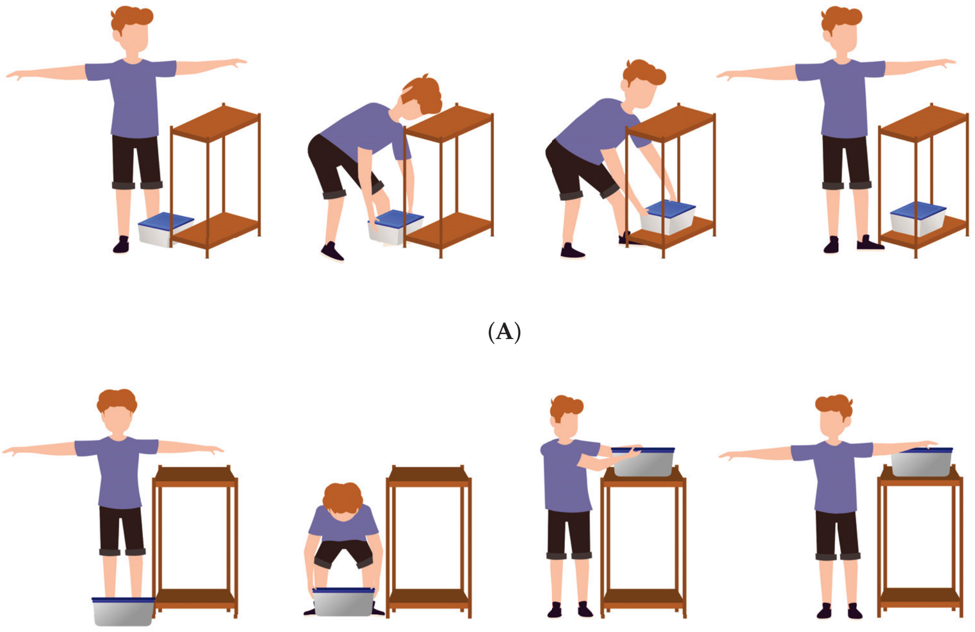


Figure 7. Wiring diagram of the generated suit (**a**) front, side, and rear view of the positioning of each of the inertial sensors that make up the sensorized suit; (**b**) descriptive diagram of data flow via Bluetooth from the suit to a computer.



(**A**)

(**B**)

Figure 8. Tasks 1 (**A**) and 2 (**B**) for ergonomic risk evaluation.

2.5. Statistical Analysis and Performance Assessment

A comprehensive analysis methodology was used to assess risk across 20 subjects. Three distinct assessment tools were used: a traditional reference method involving manual measurements with a tape measure and a digital goniometer, an optoelectronic system (OptiTrack®), and a custom BLE-based inertial sensor system. The core of our analysis was to compare the ergonomically significant variables H, V, A, and D captured by these instrumented methods.

The analysis included a statistical evaluation of the variability and accuracy of these variables, with the aim of identifying any significant differences in the data collected by the instrumented methods.

The assessment of normality or data distribution was performed using the Shapiro–Wilk test. In this case, none of the distributions of the measured metrics (H, V, A, and D) were normal.

The performance of both systems was quantitatively evaluated using four key metrics: Precision, Sensitivity, F1 Score, and Accuracy.

- **Precision** measures the proportion of correctly identified positive cases (i.e., instances where risk is present) out of all cases predicted as positive by the system. A high precision rate indicates that when the system predicts a risk, it is likely to be correct.
- **Sensitivity** (also known as recall) assesses the system's ability to correctly identify actual positive cases (i.e., instances where actual risk is present) among all cases. High sensitivity means the system effectively captures most of the at-risk tasks without missing many.
- **F1 Score** provides a balance between precision and sensitivity, considering both false positives and false negatives. This metric is particularly useful when the class distribution is uneven. Scores close to 100% indicate high precision and high sensitivity.
- **Accuracy** represents the ratio of correctly predicted observations to the total number of observations.

The analysis results and graphs were created using Matlab R2023b (MathWorks, Apple Hill Campus, Natick, MA, USA).

3. Results

The evaluation of ergonomic risks associated with manual load handling tasks was conducted using both Bluetooth Low Energy (BLE) and optical (OptiTrack®) motion capture systems. The analysis focused on comparing the distribution and statistical metrics of horizontal displacement (H), vertical displacement (V), trunk rotation (A), and vertical absolute displacement (D) captured by both systems. This section presents the results obtained from these analyses. The expected values for H, V, A, and D were predetermined based on ergonomic standards [18].

3.1. Distribution of Ergonomic Metrics

The distribution of the ergonomic metrics V, H, A, and D for the BLE Inertial MoCap and optical systems is illustrated through a series of box plots.

Figure 9 shows the variability in measurements of vertical displacement (V) and horizontal displacement (H) among participants. The BLE Inertial MoCap system displayed a wider range of values, showing a higher variability compared to the optical system. However, both systems are aligned with the expected values. The variability could reflect either inherent differences in the population or measurement technology, or a combination of both. In this case, the variability observed is due in part to participants not always staying within the requested markers, as well as differences in anthropometric assumptions made by the measurement systems. Outlier data were not treated, as the intention was to demonstrate the full and realistic functionality of our proposed system.

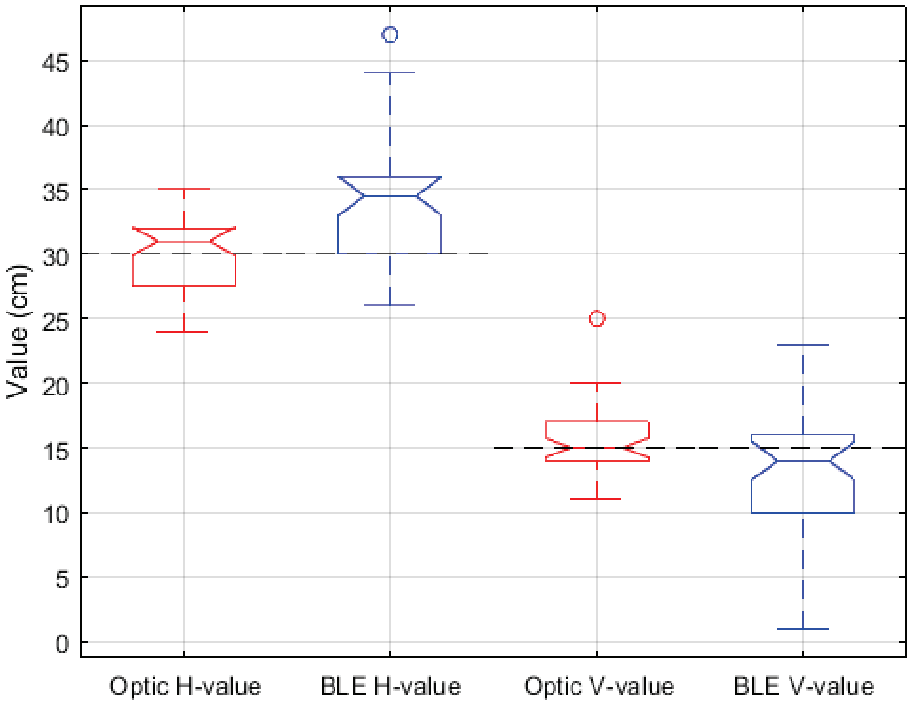


Figure 9. Distribution of ergonomic metrics V and H captured by BLE Inertial MoCap and optical system. Segmented lines correspond to expected values from Tasks 1 and 2. The box represents the interquartile range (25th to 75th percentile), with the line inside indicating the median. Whiskers extend to 1.5 times the interquartile range, and points outside are outliers.

In Figure 10, the angle of trunk rotation (A) measurements indicate an alignment between the BLE Inertial MoCap and optical systems, with both aligned with the expected task-specific values.

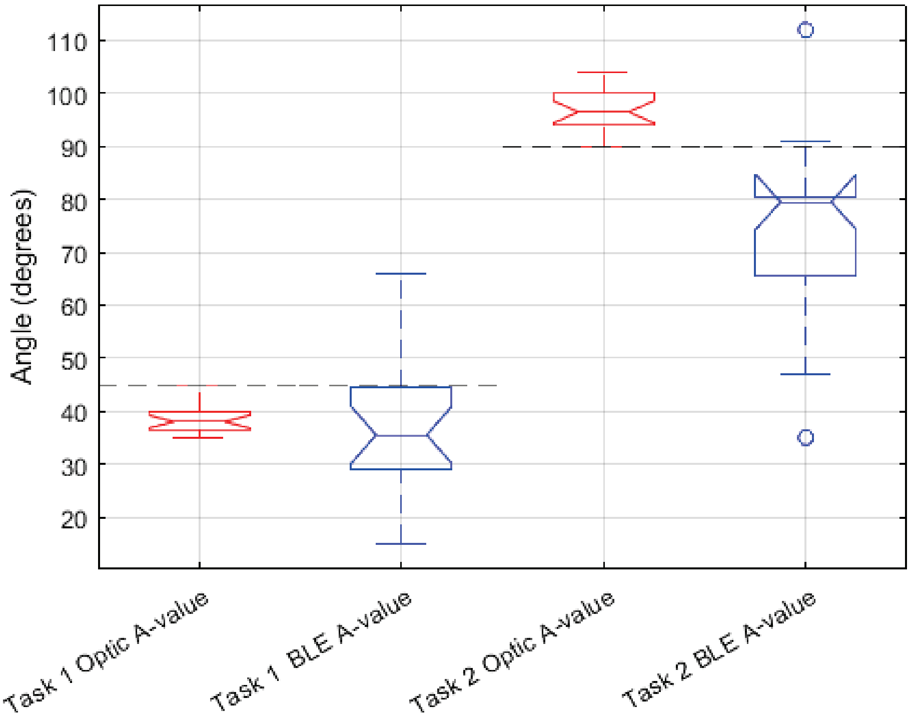


Figure 10. Distribution of ergonomic metric A (Trunk Rotation) captured by BLE and optical system. Segmented lines correspond to expected values from each task.

The box plots in Figure 11 reveal a discrepancy between the systems in measuring vertical absolute displacement (D), with the BLE Inertial MoCap system exhibiting a broader spread of values.

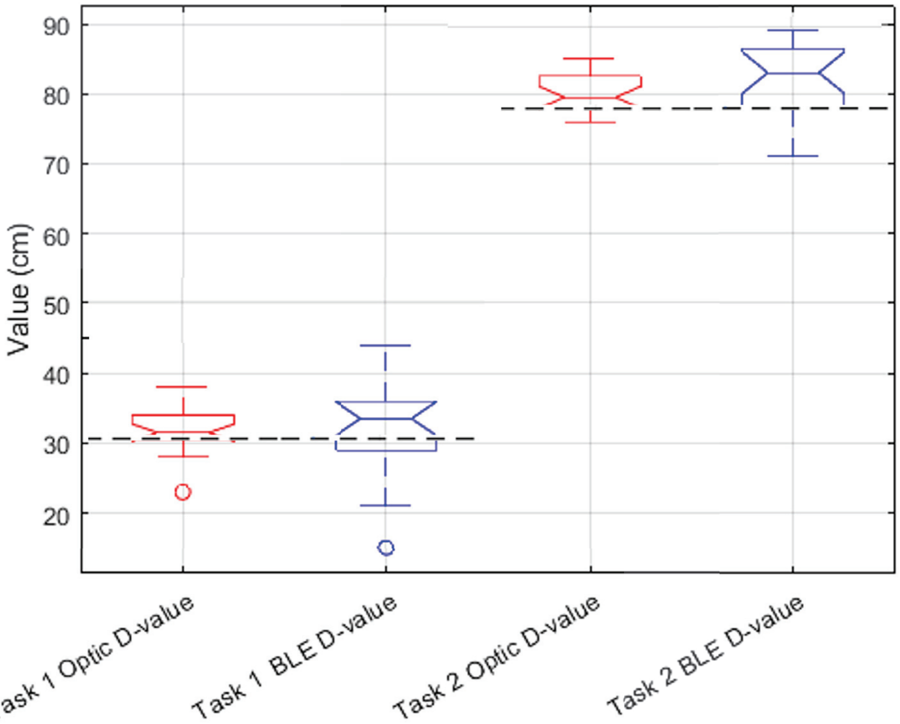


Figure 11. Distribution of ergonomic metric D captured by BLE Inertial MoCap and optical system. Segmented lines correspond to expected values from each task.

Table 5 summarizes the distribution metrics obtained from each motion capture system.

Table 5. Means and standard deviations of horizontal displacement (H), vertical displacement (V), vertical absolute displacement (D), and trunk rotations (A) for BLE Inertial MoCap and optical systems. Here, A1 denotes the results from Task 1 and A2 from Task 2.

Metric	Mean	Standard Deviation
	H (cm)	
BLE Inertial MoCap	33.87	4.46
Optical	30.12	2.91
	V (cm)	
BLE Inertial MoCap	13.17	4.75
Optical	15.67	2.63
	D (cm)	
BLE Inertial MoCap	32.05	7.36
Optical	31.80	3.25
	A1 (degrees)	
BLE Inertial MoCap	37.35	12.81
Optical	38.55	2.78
	A2 (degrees)	
BLE Inertial MoCap	73.65	16.47
Optical	96.60	4.00

3.2. Lifting Index and Recommended Weight Limit

The Lifting Index (LI) and Recommended Weight Limit (RWL) were calculated to assess ergonomic risks associated with lifting tasks.

Figure 12 visualizes the risk levels based on LI values. The majority of measurements fall within the acceptable to low risk categories, indicating that the tasks performed are within ergonomic safety limits.

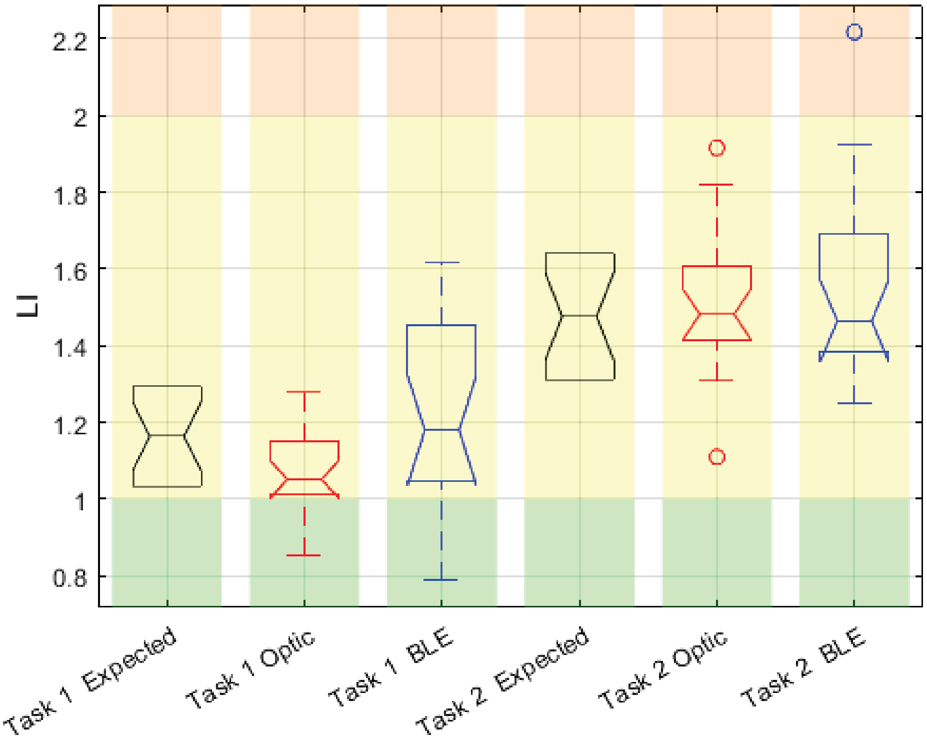


Figure 12. Distribution of the lifting index (LI) estimated by the developed platform using BVH files from each motion capture system: BLE Inertial MoCap and optical. Here, the green area is acceptable risk, yellow is low level risk, and orange is high level risk.

Similarly, Figure 13 illustrates the RWL estimations, which shows the reliability of both systems and the developed semi-automatic platform in identifying ergonomic risk levels associated with the manual handling tasks.

Tables 6 and 7 summarizes the distribution metrics obtained from each motion capture system.

Table 6. Means and standard deviations of LI1 and LI2, along with expected values, for BLE Inertial MoCap and optical systems. Here, LI1 denotes the results from Task 1 and LI2 from Task 2.

Metric	Mean	Standard Deviation
LI1 BLE	1.22	0.23
LI1 Optical	1.07	0.10
LI1 Expected	1.16	0.13
LI2 BLE	1.55	0.24
LI2 Optical	1.50	0.18
LI2 Expected	1.47	0.16

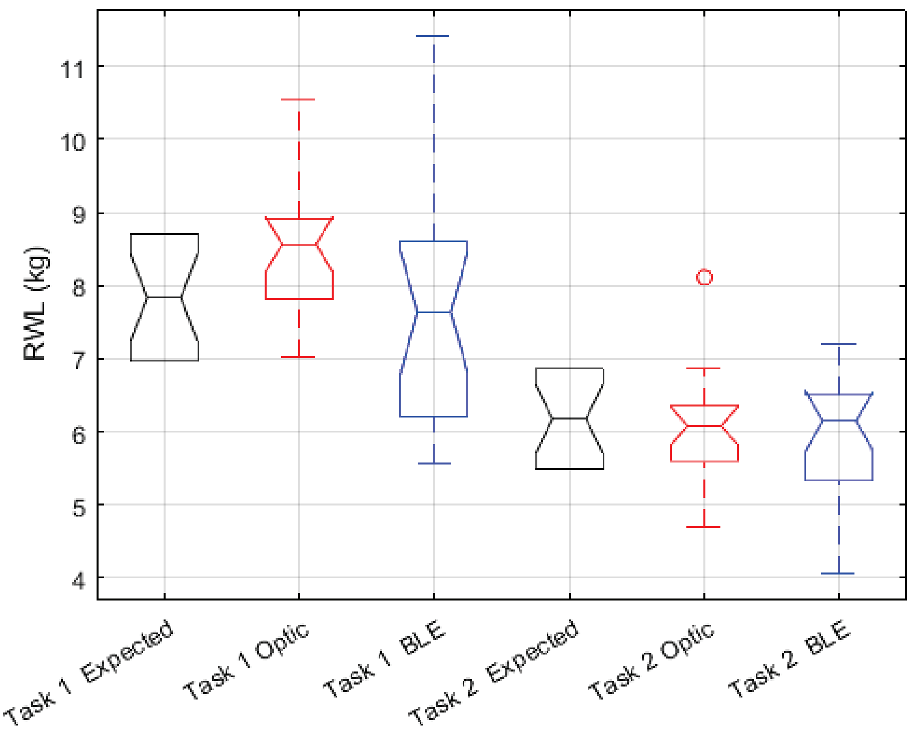


Figure 13. Distribution of the recommended weight limit (RWL) estimated by the developed platform using BVH files from each motion capture system: BLE Inertial MoCap and optical.

Table 7. Means and standard deviations of RWL1 and RWL2, along with expected values, for BLE Inertial MoCap and optical systems. Here, RWL1 denotes the results from Task 1 and RWL2 from Task 2.

Metric	Mean	Standard Deviation
RWL1 BLE (kg)	7.59	1.49
RWL1 Optical (kg)	8.44	0.82
RWL1 Expected (kg)	7.83	0.89
RWL2 BLE (kg)	5.91	0.83
RWL2 Optical (kg)	6.04	0.75
RWL2 Expected (kg)	6.17	0.70

3.3. Correlation and Statistical Significance

A Spearman correlation analysis was conducted to evaluate the agreement between the BLE Inertial MoCap and optical systems in estimating LI with the proposed platform. The analysis revealed a strong positive correlation, indicating that both systems produce consistent LI estimations (see Figure 14).

Further analysis involved calculating the lifting index (LI) and recommended weight limit (RWL) for assessing ergonomic risk. The LI and RWL values were derived for each system and compared against expected benchmarks to determine risk levels.

Statistical tests, including Mann–Whitney U, were performed to assess the significance of differences between the systems and expected values (see Table 8). The tests showed that for certain metrics (V, H, and A), there were significant differences showing areas for improvement in measurement accuracy or methodological adjustments for future assessments.

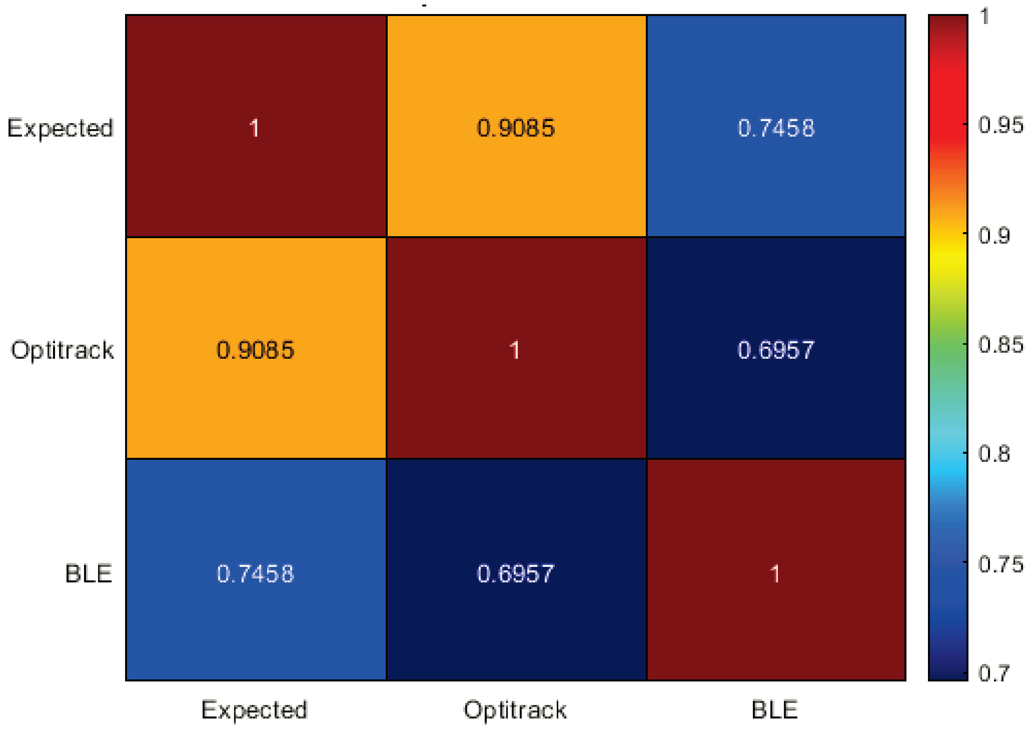


Figure 14. Spearman correlation matrix for the LI values obtained from the two motion capture systems and the expected values.

Table 8. Significance of differences in metrics between BLE Inertial MoCap and OptiTrack® systems with respect the expected values (* *p*-value < 0.05).

Metric	*p*-Value
V OptiTrack® *	0.01
V BLE *	0.05
H OptiTrack® *	0.00
H BLE *	0.08
A OptiTrack® *	0.02
A BLE *	0.08
D OptiTrack®	0.42
D BLE	0.21
LI BLE	0.21
LI OptiTrack®	0.63
RWL BLE	0.21
RWL OptiTrack®	0.63

The performance obtained due the risk classification values are shown in Table 9.

- Precision: Both BLE Inertial MoCap and OptiTrack® systems achieved a precision rate of 98.5%, indicating that the 98.5% risk categorizations they made were correct when they predicted a risk.
- Sensitivity: The BLE Inertial MoCap system exhibited a sensitivity of 97.5%, meaning it could identify 97.5% of the tasks that were actually at risk. The OptiTrack® system achieved a sensitivity of 98.7%.
- F1 Score: The BLE Inertial MoCap system's F1 score was 97.9%, and the OptiTrack® system achieved an score of 98.6%.
- Accuracy: The BLE Inertial MoCap system's accuracy was 97.5%, and the OptiTrack® system scored 98.5%.

The OptiTrack® system demonstrated consistent performance across all metrics, achieving a precision of 98.5%, sensitivity of 98.7%, F1 score of 98.6%, and accuracy of 98.5%. The BLE Inertial MoCap system, while lower in sensitivity and accuracy, still showed robust performance with a precision of 98.5%, sensitivity of 97.5%, F1 score of 97.9%, and accuracy of 97.5%. These results shows that although the BLE system had minor variations in risk detection, it still maintained a strong performance in precision and overall balance between precision and sensitivity.

Table 9. Risk categorization performance of BLE Inertial MoCap and OptiTrack® Systems from the LI values respect the expected metrics.

Metric	BLE Inertial MoCap (%)	OptiTrack® (%)
Precision	98.5	98.5
Sensitivity	97.5	98.7
F1 Score	97.9	98.6
Accuracy	97.5	98.5

4. Discussion

This study compares an optical system (OptiTrack®) and a BLE-based system with IMUs in assessing ergonomic risks through the NIOSH Lifting Index. The analysis involved the evaluation of key ergonomic variables (horizontal displacement H, vertical displacement V, trunk rotation A, and vertical absolute displacement D), which are important in calculating the Recommended Weight Limit (RWL) and Lifting Index (LI).

The findings revealed that both systems demonstrated high precision in LI-based risk categorization, with the BLE Inertial MoCap system achieving a precision rate of 98.5% and the OptiTrack® system achieving 98.5% as well (Table 9). This indicates that when risks were identified by either system, they reliably reflected ergonomic concerns. However, it is important to acknowledge that these conclusions are based on only two tasks, which cannot fully represent the wide range of real-world settings. Additionally, the expected values for Task 2 lie in the middle of the low-risk band, increasing the likelihood that any deviations in the OptiTrack® or BLE values would remain in the same band. Therefore, it is important to recognize that if the expected values were closer to the limits of the classification bands (e.g., LI = 1 or LI = 2), the metrics might not have been as strong as they are.

While the BLE Inertial MoCap system demonstrated lower sensitivity (97.5%) and accuracy (97.5%) compared to the OptiTrack® system's sensitivity (98.7%) and accuracy (98.5%), it still performed robustly in real-world settings where versatility and ease of deployment are crucial. Additionally, the BLE Inertial MoCap system's F1 score of 97.9% reflects a strong balance between precision and sensitivity, while the OptiTrack® system's F1 score of 98.6% reflects a higher overall performance. These results demonstrate the potential of both systems in ergonomic risk assessment, although the superior performance of the OptiTrack® system suggests that it may be more reliable in certain scenarios.

An observation from the results was the alignment of both systems with expected ergonomic standards for measurements of H and V (Figure 9). The BLE Inertial MoCap system exhibited higher variability in V measurements, which could be attributed to the inherent nature of wearable systems that may be subjected to more noise and movement artifacts than stationary optical systems. Despite this variability, the BLE Inertial MoCap system's measurements for D were consistent with those of the OptiTrack® system. However, the increased variability in some measurements, such as V and H, suggests that further refinement of the BLE system is necessary to enhance its measurement accuracy.

Significant differences were noted in the measurements of V, H, and A between the systems (Table 8), highlighting specific areas for improvement in the BLE-based system. These differences underline the need for ongoing refinement of BLE-based systems to better match the accuracy of optical systems like OptiTrack®, especially when applied in complex real-world environments. While both systems produced reliable LI estimations, the comparison between the systems' performance and their proximity to the expected ergonomic values shows that the BLE system may require additional adjustments to achieve the same level of precision as OptiTrack®.

The results of this study align with previous research validating inertial measurement units (IMUs) against optoelectronic systems. For instance, Robert-Lachaine et al. demonstrated that IMUs provide accurate whole-body motion analysis, comparable to optoelectronic systems [26]. However, it is important to note that these studies also emphasize certain limitations of IMUs, such as susceptibility to drift and noise, which may explain some of the variability observed in the BLE Inertial MoCap system's performance in this study.

Moreover, critical ergonomic variables H, V, A, and D were essential in the risk assessment methodology. These variables are fundamental in ergonomic risk evaluation, as corroborated by Yunus et al., who developed a portable system for real-time biomechanical risk assessment in repetitive tasks [34]. Nevertheless, the differences observed in the measurements of V, H, and A highlight that, while the BLE Inertial MoCap system has great potential for practical applications, it may still require refinement to match the level of precision achieved by more established systems like OptiTrack®.

The practical applicability of BLE Inertial MoCap systems in real-world work environments, emphasized in this study, aligns with the findings of Giannini et al., who discussed the advantages of inertial systems for field studies over optoelectronic systems, which are often hindered by occlusion issues and high costs [32]. However, it is important to remain cautious when generalizing these results, as additional testing and validation in a broader range of tasks and environments are necessary to fully understand the limitations and strengths of each system.

On the other hand, there are simpler systems that utilize straightforward methods for measuring ergonomic risks associated with manual material handling [10,35]. However, these systems incorporate complex algorithms and have only been tested in controlled and unrealistic conditions, neglecting their practical applicability in real environments and the possibility of detecting not only the presence of risk, but also identifying the specific joint or segment at risk due to the lack of full-body sensorization. This highlights the advantages of full-body motion capture systems like BLE and OptiTrack®, which provide a more comprehensive analysis of ergonomic risks.

While both the BLE Inertial MoCap and OptiTrack® systems provide valuable tools for ergonomic risk assessment, further research is needed to evaluate their performance in more diverse and complex settings. The practical implications are significant for workplace ergonomics, where rapid, accurate risk assessments are crucial for preventing musculoskeletal disorders. The semi-automatic platform developed for this study facilitates more efficient and precise risk assessments across diverse work environments using instrumented MoCap technologies.

Future work will focus on minimizing the variability and enhancing the sensitivity of BLE Inertial MoCap systems to match the reliability seen in optical systems. Additionally, it will be important to investigate the specific scenarios in which each system excels and determine the most appropriate contexts for their use.

This research advances the capabilities in ergonomic assessment, providing robust tools that combine advanced technology with practical applications in workplace safety.

5. Limitations and Future Work

This study focused on validating the semi-automatic system rather than detecting risks per se, and hence, a cohort of healthy subjects was used. Future studies should implement the system in real-world conditions and evaluate subjects who present ergonomic risks to provide a comprehensive assessment of its effectiveness in detecting and mitigating musculoskeletal disorders in diverse workplace environments.

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

Funding: This research was funded by ANID Chile and Fondef IdeA project "Technological system for the evaluation of ergonomic risks of disorders musculoskeletal following national regulations and international methodologies", Code: ID 20I10297.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki, and approved by the Vicerectory for Research and Development of the Universidad de Concepción, code CEBB 794-2020.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The original contributions presented in the 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.

Abbreviations

The following abbreviations are used in this manuscript:

MSDs	Assessing musculoskeletal disorders
BLE	Bluetooth Low Energy
IMUs	Inertial Measurement Units
H	Horizontal Displacement
V	Vertical Displacement
A	Asymmetry Angle
D	Absolute Vertical Displacement between Start and Finish
RWL	Recommended Weight Limit
LI	NIOSH Lifting Index
MoCap	Motion Capture

References

1. Gomes, J.O. El papel de la ergonomía en el cambio de las condiciones de trabajo: Perspectivas en América Latina. *Rev. Cienc. Salud* **2014**, *12*, 5–8. [CrossRef]
2. Jaffar, N.; Abdul-Tharim, A.; Mohd-Kamar, I.; Lop, N. A literature review of ergonomics risk factors in construction industry. *Procedia Eng.* **2011**, *20*, 89–97. [CrossRef]
3. Asensio Cuesta, S.; Bastante Ceca, M.J.; Diego Más, J.A. *Evaluación Ergonómica de Puestos de Trabajo*; Ediciones Paraninfo, SA: Madrid, Spain, 2012.
4. Tompa, E.; Mofidi, A.; Van Den Heuvel, S.; Van Bree, T.; Michaelsen, F.; Jung, Y.; Porsch, L.; Van Emmerik, M. The Value of Occupational Safety and Health and the Societal Costs of Work-Related Injuries and Diseases. Publications Office. Online, 2019. Available online: https://data.europa.eu/doi/10.2802/251128 (accessed on 27 September 2024).
5. Centers for Disease Control and Prevention. Musculoskeletal Disorders. Online, 2020. Available online: https://www.cdc.gov/workplacehealthpromotion/health-strategies/musculoskeletal-disorders/index.html (accessed on 13 January 2023).
6. Putz-Anderson, V.; Bernard, B.P.; Burt, S.E.; Cole, L.L.; Fairfield-Estill, C.; Fine, L.J.; Grant, K.A.; Gjessing, C.; Jenkins, L.; Hurrell, J.J., Jr.; et al. Musculoskeletal disorders and workplace factors. *Natl. Inst. Occup. Saf. Health (NIOSH)* **1997**, *104*, 97–141.
7. Rajendran, M.; Sajeev, A.; Shanmugavel, R.; Rajpradeesh, T. Ergonomic evaluation of workers during manual material handling. *Mater. Today Proc.* **2021**, *46*, 7770–7776. [CrossRef]
8. Garg, A.; Boda, S.; Hegmann, K.T.; Moore, J.S.; Kapellusch, J.M.; Bhoyar, P.; Thiese, M.S.; Merryweather, A.; Deckow-Schaefer, G.; Bloswick, D.; et al. The NIOSH lifting equation and low-back pain, Part 1: Association with low-back pain in the backworks prospective cohort study. *Hum. Factors* **2014**, *56*, 6–28. [CrossRef] [PubMed]
9. Hafez, K. The Influence of Lifting Horizontal Distance Measurement Error on NIOSH Lifting Equation Assessment Outcomes. *Phys. Ergon. Hum. Factors* **2022**, *63*, 170–178.
10. Donisi, L.; Cesarelli, G.; Coccia, A.; Panigazzi, M.; Capodaglio, E.M.; D'Addio, G. Work-related risk assessment according to the revised NIOSH lifting equation: A preliminary study using a wearable inertial sensor and machine learning. *Sensors* **2021**, *21*, 2593. [CrossRef] [PubMed]

11. Greene, R.L.; Chen, G.; Lu, M.L.; Hen Hu, Y.; Radwin, R.G. Enhancing the Revised NIOSH Lifting Equation using ComputerVision. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*; SAGE Publications Sage CA: Los Angeles, CA, USA, 2021; Online; Volume 65, pp. 467–471. Available online: https://journals.sagepub.com/doi/10.1177/1071181321651211 (accessed on 27 September 2024).
12. Spector, J.T.; Lieblich, M.; Bao, S.; McQuade, K.; Hughes, M. Automation of workplace lifting hazard assessment for musculoskeletal injury prevention. *Ann. Occup. Environ. Med.* **2014**, *26*, 15. [CrossRef] [PubMed]
13. Harari, Y.; Bechar, A.; Riemer, R. Workers' biomechanical loads and kinematics during multiple-task manual material handling. *Appl. Ergon.* **2020**, *83*, 102985. [CrossRef]
14. Akhmad, S.; Arendra, A.; Findiastuti, W.; Lumintu, I.; Pramudita, Y.D.; Mualim. Wearable IMU Wireless Sensors Network for Smart Instrument of Ergonomic Risk Assessment. In Proceedings of the 2020 6th Information Technology International Seminar (ITIS), Surabaya, Indonesia, 14–16 October 2020; pp. 213–218. [CrossRef]
15. Muller, A.; Corbeil, P. Back loading estimation during team handling: Is the use of only motion data sufficient? *PLoS ONE* **2020**, *15*, e0244405. [CrossRef]
16. Skals, S.; Bláfoss, R.; de Zee, M.; Andersen, L.L.; Andersen, M.S. Effects of load mass and position on the dynamic loading of the knees, shoulders and lumbar spine during lifting: A musculoskeletal modelling approach. *Appl. Ergon.* **2021**, *96*, 103491. [CrossRef] [PubMed]
17. Mendívil, J.A.G.; Rodríguez-Paz, M.X.; Caballero-Montes, E.; Zamora-Hernandez, I. Defining optimal lifting loads using augmented reality and internet of things. *Hum. Factors Syst. Interact.* **2023**, *84*. [CrossRef]
18. Superintendencia de Seguridad Social. *Guía Técnica: Manejo Manual de Carga*; Superintendency of Social Security, SUSESO: Santiago, Chile, 2021.
19. Retamal, G.; Gutiérrez, M.; Gómez, B.; Aqueveque, P.; Peña, G.; Baquedano, D. Desarrollo de plataforma para evaluar riesgo de trastornos musculoesqueléticos en actividades de manipulación manual de carga-resultados preliminares. *Ergon. Investig. Desarro.* **2022**, *4*, 54–67. [CrossRef]
20. Gutiérrez Henríquez, M.; Aqueveque Navarro, P.; Gómez Arias, B.; Figueroa Galindo, F. Diseño de Maniquí Informático para la Representación Gráfica de Posturas: Ergonomía y Diseño. *Atacama Journal of Health Sciences*, *1*. Online, 2022. Available online: https://salud.uda.cl/ajhs/index.php/ajhs/article/view/56 (accessed on 27 September 2024).
21. Castellucci, I.; Viviani, C.; Martínez, M. *Tablas de Antropometría de la Población Trabajadora chilena*; Universidad de Valparíaso, Mutual de Seguridad: Viña del Mar, Chile, 2017.
22. Pheasant, S.; Haslegrave, C.M. *Bodyspace: Anthropometry, Ergonomics and the Design of Work*; CRC Press: Boca Raton, FL, USA, 2018.
23. Waters, T.R.; Putz-Anderson, V.; Garg, A.; Fine, L.J. Revised NIOSH equation for the design and evaluation of manual lifting tasks. *Ergonomics* **1993**, *36*, 749–776. [CrossRef]
24. Mavor, M.P.; Ross, G.B.; Clouthier, A.L.; Karakolis, T.; Graham, R.B. Validation of an IMU suit for military-based tasks. *Sensors* **2020**, *20*, 4280. [CrossRef] [PubMed]
25. Zügner, R.; Tranberg, R.; Timperley, J.; Hodgins, D.; Mohaddes, M.; Kärrholm, J. Validation of inertial measurement units with optical tracking system in patients operated with Total hip arthroplasty. *BMC Musculoskelet. Disord.* **2019**, *20*, 52. [CrossRef]
26. Robert-Lachaine, X.; Mecheri, H.; Larue, C.; Plamondon, A. Validation of inertial measurement units with an optoelectronic system for whole-body motion analysis. *Med Biol. Eng. Comput.* **2017**, *55*, 609–619. [CrossRef]
27. Mündermann, L.; Corazza, S.; Andriacchi, T.P. The evolution of methods for the capture of human movement leading to markerless motion capture for biomechanical applications. *J. Neuroeng. Rehabil.* **2006**, *3*, 1–11. [CrossRef] [PubMed]
28. Bravo, D.; Rengifo, C.; Agredo, W. Comparación de dos Sistemas de Captura de Movimiento por medio de las Trayectorias Articulares de Marcha. *Rev. Mex. Ing. Biomed.* **2016**, *37*, 149–160.
29. Furtado, J.S.; Liu, H.H.; Lai, G.; Lacheray, H.; Desouza-Coelho, J. Comparative analysis of optitrack motion capture systems. In Proceedings of the Advances in Motion Sensing and Control for Robotic Applications: Selected Papers from the Symposium on Mechatronics, Robotics, and Control (SMRC'18)-CSME International Congress 2018, Toronto, ON, Canada, 27–30 May 2018; Springer: Berlin/Heidelberg, Germany, 2019; pp. 15–31.
30. Alarcón-Aldana, A.C.; Callejas-Cuervo, M.; Bo, A.P.L. Upper limb physical rehabilitation using serious videogames and motion capture systems: A systematic review. *Sensors* **2020**, *20*, 5989. [CrossRef]
31. Corrales, J.A.; Candelas, F.A.; Torres, F. Hybrid tracking of human operators using IMU/UWB data fusion by a Kalman filter. In Proceedings of the 3rd ACM/IEEE International Conference on Human Robot Interaction, Amsterdam, The Netherlands, 12–15 March 2008; pp. 193–200.
32. Giannini, P.; Bassani, G.; Avizzano, C.A.; Filippeschi, A. Wearable sensor network for biomechanical overload assessment in manual material handling. *Sensors* **2020**, *20*, 3877. [CrossRef] [PubMed]
33. Echeverry, L.L.G.; Henao, A.M.J.; Molina, M.A.R.; Restrepo, S.M.V.; Velásquez, C.A.P.; Bolívar, G.J.S. Human motion capture and analysis systems: A systematic review/Sistemas de captura y análisis de movimiento cinemático humano: Una revisión sistemática. *Prospectiva* **2018**, *16*, 24–34. [CrossRef]

34. Yunus, M.N.H.; Jaafar, M.H.; Mohamed, A.S.A.; Azraai, N.Z.; Hossain, M.S. Implementation of kinetic and kinematic variables in ergonomic risk assessment using motion capture simulation: A review. *Int. J. Environ. Res. Public Health* **2021**, *18*, 8342. [CrossRef] [PubMed]
35. Prisco, G.; Romano, M.; Esposito, F.; Cesarelli, M.; Santone, A.; Donisi, L.; Amato, F. Capability of Machine Learning Algorithms to Classify Safe and Unsafe Postures during Weight Lifting Tasks Using Inertial Sensors. *Diagnostics* **2024**, *14*, 576. [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.

technologies

MDPI

Review

Wearable Fall Detectors Based on Low Power Transmission Systems: A Systematic Review

Manny Villa [1,2] and Eduardo Casilari [2,*]

[1] Programa de Ingeniería Electrónica, Universidad de Investigación y Desarrollo (UDI), Bucaramanga 680001, Colombia; mvilla2@udi.edu.co

[2] Departamento de Tecnología Electrónica, Instituto TELMA, Universidad de Málaga, 29071 Málaga, Spain

[*] Correspondence: ecasilari@uma.es; Tel.: +34-952-132-755; Fax: +34-952-131-447

Abstract: Early attention to individuals who suffer falls is a critical aspect when determining the consequences of such accidents, which are among the leading causes of mortality and disability in older adults. For this reason and considering the high number of older adults living alone, the development of automatic fall alerting systems has garnered significant research attention over the past decade. A key element for deploying a fall detection system (FDS) based on wearables is the wireless transmission method employed to transmit the medical alarms. In this regard, the vast majority of prototypes in the related literature utilize short-range technologies, such as Bluetooth, which must be complemented by the existence of a gateway device (e.g., a smartphone). In other studies, standards like Wi-Fi or 3G communications are proposed, which offer greater range but come with high power consumption, which can be unsuitable for most wearables, and higher service fees. In addition, they require reliable radio coverage, which is not always guaranteed in all application scenarios. An interesting alternative to these standards is Low Power Wide Area Network (LPWAN) technologies, which minimize both energy consumption and hardware costs while maximizing transmission range. This article provides a comprehensive search and review of that works in the literature that have implemented and evaluated wearable FDSs utilizing LPWAN interfaces to transmit alarms. The review systematically examines these proposals, considering various operational aspects and identifying key areas that have not yet been adequately addressed for the viable implementation of such detectors.

Keywords: LPWAN; wearable devices; fall detection; LoRaWAN; Sigfox; NB-IoT

1. Introduction

Interest in biomedical telemonitoring research has significantly increased worldwide in recent years. This is due to its ability to monitor patients' and users' well-being remotely, enabling personalized treatments within familiar environments at a much lower cost than the traditional monitoring procedures carried out in specific facilities such as hospitals or nursing homes [1]. Technological advancements in sensors, electronic elements' miniaturization, the evolution of communication networks, and artificial intelligence have led to the development of low-cost wearable devices capable of monitoring vital and biometric signals [2]. The detection of critical, health-affecting events such as falls becomes a fundamental aspect in remote supervision of elderly individuals or those with mobility issues.

Falls occur when balance is involuntarily lost, resulting in the body impacting the ground or any other firm surface [3]. The World Health Organization (WHO) highlights that falls are a significant public health concern, being the second leading cause of accidental injury deaths worldwide. It is estimated that each year, approximately 684,000 individuals die worldwide due to falls, with the population of adults aged over 60 experiencing the highest number of fatal falls [4]. In addition, it has been projected that by 2050, the population of individuals aged 60 and above will reach around 2.1 billion, representing

Technologies **2024**, *12*, 166. https://doi.org/10.3390/technologies12090166

approximately 22% of the global population [5]. Currently, about 37.3 million falls result in severe injuries requiring medical attention. Nonetheless, timely medical intervention for individuals who have experienced a fall can lower the risk of hospitalization by 26% and reduce mortality rates by 80% [6]. For this reason, early detection of falls through remote telemonitoring systems can enhance medical care and prevent complications associated with fall accidents. In a generic manner, a fall detection system (FDS) can be defined as an architecture capable of autonomously detecting falls experienced by a particular subject and notifying caregivers as soon as these falls occur.

During the last decade, a wide variety of research related to fall detection has been developed, as evidenced in existing literature [6–34]. According to studies, a classic categorization of FDSs can be approached depending on the nature of the sensors involved in the fall detection process. In this vein, three types of sensors are normally considered: wearable devices, environmental systems, and video-based systems [26].

Video and environment-based systems, which can be grouped under the term "contextual detection systems" or "context-based systems", present similar advantages and drawbacks. Both methods use fall detection techniques involving the capture of environmental data to monitor and track body movement. Consequently, both tracking systems require the deployment and detailed configuration of sensors, cameras, and other devices in specific areas within the user's residence. Despite undeniable advancements in this type of detector, several issues that limit their effectiveness persist.

The primary limitation lies in the coverage area itself, as contextual systems demand sensor installation in constrained indoor spaces, typically within a room [10], where (in any case) dead zones or blind spots for detection may also occur. Therefore, operationality is significantly restricted to home monitoring scenarios while no ubiquitous user tracking (and freedom of movement) is permitted. Additionally, privacy concerns arise from the permanent use of cameras, while environmental sensors (such as microphones, motion detectors, etc.) can be affected by various sources of spurious noise. Furthermore, external items (furniture, pets, belongings) might fall within the tracking area and generate false alarms [35].

On the other hand, wearable fall detection devices can be seamlessly incorporated into clothing due to their reduced size. Due to the plummeting costs of wearables, they provide a more economical solution with a lower energy consumption compared to context-based systems. Typically, these devices include microcontrollers, IMU (Inertial Measurement Unit) sensors, and, in some cases, barometers, which enable fall detection based on the user's acceleration, angular velocity, orientation, or altitude. Moreover, they feature a wireless communication module, easing the remote monitoring and integration of the tracking system into IoT (Internet of Things) networked platforms [36].

Despite technological advancements in fall detection, current systems face significant challenges that limit their effectiveness in various contexts. One of the main issues is the rate of false positives [37,38], where everyday activities like sitting down quickly or bending over can be mistakenly identified as falls. This problem affects confidence in detection systems and can lead to an overload of emergency services and caregivers. Another critical challenge is user acceptance and comfort, as wearable devices need to be worn constantly to function effectively, and not all users are willing or able to adapt to this necessity [39]. Additionally, detection accuracy can be compromised in environments with electromagnetic or physical interference [33], which can affect the sensors' ability to monitor user movements properly. However, one of the major challenges is the limited autonomy of wearable devices due to data acquisition and transmission, which hampers their prolonged operation and decreases their viability for continuous monitoring without frequent recharges [40].

In this scenario, a technology emerges with significant potential for fall detection in telemonitoring contexts: LPWAN (Low Power Wide Area Network) communications are proficient in efficiently transmitting data across extensive distances while minimizing energy consumption, and hence, significantly increasing the autonomy of wearables, which are typically powered by lightweight batteries with limited capacity.

Most wearable-based FDSs integrate short-range communications, such as Bluetooth Low Energy (BLE) [41–43] or, to a lesser extent, ZigBee [44–46]. Although these technologies minimize battery drainage, they require placing a gateway or relay node in the close vicinity of the user, capable of forwarding the alarms received from the wearable to the remote monitoring point (e.g., via Wi-Fi or 3G networks). This role is usually performed by a smartphone, which the user must carry permanently, a situation that is not always possible in all the application scenarios. An alternative to a "transportable" gateway is directly integrating medium-range (Wi-Fi) long-range cellular communications, such as 3G or 4G, into the wearable. In fact, certain high-end smartwatches already incorporate these wireless interfaces. However, these technologies demand significant energy to operate [47–49]. This noticeably constrains the autonomy of the wearable, which is usually not powered by high-capacity batteries to reduce its weight. Additionally, these interfaces require being within the radio coverage of the corresponding Wi-Fi access point or cellular base station, which, depending on where the system is intended to be deployed, is not always available. In addition, the use of cellular communications adds a monthly service cost to the FDS application.

In contrast, LPWAN networks offer a low-power architecture with long-range coverage, making them particularly advantageous when deployed in outdoor environments [47,50]. LPWAN technologies, such as LoRaWAN (Long Range Wide Area Network) and Sigfox, are well-suited for IoT applications requiring extensive coverage and economical communication solutions. These networks operate in unlicensed frequency bands, significantly reducing operational costs and utilizing efficient communication protocols that allow devices to consume less energy when transmitting data [51,52]. Additionally, LPWAN solutions offer bidirectionality and the ability to establish public or private networks, providing flexibility and better adaptability to the specific needs of wearable-based monitoring systems. Each LPWAN access point or gateway can support thousands of end nodes over several kilometers, reducing implementation and maintenance expenses. This makes LPWAN architectures particularly suitable for applications in areas where traditional communication infrastructures are limited or costly to deploy [7].

In addition to the technological advancements in sensors and communication networks, the effectiveness of FDSs heavily relies on the algorithms implemented to process the data collected by wearable devices. These algorithms are designed to accurately identify falls by analyzing patterns in the inertial signals, such as acceleration, angular velocity, and orientation changes. Various algorithms, including threshold-based methods and machine-learning techniques, have been developed to enhance the accuracy and reliability of fall detection. Threshold-based methods (TBM) offer low computational complexity and can be executed directly on wearable devices without needing important hardware resources [6,8,16,26,34]. However, they may have severe limitations in distinguishing falls from other conventional activities involving energetic or fast movements. On the other hand, machine learning algorithms, including deep learning models, provide higher accuracy but require more computational resources [28,34,36,53,54]. Hybrid algorithms, which combine TBM and machine learning, leverage the advantages of both approaches to improve detection performance and energy efficiency. For instance, Yuan et al. [55] implemented a system using TBM for preliminary detection and GRU (Gated Recurrent Unit) for final classification, optimizing both accuracy and power efficiency.

In addition to technological advancements in sensors and communication networks, the effectiveness of FDSs largely depends on the algorithms implemented to process the data collected by wearable devices. These algorithms are designed to accurately identify falls by analyzing patterns in inertial signals, such as acceleration, angular velocity, and changes in orientation. Various types of algorithms have been developed for this purpose, each with its own advantages and limitations.

Threshold-based methods (TBM), which are among the simplest and most widely used, work by setting predefined limits on the inertial signals. When the data exceeds these thresholds, a fall alert is triggered. These methods offer low computational complexity [56] and can be executed directly on wearable devices without the need for significant hardware resources [6,8,16,26,34]. However, TBM methods often struggle to distinguish falls from other activities, leading to an increase in false positives and false negatives. Luque et al. [57] compared four TBM algorithms and demonstrated that simultaneously avoiding both types of errors is difficult. This is because the thresholds may be too sensitive, increasing false positives, or not sensitive enough, increasing false negatives. Adjusting these thresholds to find an optimal balance is complex and does not always guarantee consistent performance across all users and contexts.

On the other hand, machine learning algorithms have emerged as a powerful alternative to improve fall detection accuracy. These algorithms analyze large volumes of historical data to identify complex patterns and features that indicate a fall, allowing them to offer higher accuracy than threshold-based methods [28,34,36,53,54]. Among them, deep learning models, such as Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM), have proven particularly effective in fall recognition by capturing temporal dependencies in inertial signals. Salah et al. [36] developed an edge artificial intelligence FDS, achieving 95.55% accuracy using CNN and 96.78% with LSTM. However, this higher accuracy comes at a cost, as machine learning algorithms often require more computational resources [58], which can be challenging for wearable devices with limited processing and power capabilities.

To address the limitations of both approaches, hybrid algorithms have been developed that combine the strengths of TBM and machine learning. These systems typically use threshold-based methods for preliminary fall detection, taking advantage of their low energy consumption and then applying machine learning techniques for more detailed and accurate classification. However, the reverse is also possible, where the output of a Machine Learning or Deep Learning model is compared against a threshold to determine a fall. Therefore, either method can be used in any order [59]. For example, Yuan et al. [55] implemented a system that uses TBM for preliminary detection and GRU (Gated Recurrent Unit) for final classification, optimizing both accuracy and energy efficiency. These hybrid algorithms represent a trade-off between simplicity and accuracy but also introduce greater complexity in the design and implementation of the system, as they require the integration of multiple algorithmic components. While it is true that hybrid models face challenges related to selecting an appropriate threshold, these challenges can be addressed, as shown by Astriani et al. [60], who propose a method that not only relies on a simple threshold but also incorporates multiple critical features such as weightlessness, impact, post-fall immobility, and the comparison of accelerations before and after the fall. Additionally, the use of ROC (Receiver Operating Characteristic) curves is implemented to adjust the threshold dynamically, optimizing the balance between sensitivity and specificity [61]. The use of LPWAN technology in wearable devices for FDSs and biomedical telemonitoring, in general, is an ever-evolving area with noteworthy potential to enhance people's quality of life. It has clear advantages yet to be fully explored and evaluated compared to wearable systems employing short-range, low-power communications to which the literature on wearable FDSs has traditionally paid much more attention.

The primary aim of this review is to analyze and synthesize existing literature concerning the use of LPWAN technologies in wearable devices for fall detection, considering their benefits, limitations, opportunities for improvement, detection algorithms, and energy efficiency. The research focuses on the wearable devices employed, the nature of the sensors, and the algorithms that are implemented on the wearable to detect falls from the inertial signals. The main contributions of this paper are detailed below:

- It provides an overview of FDSs using wearable devices and LPWAN technologies.
- It offers a detailed examination of the predominant algorithms used in fall detection within the context of integrating LPWAN technologies.

- It conducts a comprehensive analysis of the recent state of the art, covering studies that implement LPWAN wearable technologies for fall detection and considers aspects such as the wearable devices used, their placement on the body, the sensors, and energy efficiency.
- It evaluates performance parameters such as accuracy, sensitivity, and specificity in different combinations of LPWAN technologies, detection algorithms, and sensors.
- It presents a detailed discussion on emerging trends in applying LPWAN in fall detection, as well as future research directions.

To systematically address these contributions, the paper begins by detailing the methodology used to screen the relevant literature in Section 2. This is followed by an overview of the most relevant LPWAN alternatives in Section 3. Next, Section 4 categorizes and analyzes the selected studies. Section 5 discusses the implications of the findings, while Section 6 summarizes the criticisms and limitations identified. Finally, Section 7 offers recommendations for future research and practical applications, with Section 8 presenting the main conclusions.

2. Methodology

This systematic review utilized the PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) guidelines, making specific adjustments to focus on advanced research concerning the use of LPWAN technologies in wearable devices for fall detection [62]. The methodological process consisted of the phases described in the following subsections:

2.1. Phase 1: Identification Relevant Studies (Identification)

2.1.1. Definition of the Research Question

The research question was formulated to address the topic of interest precisely: "What are the most commonly used LPWAN technologies in wearable devices for fall detection, and what is the effectiveness of their application in terms of accuracy and efficiency?".

2.1.2. Eligibility Criteria

For the selection of studies, eligibility criteria were established based on the following aspects:

Inclusion

- Full original articles published in peer-reviewed journals between 2010 and 2023.
- Research addressing the application of LPWAN technologies in wearable devices for fall detection.
- Focus on evaluating the accuracy and effectiveness of technologies in fall detection.
- Studies using inertial sensors combined with functional tests and/or daily life activities for fall detection.
- Articles that present the keywords defined in the search string in the abstract or title.

Exclusion

- Duplicated records that appear in more than one database.
- Papers not available in full-text format or not written in English.
- Works published before 2010, the year of Sigfox technology conception.
- Studies that do not present a prototype aimed at detecting falls and sending the corresponding alarm.
- Articles describing systems in which the alarm transmission technology does not involve the use of LPWAN technologies.
- Studies that do not include any type of evaluation of the developed prototype.
- Records about fall detection architecture that, although incorporating the use of LPWAN standards, are not based on wearable devices.

2.1.3. Information Sources

A comprehensive search for articles was conducted in renowned academic databases such as IEEE Xplore, MDPI, Scopus, Google Scholar, and Web of Science. While Web of Science was used to identify relevant articles, many of these were also available and downloaded from primary sources, such as IEEE Xplore.

Our search focused on articles that appeared between 2010 and 2023. The year 2010 was chosen as it marks the founding of Sigfox, the first sensor network operator with a large-scale market presence.

2.2. Phase 2: Selection of Relevant Studies (Screening)

Search and Selection of Studies

This section describes the process of searching and selecting relevant studies for the systematic review. The process is divided into several stages, starting with the design of the search string and finishing with the selection of studies that meet the established eligibility criteria.

2.2.1. Search

The terms for the bibliographic search included combinations of keywords such as ("LPWAN" OR "Low-Power Wide-Area Network" OR "LoRaWAN" OR "LoRa" OR "Symphony Link" OR "Sigfox" OR "NB-IoT" OR "LTE-M" OR "Ingenu RPMA" OR "Weightless LPWAN" OR "MIOTY" OR "DASH7") AND ("health AND fall" OR "elderly AND fall" OR "fall detection" OR "fall detection" OR "fall tracker" OR "fall detector" OR "fall sensor" OR "fall prevention" OR "fall monitoring"). These terms and logical operators, which are compatible with the search mechanisms commonly used in the databases, were alternatively introduced in the academic databases to identify the related literature.

As mentioned above, the search covered articles published in 2010, when Sigfox [63,64], the first widespread LPWAN technology, was created. However, significant research combining LPWAN and FDSs began to emerge in 2018, marking the beginning of specialized research in this specific area.

2.2.2. Title and Abstract Exploration

To evaluate the initial relevance of search results, a thorough examination of the titles and abstracts was carried out. Studies that clearly did not fit the research topic were discarded. Two researchers (M.L., E.C.P.) conducted independent reviews of the articles based on the set eligibility criteria. A consensus approach was followed since no relevant difference was found between the two independent analyses. Hence, no third reviewer was required to complete this exploration.

2.2.3. Potentially Relevant Studies Selection

The studies selected in the previous phase were evaluated in detail to determine whether they met the established inclusion criteria. Aspects such as focus on LPWAN technologies, application in wearable devices, and fall detection were considered. Studies that met these criteria progressed to the next stage.

2.3. Phase 3: Study Inclusion (Inclusion)

During this phase, the preselected studies were subjected to a comprehensive and rigorous evaluation. Additional exclusion criteria were applied to ensure that only the most pertinent studies were incorporated into the systematic review. The assessment also included an evaluation of the methodologies employed by the studies.

This multi-phase approach to the search and selection of studies ensured the systematic review was thorough and comprehensive, resulting in the inclusion of only the most relevant and high-quality studies in the final analysis.

Data Analysis and Quality Analysis of Articles

For the detailed synthesis of the selected studies, we employed several analysis methods:

- Classification of LPWAN technologies: Identifying and categorizing the LPWAN technologies (LoRaWAN, Sigfox, NB-IoT) used in each study.
- Sensor analysis: Evaluating the types of sensors (accelerometers, gyroscopes, magnetometers) and their placement on the body.
- Detection algorithm performance: Analyzing the performance of detection algorithms, focusing on accuracy, sensitivity, and specificity.
- Characteristics of the evaluation samples: Review sample sizes and the number of falls evaluated.
- Energy consumption analysis: Comparing reported battery life and power consumption of the devices.
- Comparative analysis: Highlighting strengths, weaknesses, and key findings across studies.

As mentioned above, the article selection process was divided into three fundamental stages. Figure 1 presents a visual representation of the application of these stages in the study selection process.s

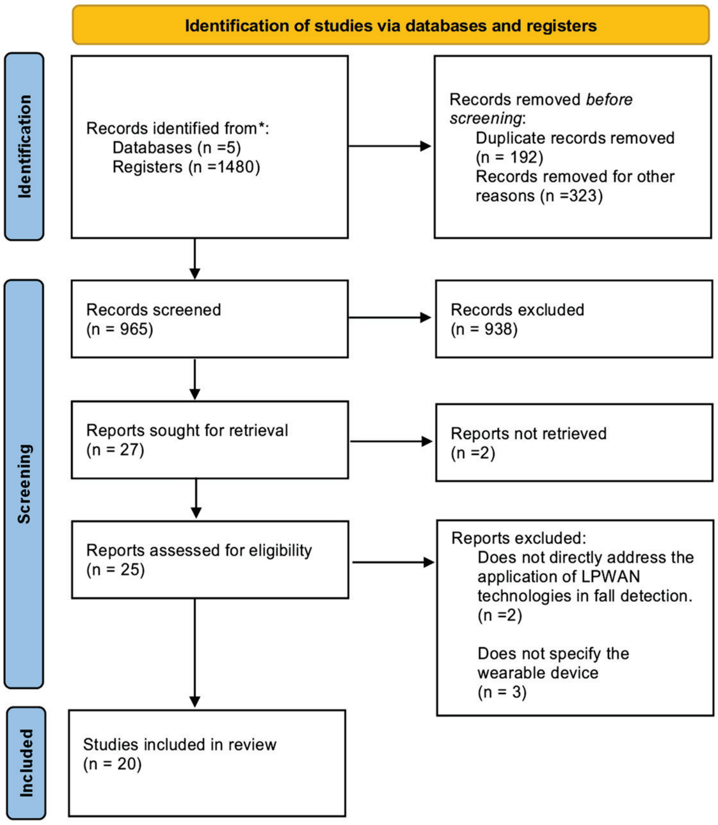

Identification of studies via databases and registers
Identification
Records identified from*:
Databases (n =5)
Registers (n =1480)
Records removed *before screening*:
Duplicate records removed (n = 192)
Records removed for other reasons (n =323)
Screening
Records screened (n = 965)
Records excluded (n = 938)
Reports sought for retrieval (n = 27)
Reports not retrieved (n =2)
Reports assessed for eligibility (n = 25)
Reports excluded:
Does not directly address the application of LPWAN technologies in fall detection. (n =2)
Does not specify the wearable device (n = 3)
Included
Studies included in review (n = 20)

Figure 1. Proposed methodology: results of the screening for each stage in the bibliographic search.

Subsequently, in the third inclusion phase, a detailed review of the 25 retrieved reports was conducted to determine their eligibility. Among these, five works were excluded: two of them for not specifying the used fall detection algorithm, as well as three papers that omitted basic information about the employed wearable device. As a definite result of this three-stage process, 20 studies were preselected for inclusion in this review [6–8,11–14,16,20,22,26–28,33,34,36,53,54,65,66].

3. Overview of the LPWAN Concept

LPWAN technologies are a set of wireless communication solutions which enable the connection of low-power and cost-effective devices over long distances. These technologies are ideal for IoT and M2M (Machine-to-machine) applications requiring wide-area connectivity and sporadic and small data transmission. Examples of such applications can be found in the fields of environmental monitoring, asset tracking, smart agriculture, smart cities, or healthcare, among others [50,67,68].

The main characteristic of LPWAN networks is the ability to connect devices that require low power, enabling them to operate with batteries or even directly powered by energy harvesting sources. This is possible because LPWAN networks employ efficient communication protocols, which allow devices to consume less energy when transmitting data [69]. Figure 2 illustrates a qualitative comparison among the different communication technologies as a function of the energy efficiency and terminal and connection costs. In this context, LPWAN stands out significantly.

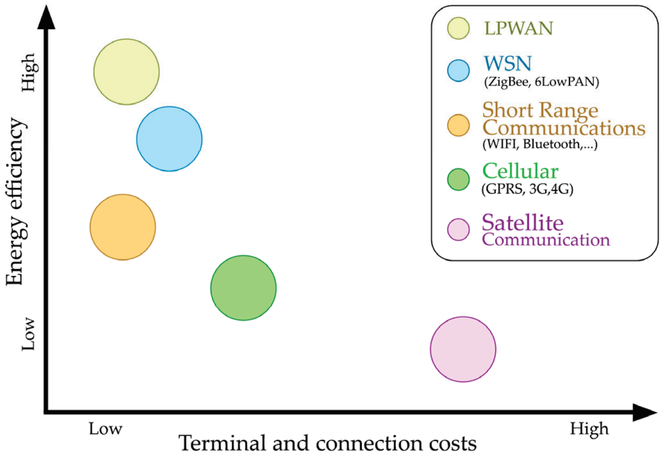


Figure 2. Comparison of energy efficiency with terminals and connection costs in various wireless communication technologies. Source: [51,69].

Apart from their low energy consumption and long-range capabilities, LPWAN networks offer extensive coverage and low implementation costs. Each access point or Gateway can support thousands of end nodes over several kilometers, thus favoring savings on network implementation costs. This makes them particularly suitable for applications in areas where traditional communication infrastructures are limited or costly to implement. IoT applications requiring the transmission of small data payloads across extensive distances while maintaining high energy efficiency are most ideally accommodated by LPWANs [70]. The former operates on licensed frequencies and is designed to function within frequency bands allocated for telecommunication services, leveraging existing infrastructures of mobile networks. In this regard, the 3GPP (3rd Generation Partnership Project) standards include several technologies that are meant for low-power, long-range, low-cost, and secure IoT applications, including NB-IoT (Narrowband- Internet-of-Things), LTE-M (Long Term Evolution for Machines), EC-GSM-IoT (Extended Coverage GSM for the Internet of Things), and 5G (5th Generation). Contrariwise, the latter uses unlicensed frequency bands. The so-called Industrial, Scientific, and Medical (ISM) radio bands are available for use without the need to pay a subscription fee. Examples of non-cellular LPWAN technologies include LoRaWAN and Sigfox. In the case of LoRaWAN, the technology can be employed to create public or private networks by incorporating new base stations that are free of license requirements [50,67].

In addition to their low energy consumption and long-range capacity, LPWAN networks are also characterized by their extended coverage and low implementation cost. LPWAN access points or gateways cost much less than any equivalent cellular base station.

Furthermore, an LPWAN gateway can support thousands of end nodes over several kilometers, which remarkably decreases the network's implementation and maintenance expenses. This makes LPWAN architectures particularly suitable for applications in areas where traditional communication infrastructures are limited or costly to deploy. Consequently, IoT applications requiring transmission of small data payloads across extensive distances (such as the medical alerts provided by FDSs) while maintaining high energy efficiency are most ideally accommodated by LPWANs [51].

Comparison of Most Popular LPWAN Technologies

The primary LPWAN technologies commonly employed in IoT applications comprise NB-IoT, LoRaWAN (Long Range Wide Area Network), and Sigfox (no relevant studies on FDSs using other LPWAN standards [47], such as Ingenu RPMA [Random Phase Multiple Access], Dash7 or Weightless LPWAN, was found). In contrast with Sigfox and LoRaWAN, NB-IoT is more focused on offering higher bandwidth and better coverage in urban areas with high node density at the cost of higher consumption and a more sophisticated protocol stack. Compared to Sigfox, LoRaWAN stands out for its greater capacity to transmit packets daily, long-lasting battery life, and lower operational costs. On the other hand, Sigfox provides a close, proprietary solution with a global network exclusively dedicated to IoT [47]. As in the case of traditional mobile telephony operators, Sigfox typically operates on a subscription-based model, according to which users can deploy their corresponding sensing nodes in an area covered by Sigfox gateways and pay for the connectivity services provided by the Sigfox network.

Table 1 provides a general comparison of these technologies, detailing aspects such as employed band, modulation, range, data rate, bidirectionality, energy consumption, and standardization.

Table 1. Summary of LPWAN Technologies: LoRaWAN, NB-IoT, and Sigfox [47,51,52].

Features	LoRaWAN	NB-IoT	Sigfox
Range	5 km (urban), 20 km (rural)	1 km (urban), 10 km (rural)	10 km (urban), 40 km (rural)
Bidirectional	Yes/half-duplex	Yes/half-duplex	Limited/half-duplex
Frequency	Unlicensed ISM band (915 MHz in North America, 433 MHz in Asia, 868 MHz in Europe)	Licensed LTE frequency	Unlicensed ISM band (868 MHz in Europe, 915 MHz in North America, 433 MHz in Asia)
Modulation	CSS	QPSK	BPSK
Maximum Bit Rate ("on-the-air")	50 kbps	250 kbps	100 bps
Standardization	LoRa Alliance	3GPP	Sigfox collaborates with ETSI on Sigfox-based network standardization
TX (Transmission) power consumption	28 mA	74–220 mA	10–50 mA
RX (Reception) power consumption	10.5 mA	46 mA	10 mA
Sleep mode power consumption	1 µA	3 µA	6 µA

Acronyms: CSS (Chirp Spread Spectrum), QPSK (Quadrature Phase Shift Keying), BPSK (Binary Phase Shift Keying), TX (Transmission), RX (Reception), ISM (Industrial, Scientific, and Medical radio bands), 3GPP (Third Generation Partnership Project), ETSI (European Telecommunications Standards Institute).

LoRaWAN and Sigfox are well-suited for IoT applications demanding extensive coverage and economical communication solutions. As aforementioned, these technologies operate in unlicensed frequency bands, which results in a significant reduction of operational costs. In addition, both architectures are based on extremely simple communication protocols with minimal handshakes, greatly simplifying their implementation in devices (such as wearables) with heavily limited hardware resources. As for the maximum data transfer rate, LoRaWAN offers up to 50 kbps, whereas Sigfox only provides up to 100 bps, rendering them suitable options for low-speed data applications (such as FDSs, which only require sending medical alerts with little data). Nevertheless, it is important to highlight that LoRaWAN may enable bidirectional or half-duplex communication, while Sigfox has more constraints regarding packet exchange rate between the end nodes and the gateways. On the other hand, the NB-IoT standard [47] offers superior performance with speeds of up to 250 kbps, although it uses licensed frequencies, which may require additional costs and regulatory challenges. In terms of energy consumption, LoRaWAN stands out for its efficiency, consuming 28 mA during transmission and 10.5 mA during reception. Conversely, NB-IoT transceivers, despite their superior performance, exhibit higher energy consumption, ranging from 74 to 220 mA during transmission and 46 mA during reception. Sigfox, on the other hand, offers moderate energy consumption, ranging from 10–50 mA during transmission and 10 mA during reception.

In summary, LoRaWAN and Sigfox are ideal for IoT applications that need wide coverage, cost-effective communication, and energy efficiency, albeit they present limitations in data speed. Conversely, NB-IoT offers superior performance at the expense of higher costs, regulatory challenges, and increased energy consumption.

4. Analysis of Selected Studies

This section presents a comprehensive analysis of the selected studies for this systematic review. Key aspects such as the LPWAN technologies used, the sensors employed, the placement of wearable devices on the body, and the fall detection algorithms implemented are examined.

4.1. Selected Studies

Table 2 summarizes the articles reviewed, showing author(s), title, type, location, LPWAN technology, application, and sensors employed in each article.

Table 2. Papers Included in the Systematic Review.

Ref.	Year	LPWAN Technology *	Sensor **
Escriba et al. [20]	2018	Sigfox	Accelerometer
Patel et al. [14]	2018	LoRaWAN	Accelerometer
Valach et al. [12]	2018	LoRaWAN	Accelerometer
Manatarinat et al. [22]	2019	NB-IoT	Accelerometer and gyroscope
Pena Queralta et al. [53]	2019	LoRaWAN	Accelerometer, gyroscope and magnetometer
Scheurer et al. [65]	2019	LoRaWAN	Accelerometer
Cai et al. [54]	2020	NB-IoT	Accelerometer and gyroscope
Chang et al. [8]	2020	LoRaWAN	Accelerometer, gyroscope and IR (Infrared)
Huynh et al. [6]	2020	LoRaWAN	Accelerometer, gyroscope and magnetometer
Lachtar et al. [11]	2020	LoRaWAN	Accelerometer, gyroscope and magnetometer
Zanaj et al. [7]	2020	LoRaWAN	Accelerometer, gyroscope and magnetometer
Liu et al. [28]	2021	NB-IoT	Accelerometer, gyroscope and magnetometer
Lousado et al. [13]	2021	LoRaWAN	Accelerometer
Fan et al. [27]	2022	NB-IoT	Accelerometer and gyroscope

Table 2. *Cont.*

Ref.	Year	LPWAN Technology *	Sensor **
Li et al. [16]	2022	LoRaWAN	Accelerometer and gyroscope
Qian et al. [26]	2022	NB-IoT	Accelerometer and gyroscope
Salah et al. [36]	2022	LoRaWAN	Accelerometer
Wong et al. [33]	2022	LoRaWAN	Accelerometer, gyroscope and magnetometer
Wu et al. [34]	2022	NB-IoT	Accelerometer and gyroscope
Pierleoni et al. [66]	2023	NB-IoT	Accelerometer, gyroscope and magnetometer

Notes: * LPWAN technologies include Sigfox, LoRaWAN (which also includes studies using LoRa), and NB-IoT, which are used for long-range, low-power wireless communication. ** Sensors mentioned include accelerometers (triaxial acceleration), gyroscopes (triaxial angular velocity), magnetometers (magnetic field), and IR sensors (infrared radiation). Some measurements (such as orientation), also used by certain FDSs, can be computed from the signals captured by the inertial sensors.

4.2. Selection of the LPWAN Technology

Among the various types of LPWAN used in biomedical telemonitoring for fall detection, LoRaWAN and NB-IoT emerge as the most prominent standards. LoRaWAN leads in preference with a total of 12 implementations, closely followed by NB-IoT with 7. Additionally, Sigfox is employed in a single study. These findings are visually summarized in Figure 3.

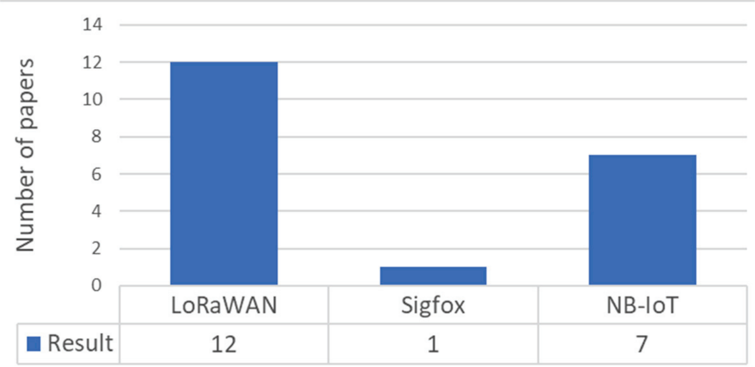


Figure 3. Distribution of LPWAN technologies used in studies of wearable fall detectors.

LoRaWAN and NB-IoT dominate the landscape of fall detection due to their balance of range and energy efficiency. However, the limited adoption of Sigfox, with only one study recorded, highlights its drawbacks. Its high latency and sensitivity to environmental conditions make it less suitable for critical applications such as fall detection [71].

4.3. Comparative Insights on LPWAN Technologies

Table 3 presents a comparative analysis of various studies on LPWAN technologies used in FDSs. This table highlights the main conclusions and limitations of each technology in the reviewed articles, providing a comprehensive overview of their application in health and safety monitoring for older adults.

The reviewed articles reveal key patterns regarding the conclusions and limitations of each LPWAN technology. The studies consistently indicate that Sigfox, while effective for low-power data transfer and suitable for fall detection, is limited by its low data rate and sensitivity to environmental conditions [20]. LoRaWAN stands out for its long-range communication capabilities and low power consumption. In fact, LoRaWAN is the most frequently used technology due to its operation in unlicensed frequency bands, which makes it a cost-effective option. However, the studies also agree that LoRaWAN can face network reliability issues, particularly in public networks and environments with physical obstacles [6–8,11–14,16,33,36,53,65]. NB-IoT, on the other hand, offers wide coverage, the ability to handle multiple connections, and low power consumption, making it suitable for health monitoring applications. The primary limitations of NB-IoT, according to the studies, relate to network coverage variability and potential data loss due to environmental interference [22,26–28,31,34,54]. Table 4 provides a summary of the performance of LPWAN technologies in fall detection across various environmental conditions (e.g., rural areas or environments with physical obstacles like cities with high building density) and mobility scenarios.

Table 3. Comparative Insights on LPWAN Technologies in Fall Detection Systems.

Ref.	Contribution	Conclusions on LPWAN Technology	LPWAN Limitations	Outcome
Escriba et al. [20]	Development of a smart wearable active patch for elderly health prevention, utilizing Sigfox for fall detection and GPS tracking.	Sigfox is effective for low-power data transfer and suitable for fall detection and geolocation.	A low data rate limits the amount of information transmitted; communication is affected by environmental conditions and device location.	The system effectively used Sigfox for low-power data transfer, with the patch maintaining communication reliability in 67.92% of tested positions, validating its capability for fall detection and location tracking.
Patel et al. [14]	Development of a LoRaWAN-based system for real-time monitoring of vital signs and fall detection, with alerts sent via LINE application.	LoRaWAN can extract data from weak signals in noisy environments, which is useful for delivering critical medical data.	No specific limitations are commented.	The system demonstrated high performance with LoRaWAN, achieving 100% sensitivity, 96.93% accuracy, 94.25% specificity, and 91.38% predictability, ensuring reliable transmission of fall alerts.
Valach et al. [12]	Research on the feasibility of using LoRaWAN in healthcare IoT devices, focusing on energy optimization and transmission reliability.	LoRaWAN is suitable for long-distance data transmission with low power consumption and is useful for patient location and monitoring vital signs.	The LoRaWAN connection was unreliable when the end node was on the ground during tests.	LoRaWAN showed reliability issues in fall detection when the end node is near the ground; using Arduino Pro Mini is suggested to reduce energy consumption.
Manatarinat et al. [22]	Development of an NB-IoT-based system for elderly healthcare, enabling automatic fall detection and alerts via the LINE application.	NB-IoT is suitable for health monitoring applications, providing efficient low-power communication with wide coverage.	No specific limitations.	Reliable fall alert communication and patient location using NB-IoT, ensuring immediate medical response. Achieved low latency and dependable alert delivery, with reliance on an NB-IoT operator for network functionality.
Pena Queralta et al. [53]	Developed and technically validated the AIDE-MOI fall detection algorithm using real-life data from older adults, utilizing LoRaWAN for effective long-range, low-power communication.	LoRaWAN is a promising option to overcome the limitations of traditional network infrastructures in remote health settings, providing long-distance, low-power data transmission.	LoRaWAN cannot support high data rate applications due to limited transmission bandwidth.	The AIDE-MOI system demonstrated significant improvement with LoRaWAN, achieving a sensitivity of 80% and a specificity of 99.9978%, ensuring reliable and efficient fall detection and data transmission in real-life environments.

Table 3. *Cont.*

Ref.	Contribution	Conclusions on LPWAN Technology	LPWAN Limitations	Outcome
Scheurer et al. [65]	Developed and technically validated the AIDE-MOI fall detection algorithm using real-life data from older adults, utilizing LoRaWAN for effective long-range, low-power communication.	LoRaWAN is used for long-range, low-power communication, allowing data transmission over several kilometers.	No specific limitations are commented.	The AIDE-MOI system using LoRaWAN achieved 80% sensitivity and 99.9978% specificity, ensuring reliable fall detection and efficient data transmission.
Cai et al. [54]	Developed a GBDT-based fall detection system using comprehensive data from posture sensors and human skeleton extraction, utilizing NB-IoT for effective data transmission to a cloud server for analysis.	NB-IoT provides wide coverage, multiple connections, low speed, and low power consumption.	No specific limitations are commented.	The system, using NB-IoT, achieved an I2 score of 0.878 on a fused dataset and 95% accuracy, demonstrating reliable and low-latency data communication.
Chang et al. [8]	This study proposes an intelligent assistive system for visually impaired individuals using smart glasses and a smart cane connected via BLE, with LoRaWAN for reliable fall detection and alert transmission.	LoRaWAN-based intelligent assistive system for aerial obstacle avoidance and fall detection for visually impaired people shows high accuracy and long-range communication.	Dependence on public LoRaWAN network coverage, which can vary based on geographical location and network infrastructure availability.	The system effectively uses LoRaWAN to transmit fall alerts, achieving long-distance coverage and low latency. The integration of both devices reduced false alarms, achieving an overall accuracy rate of 94.56%.
Huynh et al. [6]	Develops a LoRaWAN-based system for activity assessment and fall detection at home, using low-cost inertial sensors and a cloud platform for data analysis and emergency notifications.	LoRaWAN technology advantages of long-range capabilities and low power consumption. Wearable devices can operate 2–10 km from a LoRaWAN base station with an optimized battery life of one week between charges.	No specific limitations.	The system demonstrated a high specificity of 100% in fall detection during approximately 200 h of normal activities, proving the effectiveness of using LoRaWAN for reliable data transmission and emergency alerts.
Lachtar et al. [11]	Proposes a monitoring architecture using LoRaWAN and MQTT * for fall detection in elderly people within a smart city environment.	LoRaWAN technology is used for long-range, low-power communication in a smart city environment, effectively robust and efficient for elderly monitoring.	Reliance on public LoRaWAN network coverage, which can fluctuate based on geographic location and available infrastructure.	The system demonstrated efficient long-range communication with LoRaWAN, covering an average area of 6 km^2 with minimal packet loss, making it suitable for smart cities.

Table 3. *Cont.*

Ref.	Contribution	Conclusions on LPWAN Technology	LPWAN Limitations	Outcome
Zanaj et al. [7]	Proposes a wearable fall detection system integrated into shoes, using LoRaWAN for long-range, low-power communication and MQTT for alert notifications.	LoRaWAN is an attractive and promising technology for health and wellness monitoring, enabling long-range communication with low battery consumption. It has been demonstrated to be effective for long-term use and reliable coverage with a single gateway.	Packet loss can occur due to obstacles affecting the connection between the end node and gateway.	Demonstrated efficient data transmission with a 95% success rate in various environments. The system operates for approximately 23 h on a 500 mAh battery, proving LoRaWAN's viability for fall detection.
Liu et al. [28]	Proposes a wearable fall detection system utilizing a 1D CNN deep learning model and NB-IoT communication to send alerts and GPS data to the cloud.	NB-IoT is key for data transmission and alerts in a successful fall detection system based on 1D CNN.	Lack of NB-IoT service coverage can cause interference and data loss due to variability in signal quality across different geographical environments and network conditions.	The system demonstrated a fall detection accuracy of 98.85%, with a sensitivity of 98.86% and specificity of 99.84%, highlighting the effectiveness of using NB-IoT for reliable data transmission and fall alerts.
Lousado et al. [13]	Proposes a cost-effective, energy-efficient monitoring system for elderly individuals using LoRaWAN and The Things Network (TTN) for long-range communication and data processing.	LoRaWAN offers an effective and low-cost solution for monitoring the health and home conditions of elderly people, especially in remote areas with limited mobile network coverage.	Variability in LoRaWAN public network coverage, which may be insufficient in certain geographical areas due to uneven infrastructure availability.	Demonstrated high predictive accuracy (99.73%) for detecting falls and other anomalies, showcasing the viability of using LoRaWAN for reliable and continuous monitoring of elderly individuals' health and movement in areas with limited mobile network coverage.
Fan et al. [27]	Design and development of a low-power wearable fall detection device for the elderly using NB-IoT technology, capable of remote positioning, tracking, fall detection, and one-touch emergency calls.	NB-IoT communication is of paramount importance for the successful implementation of this fall detection device for the elderly, providing an effective solution for monitoring elderly safety and reducing caregiving pressure on family members.	No specific limitations.	The device demonstrated effective fall detection in different directions during simulated tests. GPS positioning accuracy showed that 94% of the time, the positioning error was within 20 m, validating the device's capability for precise location tracking using NB-IoT technology.

Table 3. *Cont.*

Ref.	Contribution	Conclusions on LPWAN Technology	LPWAN Limitations	Outcome
Li et al. [16]	Design and development of an emergency communication system for elderly people living alone, using LoRaWAN technology for long-range, low-power communication, along with a fall detection algorithm based on the channel hopping strategy.	LoRaWAN, combined with channel hopping, enhances the communication quality and efficiency of emergency systems monitoring the elderly, facilitating remote monitoring.	Communication loss rate of 2% at distances up to 1800 m due to electronic interference, physical obstacles, and simulated environmental conditions.	The device achieved a fall detection rate of over 85% in simulated tests. The average communication latency was 9.5 s, and the effective communication distance reached 1800 m, validating the efficiency of LoRaWAN for emergency communication systems.
Qian et al. [26]	Design and development of a wearable fall detection system combining MEMS * sensors and NB-IoT, capable of integrating with public health systems for real-time monitoring and timely rescue.	NB-IoT, combined with MEMS sensors and a multilevel threshold algorithm, provides an efficient, low-power solution for real-time fall detection that is suitable for integration with public communication networks.	Signals are sensitive to the environment, and disturbances and noises can cause system faults.	The system demonstrated a fall detection accuracy of 94.88%, sensitivity of 95.25%, and specificity of 94.5% in experimental tests. This validates the effectiveness of the NB-IoT-based system for real-time fall detection and location tracking in elderly individuals.
Salah et al. [36]	Design of a fall detection system based on Edge artificial intelligence, combining a microcontroller and LoRaWAN communication.	LoRaWAN technology, used with edge AI, enables long-range, low-power communication, making it suitable for real-time fall detection systems with high accuracy and low latency.	Communication range decreases when moving from a direct line of sight to a non-line of sight due to obstacle interference, affecting system performance.	The device achieved a 95.55% accuracy in fall detection using a convolutional neural network (CNN). Local inference reduced latency and improved energy efficiency, with a battery life exceeding 53 h and a communication range of up to 180 m in line-of-sight using LoRaWAN.
Wong et al. [33]	Design of a fall detection system based on LoRaWAN communication, optimizing fall detection accuracy while reducing costs by eliminating the need for mobile phones.	LoRaWAN technology is highlighted for its low power consumption, reduced operational costs, and flexible data transmission rate, making it the best option for long-distance communication in emergency situations.	Communication can be affected by obstacles, electrical and magnetic interferences, leading to system instability and data transmission issues.	The system accurately detected falls in various scenarios with set thresholds, demonstrating reliable performance in both indoor and outdoor environments. The LoRaWAN protocol ensured stable signal transmission up to 318 m with minor obstructions, proving its effectiveness for long-range communication.

Table 3. *Cont.*

Ref.	Contribution	Conclusions on LPWAN Technology	LPWAN Limitations	Outcome
Wu et al. [34]	Design and development of a fall detection module based on NB-IoT technology and MEMS systems, capable of collecting acceleration and angle data and transmitting it for precise fall analysis.	NB-IoT, combined with MEMS sensors and a GRU-based algorithm, provides an effective, low-power solution for real-time fall detection, suitable for long-distance data transmission.	No specific limitations.	The module demonstrated an accuracy of 90.1% using the threshold method and 92.9% with GRU. Data was successfully transmitted via NB-IoT, enabling alerts to be sent to family members and rescue centers in the event of a detected fall.
Pierleoni et al. [66]	Design and development of a comprehensive architecture for Ambient Assisted Living (AAL) scenarios, incorporating a cross-protocol proxy for seamless communication between different IoT protocols and a wireless wearable fall detection device based on LPWAN technologies such as NB-IoT.	NB-IoT, used in combination with CoAP * and MQTT, provides an efficient and reliable communication solution for wearable fall detection devices, ensuring low latency and high throughput.	Latency issues due to network congestion and packet loss requiring retransmissions, with latency averaging 0.4 s and occasional peaks up to 10 s.	The system demonstrated low latency (approx. 0.4 s) and high reliability in transmitting fall detection alerts using NB-IoT and MQTT. The proxy improved interoperability, and the wearable device showed a high success rate in fall detection and alerting in an AAL environment. The packet loss rate was slightly above 0.1%, with recovery through retransmissions.

* Acronyms: CoAP (Constrained Application Protocol), MQTT (Message Queuing Telemetry Transport), MEMS es Micro-Electro-Mechanical Systems.

Table 4. Performance of LPWAN Technologies in Fall Detection According to Environmental Conditions and Mobility Scenarios.

Technology	Environmental Conditions	User Mobility Scenarios	Reference
Sigfox	Sensitive to environmental conditions; ideal for rural environments	Suitable for low mobility applications where data transmission is not continuous and can tolerate delays.	[20,71]
LoRaWAN	May face reliability issues in environments with physical obstacles; offers good interference resistance	Ideal for urban and rural applications with limited to moderate mobility due to its combination of long-range and low-power consumption.	[6–8,11–14,16,33,36,53,65,71]
NB-IoT	Can be affected by network coverage variability and environmental interference	Best for urban environments where support for mobile devices and high reliability in data transmission are required, making it ideal for critical applications such as fall detection.	[22,26–28,34,54,66,71]

There is no perfect option for fall detection applications requiring immediate and reliable message delivery. Sigfox is suitable for situations where latency is not critical, and energy efficiency is prioritized. NB-IoT, with its low latency and high reliability, is ideal for critical applications in urban environments. LoRaWAN offers a good balance with its combination of long-range, low-power consumption, and better average latency, making it versatile for various environments. It is crucial to consider the specific needs of the application and the conditions of the environment when choosing the appropriate technology for fall detection.

4.4. Employed Sensors

Within the field of biomedical telemonitoring for fall detection, Inertial Measurement Unit (IMU) sensors are predominantly used to feed detection algorithms that identify falls. These IMU sensors include accelerometers, gyroscopes, and magnetometers, which are employed in various combinations across the reviewed studies (see Figure 4). The most prevalent combination is the use of accelerometer, gyroscope, and magnetometer, employed in 7 studies (35%). The combination of accelerometer and gyroscope is used in 6 studies (30%), while another six studies (30%) rely solely on the accelerometer for detection. Finally, in one case (the work by Chang et al. [8]), the use of accelerometer and gyroscope is complemented by an infrared sensor located in a second wearable (a pair of glasses).

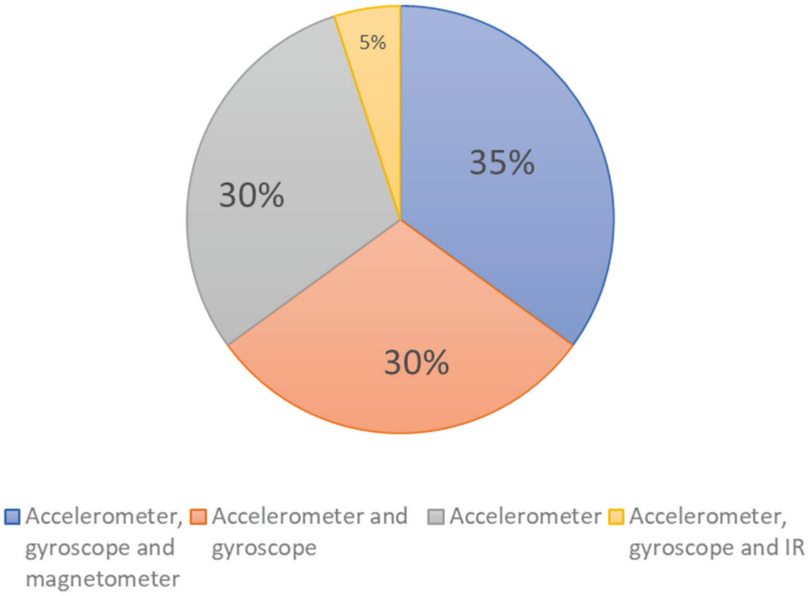


Figure 4. Types of sensors used in the selected articles.

Although IMU sensors, such as accelerometers, gyroscopes, and magnetometers, are widely used in wearable devices with LPWAN technology for fall detection, the system's lack of integration of additional sensors, such as a barometer, limits accuracy. As highlighted by Pierleoni et al. [31], the inclusion of a barometer could provide valuable information on altitude changes during a fall, improving detection reliability. The almost exclusive reliance on inertial sensors, with only occasional use of infrared sensors, suggests a missed opportunity to address more complex scenarios and enhance fall detection accuracy.

4.5. Location of the Wearable Device

The location of the wearable device is a crucial factor for accurate fall detection in LPWAN-based systems. Özdemir [39] study highlights that sensor placement can significantly influence readings and the system's ability to differentiate between normal movements and fall events. According to the study, average accuracies based on the sensor's location on the body vary considerably. The waist region is identified as the best position, with an average accuracy of 98.42% for six machine learning algorithms. The thigh sensor follows with an accuracy of 97.89%, and the ankle with 97.00%. On the other hand, the head, with an accuracy of 96.61%, and the chest, with 96.50%, also show good results. Wrist sensors have the lowest performance, with an average accuracy of 94.92%. These findings underscore the importance of carefully selecting the device's location to optimize fall detection. Table 5 presents the selected locations for wearable device placement in the LPWAN-based FDS under analysis.

Table 5. Location of the Wearable Device.

Reference	Device Position
Huynh et al. [6]	Waist or wrist
Zanaj et al. [7]	Foot (shoes)
Lachtar et al. [11]	Cane
Lousado et al. [13]	Backpack
Escriba et al. [20]	Back
Qian et al. [26]	Wrist
Fan et al. [27]	Wrist
Wu et al. [34]	Waist
Liu et al. [28]	Waist
Wong et al. [33]	Chest
Scheurer et al. [65]	Back, abdomen, or chest
Pierleoni et al. [66]	Ankle or shoe

Note: "Cane" refers to a walking stick used as an assistive device.

The studies presented in Table 5 highlight the diversity of locations used for wearable devices in fall detection. The selection of the device's location should consider both detection accuracy and user comfort and acceptance. Additionally, it is important to consider that different activities and contexts can influence the effectiveness of the sensor's location, suggesting the need to customize the device placement according to the user's individual needs. Comparing these results with Özdemir [39] findings, the accuracy on the wrist reported by Qian et al. [26] (94.88%) approximately corresponds. The accuracy on the waist reported by Liu et al. [28] (98.85%) also aligns significantly. In summary, the waist can be considered a reasonable location for a wearable fall detection device. This position not only offers high accuracy but is also close to the body's center of gravity, allowing for more precise detection of changes in body movement while minimizing interference with the user's daily activities. Although wrist devices are more comfortable and socially acceptable, similar to a conventional watch, specific arm movements can affect fall detection accuracy. On the other hand, placing the sensor on the back, abdomen, or chest, as indicated by Scheurer et al. [65], does not significantly affect fall detection effectiveness but may hamper ergonomics. Other locations, such as inside footwear or on a cane, aim to maximize comfort but at the cost of a lower accuracy of the detector.

4.6. Employed Detection Algorithms

The accuracy of fall detection in biomedical telemonitoring settings greatly relies on the detection algorithm, the "intelligence core" in charge of identifying anomalous mobility patterns from the gathered signals. However, evaluating their performance and comparing the selected studies according to a common evaluation framework becomes complex due to the significant differences in the test conditions (methodology, participants, use of fake falls, etc.) considered by the authors to validate their proposal.

In any case, a summary of outcomes from the relevant studies is presented in Table 6. In the second column, the table indicates the type of algorithm used to identify the falls.

Table 6. Summary of Performance Metrics of Fall Detection Algorithms.

Reference	Algorithm Type *	Accuracy	Sensitivity	Specificity	Sensor	Sample Size (Number of Participants)	No. of Evaluated Falls
Huynh et al. [6]	Thresholding policies	n.i.	96.3%	96.2%	Accelerometer, gyroscope and magnetometer	10	n.i.
Chang et al. [8]		98.3%	n.i.	n.i.	Accelerometer, gyroscope and IR (Infrared)	3	150
Li et al. [16]		85%	n.i.	n.i.	Accelerometer and gyroscope	6	500
Wu et al. [34]		90.1%	n.i.	n.i.	Accelerometer and gyroscope	n.i.	n.i.
Qian et al. [26]		94.88%	95.25%	94.5%	Accelerometer and gyroscope	20	400
Salah et al. [36]	K-NN (15 neighbors)	78.64%	81.07%	76.57%	Accelerometer	24	1798
Salah et al. [36]	K-NN (5 neighbors)	79.11%	80.06%	78.21%	Accelerometer	24	1798
Salah et al. [36]	CNN	95.55%	95.1%	94.86	Accelerometer	24	1798
Liu et al. [28]		98.85%	98.86%	99.84%	Accelerometer, gyroscope and magnetometer	35	1798 and 288
Salah et al. [36]	LSTM	96.78%	97.87%	95.21%	Accelerometer	24	1798
Pena Queralta et al. [53]		91.90%	95.3%	n.i.	Accelerometer, gyroscope and magnetometer	54	647
Salah et al. [36]	SVM	82.27%	87.21%	78.48%	Accelerometer	24	1798
Wu et al. [34]	GRU	92.9%	n.i.	n.i.	Accelerometer and gyroscope	n.i.	n.i.
Cai et al. [54]	GBDT (Acceleration Dataset)	89.2%	n.i.	n.i.	Accelerometer and gyroscope	10	n.i.

* Acronyms: CNN (Convolutional Neural Network), GBDT (Gradient Boosting Decision Tree), GRU (Gate Recurrent Unit), k-NN (K-Nearest Neighbors), LSTM (Long Short-Term Memory), n.i. (not indicated by the authors), SVM (Support Vector Machine).

The table shows that there are examples of the three families of algorithms typically considered to address the problem of fall detection in wearable systems: from simple thresholding strategies (which assume that a fall occurs when certain measurements—or combinations of measurements—received from the sensors exceed certain critical values) to machine learning algorithms (such as k-Nearest Neighbor, Support Vector Machine, Decision Trees, etc.) and deep learning models (such as convolutional or recurrent neural networks) [72].

To characterize the effectiveness of the classifier, the table also includes some basic performance metrics reported by the authors: accuracy (percentage of total movements that are correctly identified), sensitivity (ratio of falls properly detected), and specificity (ratio of non-falls or ADLs- Activities of Daily Living- that are adequately interpreted).

Table 6 illustrates that, for the prototypes under study, the best results are those obtained with CNNs. Liu et al. [28] conducted a study using CNN in combination with measurements provided by accelerometers, gyroscopes, and a magnetometer alongside NB-IoT technology, achieving an accuracy of 98.85%, a sensitivity of 98.86%, and a specificity of 99.84%. They utilized the SisFall [73] and Mobifall [74] datasets for training and validating their models, ensuring a comprehensive evaluation of fall detection accuracy. In a similar approach, Salah et al. [36] also implemented a detector using CNN and LoRaWAN technology but only leveraging the data from an accelerometer. The prototype achieved an accuracy of 95.55%, a sensitivity of 95.1%, and a specificity of 94.86%. Another algorithm of significant relevance is the LSTM. According to Salah et al. [36], this algorithm reached an accuracy of 96.78%, a sensitivity of 97.87%, and a specificity of 95.21% when employing an accelerometer sensor. It is worth noting that these models were also trained and validated with the SisFall dataset [73,75], providing a robust framework for evaluating the performance. These results emphasize the high potential of these algorithms for fall detection within this context.

At this point, it should be noted that many works on FDSs with wearable devices do not analyze in detail (or directly ignore) certain operational aspects of great relevance for the practical implementation of these architectures. Among these aspects, one can mention the consumption of battery, memory, and computational resources that the detection algorithms themselves demand in the wearables, which usually have significant hardware limitations.

In this context, there is a consensus on the idea that simpler algorithms, such as those based on thresholds or some machine learning solutions [76,77], require fewer resources than those based on deep learning. In some works, such as Lampoltshammer et al. [78], it is proposed to implement a very simple mechanism in the wearable (e.g., based on a simple threshold for the acceleration module), so that when a fall is suspected, the inertial signals (collected over a certain period) are sent to an external point with higher computational power for a more detailed analysis based on more complex algorithms. However, this approach may require frequent transmissions from the wearable, which can be counterproductive from an energy perspective (and impractical if LPWAN solutions with limited available bandwidth are used). Few works, such as the RNN presented by Musci et al. [79] or the Tiny CNN described by Yu et al. [41], consider a careful and optimized design of the deep learning algorithm for its implementation on low-power embedded devices. Regarding memory consumption, the recent work by Fernandez-Bermejo et al. [80] highlights the importance of minimizing the parsimony of the models used (i.e., the number of parameters that they require) to facilitate their implementability.

As for the sensor, the accelerometer is the least demanding in terms of energy and the most used in the literature for motor rehabilitation [81], so it is not surprising that most proposals are based on measuring acceleration. In this sense, the sampling rate can affect consumption. A sampling rate above 20 Hz increases recognition accuracy in HAR systems by just 1% while this quality metric stabilizes beyond 50 Hz [82]. Therefore, a sampling rate of 50 Hz is considered more than sufficient [83]. In some work, it has been proposed to adaptively modify the frequency to the user’s activity. Hence, during low activity situations, the frequency could be reduced to a minimum to moderate consumption and augmented when greater mobility is detected. However, this scheme, apart from increasing the complexity of the detector, may cause false negatives if the fall occurs from a stationary or low-movement position.

On the other hand, deep learning can benefit from the ability to work with raw signals in the time domain without needing to perform sophisticated operations to extract features. For example, Casamassima et al. [81] analyze the consumption of body area networks and conclude that the computation of features based on the FFT of accelerometry signals requires many more MCU clock cycles than those directly obtained from the time series.

Activating the GPS to report the position of the faller can also be a significant cause of battery drain in wearables. Gharghan et al. [84] propose activating the GPS only when a fall is suspected.

In any case, regarding the specific literature on fall detectors using LPWAN technologies, it should be noted that these operational aspects are practically overlooked, which is paradoxical since one of the main objectives of using LPWAN communications is to minimize consumption and hardware complexity.

4.7. Energy Consumption

One of the crucial aspects to consider when designing a wearable fall detector is battery life. However, battery lifetime is only investigated in some works of the related literature (synopsized in Table 7). As it can be appreciated, results indicate that battery lifespan can significantly vary based on the algorithm used, LPWAN technology, and sensors employed. None of the selected works explicitly study which of these factors is the main cause of battery drain. Only the work by Salah et al. in [36] showed that implementing the CNN model on a LoRaWAN sensing unit results in an improvement of battery life: 53 h in contrast with the 38 h of lifetime attained when a similar prototype combines BLE and a CNN-based detector. This highlights the potential savings in battery consumption that can be achieved with LPWAN technology when compared to other wireless transmission standards traditionally used in FDSs. In any case, when examining the results in the table, a certain relationship between the type of detection algorithm used and battery life technology becomes evident. Simpler detection algorithms, such as those considered in the prototypes described by Huyn et al. [6], tend to contribute to longer battery life, whereas more complex algorithms like CNN [36], may require more resources and thus consume more energy.

Table 7. Energy Consumption in Fall Detectors using LPWAN Technology.

Reference	Battery Life *	Battery Capacity **	Algorithm Type	LPWAN Technology	Employed Sensor
Huynh et al. [6]	1 week–1 month	Not specified	Thresholding	LoRaWAN	Accelerometer, gyroscope and magnetometer
Zanaj et al. [7]	23 h (500 mA/h), 36 h (800 mA/h)	500 mAh 800 mAh	Thresholding	LoRaWAN	Accelerometer, gyroscope and magnetometer
Salah et al. [36]	More than 53 h	2000 mAh	CNN	LoRaWAN	Accelerometer
Escriba et al. [20]	3 days (low-power mode)–13 h (GPS tracking)	30 mAh	Not indicated	Sigfox	Accelerometer

Notes: * Battery life varies depending on the operational mode; for example, Escriba et al. reported different durations for. low-power mode versus GPS tracking mode. ** Battery capacity is measured in milliampere-hours (mAh).

In any case, it is noteworthy that a significant number of studies employing LPWAN technology with fall detectors do not analyze battery life or consumption of the prototype through field tests. This becomes particularly striking when considering that one of the major benefits of these long-range wireless communication standards is energy efficiency.

4.8. LPWAN Transceivers

A comprehensive understanding of the transceivers used in FDSs with LPWAN technology is of great interest for current research, as it enables the identification of trends in the design of these devices, especially considering that the radio transceiver is one of the components with the highest energy consumption [85]. Reducing the energy of wireless communication is crucial for improving the energy efficiency of wearable devices. Table 8 compiles the transceivers used by the reviewed works, detailing the associated LPWAN technology and their references.

Table 8. LPWAN Transceivers.

Reference	Transceiver	LPWAN Technology *
Zanaj et al. [7]	SX1257	LoRaWAN
Salah et al. [36]	RFM95W	LoRaWAN
Lachtar et al. [11]	RFM95/96/97/98(W)	LoRaWAN
Valach et al. [12]	RFM95W	LoRaWAN
Lousado et al. [13]	SX1276	LoRaWAN
Qian et al. [26]	BC-95	NB-IoT
Fan et al. [27]	M5310A	NB-IoT
Wong et al. [33]	SX1278 RA-02	LoRaWAN
Pierleoni et al. [66]	nRF9160	NB-IoT

Notes: * LPWAN technologies include Sigfox, LoRaWAN, and NB-IoT, which are used for long-range, low-power wireless communication.

The review identified a clear preference for LoRaWAN technology in FDSs, due to its balance between range, energy efficiency, and the use of unlicensed spectrum bands [7,11–13,33,36]. Transceivers such as the SX1257, RFM95W, and SX1276, with current consumption of 58–85 mA and 120 mA respectively [86–88] for the transmission mode, demonstrate their suitability for battery-dependent long-range wearable devices. Additionally, the presence of NB-IoT technologies is noted, albeit to a lesser extent, represented by transceivers like the BC-95, M5310A, and nRF9160 [89–91], showing transmission power consumption of up to 220 mA. NB-IoT, which operates in licensed bands, may involve higher operational costs due to this requirement, despite offering more robust coverage. The nRF9160 stands out for its efficiency in idle mode, with consumption as low as 2.7 µA in PSM (Power Saving Mode) and 18–37 µA in eDRX (Extended Discontinuous Reception) [91]. Although NB-IoT provides greater coverage, its higher energy consumption during transmission may limit the battery life of wearable devices, underscoring the need of optimizing energy efficiency for wearable applications.

5. Discussion

The results of this systematic review provide insight into the application of LPWAN technologies in wearable devices for fall detection. This field is growing significantly, and our analysis highlights current and future trends.

As a first observation, there is a notable concentration of research in China (6 studies) [16,26–28,34,54] and Taiwan (2 studies) [8,33], with other countries contributing one manuscript each [6,7,11–14,20,22,36,53,65,66]. This reflects the leadership of these regions in the research and use of low cost transmission technologies in IoT appliances. Although the review initially considered articles from 2010 due to the conception of Sigfox, it focused on studies from 2018 to 2023 due to the increase in relevant research during this period. The results do not show a clear temporal pattern, although an increase was noted in 2020 and 2022. The review shows a balance between journal and conference papers, with 10 references of each type, indicating attention to both established and recent research. The focus from 2018 onwards was due to the emergence of articles combining LPWAN with fall detection. Additionally, this review covered various LPWAN technologies, including LoRaWAN, Sigfox, and NB-IoT, for a comprehensive assessment [6–8,11–14,16,20,22,26–28,33,34,36,53,54,65,66].

The choice of LPWAN technologies is regarded by the authors of these papers as a acritical factor in implementing effective and efficient FDSs. Our findings support the predominance of LoRaWAN and NB-IoT in this field, a trend which has remained consistent and is sup-ported by previous research (Mekki et al. [47]). Nonetheless, the analysis conducted delves deeper into comparing these technologies, particularly regarding their energy efficiency and transmission speed. Particularly, LoRaWAN stands out in its energy efficiency, surpassing NB-IoT, a highly relevant aspect due to its potential impact on wearable devices' battery life [51]. This characteristic becomes critically important to ensure the continuous and effective operation of these devices. On the other hand, NB-IoT exhibits notable advantages in data transmission speed according to Sinha et al. [92], which could be relevant in applications requiring fast and reliable transmission of biomedical information.

As it refers to energy consumption, the main problem is that in most works on FDS based on LPWAN (as those using other standards), battery consumption is either not assessed or is analyzed in a very superficial way. Just one of the reviewed works (Salah et al. in [36]) provides evidence of the potential advantages of replacing a short-range, low-power standard with LPWAN technologies.

Regarding the sensors employed in wearable devices for fall detection, our findings align with prior research by Bet et al. [93], emphasizing the significance of Inertial Measurement Unit (IMU) sensors in this domain. The analysis conducted has identified a diversification in the combinations of these sensors used, suggesting that the selection of sensors might depend on specific contexts and precision requirements. Moreover, studies [6,7,13,20,26–28,33,34,65,66] have observed various locations for fall detection devices, ranging from the waist to the wrist, including the feet and different areas of the torso. Overall, the waist and chest are the most effective locations, offering higher accuracy. However, other studies note that placing the device in other areas of the torso does not significantly affect its detection capability, thus, providing more flexibility in placement [6,65]. Currently, researchers are exploring alternative positions such as the foot 'within shoes' or an auxiliary element like a walking stick, aiming to maximize comfort without compromising accuracy [7,11,66].

The analysis of fall detection algorithms has revealed consistency with prior research, such as the study by Liu et a. [28], emphasizing the advantages of employing neural network-based approaches, including CNN, and Salah et al. [36] using an LSTM architecture. However, our analysis has provided a deeper level of contextualization by considering the performance of these algorithms concerning LPWAN technologies and the sensors used. It is worth mentioning that while these approaches exhibit high potential in terms of accuracy, they may also be computationally resource-intensive, posing challenges in energy efficiency for resource-limited wearable devices [36].

In the study by Valach et al. [12], the reliability of fall detection was questioned when devices were placed in low-visibility areas like on the floor or under tables. During testing, the LoRaWAN base stations of the TTN (The Things Network) network failed to receive transmissions from these positions, which could jeopardize the detection of many falls. This suggests that further investigation into the shielding effects of the device's board and its impact on signal transmission is necessary to enhance reliability in critical scenarios. Nonetheless, compared to BLE and WiFi, LPWAN technology like LoRa offers better signal penetration through physical obstacles due to its sub-GHz frequencies, resulting in better coverage in challenging environments [94]. The sub-GHz band provides superior signal quality over wider areas and longer distances, experiencing less attenuation and multipath fading caused by obstacles and dense surfaces like concrete walls [95]. The actual reception ratio of LPWAN-transmitted alarms generated by wearable fall detectors should be carefully investigated in real application scenarios.

A critical aspect of wearable device design is user comfort, social acceptability, and ease of use, elements that were not considered by the authors reviewed [6–8,11–14,16,20,22,26–28,33,34,36,53,54,65,66], who focused exclusively on the technical aspect. According to Özdemir [39], the current elderly population avoids technology due to discomfort with wearable devices. This highlights the importance of user-centered designs to enhance their adoption. Studies like that of Gemperle et al. [96] have identified less intrusive areas of the body for wearable placement, such as the upper back of the arm, the waist area, and the back of the torso. Social acceptability is also crucial; Pasher et al. [97] found that the acceptance of wearable devices significantly depends on their ergonomics and how users perceive their comfort and aesthetics, often outweighing privacy concerns. Finally, Thilo et al. [98] emphasize that involving end-users in the design and development stage of a wearable prototype is well-received by older adults and allows for the exploration of their needs and preferences, indicating that older adults' perceptions of activity, independence, and familiarity should be considered to ensure the devices are functional, comfortable, and usable.

6. Criticisms and Limitations

- From the study conducted, it can be inferred that the application of LPWAN technologies to wearable FDSs is still in an embryonic state, having neither exploited nor systematically analyzed its main benefits. Therefore, it is necessary to remark on the following weaknesses in the reviewed literature:
- Limited coverage: Signal reliability can be challenging in low-visibility areas. However, the developed prototypes do not investigate realistic usage scenarios where coverage might fail.
- Energy consumption: When compared to short-range and local area technologies, LPWAN standards may excel in energy efficiency, which is a key factor in increasing the battery life of wearables. Nevertheless, the existing studies to date have barely evaluated the quantitative benefits of using LPWAN compared not only to other transmission technologies but also to other sources of consumption present in the detector, mainly the detection algorithm.
- Implementation difficulty and reliability: LPWAN networks offer greater flexibility than other types of networks, but they still involve certain operational costs (subscription to operators) or deployment costs (e.g., location, installation, and management of base stations) that the literature does not evaluate. Likewise, the low bandwidth provided for LPWAN networks may be sufficient to send certain types of alarms (encodable in a few bytes) but may be clearly insufficient if more complex information needs to be sent to an external node in real-time (such as long samples of inertial signals). This limitation is scarcely addressed in the reviewed works.
- Lack of large-scale and clinical studies: In any case, as with almost all literature dedicated to FDS systems (whether based on contextual or wearable sensor systems), there is an almost complete lack of large-scale clinical trials.

7. Recommendations

- Based on the limitations and deficiencies identified, we outline the main areas of action that future proposals for LPWAN-based FDS should consider to develop viable products for real-world applications:
- Analyze coverage: Prototypes should be tested in realistic environments to demonstrate that users can be monitored over long distances (both outdoors and indoors) without compromising signal quality.
- Characterize energy consumption: The energy consumption of prototypes should be 'budgeted' and characterized in detail to demonstrate that using LPWAN technologies contributes to a significant increase in wearable autonomy. In this regard, the consumption (not only of energy but also of hardware resources) of the transmission system should be compared with that required by the detection algorithm.
- Conduct large-scale clinical studies: Develop large-scale longitudinal studies to appraise the actual effectiveness of LPWAN-based FDSs. In that sense, clinical field tests and massive evaluations of the long-term performance of the detectors in real-world scenarios should be prioritized.
- Integrate with emerging technologies: Investigate the integration of LPWAN technologies with emerging technologies such as 5G and edge computing to enhance data processing capabilities and reduce latency. This could improve the overall performance and responsiveness of FDSs.

- Develop user-centric designs: Evaluate the prototypes not only from a purely technical point-of-view but also consider a user-centric perspective that does not neglect key aspects such as ergonomics or usability. Focus on developing user-friendly wearable devices that ensure comfort and social acceptability. Consider different placement locations on the body and the specific needs of various user groups to maximize the practicality and adoption of these devices.
- Remote Rural Areas: LPWAN technologies such as LoRaWAN and Sigfox are highly beneficial in remote rural areas where traditional communication infrastructures, such as cellular networks, are limited or nonexistent. The long-range and low-power characteristics of LPWAN make it an ideal choice for monitoring fall detection in environments where other communication networks might be impractical or too costly to implement.
- Urban Environments with High Building Density: NB-IoT is particularly well-suited for urban environments with high building density and potential signal interference. Its ability to penetrate buildings and provide reliable communication in densely populated areas makes it a valuable asset for FDSs deployed in cities, ensuring consistent performance even in challenging urban landscapes.
- Elder Care in Low-Cost Public Health Initiatives: In scenarios where cost-effectiveness is crucial, such as large-scale public health initiatives aimed at elder care in developing regions, the low operational costs and minimal infrastructure requirements of LPWAN provide a scalable solution for deploying FDSs to a broader population, making healthcare more accessible.

8. Conclusions

This systematic review provides a comprehensive analysis of LPWAN technologies in wearable FDSs, focusing on enhancing transmission reliability, energy efficiency, and sensor precision. LPWAN technologies, especially LoRaWAN, offer superior energy efficiency and better signal penetration than WiFi and BLE, resulting in improved coverage in challenging environments. However, low-visibility scenarios, such as devices placed on the floor, still pose challenges.

The review highlights the use of various IMU sensors (accelerometers, gyroscopes, magnetometers) and advanced detection algorithms (CNN, LSTM), which improve detection accuracy and system reliability. While threshold-based algorithms offer greater energy efficiency, machine learning and neural network-based algorithms enhance detection accuracy, albeit with higher energy consumption.

Balancing energy efficiency and data transmission speed is crucial when choosing between LoRaWAN and NB-IoT. Despite their potential, LPWAN technologies have not been fully investigated in FDSs, with limited studies on their benefits. Further research is needed to explore these aspects comprehensively, emphasizing large-scale field tests and clinical evaluations to fully realize the potential of LPWAN technologies in wearable FDSs.

Overall, enhancing transmission reliability, optimizing energy use, and improving sensor precision are vital for advancing FDSs. This review guides future developments in enhancing the precision, energy efficiency, and global effectiveness of these systems.

Future research should focus on conducting large-scale field tests and clinical evaluations to validate the long-term effectiveness of LPWAN-based FDSs. Additionally, exploring hybrid detection algorithms and integrating LPWAN technologies with emerging technologies like 5G and edge computing could further enhance system performance. Practical applications include not only FDSs but also remote health monitoring, smart home integration, and use in rehabilitation and assisted living facilities to improve elderly care and emergency response times.

Author Contributions: Conceptualization, M.V. and E.C.; methodology, M.V.; formal analysis, M.V. and E.C.; investigation, M.V.; resources, M.V. and E.C.; writing—original draft preparation, M.V. and E.C.; writing—review and editing, M.V. and E.C.; supervision, E.C.; project administration,

E.C.; funding acquisition, E.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Spanish Ministry of Science, Innovation, and Universities (MCIN/AEI/10.13039/501100011033) and NextGenerationEU/PRTR Funds under grant ED2021-130456B-I00, by Universidad de Málaga, Campus de Excelencia Internacional Andalucia Tech (grant B4-2023-12), DIANA PAIDI research group and Universidad de Investigación y Desarrollo (UDI, Bucaramanga -Colombia-).

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

References

1. Aledhari, M.; Razzak, R.; Qolomany, B.; Al-Fuqaha, A.; Saeed, F. Biomedical IoT: Enabling Technologies, Architectural Elements, Challenges, and Future Directions. *IEEE Access* **2022**, *10*, 31306–31339. [CrossRef] [PubMed]
2. Costin, H.; Rotariu, C.; Adochiei, F.; Ciobotariu, R.; Andruseac, G.; Corciova, F. Telemonitoring of vital signs—An effective tool for ambient assisted living. *IFMBE Proc.* **2011**, *36*, 60–65. [CrossRef]
3. World Health Organization. WHO Global Report on Falls Prevention in Older Age. Available online: https://apps.who.int/iris/handle/10665/43811 (accessed on 26 February 2023).
4. World Health Organization. Falls. Available online: https://www.who.int/en/news-room/fact-sheets/detail/falls (accessed on 26 February 2023).
5. World Health Organization. Ageing and Health. Available online: https://www.who.int/news-room/fact-sheets/detail/ageing-and-health (accessed on 10 April 2023).
6. Huynh, Q.T.; Nguyen, U.D.; Tran, B.Q. A Cloud-Based System for In-Home Fall Detection and Activity Assessment. In *IFMBE Proceedings, Proceedings of the 7th International Conference on the Development of Biomedical Engineering in Vietnam (BME 7), Ho Chi Minh City, Vietnam, 27–29 June 2018*; Springer: Singapore, 2020; Volume 69, pp. 103–108. [CrossRef]
7. Zanaj, E.; Disha, D.; Spinsante, S.; Gambi, E. A wearable fall detection system based on LoRa LPWAN technology. *J. Commun. Softw. Syst.* **2020**, *16*, 232–242. [CrossRef]
8. Chang, W.J.; Chen, L.B.; Chen, M.C.; Su, J.P.; Sie, C.Y.; Yang, C.H. Design and Implementation of an Intelligent Assistive System for Visually Impaired People for Aerial Obstacle Avoidance and Fall Detection. *IEEE Sens. J.* **2020**, *20*, 10199–10210. [CrossRef]
9. Newaz, N.T.; Hanada, E. The Methods of Fall Detection: A Literature Review. *Sensors* **2023**, *23*, 5212. [CrossRef]
10. Tanutama, L.; Wijaya, H.; Ardianti, D. Elderly Fall Detection and Warning System. In *IOP Conference Series: Earth and Environmental Science, Proceedings of the 4th International Conference on Eco Engineering Development 2020, Banten, Indonesia, 10–11 November 2020*; IOP Publishing Ltd.: Bristol, UK, 2021; Volume 794, p. 794. [CrossRef]
11. Lachtar, A.; Val, T.; Kachouri, A. Elderly monitoring system in a smart city environment using LoRa and MQTT. *IET Wirel. Sens. Syst.* **2020**, *10*, 70–77. [CrossRef]
12. Valach, A.; Macko, D. Exploration of the LoRa Technology Utilization Possibilities in Healthcare IoT Devices. In Proceedings of the 2018 16th International Conference on Emerging eLearning Technologies and Applications (ICETA), Stary Smokovec, Slovakia, 15–16 November 2018. [CrossRef]
13. Lousado, J.P.; Pires, I.M.; Zdravevski, E.; Antunes, S. Monitoring the health and residence conditions of elderly people, using lora and the things network. *Electronics* **2021**, *10*, 1729. [CrossRef]
14. Patel, W.D.; Ramani, B.; Pandya, S.; Bhaskar, S.; Koyuncu, B.; Ghayvat, H. NXTGeUH: LoRaWAN based NEXT Generation Ubiquitous Healthcare System for Vital Signs Monitoring & Falls Detection. In Proceedings of the 2018 IEEE Punecon, Pune, India, 30 November–2 December 2018. [CrossRef]
15. Wu, H.H.; Lee, D.H. Monitoring of driver's biomedical signals using LoRa-based wireless communications. *IEICE Electron. Express* **2021**, *18*, 20210104. [CrossRef]
16. Li, Y.; Lin, Z.; Huang, Z.; Cai, Z.; Huang, L.; Wei, Z. A Channel Hopping LoRa Technology Based Emergency Communication System for Elderly People Living Alone. In Proceedings of the 2022 21st International Symposium on Communications and Information Technologies ISCIT, Xi'an, China, 27–30 September 2022; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2022; pp. 19–26. [CrossRef]
17. Han, J.; Song, W.; Gozho, A.; Sung, Y.; Ji, S.; Song, L.; Wen, L.; Zhang, Q. LoRa-Based smart iot application for smart city: An Example of Human Posture Detection. *Wirel. Commun. Mob. Comput.* **2020**, *2020*, 8822555. [CrossRef]
18. Fernandes Carvalho, D.; Ferrari, P.; Sisinni, E.; Bellitti, P.; Lopomo, N.F.; Serpelloni, M. Using LPWAN Connectivity for Elderly Activity Monitoring in Smartcity Scenarios. *Lect. Notes Electr. Eng.* **2020**, *627*, 81–87. [CrossRef]
19. Vimal, S.; Robinson, Y.H.; Kadry, S.; Long, H.V.; Nam, Y. IoT Based Smart Health Monitoring with CNN Using Edge Computing. *J. Internet Technol.* **2021**, *22*, 173–185. [CrossRef]
20. Escriba, C.; Roux, J.; Hajjine, B.; Fourniols, J.Y. Smart wearable active patch for elderly health prevention. In Proceedings of the 2018 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 2018, 12–14 December 2018; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2018; pp. 1040–1043. [CrossRef]

21. Much, M.D.; Marcon, C.; Hessel, F.; Cataldo Neto, A. LifeSenior—A Health Monitoring IoT System Based on Deep Learning Architecture. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Proceedings of the 7th International Conference, ITAP 2021, Washington, DC, USA 24–29 July 2021*; Springer Science and Business Media Deutschland GmbH: Berlin, Germany, 2021; Volume 12787, pp. 293–306. [CrossRef]
22. Manatarinat, W.; Poomrittigul, S.; Tantatsanawong, P. Narrowband-internet of things (NB-IoT) system for elderly healthcare services. In Proceedings of the 2019 5th International Conference on Engineering, Applied Sciences and Technology (ICEAST), Luang Prabang, Laos, 2–5 July 2019; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2019. [CrossRef]
23. Islam, M.S.; Islam, M.T.; Almutairi, A.F.; Beng, G.K.; Misran, N.; Amin, N. Monitoring of the Human Body Signal through the Internet of Things (IoT) Based LoRa Wireless Network System. *Appl. Sci.* **2019**, *9*, 1884. [CrossRef]
24. Dammak, B.; Turki, M.; Cheikhrouhou, S.; Baklouti, M.; Mars, R.; Dhahbi, A. LoRaChainCare: An IoT Architecture Integrating Blockchain and LoRa Network for Personal Health Care Data Monitoring. *Sensors* **2022**, *22*, 1497. [CrossRef] [PubMed]
25. Song, W.; Liao, J.; Han, J. A Real-Time Human Posture Recognition System Using Internet of Things (IoT) Based on LoRa Wireless Network. In *Lecture Notes in Electrical Engineering, Proceedings of the CSA-CUTE 2019, Macau, China, 18–20 December 2019*; Springer Science and Business Media Deutschland GmbH: Berlin, Germany, 2021; Volume 715, pp. 379–385. [CrossRef]
26. Qian, Z.; Lin, Y.; Jing, W.; Ma, Z.; Liu, H.; Yin, R.; Li, Z.; Bi, Z.; Zhang, W. Development of a Real-Time Wearable Fall Detection System in the Context of Internet of Things. *IEEE Internet Things J.* **2022**, *9*, 21999–22007. [CrossRef]
27. Fan, X.; Li, Z.; Zhang, L. Design and Implementation of Fall Detection Equipment for the Elderly Based on NB-IoT. In Proceedings of the 2022 International Conference on Artificial Intelligence and Computer Information Technology (AICIT), Yichang, China, 16–18 September 2022. [CrossRef]
28. Liu, P.; Pan, J.; Zhu, H.; Li, Y. A Wearable Fall Detection System Based on 1D CNN. In Proceedings of the 2021 2nd International Conference on Artificial Intelligence and Computer Engineering (ICAICE), Hangzhou, China, 5–7 November 2021; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2021; pp. 200–203. [CrossRef]
29. Saleh Alhassoun, N. Cross-Layer Energy Optimization for IoT-Enabled Smart Spaces. In Proceedings of the 2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), Austin, TX, USA, 23–27 March 2020.
30. Chen, X.; Jiang, S.; Lo, B. Subject-Independent Slow Fall Detection with Wearable Sensors via Deep Learning. In Proceedings of the 2020 IEEE Sensors, Rotterdam, Netherlands, 25–28 October 2020; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2020. [CrossRef]
31. Pierleoni, P.; Belli, A.; Maurizi, L.; Palma, L.; Pernini, L.; Paniccia, M.; Valenti, S. A Wearable Fall Detector for Elderly People Based on AHRS and Barometric Sensor. *IEEE Sens. J.* **2016**, *16*, 6733–6744. [CrossRef]
32. Makma, J.; Thanapatay, D.; Isshiki, T.; Chinrungrueng, J.; Thiemjarus, S. Toward Accurate Fall Detection with a Combined Use of Wearable and Ambient Sensors. In Proceedings of the 2022 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT & NCON), Chiang Rai, Thailand, 26–28 January 2022; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2022; pp. 298–301. [CrossRef]
33. Wong, W.-K.; Hou, L.-Y.; Pan, T.; Wu, C.-C.; Chen, Y.-H. An IoT Application Based on LoRa Data Transmission. *InternatIonal J. IntellIgent Technol. Appl. Stat.* **2022**, *15*, 87–100. [CrossRef]
34. Wu, Y.; Zeng, P.; Ge, H. A Research of Fall Detection Module Based on NB-IOT. In Proceedings of the 2022 7th International Conference on Computer and Communication Systems (ICCCS), Wuhan, China, 22–25 April 2022; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2022; pp. 197–201. [CrossRef]
35. Ren, L.; Peng, Y. Research of fall detection and fall prevention technologies: A systematic review. *IEEE Access* **2019**, *7*, 77702–77722. [CrossRef]
36. Salah, O.Z.; Selvaperumal, S.K.; Abdulla, R. Accelerometer-based elderly fall detection system using edge artificial intelligence architecture. *Int. J. Electr. Comput. Eng.* **2022**, *12*, 4430–4438. [CrossRef]
37. Fanca, A.; Puscasiu, A.; Gota, D.I.; Valean, H. Methods to minimize false detection in accidental fall warning systems. In Proceedings of the 2019 23rd International Conference on System Theory, Control and Computing (ICSTCC), Sinaia, Romania, 9–11 October 2019; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2019; pp. 851–855. [CrossRef]
38. Igual, R.; Medrano, C.; Plaza, I. Challenges, issues and trends in fall detection systems. *Biomed. Eng. Online* **2013**, *12*, 66. [CrossRef]
39. Özdemir, A.T. An analysis on sensor locations of the human body for wearable fall detection devices: Principles and practice. *Sensors* **2016**, *16*, 1161. [CrossRef]
40. Nguyen Gia, T.; Sarker, V.K.; Tcarenko, I.; Rahmani, A.M.; Westerlund, T.; Liljeberg, P.; Tenhunen, H. Energy efficient wearable sensor node for IoT-based fall detection systems. *Microprocess. Microsyst.* **2018**, *56*, 34–46. [CrossRef]
41. Yu, X.; Park, S.; Kim, D.; Kim, E.; Kim, J.; Kim, W.; An, Y.; Xiong, S. A practical wearable fall detection system based on tiny convolutional neural networks. *Biomed. Signal Process. Control* **2023**, *86*, 105325. [CrossRef]
42. De Raeve, N.; Shahid, A.; De Schepper, M.; De Poorter, E.; Moerman, I.; Verhaevert, J.; Van Torre, P.; Rogier, H. Bluetooth-Low-Energy-Based Fall Detection and Warning System for Elderly People in Nursing Homes. *J. Sens.* **2022**, *2022*, 9930681. [CrossRef]
43. Freitas, R.; Terroso, M.; Marques, M.; Gabriel, J.; Marques, A.T.; Simoes, R. Wearable sensor networks supported by mobile devices for fall detection. *Proc. IEEE Sens.* **2014**, *2014*, 2246–2249. [CrossRef]

44. Cruz, F.R.G.; Sejera, M.P.; Bunnao, M.B.G.; Jovellanos, B.R.; Maaño, P.L.C.; Santos, C.J.R. Fall Detection Wearable Device Interconnected Through ZigBee Network. In Proceedings of the 2017 IEEE 9th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM), Manila, Philippines, 1–3 December 2017. [CrossRef]
45. Rao Gannapathy, V.; Fayeez, A.; Ibrahim, B.T.; Zakaria, Z.B.; Rani, A.; Othman, B.; Latiff, A.A. Zigbee-based smart fall detection and notification system with wearable sensor (e-safe). *IJRET: Int. J. Res. Eng. Technol.* **2013**, *2*, 337–344. [CrossRef]
46. Huang, C.N.; Chan, C.T. A ZigBee-Based Location-Aware Fall Detection System for Improving Elderly Telecare. *Int. J. Environ. Res. Public. Health* **2014**, *11*, 4233–4248. [CrossRef]
47. Mekki, K.; Bajic, E.; Chaxel, F.; Meyer, F. A comparative study of LPWAN technologies for large-scale IoT deployment. *ICT Express* **2019**, *5*, 1–7. [CrossRef]
48. Devalal, S.; Karthikeyan, A. LoRa Technology—An Overview. In Proceedings of the 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 29–31 March 2018; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2018; pp. 284–290. [CrossRef]
49. Alkhayyal, M.; Mostafa, A. Recent Developments in AI and ML for IoT: A Systematic Literature Review on LoRaWAN Energy Efficiency and Performance Optimization. *Sensors* **2024**, *24*, 4482. [CrossRef] [PubMed]
50. Chaudhari, B.S.; Zennaro, M.; Borkar, S. LPWAN technologies: Emerging application characteristics, requirements, and design considerations. *Future Internet* **2020**, *12*, 46. [CrossRef]
51. Chilamkurthy, N.S.; Pandey, O.J.; Ghosh, A.; Cenkeramaddi, L.R.; Dai, H.N. Low-Power Wide-Area Networks: A Broad Overview of Its Different Aspects. *IEEE Access* **2022**, *10*, 81926–81959. [CrossRef]
52. Nolan, K.E.; Guibene, W.; Kelly, M.Y. An evaluation of low power wide area network technologies for the Internet of Things. In Proceedings of the 2016 International Wireless Communications and Mobile Computing Conference (IWCMC), Paphos, Cyprus, 5–9 September 2016; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2016; pp. 439–444. [CrossRef]
53. Pena Queralta, J.; Gia, T.N.; Tenhunen, H.; Westerlund, T. Edge-AI in LoRa-based health monitoring: Fall detection system with fog computing and LSTM recurrent neural networks. In Proceedings of the 2019 42nd International Conference on Telecommunications and Signal Processing (TSP), Budapest, Hungary, 1–3 July 2019; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2019; pp. 601–604. [CrossRef]
54. Cai, W.Y.; Guo, J.H.; Zhang, M.Y.; Ruan, Z.X.; Zheng, X.C.; Lv, S.S. GBDT-Based Fall Detection with Comprehensive Data from Posture Sensor and Human Skeleton Extraction. *J. Healthc. Eng.* **2020**, *2020*, 8887340. [CrossRef] [PubMed]
55. Yuan, J.; Tan, K.K.; Lee, T.H.; Koh, G.C.H. Power-efficient interrupt-driven algorithms for fall detection and classification of activities of daily living. *IEEE Sens. J.* **2015**, *15*, 1377–1387. [CrossRef]
56. Siong Jun, S.; Rashidi Ramli, H.; Che Soh, A.; Ain Kamsani, N.; Kamil Raja Ahmad, R.; Anom Ahmad, S.; Juraiza Ishak, A. Development of fall detection and activity recognition using threshold based method and neural network. *Indones. J. Electr. Eng. Comput. Sci.* **2020**, *17*, 1338–1347. [CrossRef]
57. Luque, R.; Casilari, E.; Morón, M.J.; Redondo, G. Comparison and Characterization of Android-Based Fall Detection Systems. *Sensors* **2014**, *14*, 18543–18574. [CrossRef] [PubMed]
58. Kausar, F.; Mesbah, M.; Iqbal, W.; Ahmad, A.; Sayyed, I. Fall Detection in the Elderly using Different Machine Learning Algorithms with Optimal Window Size. *Mob. Netw. Appl.* **2023**, *1*, 1–11. [CrossRef]
59. Chai, X.; Lee, B.G.; Pike, M.; Wu, R.; Chieng, D.; Chung, W.Y. Pre-Impact Firefighter Fall Detection Using Machine Learning on the Edge. *IEEE Sens. J.* **2023**, *23*, 14997–15009. [CrossRef]
60. Astriani, M.S.; Bahana, R.; Kurniawan, A.; Yi, L.H. Threshold-based low power consumption human fall detection for health care and monitoring system. In Proceedings of the Proceedings of 2020 International Conference on Information Management and Technology (ICIMTech), Bandung, Indonesia, 13–14 August 2020; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2020; pp. 853–857. [CrossRef]
61. Šeketa, G.; Vugrin, J.; Lacković, I. Optimal threshold selection for acceleration-based fall detection. *IFMBE Proc.* **2018**, *66*, 151–155. [CrossRef]
62. Page, M.J.; McKenzie, J.E.; Bossuyt, P.M.; Boutron, I.; Hoffmann, T.C.; Mulrow, C.D.; Shamseer, L.; Tetzlaff, J.M.; Akl, E.A.; Brennan, S.E.; et al. The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *BMJ* **2021**, *372*. [CrossRef]
63. Sigfox, S.A. Sigfox 0G Technology. Available online: https://www.sigfox.com/what-is-sigfox/ (accessed on 16 May 2024).
64. Mekki, K.; Bajic, E.; Chaxel, F.; Meyer, F. Overview of Cellular LPWAN Technologies for IoT Deployment: Sigfox, LoRaWAN, and NB-IoT. In Proceedings of the 2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), Athens, Greece, 19–23 March 2018; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2018; pp. 197–202. [CrossRef]
65. Scheurer, S.; Koch, J.; Kucera, M.; Bryn, H.; Bärtschi, M.; Meerstetter, T.; Nef, T.; Urwyler, P. Optimization and technical validation of the AIDE-MOI fall detection algorithm in a real-life setting with older adults. *Sensors* **2019**, *19*, 1357. [CrossRef]
66. Pierleoni, P.; Belli, A.; Palma, L.; Concetti, R.; Sabbatini, L.; Raggiunto, S. A complete architecture for Ambient Assisted Living scenarios using a cross protocol proxy. *J. Ambient. Intell. Humaniz. Comput.* **2023**, *15*, 2757–2764. [CrossRef]

67. Kautsarina; Kusumawati, D. The Potential Adoption of the Internet of Things in Rural Areas. In Proceedings of the 2018 International Conference on ICT for Rural Development: Rural Development through ICT: Concept, Design, and Implication (IC-ICTRuDEv), Badung, Indonesia, 17–18 October 2018; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2018; pp. 124–130. [CrossRef]
68. Lykov, Y.; Paniotova, A.; Shatalova, V.; Lykova, A. Energy Efficiency Comparison LPWANs: LoRaWAN vs Sigfox. In Proceedings of the 2020 IEEE International Conference on Problems of Infocommunications Science and Technology (PIC S & T), Kharkiv, Ukraine, 6–9 October 2020; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2021; pp. 485–490. [CrossRef]
69. Qadir, Q.M.; Rashid, T.A.; Al-Salihi, N.K.; Ismael, B.; Kist, A.A.; Zhang, Z. Low power wide area networks: A survey of enabling technologies, applications and interoperability needs. *IEEE Access* **2018**, *6*, 77454–77473. [CrossRef]
70. Rama, Y.; Alper Özpınar, M. A Comparison of Long-Range Licensed and Unlicensed LPWAN Technologies According to Their Geolocation Services and Commercial Opportunities. In Proceedings of the 2018 IEEE 18th Mediterranean Microwave Symposium (MMS), Istanbul, Turkey, 31 October–2 November 2018; pp. 398–403. [CrossRef]
71. Stanco, G.; Botta, A.; Frattini, F.; Giordano, U.; Ventre, G. On the performance of IoT LPWAN technologies: The case of Sigfox, LoRaWAN and NB-IoT. In Proceedings of the IEEE International Conference on Communications, Seoul, Republic of Korea, 16–20 May 2022; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2022; Volume 2022, pp. 2096–2101. [CrossRef]
72. Xu, T.; Se, H.; Liu, J. A fusion fall detection algorithm combining threshold-based method and convolutional neural network. *Microprocess. Microsyst.* **2021**, *82*, 103828. [CrossRef]
73. Sucerquia, A.; López, J.D.; Vargas-Bonilla, J.F. SisFall: A fall and movement dataset. *Sensors* **2017**, *17*, 198. [CrossRef]
74. Vavoulas, G.; Pediaditis, M.; Spanakis, E.G.; Tsiknakis, M. The MobiFall dataset: An initial evaluation of fall detection algorithms using smartphones. In Proceedings of the 13th IEEE International Conference on BioInformatics and BioEngineering, Chania, Greece, 10–13 November 2013. [CrossRef]
75. Casilari, E.; Santoyo-Ramón, J.A.; Cano-García, J.M. Analysis of public datasets for wearable fall detection systems. *Sensors* **2017**, *17*, 1513. [CrossRef]
76. Saleh, M.; Jeannes, R.L.B. Elderly Fall Detection Using Wearable Sensors: A Low Cost Highly Accurate Algorithm. *IEEE Sens. J.* **2019**, *19*, 3156–3164. [CrossRef]
77. Guvensan, M.A.; Kansiz, A.O.; Camgoz, N.C.; Turkmen, H.I.; Yavuz, A.G.; Karsligil, M.E. An Energy-Efficient Multi-Tier Architecture for Fall Detection on Smartphones. *Sensors* **2017**, *17*, 1487. [CrossRef]
78. Lampoltshammer, T.J.; de Freitas, E.P.; Nowotny, T.; Plank, S.; da Costa, J.P.C.L.; Larsson, T.; Heistracher, T. Use of Local Intelligence to Reduce Energy Consumption of Wireless Sensor Nodes in Elderly Health Monitoring Systems. *Sensors* **2014**, *14*, 4932–4947. [CrossRef]
79. Musci, M.; De Martini, D.; Blago, N.; Facchinetti, T.; Piastra, M. Online Fall Detection Using Recurrent Neural Networks on Smart Wearable Devices. *IEEE Trans. Emerg. Top. Comput.* **2021**, *9*, 1276–1289. [CrossRef]
80. Fernandez-Bermejo, J.; Martinez-del-Rincon, J.; Dorado, J.; del Toro, X.; Santof imia, M.J.; Lopez, J.C. Edge computing transformers for fall detection in older adults. *Int. J. Neural Syst.* **2024**, *34*, 2450026. [CrossRef]
81. Casamassima, F.; Farella, E.; Benini, L. Context aware power management for motion-sensing body area network nodes. In Proceedings of the 2014 Design, Automation & Test in Europe Conference & Exhibition (DATE), Dresden, Germany, 24–28 March 2014; pp. 1–6. [CrossRef]
82. Gao, L.; Bourke, A.K.; Nelson, J. Evaluation of accelerometer based multi-sensor versus single-sensor activity recognition systems. *Med. Eng. Phys.* **2014**, *36*, 779–785. [CrossRef]
83. Noor, M.H.M.; Salcic, Z.; Wang, K.I.K. Dynamic sliding window method for physical activity recognition using a single tri-axial accelerometer. In Proceedings of the 2015 10th IEEE Conference on Industrial Electronics and Applications (ICIEA), Auckland, New Zealand, 15–17 June 2015; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2015; pp. 102–107. [CrossRef]
84. Gharghan, S.K.; Fakhrulddin, S.S.; Al-Naji, A.; Chahl, J. Energy-efficient elderly fall detection system based on power reduction and wireless power transfer. *Sensors* **2019**, *19*, 4452. [CrossRef] [PubMed]
85. Ballerini, M.; Polonelli, T.; Brunelli, D.; Magno, M.; Benini, L. NB-IoT Versus LoRaWAN: An Experimental Evaluation for Industrial Applications. *IEEE Trans. Industr Inform.* **2020**, *16*, 7802–7811. [CrossRef]
86. Semtech SX1257 Datasheet. Available online: https://www.semtech.com/products/wireless-rf/lora-core/sx1257 (accessed on 22 August 2024).
87. HopeRF. RFM95/96/97/98(W) LoRa Transceiver Module Datasheet. Available online: https://www.hoperf.com/modules/lora/RFM95W.html (accessed on 22 August 2024).
88. Semtech. SX1276/77/78/79 Datasheet. Available online: https://www.semtech.com/products/wireless-rf/lora-connect/sx1276 (accessed on 22 August 2024).
89. Quectel. BC95 NB-IoT Module Datasheet. Available online: https://www.es.co.th/Schemetic/PDF/QUECTEL_BC95B.PDF (accessed on 22 August 2024).

90. China Mobile IoT Company. M5310A AT Command Manual. Available online: https://iot.10086.cn/Uploads/file/product/20180827/M5310A%20AT%20%E5%91%BD%E4%BB%A4%E7%94%A8%E4%B9%A6%E4%BD%BF%E7%94%A8%E6%89%8B%E5%86%8C_V1_20180827154312_20506.pdf (accessed on 22 August 2024).
91. Nordic Semiconductor. nRF9160 System-in-Package Datasheet. Nordic Semiconductor. Available online: https://www.nordicsemi.com/Products/nRF9160 (accessed on 22 August 2024).
92. Sinha, R.S.; Wei, Y.; Hwang, S.H. A survey on LPWA technology: LoRa and NB-IoT. *ICT Express* **2017**, *3*, 14–21. [CrossRef]
93. Bet, P.; Castro, P.C.; Ponti, M.A. Fall detection and fall risk assessment in older person using wearable sensors: A systematic review. *Int. J. Med. Inform.* **2019**, *130*, 103946. [CrossRef] [PubMed]
94. Ferreira, C.M.S.; Oliveira, R.A.R.; Silva, J.S. Low-energy smart cities network with lora and bluetooth. In Proceedings of the 2019 7th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud), Newark, CA, USA, 4–9 April 2019; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2019; pp. 24–29. [CrossRef]
95. Muteba, F.; Djouani, K.; Olwal, T. A comparative Survey Study on LPWA IoT Technologies: Design, considerations, challenges and solutions. *Procedia Comput. Sci.* **2019**, *155*, 636–641. [CrossRef]
96. Gemperle, F.; Kasabach, C.; Stivoric, J.; Bauer, M.; Martin, R. Design for Wearability. In Proceedings of the 2nd IEEE International Symposium on Wearable Computers, Pittsburgh, PA, USA, 19–20 October 1998. [CrossRef]
97. Pasher, E.; Popper, Z.; Raz, H.; Lawo, M. WearIT@work: A wearable computing solution for knowledge-based development. *Int. J. Knowl.-Based Dev.* **2010**, *1*, 346–360. [CrossRef]
98. Thilo, F.J.S.; Bilger, S.; Halfens, R.J.G.; Schols, J.M.G.A.; Hahn, S. Involvement of the end user: Exploration of older people's needs and preferences for a wearable fall detection device—A qualitative descriptive study. *Patient Prefer. Adherence* **2017**, *11*, 11–22. [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.

technologies

Article

Enhancing Diagnostic Accuracy for Skin Cancer and COVID-19 Detection: A Comparative Study Using a Stacked Ensemble Method

Hafza Qayyum [1,*], Syed Tahir Hussain Rizvi [2], Muddasar Naeem [3,*], Umamah bint Khalid [1], Musarat Abbas [1] and Antonio Coronato [3]

[1] Department of Electronics, Quaid-i-Azam University, Islamabad 45320, Pakistan
[2] Department of Electrical Engineering and Computer Science, University of Stavanger, 4036 Stavanger, Norway
[3] Università Department of Computer Engineering, Università Telematica Giustino Fortunato, 82100 Benevento, Italy
* Correspondence: qayyumhafza@gmail.com (H.Q.); m.naeem@unifortunato.eu (M.N.)

Abstract: In recent years, COVID-19 and skin cancer have become two prevalent illnesses with severe consequences if untreated. This research represents a significant step toward leveraging machine learning (ML) and ensemble techniques to improve the accuracy and efficiency of medical image diagnosis for critical diseases such as COVID-19 (grayscale images) and skin cancer (RGB images). In this paper, a stacked ensemble learning approach is proposed to enhance the precision and effectiveness of diagnosis of both COVID-19 and skin cancer. The proposed method combines pretrained models of convolutional neural networks (CNNs) including ResNet101, DenseNet121, and VGG16 for feature extraction of grayscale (COVID-19) and RGB (skin cancer) images. The performance of the model is evaluated using both individual CNNs and a combination of feature vectors generated from ResNet101, DenseNet121, and VGG16 architectures. The feature vectors obtained through transfer learning are then fed into base-learner models consisting of five different ML algorithms. In the final step, the predictions from the base-learner models, the ensemble validation dataset, and the feature vectors extracted from neural networks are assembled and applied as input for the meta-learner model to obtain final predictions. The performance metrics of the stacked ensemble model show high accuracy for COVID-19 diagnosis and intermediate accuracy for skin cancer.

Keywords: COVID-19; skin cancer; stacking; feature vector; ensemble learning; machine learning algorithms

1. Introduction

In recent times, there has been notable and efficient advancement in the domain of the automated analysis of medical images [1]. Modern imaging techniques rely on images with a high resolution to give radiologists multifaceted views to aid in clinical diagnoses, precise predictions, and patient treatment. Ultrasound, endoscopy, X-ray, computed tomography (CT), and magnetic resonance imaging (MRI) are the most common methods for capturing medical images [2]. Currently, numerous studies have emerged concerning the categorization and identification of illnesses through medical imaging. Even though these models have demonstrated promising outcomes, the medical domain still demands enhanced precision [3].

The ongoing COVID-19 pandemic and skin cancer have emphasized the need for accurate and efficient diagnostic tools to identify infected individuals and prevent further transmission [4]. The focus of the proposed study is on utilizing stacked ensemble learning techniques to enhance the accuracy of COVID-19 and skin cancer detection. The main aim of this research is to develop a generalized model using an ensemble learning methodology resulting in precise and accurate predictions for grayscale and RGB medical images.

Artificial intelligence (AI) involves methods and algorithms for performing tasks smartly by learning from previous data or examples, like planning and learning from language [5]. Machine learning (ML) and deep learning (DL) are pivotal branches of AI. ML methods involve training algorithms to learn from data and make predictions and decisions based on patterns identified during training [6]. Deep learning (DL), a subset of ML, employs intricate neural networks with multiple layers to automatically extract complex features from data [7]. This facilitates advancements in tasks like image recognition and natural language processing [8]. The use of ML and DL technologies for the detection and diagnosis of COVID-19 disease has significant effects and has been used in several investigations [9]. These AI-empowered techniques have a considerable tangible capacity for providing an accurate and efficient intelligent system for detecting and estimating the severity of COVID-19. The performance of AI may further be improved by considering safety features of underlying environments [10]. Moreover, AI-based systems can be combined with other technologies such as 5G, cloud storage, and the Internet of Things (IoT) for other variants of COVID-19 epidemics to eliminate geographical issues in the rapid estimation of disease severity, lower treatment costs, and perform epidemic management and immediate epidemic control [11]. Both ML and DL play vital roles in transforming industries by enabling computers to learn and adapt from experience, enhancing their performance over time [12].

The purposeful design and mixing of many models, such as learners, to address specific computational intelligence challenges is known as ensemble learning (EL) [13]. It is a method for combining numerous models to achieve better generalization ability [14]. The ensemble machine learning technique combines different classification algorithms, called base learners (also called base models), to produce a single improved classification model. The main idea is that the final prediction is made by the meta-learner (meta model) based on the base learners. The meta-learner is an approach that works to reduce the base learner's error in prediction. The predicted output of the base learners is utilized as input for the meta-learner. This generalized ability and the accuracy of prediction results obtained using this technique beat the results of a single ML setup [15]. Ensemble strategies have changed with time to improve model generalization in learning. These strategies may be divided into three categories, bagging, boosting, and stacking, as follows:

- Bagging, often referred to as bootstrap aggregating, serves as a commonly employed method for creating ensemble-based algorithms. The core concept behind bagging involves creating a set of independent datasets from the original data. Bagging introduces two key steps to the original models: firstly, the creation of bagging samples and their subsequent presentation to the base-learner models, and secondly, the approach to merge the predictions from multiple predictors.
- Boosting is an ML algorithm that involves training multiple models sequentially, with each subsequent model focusing on correcting the errors made by the previous model. Boosting stands as a robust approach that effectively mitigates overfitting [14].
- The stacked ensemble ML method is a technique that combines multiple classification methods that are homogeneous or heterogeneous, known as base learners, to produce a single superior-performing classification model. The key idea of this method is that the meta-learner generates its final predictions according to the base-learner predictions. The meta-learner is a model that aims to reduce the prediction mistakes of the base learners [15].

Contribution of Research Work

This research work proposed a novel approach for the classification of COVID-19 and skin cancer. The main contributions and the uniqueness of this work are as follows:

- To ensure a reliable evaluation of the proposed method, a customized distribution strategy is implemented for sampling each dataset. Departing from the standard data split method, the approach involves a balanced division of data at each stage of model development. This tailored strategy demonstrates its efficacy in enhancing the overall performance of the model.

- Additionally, the most effective and high-performing CNN variants with a default input size of 224 × 224 (DenseNet-121, ResNet101, VGG16, and a combination of these networks) are utilized as feature extractors. These pretrained models are configured to exclude their fully connected layers to make them suitable for feature extraction. The extracted high-level features from these models are combined as a feature vector that can be valuable for subsequent classification tasks and that also reduces model complexity due to the removal of computationally and memory-intensive fully connected layers.
- The generated feature vector is then further used as input of base-learner modelsto train them. Five different base-learner models (support vector machine (SVM), linear regression (LR), decision tree (DT), random forest (RF), and naive Bayes (NB)) are used in this work.
- The predictions from the base-learner models are integrated with the initial feature vector and the ensemble validation set to generate a very rich and informative fused vector. This fused feature vector is finally fed to the meta-learner model, which provides a very precise final prediction. Five different meta-learner models (RF, DT, LR, NB, and SVM) are compared by training on the same fused feature vector.
- Most of the state-of-the-art methods apply only the base-learner prediction as the input of the meta-learner. The proposed ensemble technique combines different types and levels of features to improve the generalization capability of the final predictors.

This manuscript is organized as follows: Section 1 introduces the field of ensemble learning and the research question. Section 2 discusses work related to COVID-19 and skin cancer; this section also discusses the concept of ensemble learning. Section 3 presents the proposed pipeline, including the datasets, preprocessing methods, deep convolutional neural network architectures, ensemble learning strategies, and machine learning algorithms. Section 4 reports the experimental results, and a discussion of these results can be found in Section 5. Section 6 concludes this paper and give insights on future work.

2. Related Work

Several methods have been proposed in the literature to categorize diseases through medical imaging, with some offering solutions for disease identification. The classification of medical images is a growing field that primarily relies on DL and ML techniques. This section reviews the latest research on medical image classification, especially focusing on datasets related to COVID-19 and skin cancer.

2.1. COVID-19

A pretrained VGGNet model and SVM algorithm are employed in [16] to classify X-ray images into three distinct categories: COVID-19, healthy images, and pneumonia. This provided an accuracy of 95.81%.

Similarly, the authors in [17] employed an ensemble architecture to classify CT scan images. Their approach combined various deep learning models, including VGG16, ResNet50, VGG19, ResNet50V2, InceptionV3, InceptionResNetV2, MobileNet, and Xception. A VGG model having 19 layers achieved the highest accuracy of 92%, while all other models were confined to below 90% accuracy.

2.2. Skin Cancer

The work in [18] provided a comprehensive systematic review of the latest research on the usage of ML algorithms for dermoscopic image categorization of skin cancer. Overall, the review provided valuable insights into the latest research on using ML for skin cancer diagnosis. It can be a useful resource for researchers and practitioners in the field. The results of these studies vary depending on the used algorithm, dataset, and evaluation metrics. However, the review explored that the recent ML and DL models show high potential in skin lesion classification.

The paper [19] introduced a novel approach to combine ML and DL techniques for skin cancer detection that outperformed both expert dermatologists and other state-of-the-art deep learning models. Input features for the classifiers were synthesized from image processing techniques such as contourlet transform and local binary pattern histogram. Pretrained CNNs were used for transfer learning and automatic feature extraction. This method achieved good detection accuracy for benign and malignant forms of cancer.

Moreover, the authors in [20] proposed a deep neural network-based system for skin lesion classification. They used five state-of-the-art architectures as base models, including ResNeXt, SeResNeXt, ResNet, Xception, and DenseNet. They used average and weighted average ensemble models. This method achieved excellent performance for highly imbalanced seven-class datasets. The ensemble models showed higher distinguishing capability compared to others. The average ensemble model achieved a recall score of 93%, and the weighted average ensemble model achieved a recall score of 94%.

2.3. COVID-19 and Skin Cancer

A reproducible pipeline for medical image classification was employed by [21] that they used to analyze the effect of various ensemble learning approaches on the performance. That research work used a dataset related to skin and COVID-19 on various ensemble learning techniques. The paper utilized an ensemble learning approach that combines multiple deep CNNs to improve the accuracy of classification. The outcomes indicated that the stacking exhibited a substantial enhancement in performance, with a notable increase of up to 13% in the F1-score.

A novel approach was proposed in [15] for feature extraction by combining transfer learning from the DenseNet model and six handcrafted techniques to catch more comprehensive and intricate features. In the evaluation of the ISIC dataset and COVID-19 CT Scan data, RF exhibited the highest efficiency for skin detection, and LR outperformed the other algorithms across every evaluation metric during COVID-19 detection.

During the literature review, we observed that there is no nasal association between COVID-19 and skin cancer. However, from a clinical point of view, both are highly associated with the immune system. COVID-19 affects the immune system, and people with low immunity are more prone to the recurrence of melanoma. Melanoma is an aggressive type and can recur even after surgery. During the COVID-19 pandemic, many skin infections were reported which were initially considered as a melanoma [22]. Later, the findings reveal that it was not a melanoma.

2.4. Ensemble Learning

Traditional machine learning methods fail when dealing with complex data because they cannot capture multiple characteristics and underlying structures. Building an efficient model for knowledge discovery is crucial in data mining. Ensemble learning aims to integrate data fusion, data modeling, and data mining into a unified framework. In ensemble learning, a set of features is first extracted using various transformations. Finally, ensemble learning combines the valuable knowledge from the extracted results to achieve knowledge discovery and improve predictive performance through adaptive voting schemes [23].

The ensemble approach to ML uses several homogeneous or heterogeneous algorithms for classification, known as base learners. The base learners are called weak learners and are obtained by a base learning algorithm based on training data to create a classification model [24]. The outputs obtained from base learners are then fed as input to the meta-learners. The meta-learner is an algorithm that learns to minimize the base learners' prediction mistakes. The prediction output of base learners is used as input to the meta-learner, as depicted in Figure 1.

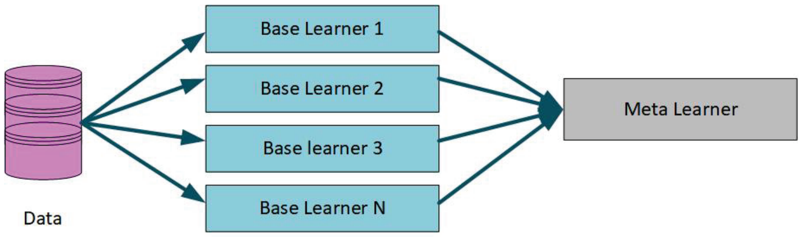


Figure 1. Ensemble learning approach.

2.5. Types of Ensemble Learning

There is no definitive taxonomy of ensemble learning. Usually, four methods used for ensemble learning are described: bagging, boosting, stacking, and error-correcting output codes [25].

- Bagging: It is the first effective and the simplest method of ensemble learning. It was originally designed for classification and is usually applied to decision tree models, but it can be used for regression. The method involves using multiple versions of a training set through the process of bootstrapping. Each of these datasets is used to train a different model. The outputs of the models are combined by averaging (for regression) or voting (for classification) to create a single output. Bagging is only effective for non-linear models.
- Boosting: This model-averaging method is a widely used ensemble technique. It can be used for both classification and regression tasks. In this method, weak classifiers are created iteratively, with each one trained on a dataset where misclassified points are given more weight. The final model is created by combining the outputs of the successive models using voting or averaging. Boosting can sometimes fail to generate a classifier as accurate as a single classifier built from the same data, indicating overfitting.
- Stacking: It is a distinct way of combining multiple models that introduces the concept of a meta-learner. It is less widely used compared to bagging and boosting. Stacking is normally used to combine models of different types. It splits the training set into two disjoint sets. The method trains several base learners on the first part and tests on the second part. The predictions of the base learners are taken as the inputs to train the meta-learners.
- Error-correcting output codes: It is a technique that enhances the performance of classification algorithms in multiclass learning scenarios. In this approach, the multiclass dataset is broken down into multiple independent two-class problems. The algorithm is then applied to each of these problems, and the outputs from the resulting classifiers are combined to make the final prediction.

3. Methodology Overview

The main objective of the proposed method is to maximize the performance of the required task of detection by training various models on a particular dataset. In the proposed approach, a combined feature extraction stage and a stacked ensemble technique are employed to address the challenges of identifying COVID-19 and skin cancer disease.

A comprehensive overview of the different sequential stages of the proposed methodology is illustrated in Figure 2. Commencing with image acquisition, the input data are first passed to the feature extraction stage, employing various pretrained learning models. Subsequently, a feature vector is generated through the combination of features derived from these extractions, which is then passed to the base learners. These base learners are trained on this feature vector, encompassing five diverse ML algorithms: RF, DT, LR, NB, and SVM. Conclusively, the predictions originating from the base learners are integrated with the initial feature vector and the ensemble validation set. The final prediction is executed through the application of the stacking ensemble technique.

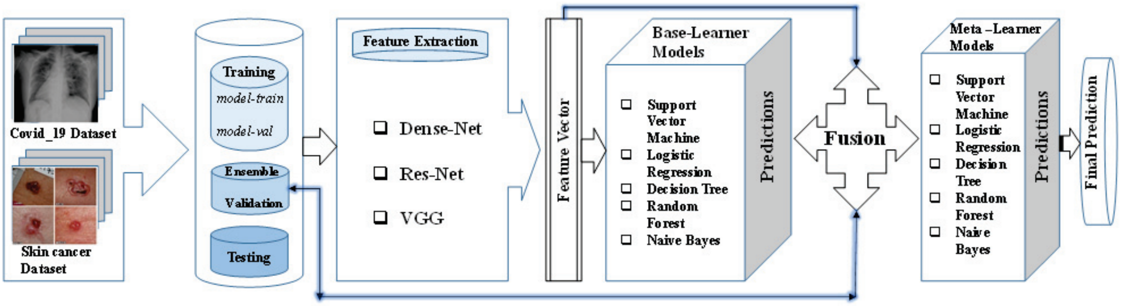
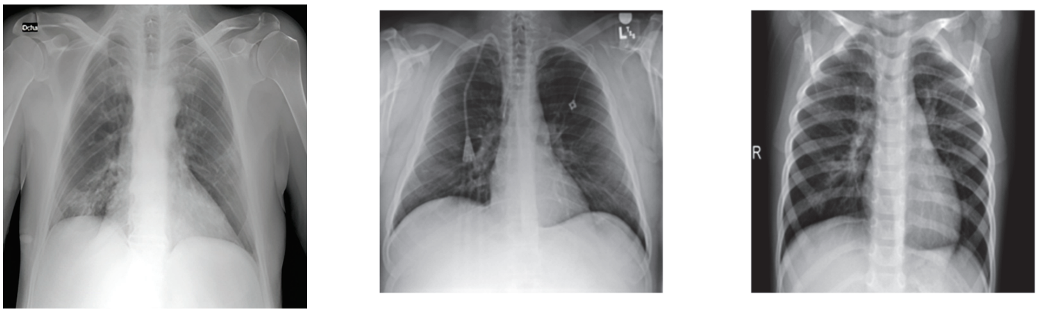


Figure 2. Proposed architecture.

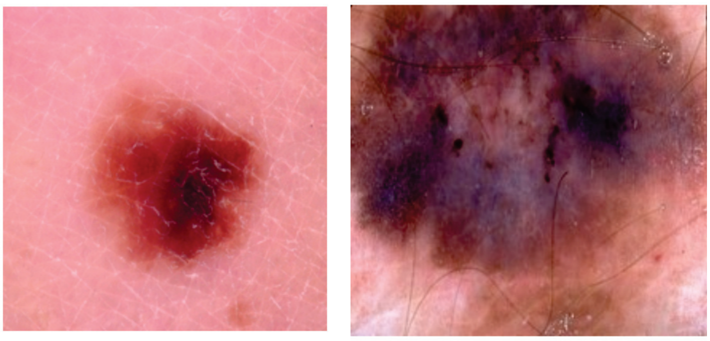
3.1. Datasets

We utilized publicly accessible image datasets for the performance evaluation of the proposed methodology: SARS-CoV-2 CT scan dataset [26] and ISIC Archive dataset [27]. Visual examples showcasing images from each dataset, encompassing different classes, are shown in Figures 3 and 4.



(**a**) COVID (**b**) Normal (**c**) Viral Pneumonia

Figure 3. Sample images from the SARS-CoV-2 X-ray dataset illustrating instances of COVID-19 cases, healthy control cases and viral pneumonia.

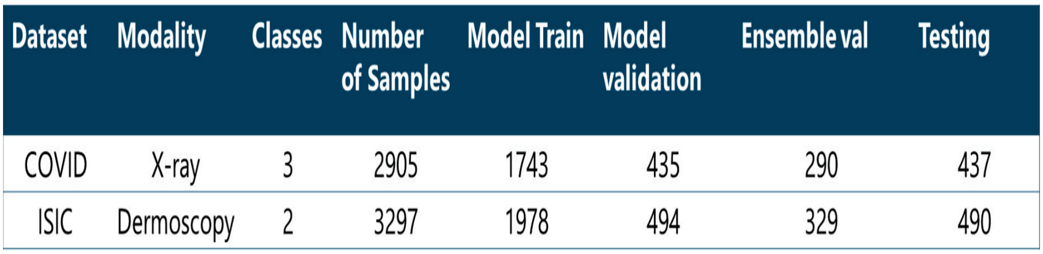


(**a**) Benign (**b**) Malignant

Figure 4. Sample images from the ISIC Archive dataset showing both benign and malignant cases.

3.1.1. COVID-19 Radiography Database

X-ray images are important for visualizing medical issues, and they are used a lot in healthcare. They are also used instead of certain tests for COVID-19. Researchers from different places like Qatar, Doha, Dhaka, Bangladesh, Pakistan, and Malaysia collected X-ray images of people's chests who had COVID-19, as well as some who were healthy and others with viral pneumonia. This dataset has 2905 grayscale pictures. There are 220 pictures of people with COVID-19, 1345 with viral pneumonia, and 1340 healthy pictures. The distribution of these data for our experimentation is shown in Figure 5.



Dataset	Modality	Classes	Number of Samples	Model Train	Model validation	Ensemble val	Testing
COVID	X-ray	3	2905	1743	435	290	437
ISIC	Dermoscopy	2	3297	1978	494	329	490

Figure 5. Overview of the datasets used along with their descriptions and how the samples were distributed.

3.1.2. Skin Lesion Images for Melanoma Classification

Melanoma is a serious health issue where colored spots appear on the skin. It causes a lot of people to get sick, with over 300,000 new cases every year, and sadly, many people die from it. Dermoscopy is a way to find melanoma early. This can be performed by experts looking closely at the skin or by using special cameras that take really detailed pictures. The International Skin Imaging Collaboration (ISIC) has a big collection of pictures of skin spots that people can use to learn and study [21]. The ISIC Archive collection includes 3297 skin cancer images, 1800 of which are classified as benign and 1497 of which are malignant. An overview of the total number of samples and the distribution of the samples are shown in Figure 5.

3.2. Data Splitting

To ensure an assessment of the proposed scheme, we employed the following distribution strategy for the sampling of each dataset, as shown in Figure 6. For the training of base-learner models, 75% of the total dataset was divided into two parts (80% for training of base-learner models (called 'model-train') and 20% for validation (called 'model-val')). For possible training of the meta-model, an additional 10% of the total dataset was set aside (called 'ensemble-validation'). The remaining 15% of the overall dataset was sampled as a testing set (called 'testing') for the final predictions.

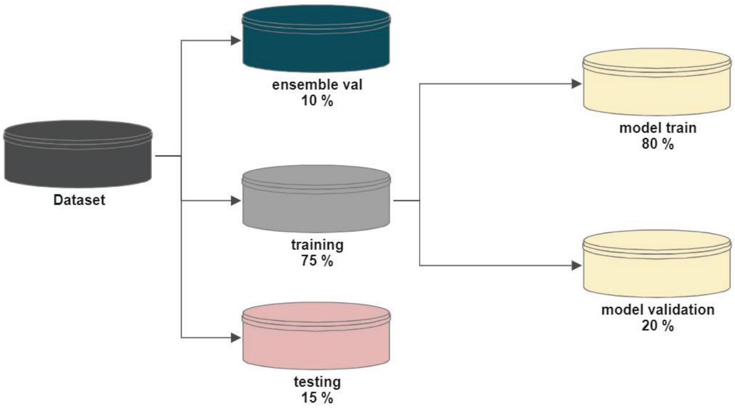


Figure 6. Dataset distribution strategy.

3.3. Data Preprocessing

Preprocessing plays a crucial role in computer vision applications, serving various purposes such as performing noise reduction, highlighting relevant image features for recognition tasks, and aiding in the training of learning models. In this study, a straightforward approach involving the normalization of pixel intensities within the [0, 1] range was employed. This preprocessing step is essential to ensure the model's convergence during the training phase [28]. The configuration for image data preparation includes the techniques such as zoom, brightness, and normalization. Furthermore, the image sizes were scaled down to 224 × 224 pixels, the standard input size of selected feature extractors.

Labels were assigned to three classes in the context of COVID-19 classification, 'COVID', 'Normal', and 'Viral Pneumonia', which are represented numerically as [0, 1, 2]. These labels are used to categorize various circumstances within the dataset.

In the context of skin cancer classification, there are two categories, 'benign' and 'malignant', which are labeled numerically [0, 1]. These numeric identifiers are used to distinguish between benign (non-cancerous) and malignant (cancerous) skin lesions.

3.4. Feature Extraction Technique

CNNs have become the prevailing method for performing tasks such as feature extraction, segmentation, and classification in the field of image processing. In this research work, some of the most effective and high-performing CNN variants (DenseNet-121, ResNet101, and VGG16) were utilized. These models were already trained on the ImageNet dataset. These models were fine-tuned on our dataset using previous weights; this technique is called transfer learning.

Transfer learning focuses on the idea of preserving the acquired knowledge from one problem and leveraging it for solving distinct yet related problems. TL enables us to efficiently employ pretrained deep learning models that have been trained on extensive and publicly accessible datasets. We used pretrained VGG16, DenseNet-121, and ResNet101 models, loaded using TensorFlow and Keras. We configured these models for our approach by excluding their fully connected layers, rendering them suitable for feature extraction. These extracted high-level features are used as valuable inputs for subsequent classification tasks using base-learner models.

3.5. Fusion Technique

The process of combining several feature vectors produced from diverse methodologies in disciplines such as computer vision and other ML applications is known as feature fusion. Previously extracted deep features from three different pretrained models (DenseNet121, ResNet101, and VGG16) were used in this context. We used these distinct feature vectors and their combinations to generate a feature vector. Different base-learner models were trained on this vector. Finally, the meta-learner model was trained on the fused feature vector, made up of a concatenated feature vector derived from the predictions of five base-learner models as well as ensemble validation.

3.6. Classification Models

In this research work, we employed five foundational ML classification algorithms (RF, DT, NB, LR, and SVM) as base-learner models. Afterwards, we used a stacking technique at level 1 which combined all five of these algorithms. Ensemble validation was used at level 2 for the stacking of meta-learners. This implementation was carried out using the Sklearn library.

3.7. Stacked Ensemble Learning

The ensemble approach to ML uses several homogeneous or heterogeneous algorithms for classification, known as base learners, which work together to create a classification model that achieves superior performance. The fundamental concept is the idea that the meta-learner produces its final prediction depending on the base-learner models. The meta-learner is an algorithm that learns to minimize the loss of base-learner models. The prediction output of base learners is used as input to the meta-learner as depicted in Figure 1.

In this research work, we used a first-level stacking strategy for the base-learner phase using five ML algorithms. Furthermore, we used second-level stacking in the meta-learner phase. Previous works solely used base-learner predictions to direct the meta-learner's final decision. In this work, we combined the predictions of the base learners with ensemble validation data to serve as input for the meta-learner. All five base-learner algorithms were used and evaluated for the meta-learner to make a final prediction. In this analysis, we excluded both boosting and bagging techniques. Boosting is not feasible to apply for image classification because of the dramatic increase in training hours and the complexity of the model, which makes it harder to interpret. According to [21], stacking is a highly effective technique for image classification that can lead to a substantial improvement in performance. Additionally, cross-validation-based bagging has also demonstrated a significant enhancement in performance in some research works, closely competing with stacking. This suggests that stacking is the preferred approach, with bagging being a strong contender.

Another research work [29] verified the superiority of the stacking method as the optimal choice for achieving superior ensemble-based results. Therefore, in this work, we utilized stacking while paying close attention to high-performing strategies, aligning with the findings of both studies and ensuring the originality in this work.

3.8. Performance Evaluation Metrics

In this research, we employed various performance evaluation metrics to assess the effectiveness of the models. These metrics included accuracy (*Ac*), precision (*Pr*), recall (*Re*), and the *F*1-*score*. Accuracy describes the overall reliability of the model's performance by computing the ratio of correctly classified class labels to the overall number of data points in the dataset, shown in Equation (1). Precision, as defined in Equation (2), represents the positive predictive value, which represents the fraction of true positives divided by the overall number of actual true instances. Conversely, recall, which is also referred to as sensitivity, is expressed in Equation (3); it measures the fraction of true positives relative to the total number of predicted true instances. Finally, Equation (4) outlines the formula for the *F*1-*score*, a metric that strikes a balance between precision and recall by computing their harmonic mean. We used the Scikit-learn library for evaluation purposes.

$$Ac = \frac{TruePositive + TrueNegative}{TruePositive + FalsePositive + FalseNegative + TrueNegative} \tag{1}$$

$$Pr = \frac{TruePositive}{TruePositive + FalsePositive} \tag{2}$$

$$Re = \frac{TruePositive}{TruePositive + FalseNegative} \tag{3}$$

$$F1\text{-}score = 2 \cdot \frac{Pr \cdot Re}{Pr + Re} \tag{4}$$

4. Results and Analysis

In this section, we present a comprehensive analysis of the experimental outcomes achieved through the application of the proposed scheme on two distinct datasets: SARS CoV-2 CT scans and the ISIC dataset. The proposed methodology comprises two pivotal stages: the base-learner and the meta-learner stages. Across both datasets, we conducted training and evaluation processes on five distinct algorithms, namely, RF, DT, LR, SVM, and NB, within both the base-learner and meta-learner phases with different feature extraction models.

The initial step in the proposed method involves the construction of a feature space. This is accomplished by using three distinct pretrained deep learning models: DenseNet-121, ResNet-101, and VGG16. Additionally, we explored the combinations of these models, namely, DenseNet-121 and ResNet-101; DenseNet-121 and VGG16; ResNet-101 and VGG16; and the combination of all three models, DenseNet-121, ResNet-101, and VGG16. We performed feature extraction using these various pretrained models and combinations of these to assess their impact on classification accuracy. The results of these evaluations, specifically the accuracy of the five classification algorithms, are visually represented in Tables 1 and 2 for COVID-19 and in Tables 3 and 4 for skin cancer detection, respectively.

Table 1. Accuracy of five classification algorithms for different combinations of pretrained learning feature extraction models and base-learner prediction performance results on COVID-19 dataset. **Bold represents the best-performing configuration.**

Models	VGG16	DENSENET121	RESNET101	DENSENET121 + RESNET101	DENSENET121 + VGG16	RESNET101 + VGG16	DENSENET121 + RESNET101 + VGG16
SVM	95.63	91.72	95.40	92.64	94.25	91.49	94.48
LR	**96.78**	94.71	96.09	95.40	96.55	94.48	95.86
DT	83.91	87.13	85.06	88.74	82.53	86.44	81.15
RF	92.18	91.03	92.18	91.03	91.03	91.95	89.89
NB	91.26	84.83	90.34	90.57	91.26	90.34	91.03

Table 2. Accuracy of five classification algorithms for different combinations of pretrained deep learning feature extraction models and meta-learner prediction performance results on COVID-19 dataset. **Bold represents the best-performing configuration.**

Models	VGG16	DENSENET121	RESNET101	DENSENET121 + RESNET101	DENSENET121 + VGG16	RESNET101 + VGG16	DENSENET121 + RESNET101 + VGG16
SVM	97.01	95.17	**97.24**	94.94	96.78	96.78	96.32
LR	97.01	94.71	96.09	95.40	96.55	96.09	96.32
DT	94.02	91.03	95.17	89.43	94.25	90.57	91.49
RF	96.55	93.56	96.78	94.02	96.55	94.48	93.79
NB	96.05	92.64	93.33	90.57	93.33	90.34	95.63

Table 3. Accuracy of five classification algorithms for different combinations of pretrained deep learning feature extraction models and base-learner prediction performance results on ISIC dataset. **Bold represents the best-performing configuration.**

Models	VGG16	DENSENET121	RESNET101	DENSENET121 + RESNET101	DENSENET121 + VGG16	RESNET101 + VGG16	DENSENET121 + RESNET101 + VGG16
SVM	83.40	82.39	86.64	81.17	83.60	82.19	80.16
LR	82.19	83.00	**87.85**	84.82	80.77	84.41	81.98
DT	70.04	75.91	71.05	69.84	71.26	72.27	76.32
RF	80.97	81.98	84.21	82.79	81.17	83.81	78.95
NB	79.35	77.94	79.51	78.74	78.14	78.14	76.52

Table 4. Accuracy of five classification algorithms for different combinations of pretrained deep learning feature extraction models and meta-learner prediction performance results on ISIC dataset. **Bold represents the best-performing configuration.**

Models	VGG16	DENSENET121	RESNET101	DENSENET121 + RESNET101	DENSENET121 + VGG16	RESNET101 + VGG16	DENSENET121 + RESNET101 + VGG16
SVM	84.50	83.28	86.79	82.37	83.60	84.19	80.85
LR	82.37	83.59	**87.89**	85.72	82.37	83.59	81.99
DT	77.81	76.90	82.67	79.94	76.90	79.94	77.81
RF	80.99	81.98	84.27	83.37	81.72	83.91	78.98
NB	54.71	54.71	54.71	54.71	54.71	54.71	54.71

Table 1 demonstrates the effectiveness of the base-learner models in detecting COVID-19 using the SARS-CoV-2 CT scan database. First, we examined the influence of both the individual pretrained models and their combined usage for feature extraction on the performance of all five classification algorithms. Thus, it can be observed that the individual VGG16 and combination of VGG16 with DenseNet121 yield superior performance on LR compared to the other pretrained models. LR exhibits superior performance relative to other ML algorithms during the base-learner stage. Accuracies of 95.63%, **96.78%**, 83.91%, 92.18%, and 91.26% are observed in the base-learner stage from SVM, **LR (Best)**, DT, RF, and NB using VGG16. During the base-learner phase with the DenseNet121 model, the recorded accuracies for different classifiers are 91.72%, 94.71%, 87.13%, 91.03%, and 84.83% for SVM, LR, DT, RF, and NB, respectively. In the initial training phase with the ResNet101 model, results are 95.40%, 96.09%, 85.06%, 92.18%, and 90.34% for SVM, LR, DT, RF, and NB. Moreover, the performance of all pretrained models and their combination in the base-learner stage is also an excellent outcome; the LR scores were the highest in all the implemented pretrained model feature extraction techniques.

Certainly, in summary, LR consistently yielded high accuracy on applied pretrained models in base-learner stage. However, when further analyzed, it was observed that LR achieved its **highest accuracy, 96.78%**, for VGG16 as compared to other feature extraction methods; the second highest was a 96.55% accuracy on a specific combination of pretrained models, that is, DenseNet-121 and VGG-16.

Table 2 illustrates the effectiveness of the meta-learner models for detecting COVID-19 using base-learner prediction and ensemble validation as input. As Table 2 shows, the algorithm's performance for all cases in the meta-learner phase is better than in the base-learner phase. This implies that using the stacking ensemble method is a promising technique to improve classification accuracy. Moreover, the performance of all pretrained models and their combination in the meta-learner stage is also an excellent outcome. SVM and LR scores the highest in all the implemented pretrained model feature extraction techniques.

In summary, SVM consistently yielded high accuracy on applied pretrained models in the meta-learner phase. However, it can also be observed that SVM achieved its **highest accuracy, 97.24%**, for ResNet-101 as compared to other feature extraction methods.

Figure 7 illustrates the evolution of the accuracy of ML algorithms in each feature extraction method from the base-learner to the meta-learner stage.

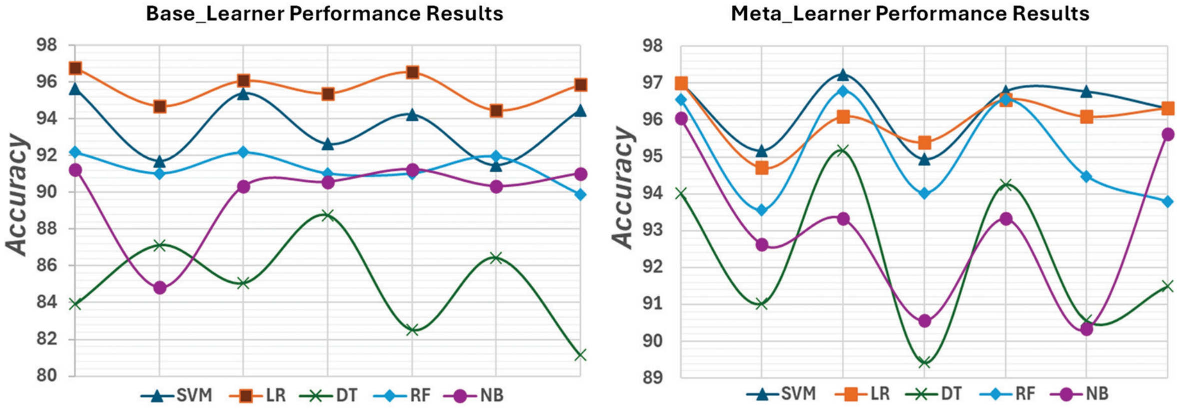


Figure 7. Graphical description of the performance results for all implemented ML algorithms using the SARS-CoV-2 CT scan.

The same experiment was carried out utilizing the ISIC Archive dataset to identify skin cancer cases at the base-learner level. Results of this experiment are listed in Table 3. In the case of the base-learner model, LR again demonstrated superior performance compared to other combinations of models, similar to the SARS-CoV-2 CT scan dataset; furthermore, SVM also excelled in the ISIC Archive dataset. LR tends to perform better when there are fewer noisy variables compared to explanatory factors or when the number of noisy variables is equal to or lower than the number of explanatory factors. SVM's strength lies in its ability to handle high-dimensional data effectively. The main objective in utilizing two distinct datasets to assess this methodology was to evaluate how the algorithms perform in the same scenarios. Upon closer examination of base-learner models, it became evident that **LR** and SVM achieved their highest accuracy rates, reaching **87.85%** and 86.64%, respectively, when applied to a particular pretrained model, namely, ResNet-101. This performance surpasses other feature extraction techniques and classification algorithms.

Table 4 illustrates the effectiveness of the meta-learner models in detecting skin cancer and shows that the algorithm's performance in the meta-learner phase is better than in the base-learner phase for every utilized algorithm except NB. This implies that using the two-level stacking method is a promising technique to improve classification accuracy, but the meta-learning stage is unsuitable for the NB algorithm. Moreover, SVM and LR score the highest in all the implemented pretrained model feature extraction techniques. It can be observed that LR achieved its highest accuracy, **87.89%**, for ResNet-101 as compared to other feature extraction methods.

Figure 8 illustrates the evolution of accuracy among ML algorithms at each feature extraction, from the base learner to the meta-learner, for the ISIC dataset. It demonstrates that not all ML algorithms exhibit improvement in accuracy. For instance, the accuracy of the NB algorithm decreases as it progresses from the base-learner to the meta-learner stage because of its simple probabilistic approach that assumes independence between features. As the algorithm progresses to more advanced stages like meta-learning, where it may encounter more complex feature interactions, its accuracy further decreases.

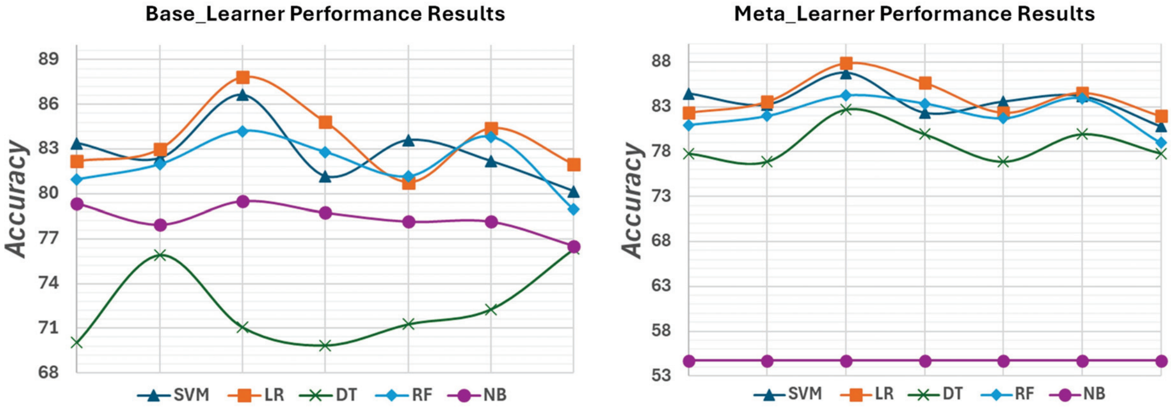


Figure 8. Graphical description of the performance results for all implemented ML algorithms using the ISIC archive dataset.

This experiment followed the same technique as the previous one on the SARS-CoV-2 CT scan dataset. However, in contrast to the outcomes observed on the ISIC dataset, it is important to note that the LR algorithm did not consistently outperform the other five algorithms during the base-learner phase. This highlights the fact that there is not a single classifier that consistently excels in all scenarios or across various datasets. The varying performance of different algorithms across datasets can be attributed to the specific traits of the algorithms employed and the inherent characteristics of the datasets themselves.

As per Figure 9, when rigorously examined, it shows that in COVID-19 and skin cancer databases, ResNet101 outperformed all models for feature extraction. Likewise, in the field of COVID-19 classification, ResNet101 in conjunction with the SVM algorithm obtained the best accuracy over the other ML algorithms. However, in skin cancer classification, ResNet101 embedding LR showed the best accuracy.

The total training time for the complete analysis can be visualized from the following distribution chart: Experiments relevant to SARS CoV-2 CT scans took a total of 16 hours. Experiments relevant to ISIC took less than 12 h. It has to be noted that the stacking techniques with the ML algorithm and pretrained model do not require extensive additional training time. DenseNet-121 and ResNet-101 revealed a high training time across all pretrained models of 6 hours and 53 min for the SARS CoV-2 CT scans dataset, whereas the VGG16 model had the lowest training time across the SARS CoV-2 CT scans dataset at 16 min. For the skin cancer dataset, the same as in COVID-19 detection, DenseNet-121 and ResNet-101 revealed a high training time across all pretrained models of 9 h and 23 min, whereas the VGG16 model had the lowest training time of 30 min. Further details on training times for all feature extraction models with distributive datasets are found in Figure 10.

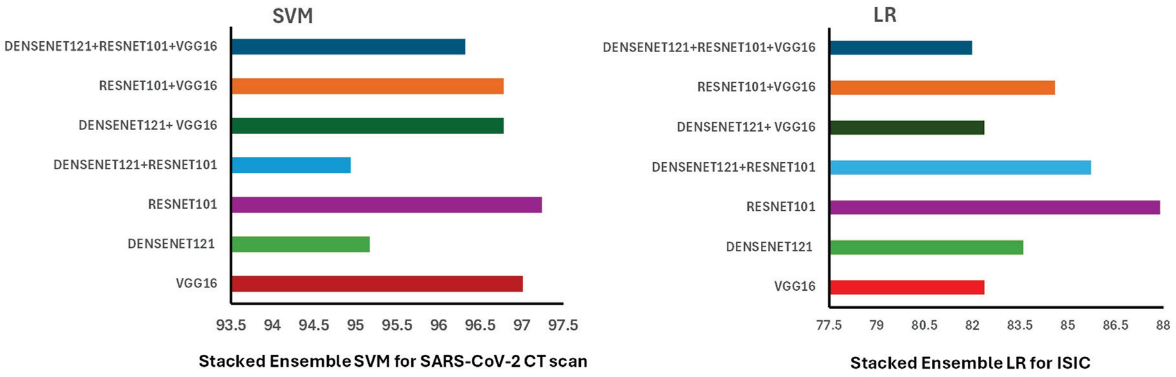


Figure 9. Comparative analysis of feature extraction models' performance in COVID-19 and skin cancer datasets.

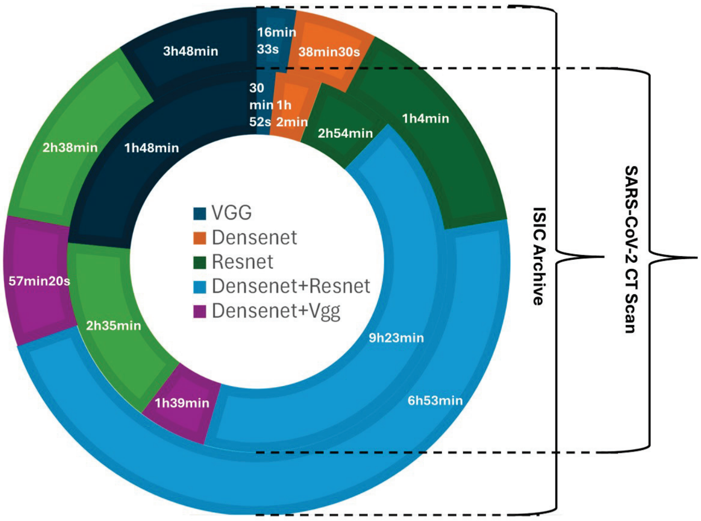


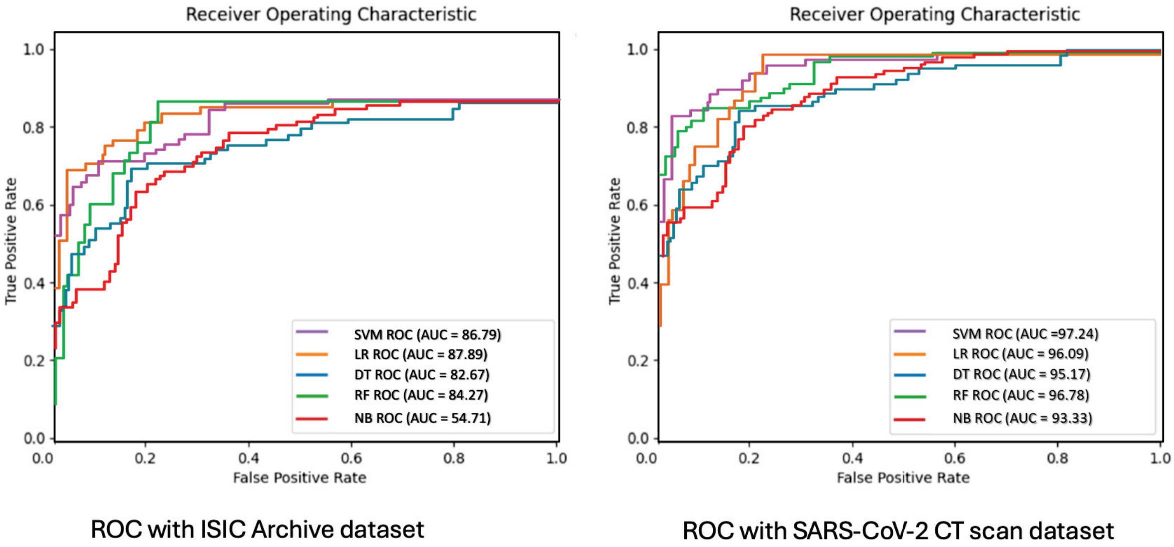
Figure 10. Training times for all feature extraction models with datasets of COVID-19 and skin cancer.

Table 5 shows a detailed comparison of all evaluation matrices (accuracy, precision, recall and F1-score) of the final combination of the proposed scheme on the COVID-19 (SARS-CoV-2 CT) and cancer (ISIC) datasets. As mentioned earlier, ResNet101 outperformed all other pretrained models in the task of feature extraction. Furthermore, ResNet101 in conjunction with the SVM algorithm obtained the best accuracy in the field of COVID-19. However, in the case of skin cancer classification, ResNet101 showed the best accuracy when combined with the linear regression model.

Table 5. Comparison of best-performing feature extractor and final output of meta-learner models on COVID-19 and ISIC datasets. **Bold represents the best-performing configuration.**

Base Model	SARS-CoV-2 CT Scan Database				ISIC-Archive Dataset			
	ResNet101				ResNet101			
	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score
SVM	**97.24**	**97.78**	**96.89**	**97.55**	86.79	87.45	86.99	86.23
LR	96.09	96.67	95.84	96.27	**87.89**	**88.56**	**87.29**	**87.92**
DT	95.17	96.45	95.23	95.05	82.67	82.88	82.68	82.79
RF	96.78	97.01	96.55	96.55	84.27	84.98	84.43	84.99
NB	93.33	93.99	92.56	93.23	54.71	53.21	54.27	54.23

Figure 11 shows the receiver operating characteristic (ROC)–area under curve (AUC) classification metric. It is an excellent metric to observe the performance of a model. If the AUC is near to 1 or 100 percent, then it means that the model has a good measure of separability. A poor model shows an AUC near 0. We can also verify from this graph that for the ISIC dataset, the AUC of the naive Bayes (NB) algorithm is around 0.5 (54.71 percent), wich is very poor compared to other models. As mentioned earlier, in the case of progression to more advanced stages like meta-learning, where more complex features are required to be distinguished or identified, the accuracy of NB shows a significant decrease.



ROC with ISIC Archive dataset

ROC with SARS-CoV-2 CT scan dataset

Figure 11. ROC-AUC classification evaluation metric of final meta-learners for COVID-19 and ISIC datasets.

5. Discussion

The primary objective of the current study is to develop a comprehensive model capable of delivering precise and accurate predictions for both grayscale and RGB images within medical datasets. The aims are to achieve this using an ensemble learning approach and to assess the model's performance in detecting both COVID-19 and skin cancer diseases. In pursuit of this goal, the proposed research enhanced both the feature extraction and classification components, which are pivotal aspects in the field of medical image processing.

This approach involves a fused feature extraction technique, combining features extracted from three different pretrained feature extractor methods with various combinations. Additionally, this study introduces a unique stacked ensemble classification method that incorporates the original feature maps, ensemble validation data, and base-learner predictions as inputs for the meta-learner. The experimental results demonstrate that this method achieves the highest performance levels.

In both datasets, training and evaluation procedures used five different algorithms: RF, DT, LR, SVM, and NB. These algorithms were applied in both the base-learner and meta-learner phases, employing various feature extraction models. This work conducts feature extraction using a variety of pretrained models and combinations to evaluate how they influence classification accuracy. In the context of the SARS-CoV-2 CT scan dataset, this study investigates the impact of using individual pretrained models and their combined application in feature extraction on the performance of all five classification algorithms. Consequently, it becomes evident that when compared to other pretrained models, ResNet-101 exhibits superior performance in the context of SVM. In this study, SVM showcased exceptional performance when applied to the SARS-CoV-2 CT scan dataset, whereas LR and SVM demonstrated outstanding results when dealing with the ISIC

Archive dataset. Certainly, it is evident that LR achieved the highest accuracy rates with ResNet-101, respectively.

6. Conclusions

This research introduces a feature extraction and classification model designed to optimize the accurate detection of COVID-19 and skin cancer within the context of image datasets. This approach for feature extraction techniques harnesses deep learning features obtained from pretrained models. Subsequently, the resulting fused feature vector is integrated into the stacked ensemble approach, particularly when utilizing combinations of two or three pretrained models. Initially, base-learner predictions are made, followed by concatenation of the original feature map and ensemble validation data, ultimately feeding into the meta-learner stage for the final prediction.

The learner was trained and tested with a pair of datasets: the SARS-CoV-2 CT Scan and the ISIC Archive. Employing a stacked method, the SVM technique yielded the highest classification accuracy for the SARS-CoV-2 CT Scan dataset, whereas LR approaches outperformed others for the ISIC Archive dataset. These results were achieved when both datasets yielded feature extraction using ResNet-101.

Future Work

This research presented a stacked ensemble methodology for the classification of COVID-19 and skin cancer disease. Certain limitations in this study will be subject to future improvements. Here are the constraints in this work:

- This work exclusively underwent testing on COVID-19 and skin cancer datasets. Therefore, in order to extend its applicability to further datasets, additional research and investigation are required.
- This study approach relied on five established ML models to construct this stacked ensemble model. Additionally, we utilized pretrained models with a fixed input size of 224 × 224 for feature extraction. Nevertheless, in future research, there is potential to broaden the scope of this method by adapting it to various pretrained CNN architectures that may have different input sizes.

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

Funding: This research has been partially supported by the Project of National Relevance "Innovative mathematical modelling for cell mechanics: global approach from micro-scale models to experimental validation integrated by reinforcement learning", financed by European Union—Next-GenerationEU—National Recovery and Resilience Plan—NRRP-M4C2-I 1.1, CALL PRIN 2022 PNRR D.D. 1409 14-09-2022—(Project code P2022MXCJ2, CUP D53D23018940001) granted by the Italian MUR.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Conflicts of Interest: Authors declare no conflict of interest.

References

1. Kaliraman, B.; Duhan, M. A new hybrid approach for feature extraction and selection of electroencephalogram signals in case of person recognition. *J. Reliab. Intell. Environ.* **2021**, *7*, 241–251. [CrossRef]
2. Umair, A.; Masciari, E.; Ullah, M.H. Vaccine sentiment analysis using BERT + NBSVM and geo-spatial approaches. *J. Supercomput.* **2023**, *79*, 17355–17385. [CrossRef]

3. Hovorushchenko, T.; Moskalenko, A.; Osyadlyi, V. Methods of medical data management based on blockchain technologies. *J. Reliab. Intell. Environ.* **2023**, *9*, 5–16. [CrossRef] [PubMed]
4. Umair, A.; Masciari, E. Sentimental and spatial analysis of covid-19 vaccines tweets. *J. Intell. Inf. Syst.* **2023**, *60*, 1–21. [CrossRef] [PubMed]
5. Naeem, M.; Coronato, A. An AI-empowered home-infrastructure to minimize medication errors. *J. Sens. Actuator Netw.* **2022**, *11*, 13. [CrossRef]
6. Naeem, M.; Coronato, A.; Paragliola, G. Adaptive treatment assisting system for patients using machine learning. In Proceedings of the 2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS), Granada, Spain, 22–25 October 2019; IEEE: New York, NY, USA, 2019; pp. 460–465.
7. Haseeb, A.; Khan, M.A.; Shehzad, F.; Alhaisoni, M.; Khan, J.A.; Kim, T.; Cha, J. Knee Osteoarthritis Classification Using X-Ray Images Based on Optimal Deep Neural Network. *Comput. Syst. Sci. Eng.* **2023**, *47*, 2397–2415. [CrossRef]
8. Shah, S.I.H.; Coronato, A.; Naeem, M.; De Pietro, G. Learning and assessing optimal dynamic treatment regimes through cooperative imitation learning. *IEEE Access* **2022**, *10*, 78148–78158. [CrossRef]
9. Gheisari, M.; Taami, T.; Ghaderzadeh, M.; Li, H.; Sadeghsalehi, H.; Sadeghsalehi, H.; Abbasi, A.A. Mobile applications in COVID-19 detection and diagnosis: An efficient tool to control the future pandemic; a multidimensional systematic review of the state of the art. *JMIR MHealth UHealth* **2023**, *12*, e44406.
10. Fiorino, M.; Naeem, M.; Ciampi, M.; Coronato, A. Defining a Metric-Driven Approach for Learning Hazardous Situations. *Technologies* **2024**, *12*, 103. [CrossRef]
11. Ghaderzadeh, M.; Asadi, F.; Ramezan Ghorbani, N.; Almasi, S.; Taami, T. Toward artificial intelligence (AI) applications in the determination of COVID-19 infection severity: Considering AI as a disease control strategy in future pandemics. *Iran. J. Blood Cancer* **2023**, *15*, 93–111. [CrossRef]
12. Coronato, A.; Naeem, M. A reinforcement learning based intelligent system for the healthcare treatment assistance of patients with disabilities. In Proceedings of the International Symposium on Pervasive Systems, Algorithms and Networks, Naples, Italy, 16–20 September 2019; Springer: Cham, Switzerland, 2019; pp. 15–28.
13. Xue, D.; Zhou, X.; Li, C.; Yao, Y.; Rahaman, M.M.; Zhang, J.; Chen, H.; Zhang, J.; Qi, S.; Sun, H. An Application of Transfer Learning and Ensemble Learning Techniques for Cervical Histopathology Image Classification. *IEEE Access* **2020**, *8*, 104603–104618. [CrossRef]
14. Ganaie, M.A.; Hu, M.; Malik, A.; Tanveer, M.; Suganthan, P. Ensemble deep learning: A review. *Eng. Appl. Artif. Intell.* **2022**, *115*, 105151. [CrossRef]
15. Shekar, B.; Hailu, H. An efficient stacked ensemble model for the detection of COVID-19 and skin cancer using fused feature of transfer learning and handcrafted methods. *Comput. Methods Biomech. Biomed. Eng. Imaging Vis.* **2023**, *11*, 878–894. [CrossRef]
16. Rajagopal, R. Comparative analysis of COVID-19 X-ray images classification using convolutional neural network, transfer learning, and machine learning classifiers using deep features. *Pattern Recognit. Image Anal.* **2021**, *31*, 313–322. [CrossRef]
17. Shaik, N.S.; Cherukuri, T.K. Transfer learning based novel ensemble classifier for COVID-19 detection from chest CT-scans. *Comput. Biol. Med.* **2022**, *141*, 105127. [CrossRef]
18. Grignaffini, F.; Barbuto, F.; Piazzo, L.; Troiano, M.; Simeoni, P.; Mangini, F.; Pellacani, G.; Cantisani, C.; Frezza, F. Machine Learning Approaches for Skin Cancer Classification from Dermoscopic Images: A Systematic Review. *Algorithms* **2022**, *15*, 438. [CrossRef]
19. Tembhurne, J.V.; Hebbar, N.; Patil, H.Y.; Diwan, T. Skin cancer detection using ensemble of machine learning and deep learning techniques. *Multimed. Tools Appl.* **2023**, *82*, 1–24. [CrossRef]
20. Rahman, Z.; Hossain, M.S.; Islam, M.R.; Hasan, M.M.; Hridhee, R.A. An approach for multiclass skin lesion classification based on ensemble learning. *Inform. Med. Unlocked* **2021**, *25*, 100659. [CrossRef]
21. Müller, D.; Soto-Rey, I.; Kramer, F. An analysis on ensemble learning optimized medical image classification with deep convolutional neural networks. *IEEE Access* **2022**, *10*, 66467–66480. [CrossRef]
22. Shinkai, K.; Bruckner, A.L. Dermatology and COVID-19. *JAMA* **2020**, *324*, 1133–1134. [CrossRef]
23. Goyal, M.; Pandey, M. Ensemble-based data modeling for the prediction of energy consumption in hvac plants. *J. Reliab. Intell. Environ.* **2021**, *7*, 49–64. [CrossRef]
24. Rincy, T.N.; Gupta, R. Ensemble learning techniques and its efficiency in machine learning: A survey. In Proceedings of the 2nd International Conference on Data, Engineering and Applications (IDEA), Bhopal, India, 28–29 February 2020; IEEE: New York, NY, USA, 2020; pp. 1–6.
25. Witten, I.H.; Frank, E.; Hall, M.A.; Pal, C.J.; Data, M. Practical machine learning tools and techniques. In *Proceedings of the Data Mining*; Elsevier: Amsterdam, The Netherlands, 2005; Volume 2, pp. 403–413.
26. Chowdhury, M.E.; Rahman, T.; Khandakar, A.; Mazhar, R.; Kadir, M.A.; Mahbub, Z.B.; Islam, K.R.; Khan, M.S.; Iqbal, A.; Al Emadi, N.; et al. Can AI help in screening viral and COVID-19 pneumonia? *IEEE Access* **2020**, *8*, 132665–132676. [CrossRef]
27. Fanconi, C. Skin Cancer: Malignant vs. Benign-Processed Skin Cancer Pictures of the ISIC Archive. 2019. Available online: https://www.kaggle.com/datasets/fanconic/skin-cancer-malignant-vs-benign (accessed on 15 June 2024).

28. Silva, P.; Luz, E.; Silva, G.; Moreira, G.; Silva, R.; Lucio, D.; Menotti, D. COVID-19 detection in CT images with deep learning: A voting-based scheme and cross-datasets analysis. *Inform. Med. Unlocked* **2020**, *20*, 100427. [CrossRef]
29. Mahajan, P.; Uddin, S.; Hajati, F.; Moni, M.A. Ensemble Learning for Disease Prediction: A Review. *Healthcare* **2023**, *11*, 1808. [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.

technologies

MDPI

Article

Real-Time Machine Learning for Accurate Mexican Sign Language Identification: A Distal Phalanges Approach

Gerardo García-Gil, Gabriela del Carmen López-Armas *, Juan Jaime Sánchez-Escobar *, Bryan Armando Salazar-Torres and Alma Nayeli Rodríguez-Vázquez

Technical Industrial Teaching Center, Department of Investigation, Software Design and Development/Biomedical, Nueva Escocia Street 1885, Guadalajara CP 44638, Jalisco, Mexico; ggarcia@ceti.mx (G.G.-G.); nrodriguez@ceti.mx (A.N.R.-V.)
* Correspondence: glopez@ceti.mx (G.d.C.L.-A.); jjsanchez@ceti.mx (J.J.S.-E.)

Abstract: Effective communication is crucial in daily life, and for people with hearing disabilities, sign language is no exception, serving as their primary means of interaction. Various technologies, such as cochlear implants and mobile sign language translation applications, have been explored to enhance communication and improve the quality of life of the deaf community. This article presents a new, innovative method that uses real-time machine learning (ML) to accurately identify Mexican sign language (MSL) and is adaptable to any sign language. Our method is based on analyzing six features that represent the angles between the distal phalanges and the palm, thus eliminating the need for complex image processing. Our ML approach achieves accurate sign language identification in real-time, with an accuracy and F1 score of 99%. These results demonstrate that a simple approach can effectively identify sign language. This advance is significant, as it offers an effective and accessible solution to improve communication for people with hearing impairments. Furthermore, the proposed method has the potential to be implemented in mobile applications and other devices to provide practical support to the deaf community.

Keywords: MediaPipe; OpenCV; decision tree; Gini; machine learning; Mexican sign language

1. Introduction

Sign language recognition (SLR) has become crucial for bridging the communication gap between hearing and deaf people, thus facilitating assistive technologies, primarily through mobile applications [1–3]. In recent decades, advances in computer vision and machine learning (ML) have led to significant progress in SLR [4,5]. At the same time, research efforts have focused on developing technology-based solutions that improve communication and overall quality of life for the hearing-impaired community [6].

This study focuses on Mexican sign language (MSL) recognition, although it can be applied to other sign language, such as American sign language (ASL), using hand-angle analysis [7,8]. A MediaPipe hand skeleton descriptor is used for training, and the set of six angles for each letter is plotted using OpenCV [9–11]. These data are stored in a dataset and used to train a decision tree (DT C4.5) and label the predicted letter. To validate the model, the angles of the hand signals are compared to the training dataset. If a match is found, the model classifies it. Otherwise, it makes a prediction. A schematic of the hand signals of the MSL letters is shown in Figure 1.

This method differs from other ML-based approaches by requiring a relatively modest amount of input data, significantly reducing computational time for accurate MSL interpretation. This demonstrates that classical ML approaches can be more effective for real-time classification problems than some state-of-the-art ML algorithms [12,13]. The main contributions of our method can be summarized as follows:

Technologies **2024**, *12*, 152. https://doi.org/10.3390/technologies12090152

i. only six measurements (angles) are used as feature descriptors;
ii. efficient and accurate prediction;
iii. relatively small training dataset;
iv. high efficiency in letter prediction.

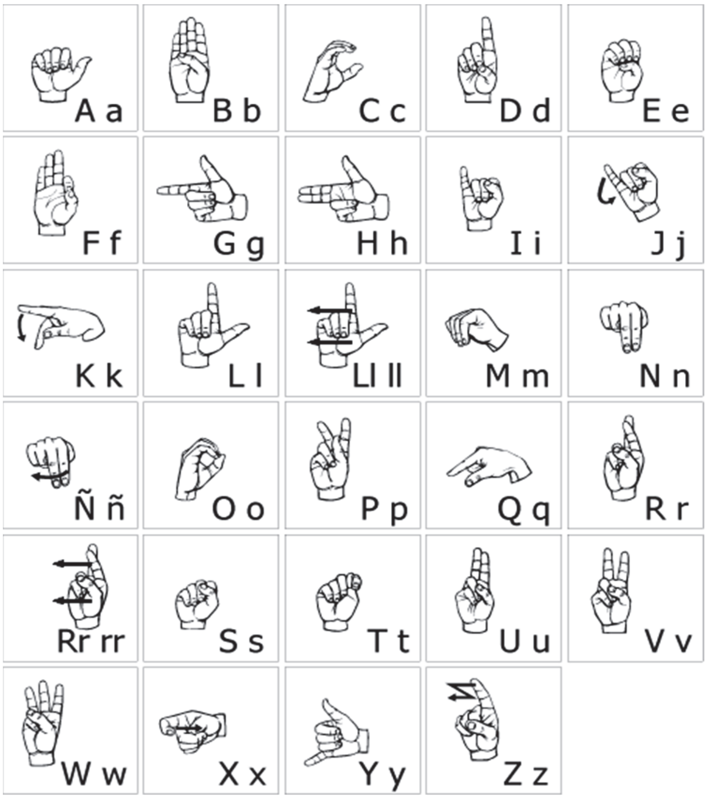


Figure 1. MSL sign language.

The proposed method based on hand-angle analysis achieves significantly higher accuracy and efficiency in MSL recognition than other methods reported in the field, as demonstrated and mentioned below. A relatively small training dataset was used, and feature extraction using six specific hand angles allows for accurate and fast interpretation of sign language gestures, which improves communication and quality of life for the hearing impaired.

The following sections of this paper are organized as follows. Section II discusses the previous related work. Section 3 provides the materials and methods. Section IV presents the results of experimental performance metrics and their analysis. Section 5 presents the discussion, and Section 6 contains the conclusions.

2. Related Work

Sign alphabet recognition is a form of communication that involves images or videos depicting one or two hands. Experts have devoted efforts to predicting letters, words, and ideas in sign languages to assist people with hearing impairments using advanced technologies and artificial intelligence algorithms. Despite the many types of research in this field, there are limitations and opportunities for improving communication between hearing and non-hearing individuals. In this section, we will focus on exploring one-handed SRL. To this end, we present a brief literature review of recent works on SRL using deep-learning techniques, particularly convolutional neural networks (CNNs) and ML based on digital image processing and sensors [14–17].

Among the reviewed works, Ameen et al. (2016) explored the applicability of deep learning for sign language interpretation by developing a CNN to classify images based on manual spelling. They achieved 82% accuracy and 80% recall using intensity and depth data [18]. Thongtawee et al. (2018) presented an efficient feature-extraction method and algorithm to distinguish American sign language (ASL) from static and dynamic gestures, achieving 95% recognition using an artificial neural network (ANN) [19]. Rastgoo et al. (2020) addressed the challenge of real-time SLR using extra spatial hand relation (ESHR) and hand pose (HP) features, a 2D CNN, singular value decomposition (SVD), and long short-term memory (LSTM) with an accuracy of 86.32% [20,21].

Sharma et al. (2020) proposed a systematic statistical analysis and evaluated previously trained deep models for static Indian sign language (ISL) recognition. They achieved 99.0% and 97.6% recognition accuracy for numbers and letters, respectively, using a public ISL dataset [22]. Katoch et al. (2022) reported a technique using the bag-of-visual-words (BOVW) model to identify ISL alphabets and digits in a live video stream, achieving an average accuracy of 89.24% using support vector machine (SVM) and CNN [23–25]. Subramanian et al. (2022) proposed an optimized hand skeleton descriptor integrating the model of closed recurrent unit (MOPGRU) for ISL recognition, achieving an average accuracy of 95% using the bidirectional long short-term memory network (BiLSTM) [26]. Sundar et al. performed a hybrid with long short-term memory (LSTM), achieving an accuracy of 99% [27,28]. Pathan et al. investigated SLR using CNN and an ASL image dataset, achieving a test accuracy of 98.98% [29]. Sanchez et al. investigated a word-level SLR methodology on the Corpus LIBRAS dataset (Brazilian sign language), obtaining an accuracy of 94.33% using BiLSTM [30]. Mohsin et al. focused on letter and number recognition in ASL and achieved 96% accuracy using InceptionV3 [31]. Amangeldy et al. proposed an improved method for continuous recognition of Kazakh sign language, achieving an average accuracy of 97% [32]. Finally, Wali et al. comprehensively reviewed emerging frameworks and algorithms in SLR, identifying state-of-the-art techniques and suggesting new research directions [33].

2.1. The Taxonomy of Sign Language Recognition

Farooq et al. (2021) propose a taxonomy of SLR that includes applications, avatar technology, gesture recognition, natural language to sign language translation, and repositories of written text units, such as letters, words, or sentences. Individual signs in sign language consist of gestures or hand movements, each representing a specific word. They can also be considered static images, where one image corresponds to one word. Signed phrases, or running signs, contain multiple words and are considered sign alphabets or static signs. They are easier to recognize than full sentences. This section reviews several studies in this area [31,34].

Research on sign language translation identifies several possibilities. These efforts aim to improve communication between people with and without hearing impairments. Our proposed approach is highlighted in the white box in Figure 2.

2.2. Machine Learning Models in Sign Languages

Among the ML models similar to ours, MediaPipe was used by Bajaj et al. (2010) to experiment with different combinations of landmarks and classification algorithms, such as K-nearest neighbors (KNN), random forests, and neural networks. After preprocessing to bring the landmarks into a single reference frame, these researchers obtained average accuracies of 82.19% for KNN, 85.30% for random forests, and 90.95% for neural networks, respectively [35–38]. Sahoo et al. (2014) used various classification techniques, such as position and motion-based SLR using MultiStream (HMM), the Boost Map method (BoostMap), and neural networks, for sign language-related gestures. Their approach focused on determining the shape and center of gravity of the hands in images captured by a digital camera. They used landmarks to train signers and achieved satisfactory accuracy [39–41].

In 2021, Shah et al. used SVM multiple-kernel learning classification to recognize static Pakistani sign language alphabets. They extracted visual features from images, such as local binary patterns (LBP), histogram of oriented gradient (HOG), and speeded-up robust features (SURF). They classified them using multiple kernels, achieving an average accuracy of 89.24% [42–44]. Hussain et al. (2022) focused on ASL and Irish sign language (ISL) annotation, using MediaPipe to extract features and classify hand gestures. They achieved a maximum accuracy of 93% and 96.7%, respectively, with different classifiers [45]. Despite the extensive existing research, in this study, we will compare ML methods such as SVM, naive Bayes, KNN, random forests, XGBoost, and LightGBM, in addition to DT C4.5 [46–49].

Unlike previous research, our approach is characterized by its simplicity. It is based on only six features in real-time SLR. The careful feature selection and efficiency of the model support its applicability in practical situations.

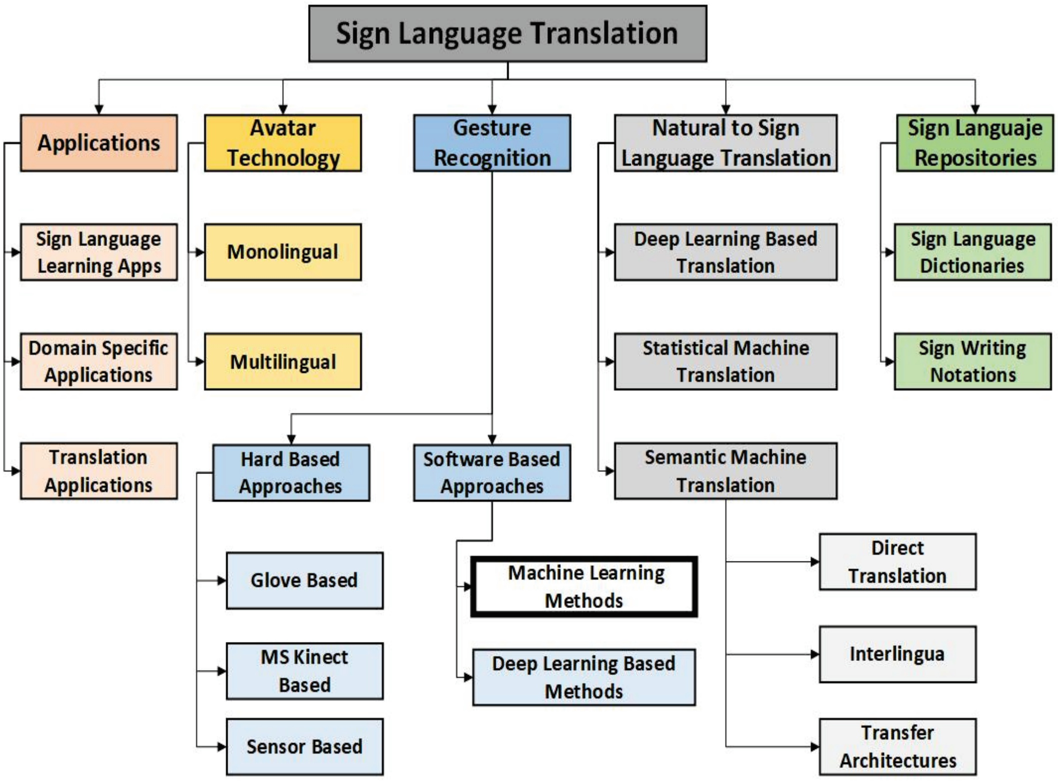

Sign Language Translation
Applications
Avatar Technology
Gesture Recognition
Natural to Sign Language Translation
Sign Languaje Repositories
Sign Language Learning Apps
Monolingual
Deep Learning Based Translation
Sign Language Dictionaries
Domain Specific Applications
Multilingual
Statistical Machine Translation
Sign Writing Notations
Translation Applications
Hard Based Approaches
Software Based Approaches
Semantic Machine Translation
Glove Based
Machine Learning Methods
Direct Translation
MS Kinect Based
Deep Learning Based Methods
Interlingua
Sensor Based
Transfer Architectures

Figure 2. The taxonomy of sign language recognition.

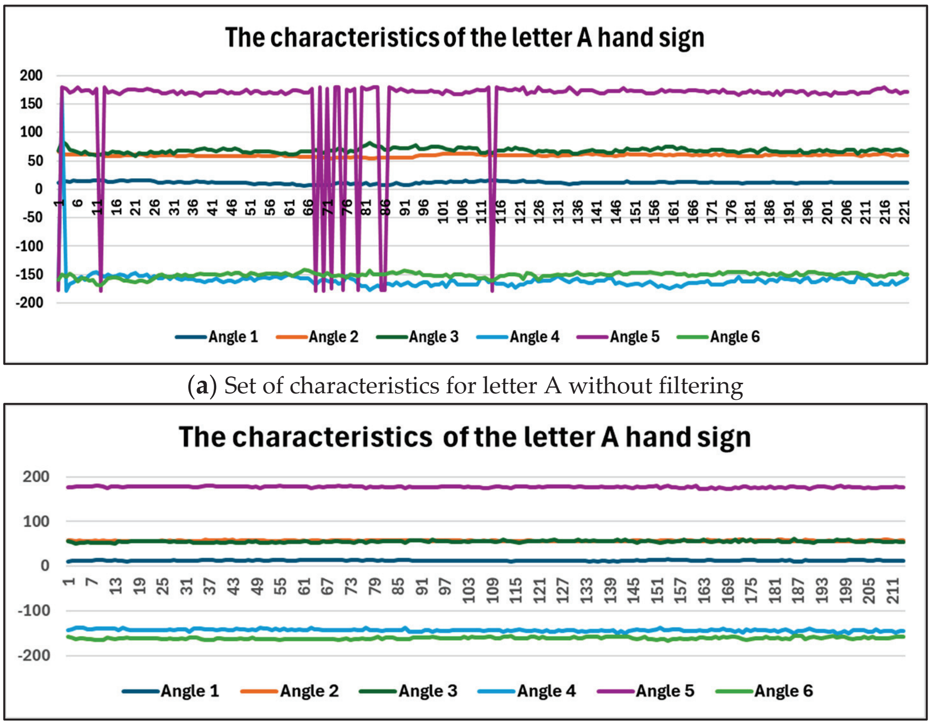
3. Materials and Methods

3.1. Software and Hardware Characteristics

For this work, a CPU with an AMD Ryzen 5 5600 G processor and Radeon graphics at 3.90 GHz, 16 GB of RAM, and a Logitech Model C920 HD Pro 1080p 960-000764 webcam were used. The model was built using the MediaPipe classifier as a descriptor of the hand skeleton, OpenCV to measure the internal angles of the hand, and the Python sklearn library (version 9.11.9) to construct the C4.5 decision-tree classifier [50].

3.2. Data Loading and Dataset Preparation

The process began with collecting a dataset explicitly designed for our task, extracted from a proprietary database stored in an Excel file with 5690 records collected and trained by four people of different genders and ages. Furthermore, the database can be downloaded from https://github.com/gggvamp/pdi/blob/main/datosO.xlsx (accessed on 4 April 2024), since we focus on classification and prediction in the language domain. The last column of the dataset contains the class labels corresponding to the sign language letters, while the previous six columns contain the relevant predictive features of the hand, which can be seen in Figure 3a,b. The training set was used to fit the model letter by letter. We apply essential preprocessing techniques to ensure the quality and adequacy of the data. This process included the identification and removal of outliers, which could bias the results of the analysis, and the normalization and standardization of the characteristics were also carried out. We adjusted the scales of the six features to achieve a standard distribution, which facilitates the comparison of different hand signs (letters) and improves the stability of the analysis models. This is conducted for each letter [51,52]. See Figure 3a,b.



(**a**) Set of characteristics for letter A without filtering

(**b**) Set of filtered letter A characteristics

Figure 3. shows the measurements of the 6 characteristics, including the letter A in this case. The unfiltered data are displayed in (**a**), while the filtered (noise-reduced) data using the procedures mentioned above are shown in (**b**).

We split the data into training and test sets, allocating 20% of the data to the test set and 80% to the training set to ensure adequate representation of the algorithms [53]. Additionally, we used the numerical value of 42 to initialize the random number generator, ensuring the reproducibility of the data splitting. This separation allows for adequate model training with independent data before the evaluation of the test set, avoiding overfitting and providing a more accurate assessment of model performance under real-time and user-independent conditions [54].

Once the data have been collected, the requisite preprocessing is initiated to ensure the quality and suitability of the data for subsequent analysis. This process involves several crucial steps designed to preserve the integrity and consistency of the data. These steps include identifying and eliminating outliers that could skew the analysis results. Additionally, the features are normalized or standardized, adjusting their scales to achieve a standard distribution. This step facilitates the comparison of different variables and improves the stability of the analysis models.

3.3. The Performance Evaluation of Trained Models

The ML algorithms were evaluated using several metrics, including prediction robustness, completeness, sensitivity, specificity, precision, recall, and F1 score (a combination of precision and recall), in addition to time and accuracy in prediction, training, and validation.

Once the models were trained, their performances were evaluated using the test suite, which involved calculating numerous metrics designed to provide a complete and detailed understanding of each model's performance. Among these metrics are confusion matrices, which provide a robust evaluation of each classification method on datasets containing 21 classes (letters).

The confusion matrix is a valuable source of information about the classifier's predictions. To improve clarity, it is common practice to normalize the confusion matrix by converting absolute counts to proportions. This normalization allows for comparing model performance between classes with different sample sizes. Additionally, the matrix is presented as a heat map, where darker shades represent higher values. This visualization helps to identify areas of high confusion, as shown in Table 1, and provides a clear understanding of model performance.

Table 1. Description of a confusion matrix. The confusion matrix evaluates classification accuracy and determines a classifier's overall performance. It defines key concepts, such as precision (P), recall (R), true positives (TP), false positives (FP), false negatives (FN), and true negatives (TN).

		Actual Class		
		Positive	**Negative**	
Predicted Class	**Negative**	False Negative (FN)	True Negative (TN)	**Neg Pre. Value** TN/(TN + FN)
	Positive	True Positive (TP)	False Positive (FP)	**Precision** TP/(TP + FP)
	F1-score (2xPxR)/P + R	**Recall** TP/(TP + FN)	**Specificity** TN/(TN + FP)	**Accuracy** TP + TN/(TP + TN + FP + FN)

3.4. Selection of Algorithms to Compare

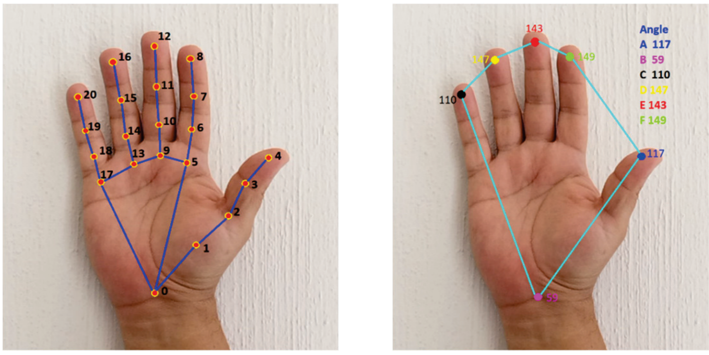
Several ML algorithms have been selected for comparison to assess their performances in addressing the classification and prediction problem. The selected algorithms are:

- **SVM:** data classification is achieved by identifying the optimal separating hyperplane between classes in a multidimensional space;
- **Naive Bayes:** this approach is based on Bayes' theorem and assumes independence between characteristics and the classes;
- **KNN:** data classification is performed by assigning labels based on nearest neighbor labels;
- **Decision trees:** this method classifies data using a decision tree, where each node represents a feature and each leaf a label;
- **Random forests:** combine multiple decision trees to classify data, reducing overfitting;
- **XGBoost:** implements gradient boosting to improve model accuracy using sequential decision trees with regularization and parallelization;
- **LightGBM is another efficient implementation of gradient boosting. it uses** sampling techniques to build decision trees faster and with lower memory usage;
- **CatBoost** is an ML algorithm developed by Yandex for gradient boosting on decision trees. It is particularly effective at handling categorical features directly, preventing overfitting, and offering high performance with both CPU and GPU support;
- **RNNs:** recurrent neural networks are designed for processing sequential data by maintaining a hidden state that captures information from previous time steps. They are widely used in applications like time-series forecasting and natural language processing.

The selected algorithms were implemented at this stage using well-known ML libraries, such as Scikit-learn 1.4.2, XGBoost, and LightGBM. Each model was carefully configured and trained using the training set prepared during the data-preprocessing phase. During the training process, rigorous monitoring was conducted by recording the time required for each model's training. Several evaluation metrics were calculated and thoroughly analyzed, including training accuracy, a crucial measure of each model's predictive ability and fit. It should be noted that all algorithms compared were subjected to the same circumstances and initial conditions. This meticulous approach ensured a thorough understanding of the models' performance and allowed the identification of areas for improvement to optimize their performance in future phases of the project [55,56].

3.5. Proposed Work

This work uses a hand-feature extractor called the MediaPipe Hand Landmarker. This tool identifies key points on the hands in an image. These points can be used to detect significant locations on the hands and apply visual effects to them. See Figure 4a,b.



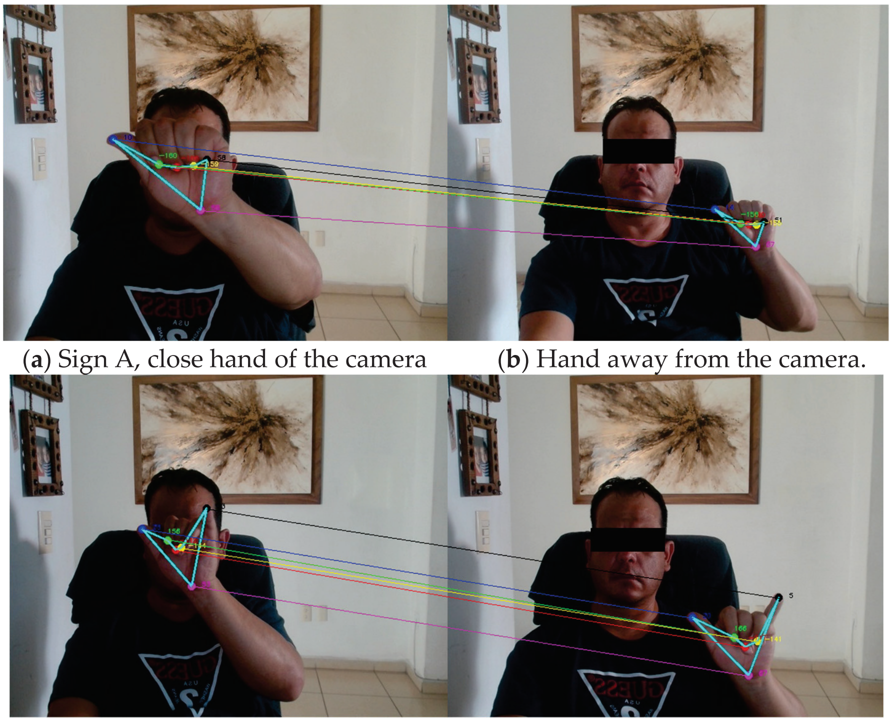
(**a**) Descriptor points (**b**) Reduction to 6 distal angles

Figure 4. The task is effective on static image data and videos. (**a**) Twenty-one landmarks of the hands in image coordinates. (**b**) The angles between the distal phalanges and the palm are classified as internal angles ∝A, ∝B, ∝C, ∝D, ∝E, and ∝F.

Measuring the internal angles between the distal phalanges and the palm is crucial to our methodology. These angles capture distinctive features without relying on correlation and convolution processes. Calculating these angles provides a concise representation of the hand signals and, through OpenCV, stabilizes the plot regardless of the position or distance of the hand from the camera. This stability is vital for improving the accuracy and speed of letter prediction in sign language. The angle is calculated using the direction vectors of the two lines $\vec{\boldsymbol{u}} = (u_1, u_2)$ and $\vec{\boldsymbol{v}} = (v_1, v_2)$, and the angle formed by these two lines can be calculated using Equation (1):

$$\cos\alpha = \left(\frac{\left| \vec{u} \cdot \vec{v} \right|}{\left| \vec{u} \right| \left| \vec{v} \right|} \right) = \frac{|u_1 v_1 + u_2 v_2|}{\sqrt{u_1^2 + u_2^2} + \sqrt{v_1^2 + v_2^2}} \tag{1}$$

where $\left| \vec{\boldsymbol{u}} \right|$ and $\left| \vec{v} \right|$ are the modules of vectors $\boldsymbol{u}$ and $\boldsymbol{v}$, respectively, the angles between the distal phalanges and the palm are obtained. It should be noted that this contribution is particularly significant, since not only are the features reduced from 21 to 6 dimensions but also, by focusing on distal angles, the values of these angles are consistent regardless of the position or distance relative to the camera. This is a significant advantage in processing, classifying, and, above all, predicting MSL letter labels. See Figure 5a–d.



(**a**) Sign A, close hand of the camera (**b**) Hand away from the camera.

(**c**) Y sign, incorrect hand position (**d**) Y sign, correct hand position

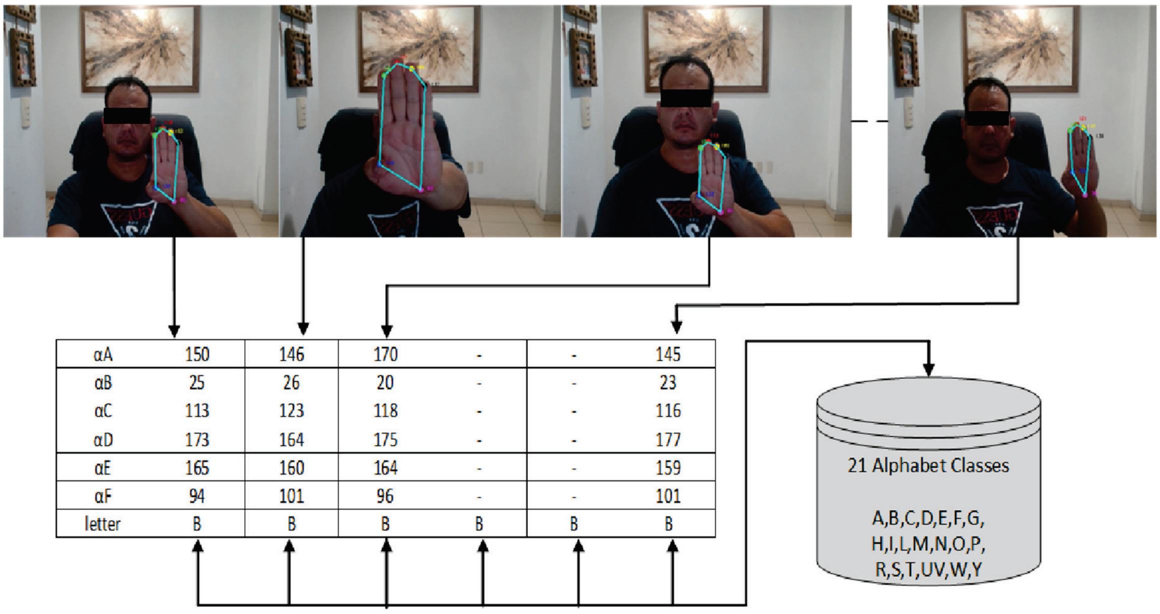
Figure 5. Regardless of the hand's location, the algorithm detects it. (**a**) Training the MSL character 'A' near the camera. (**b**) Training the MSL character 'A' away from the camera. (**c**) Training the character 'Y' in the incorrect position. (**d**) Training the character 'Y' in the correct position. The process begins with two stages, namely the training stage and the validation stage.

3.5.1. Training Stage

During the training stage, real-time video recordings are performed for each letter of the Spanish alphabet using the MSL variant. The training begins by labeling each sign with its corresponding letter, starting with the sign for the letter A and ending with the sign for Y, excluding the letters J, K, Ñ, Q, X, and Z due to their movement nature. The six most critical characteristic points are plotted using Equation (1) to derive and record the six angles, from which 21 of the 27 MSL letters are obtained.

These data are stored in a dataset containing each letter of the corresponding alphabet. For example, in Figure 6, the letter 'B' is trained. Records of the features of each letter are stored in a more extensive dataset called 'alphabet classes', which is accessible in the training and validation phase. See Figure 6.

The training algorithm outlines a system that uses a camera to detect and track hands in real time. The process begins with importing the necessary libraries for image processing and hand detection. Video capture from the camera is set in a continuous loop, where the algorithm captures a frame and verifies the success of the capture. The algorithm then searches for the user's hand within the frame. If a hand is detected, the algorithm analyzes it and calculates the coordinates of the hand landmarks. These coordinates are used to draw lines on the frame, representing the hand. Additionally, the algorithm calculates six angles using the hand landmark coordinates, which are stored. The modified frame with the detected hand and calculated angles is displayed on the screen. The loop continues until the user stops it, at which point the camera resources are released. For a detailed description of this phase, refer to Table 2.



αA	150	146	170	-	-	145
αB	25	26	20	-	-	23
αC	113	123	118	-	-	116
αD	173	164	175	-	-	177
αE	165	160	164	-	-	159
αF	94	101	96	-	-	101
letter	B	B	B	B	B	B

Figure 6. Training and labeling process of the 21 static letters of the MSL.

Table 2. Training and labeling process.

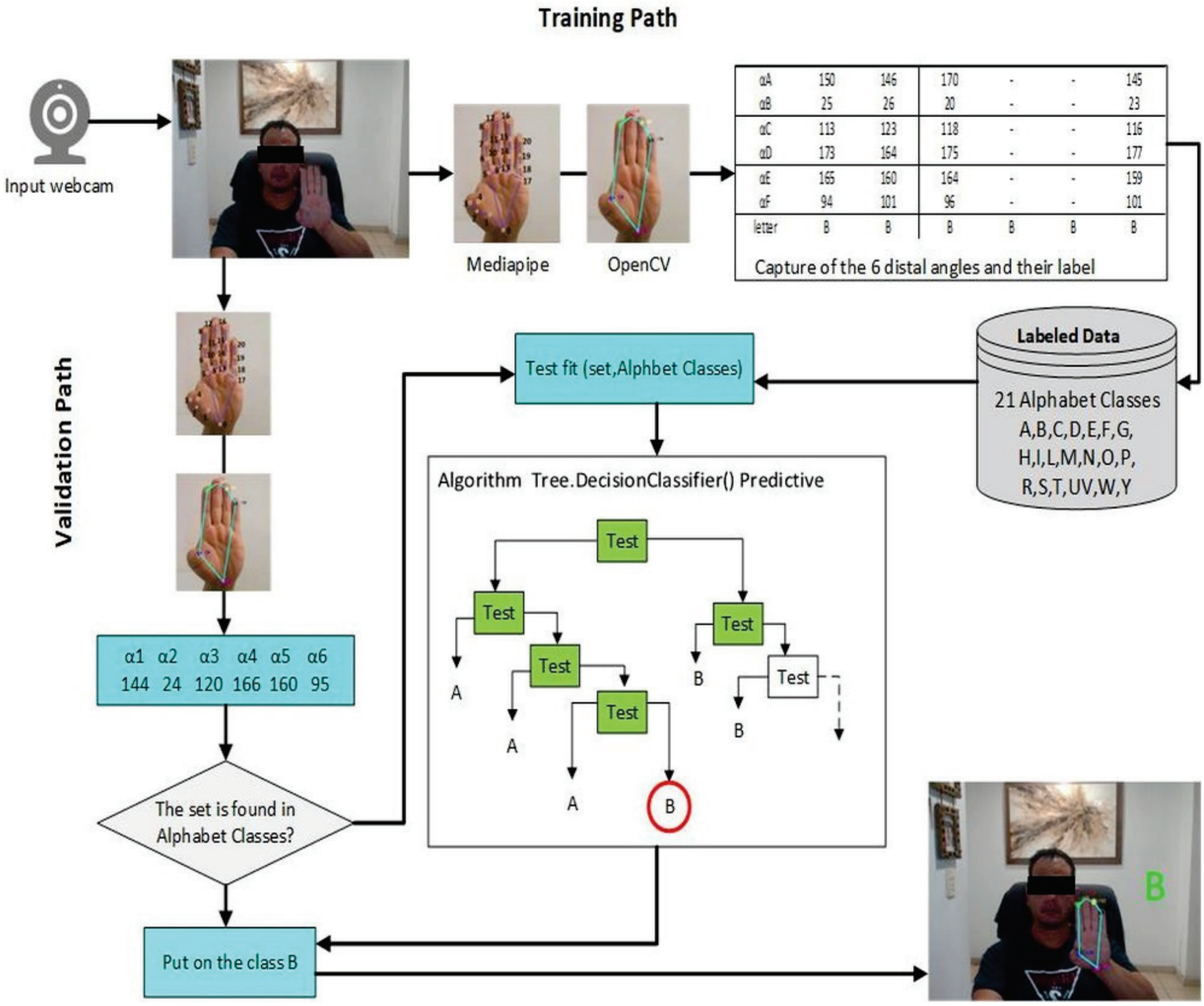
Pseudocode 1 Hand Angle Data Collection and Training

```
Start
Import necessary libraries (cv2, mediapipe, numpy, pandas)
Configure mediapipe for hand detection and capture video from the camera
Write data into an Excel file
Initialize counter to zero
While true:
    Capture a frame from the camera
    If the frame was captured successfully:
        Flip the frame horizontally
        Convert the frame to RGB format
        Process the frame with the hand-detection model
        If hands are detected in the frame:
            For each detected hand:
                For each hand landmark:
                    Calculate the coordinates of finger landmarks and wrist
                    Draw circles at finger landmarks and wrist
                For each pair of landmarks forming a hand:
                    Calculate the angles between fingers and wrist
                    Draw lines between landmarks to represent the hand
                    Display the calculated angles near the landmarks
                    Print the calculated angles
        Show the frame with detected hands and calculated angles
    If the 'ESC' key is pressed:
        Exit the loop
Release camera resources and close all windows
End
```

The process begins with the importation of the requisite libraries and the configuration of hand detection. This is followed by writing data to an Excel file and initiating an infinite loop to acquire and process images captured by the camera. Within this loop, the algorithm detects the hand in the image, calculates the reference angles, displays the results, and stores the calculated angles.

3.5.2. Validation Phase

Validation is a crucial step to ensuring the accuracy and reliability of the character recognition system. For this purpose, the DT C4.5 ML method, specifically the C4.5 algorithm, is employed [57]. The validation process comprises several key steps. Initially, hand signals are processed analogously to those used during training. Six internal angles of the hand are extracted as features for validation. These angles are compared to a set of "alphabet classes," which contain information about the signs corresponding to each of the 21 letters of the alphabet. The objective is to determine if the captured hand signal corresponds to one of the previously trained alphabet classes. If no direct correspondence with the alphabet classes is found, the DT C4.5 algorithm is activated. This algorithm is renowned for its capacity to construct decision trees based on the information provided by the training data. These decision rules may be employed to classify unknown records from the captured hand signal. In this manner, the DT C4.5 functions as a predictor, determining which letter of the alphabet the signal belongs to and labeling it correctly. For further clarification, please refer to Figure 7.



αA	150	146	170	-	-	145
αB	25	26	20	-	-	23
αC	113	123	118	-	-	116
αD	173	164	175	-	-	177
αE	165	160	164	-	-	159
αF	94	101	96	-	-	101
letter	B	B	B	B	B	B

Figure 7. Complete block diagram of the training and validation stages.

This system employs a camera and a hand recognition model to capture and process gestures. The camera initiates video capture, while hand-gesture data are read from an Excel file, potentially containing information from a previous training session. An infinite loop is established to capture frames from the camera continuously. Each frame is processed with the hand recognition model, extracting the coordinates of the landmarks and calculating the angles between them. The processing cycle persists until the user elects to terminate it. At this juncture, the DT C4.5 model categorizes the captured signals. The image depicts the classified signal, and the resulting video with the gesture labels is displayed for future reference. Table 3 provides a detailed description of the validation process.

Table 3. Validation process for hand-gesture recognition system with DT C4.5.

Pseudocode 2 Real-Time Hand Gesture Recognition Validation

```
Start
Import necessary libraries (cv2, mediapipe, matplotlib, numpy, pandas, sklearn)
Configure mediapipe for hand detection and capture video from the camera
Read hand gesture data from an Excel file
Initialize necessary variables and data structures
With mediapipe.Hands(
    static_image_mode = False,
    max_num_hands = 2,
    min_detection_confidence = 0.5) as hands:
    While True:
        Read a frame from the camera
        If the frame was read successfully:
            Process the frame with the hand detection model
            If hands are detected in the frame:
                For each detected hand:
                    Extract coordinates of finger landmarks and wrist
                    Calculate angles between finger landmarks and wrist
                    Visualize landmarks and hand lines on the frame
                    Display calculated angles near the landmarks
                    Store the calculated angles
            Show the frame with detected hands and calculated angles
        If the 'ESC' key is pressed:
            Perform gesture classification using a decision tree model
            Display the predicted gesture label on the frame
            If needed, save the output video with gesture labels
            Exit the loop
Release camera resources and close all windows
End
```

The pseudocode presented elucidates the real-time validation of hand-gesture recognition. As stated, it imports libraries and reads gesture data from an Excel file. After initializing the requisite variables, it initiates an infinite loop to capture and process images from the camera.

3.6. Decision Tree

This algorithm needs to be explained because it performed best compared to the ML algorithms in the results section; this decision algorithm and its possible results are presented as a tree. It is used for classification and regression tasks, especially DT C4.5, which has served as the basis for several variants and improvements in the design of decision-tree algorithms, such as random forests [58,59]. In this context, the tool predicts the category (letter) of the hand signs not found in the previously trained set of alphabet classes. Although this tool is not new, our experiments have shown extraordinary efficiency, responsiveness, and accuracy despite its simplicity of application. Its use in this work is further detailed by explaining the parts that make up this algorithm.

3.6.1. Entropy

Entropy is a physical quantity applied to a thermodynamic system in equilibrium. Its function is to measure the number of microstates compatible with the macroequilibrium state of the system. This measurement can be understood as measuring the degree of organization present in the system in that state. In our case, when the entropy level is zero, it represents the maximum order. The decision tree is created based on the gain of information obtained from the training examples and is then used to classify the test set.

The classification task is typically performed with nominal attributes and no missing values in the dataset. If a probability distribution $P = (p1,\ p2,\ \ldots,\ pn)$ is provided, then the information carried by this distribution is known as entropy and is calculated by:

$$Entropy\ P = -\sum\nolimits_{i=1}^{n} p_i \cdot \log(p_i) \tag{2}$$

3.6.2. Information Gain

To select the attribute of a given node at any position in the tree under construction, it is necessary to determine the gain for a *t*-test and a position p at that node using the following equation:

$$Gan(p,\ T) = EntrP - \sum\nolimits_{j=1}^{n} (pj \cdot Entr)(pj) \tag{3}$$

The values (pj) represent the set of all possible values for the attribute. This measure given in Equation (3) can be used to determine which attribute is better and to construct the decision tree by considering the node that possesses the attribute with the highest information gain of all the attributes that have not yet been considered in the route from the root node.

3.6.3. Algorithm C4.5

The C4.5 algorithm is an extension of the ID3 algorithm proposed by Quinlan to address some of the deficiencies of ID3, in that it was not designed for numeric attributes and does not use pruning to reduce overtraining. To solve these problems, algorithm C4.5 uses a new calculation to measure the gain ratio [60]. Thanks to the introduction of this new calculation, it is possible to calculate a gain ratio:

$$RelGan(p,\ T) = \frac{Gan(p,\ T)}{infDiv(p,\ T)} \tag{4}$$

where

$$\text{infDiv}(\text{p}, \text{test}) = -\sum\nolimits_{j=1}^{k} p\left(\frac{j}{p}\right) \cdot \log(p'\left(\frac{j}{p}\right)) \tag{5}$$

$\text{p}'(j/p)$ is the proportion of elements present at position p, calculated from the umpteenth test. Unlike the entropy in ID3, the gain ratio is independent of the objects distributed in different classes. C4.5 handles attributes with unknown values more effectively by evaluating the gain ratio for these attributes by considering only the datasets for which that dataset is defined. To accomplish this task, the algorithm estimates the probabilities of different outcomes. Then, the new gain criterion takes the form.

$$Gan(p) = F(Info(T) - Info(p,\ T)) \tag{6}$$

where F is the number of examples in the dataset with known values for the number of examples in an attribute dataset.

$$\text{Info(T)} = \sum\nolimits_{i=1}^{n} ((p_j) Entropia(p_j)) \tag{7}$$

If we first partition T into sets p_1, $p_2 \ldots$, p_n based on the value of a non-categorical attribute p, then the information needed to identify the class of an element of T becomes the weighted average of the information required to specify the class of a component of T_i, i.e., the weighted average of Info (T_i). It also handles attributes with continuous values. Let p_j be an attribute with a continuous value in a range of constant values. The values of these attributes are examined in the training set. The gain for each partition is calculated, and the gain-maximizing partition is selected. An alternative approach used in the C4.5 algorithm is the technique of post-pruning. The algorithm does not stop during execution. Therefore, it also allows over-fitting, and only at the end are pruning rules applied to improve the generalization ability. Another difficulty is handling continuous value attributes, such as real numbers [61]. Lastly, C4.5 uses a pruning technique to minimize the error rate. This technique reduces the size of the tree by removing parts that may be due to incorrect or missing data, thereby reducing the complexity of the tree and improving its classification performance.

3.6.4. Measure of Gini Impurity

Gini impurity is a metric used to create classification trees, providing more insight into the data distribution per node than classification accuracy alone. It is calculated by considering the proportion of each target category among all the records at a node. The Gini impurity is computed as the sum of the squares of these proportions subtracted from one. For example, when splitting a node, the algorithm seeks the split that maximizes the reduction in impurity, defined as the impurity of the parent node minus the weighted average impurity of the child nodes. The overall objective is to minimize the Gini impurity.

$$Gini\ (Xq) = 1 - \sum_{k=1}^{k} \left(p_{k,q}\right)^2 \tag{8}$$

This is used for categorical attributes. This criterion attempts to estimate the information provided by each attribute based on "information theory". Entropy is a measure of the uncertainty or randomness of a random variable "x". By calculating the entropy for each attribute, the information gain of the tree can be determined.

4. Results

4.1. Evaluation Metrics and Their Relevance

Various metrics have been used to evaluate model performance, including training and validation accuracy and training and prediction time. These metrics are essential for understanding different aspects of model performance. Accuracy provides a measure of the quality of predictions, while training and prediction time give information about the computational efficiency of the algorithms, which is critical for real-time or large-scale applications.

Several ML models have been trained using algorithms, including random forest, SVM, naive Bayes, KNN, decision tree C4.5, LightGBM, XGBoost, CatBoost, and RNNs. This broad range allows for a comprehensive comparison between different modeling paradigms, which is crucial for understanding which approaches best fit a dataset and the sign language predictor. The importance of this analysis lies in its ability to determine the algorithm that offers the best performance in terms of accuracy, computational efficiency, and generalizability, all of which are essential for the success and utility of our sign language predictor. The results illustrate how the accuracy of the models varies with increasing the training time. Table 4 presents the performance comparison of the ML models.

Throughout this process, we closely examined how performance metrics, such as accuracy, varied between the training and test sets, and how the performances of the different classification models were reflected in terms of precision, recall, F1 score, and accuracy. Table 5 presents the training and prediction time metrics.

The decision tree, naive Bayes, and k-NN are fast at training and prediction. SVM and random forest have moderate training and prediction times. XGBoost and CatBoost are quick predictors, but CatBoost has a longer training time. LightGBM is efficient for large datasets but slower in prediction. RNNs have long training and prediction times. The choice of algorithm depends on the problem requirements and available computational resources. Cross-validation scores are used to assess the generalization ability of a machine-learning model; see Table 6. Cross-validation is a technique that helps ensure that the model performs well on the training dataset and previously unseen new data.

Table 4. Performance Models.

Model Results									
Metric/Method	**D T C4.5**	**SVM**	**N B**	**k-NN**	**RF**	**XGBoost**	**LightGBM**	**CatBoost**	**RNNs**
Training Accuracy	**1.00**	0.98	0.97	0.99	1.00	**1.00**	**1.00**	**1.00**	0.92
Training Loss	**0.00**	0.02	0.03	0.01	**0.00**	**0.00**	**0.00**	**0.00**	0.27
Testing Accuracy	0.99	0.96	0.96	0.98	0.99	0.99	0.99	**1.00**	0.96
Testing Loss	0.01	0.04	0.04	0.02	0.01	0.01	0.01	**0.00**	0.04
Accuracy	0.99	0.96	0.96	0.98	0.99	0.99	0.99	**1.00**	0.96
Precision	0.99	0.96	0.97	0.98	**1.00**	0.99	0.99	**1.00**	0.96
Recall	0.99	0.96	0.96	0.98	0.99	0.99	0.99	**1.00**	0.96
F1 Score	0.99	0.96	0.96	0.98	0.99	0.99	0.99	**1.00**	0.96

The numbers marked in red are the ones with the best performance.

Table 5. Training time and prediction.

Execution Times									
Time/Method	**D T C4.5**	**SVM**	**NB**	**k-NN**	**RF**	**XGBoost**	**LightGBM**	**CatBoost**	**RNNs**
Training Time	0.01 s	0.05 s	**0.00 s**	0.02 s	0.36 s	0.31 s	0.51 s	2.42 s	7.09 s
Prediction Time	**0.00 s**	0.03 s	0.01 s	0.03 s	0.02 s	**0.00 s**	0.10 s	0.00 s	0.12 s

The numbers marked in red are the ones with the best time.

Table 6. Indicates the cross-validation metrics.

Cross-Validation Scores									
Metric/Method	**D T C4.5**	**SVM**	**NB**	**k-NN**	**RF**	**XGBoost**	**LightGBM**	**CatBoost**	**RNNs**
Cross-Validation Time	0.07 s	0.45 s	**0.02 s**	0.24 s	1.78 s	1.58 s	2.41 s	11.17 s	29.00 s
CV Scores	[0.977, 0.985, 0.984, 0.984, 0.985]	[0.946, 0.960, 0.932, 0.952, 0.951]	[0.954, 0.963, 0.936, 0.926, 0.976]	[0.965, 0.982, 0.954, 0.944, 0.973]	[0.984, 0.992, 0.971, 0.974, 0.976]	[0.937, 0.988, 0.967, 0.976, 0.981]	[0.937, 0.989, 0.967, 0.976, 0.981]	[0.990, 0.992, 0.991, 0.995, 0.993]	[0.967, 0.970, 0.967, 0.968, 0.980]
CV Mean	0.98	0.96	0.95	0.96	0.98	0.97	0.97	**0.99**	0.97
CV Standard Deviation	**0.00**	0.01	0.02	0.01	0.01	0.02	0.02	**0.00**	0.01

The numbers marked in red are the ones with the best performance.

All models have a high overall accuracy, ranging from 0.96 to 0.99. This indicates that most of the models' predictions are correct compared to the total predictions. Accuracy measures the proportion of correct positive predictions. The models' accuracy ranges from 0.96 to 0.99, indicating a very high ability to predict instances correctly. State-of-the-art algorithms, such as CatBoost, XGBoost, and LightGBM, are the most accurate, although they struggle with training time. Metrics using clustered bar graphs facilitate visual comparisons between the different models. However, it is essential to interpret the results with caution. For example, a model with a high training accuracy but a low validation accuracy may indicate overfitting, while longer training and prediction times can be problematic in time-critical applications. See Figure 8.

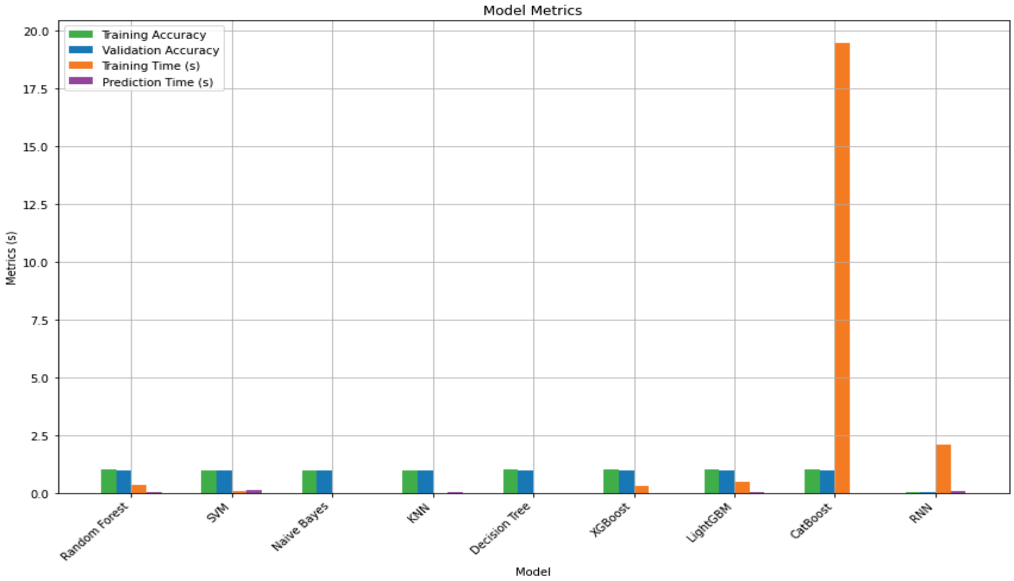


Figure 8. The metrics of the compared algorithms with the respective accuracy and time.

It is essential to contextualize the results concerning the specific problem and data under consideration. What works well in one dataset may not apply to another. In addition, evaluation results should be cautiously generalized and validated on independent datasets to ensure their robustness and reliability. The implementation provides a valuable exploration of the performance of different ML algorithms. However, to describe the performance of all models based on the graph provided, we note the following:

i. **Random forest:** Although random forest is known for its ability to handle large datasets with many features, it appears to be more time-consuming in training and prediction compared to other models. Nonetheless, it provides good accuracy on both training and test sets;

ii. **SVM:** it has longer training and prediction times than other models, and its accuracy is not the highest on this dataset;

iii. **Naive Bayes:** Although it has shorter training and prediction times, its accuracy is lower than other models. However, it could be a good choice if speed is a priority, and the required accuracy is reasonably high but not critical;

iv. KNN: It shows very short training times, but the prediction times are longer. Its accuracy is relatively high, but its distance-based nature may not be optimal for large datasets;

v. **DT C4.5:** it shows shorter training and prediction times compared to other models, and its accuracy is comparable to and even better than that of other more complex models on this dataset;

vi. **XGBoost and LightGBM:** These gradient-boosting models have good accuracy results but longer training times than DT C4.5. However, their prediction times are shorter than those of random forest and SVM;

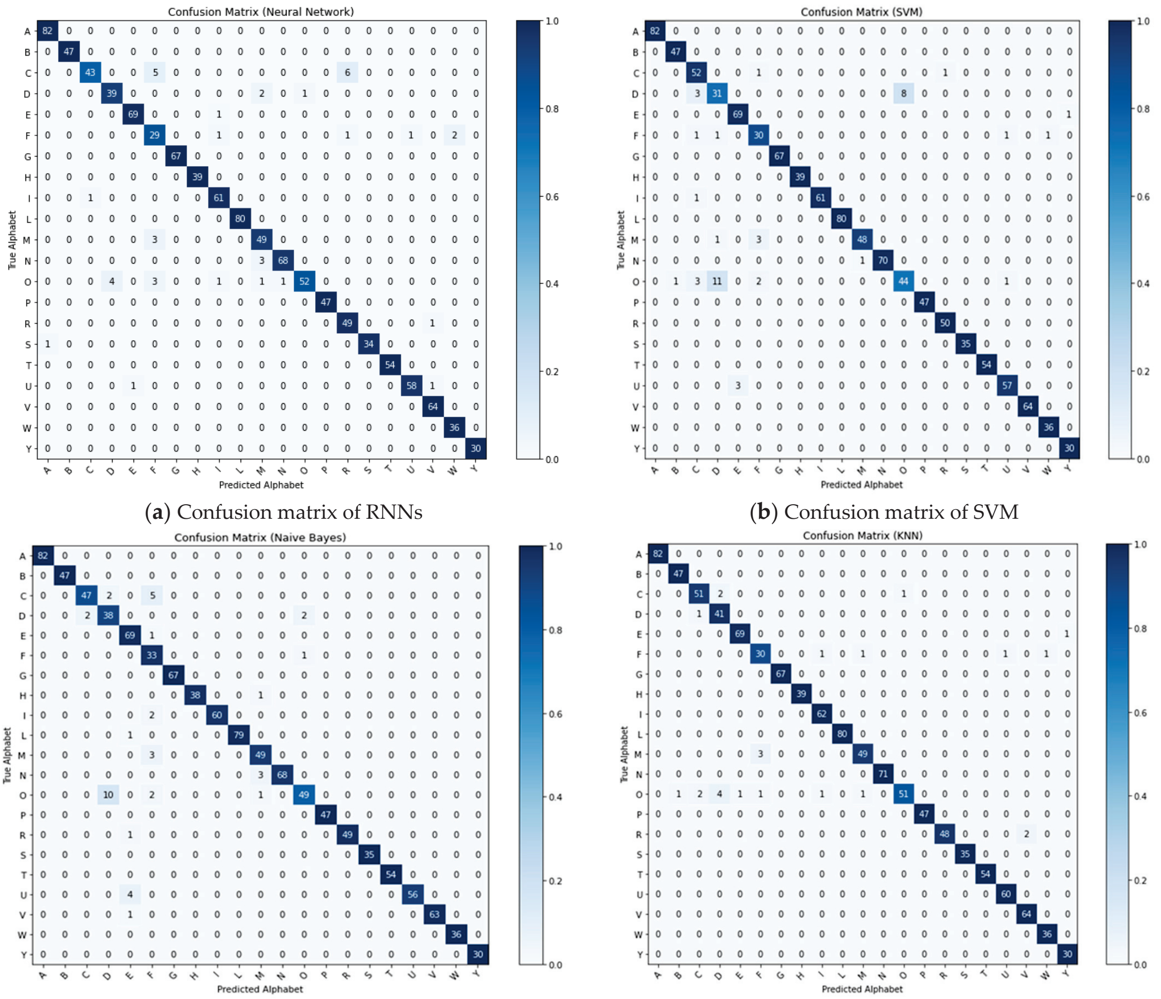
vii. **CatBoost:** It performs excellently, with an accuracy of 1.00. However, its training time is significantly higher (2.42 s), and the cross-validation time is also longer (11.17 s);

viii. **Neural networks:** although neural networks have good accuracy and recall (0.96), their training times (7.09 s) and cross-validation (29.00 s) are much longer compared to the decision tree and boosting models.

4.2. Confusion Matrix of the Compared Models

The confusion matrix, a fundamental component in evaluating the performance of the classification model, serves to quantify the model's accuracy by showing the number of correct and incorrect predictions for each class (letter). The confusion matrix is visualized, highlighting the relationships between classes and facilitating the identification of patterns of classification errors. Below, you can see the confusion matrices that were compared to determine which sign language predictor to choose. See Figure 9a–i.

The last matrix, DT C4.5, shows the number of correct and incorrect predictions for each class in a tabular format. It provides a detailed understanding of how the model classifies instances in each class. The model is accurate overall, with many correct predictions for most classes. The accuracy varies between the classes, indicating that the model may perform better for some classes than others. The high-performing classes are A, C, G, P, T, W, and Y, mostly with correct predictions with no significant false positives or negatives. This indicates that the model classifies these classes effectively. Some classes, such as M, R, and U, show false positives or negatives. These classes present opportunities for improvement to enhance the model's accuracy in classifying these instances. Overall, the confusion matrix results provide valuable information about the performance of the DT C4.5 model, highlighting areas of strength and areas that need improvement for more accurate classification. Based on these comparative analyses, we could decide which algorithm best balances accuracy and computational efficiency for our sign language predictor. This evaluation allowed us to select the most appropriate model to meet the problem's requirements and ensure optimal performance in practical situations.



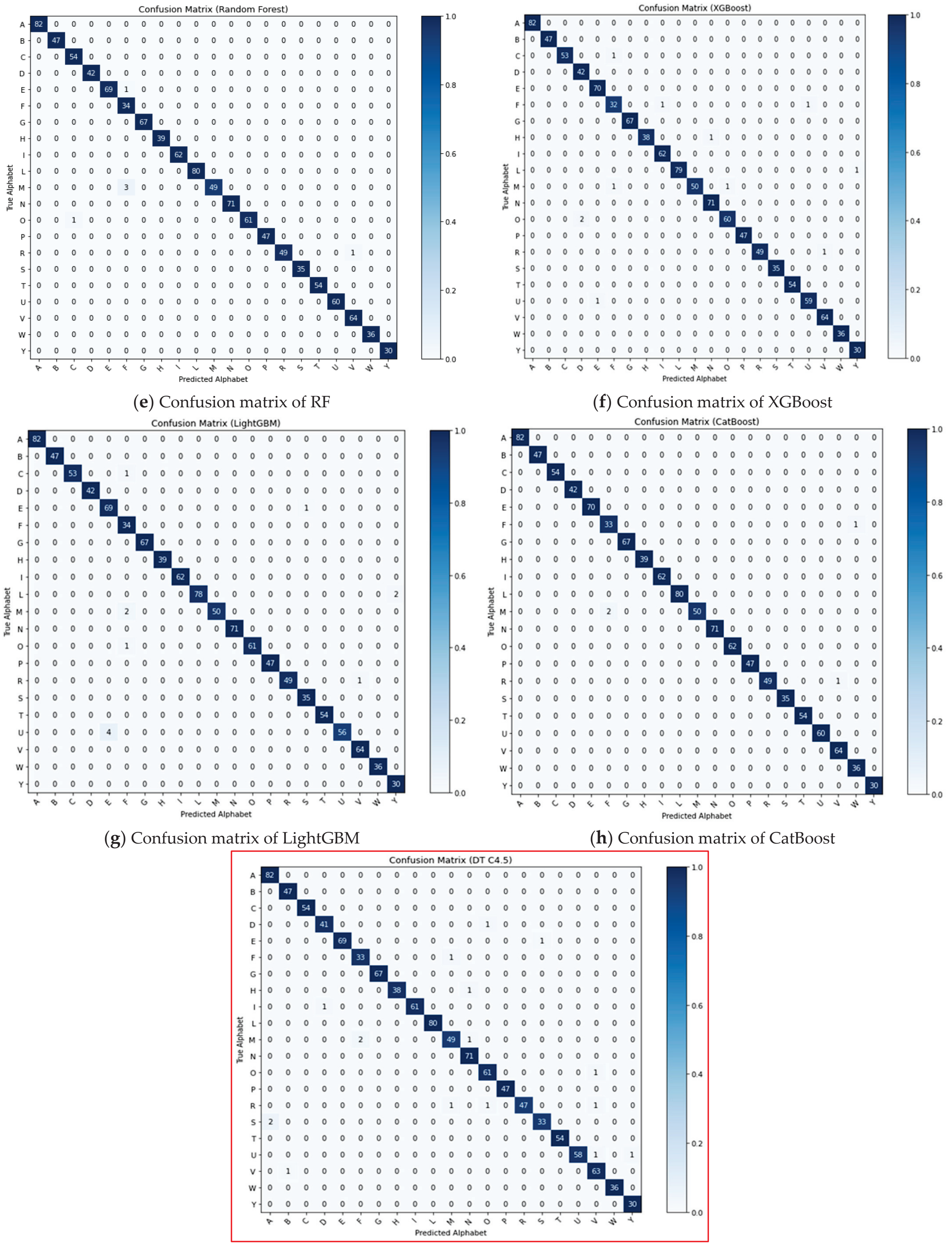
(**a**) Confusion matrix of RNNs

(**b**) Confusion matrix of SVM

(**c**) Confusion matrix of naïve Bayes

(**d**) Confusion matrix of KNN

Figure 9. *Cont.*



(**e**) Confusion matrix of RF

(**f**) Confusion matrix of XGBoost

(**g**) Confusion matrix of LightGBM

(**h**) Confusion matrix of CatBoost

(**i**) Confusion matrix of DT C4.5

Figure 9. (**a–i**): Confusion matrices for various sign language prediction models. Each confusion matrix illustrates the performance of a specific model in correctly classifying sign language gestures. The models compared include (**a**) recurrent neural networks (RNNs), (**b**) support vector machines (SVM), (**c**) naïve Bayes, (**d**) K-nearest neighbors (KNN), (**e**) random forest (RF), (**f**) XGBoost, (**g**) LightGBM, (**h**) CatBoost, and (**i**) decision tree (DT C4.5). These matrices help identify the model with the highest accuracy and best classification performance for SLR.

4.3. Performance of the DT C4.5 Classification Model

The following metrics provide a clear and concise evaluation of the proposed method's performance with DT C4.5 for classifying different letters. These metrics, including accuracy, precision, recall, and F1 score, are based on the most recent results and are presented straightforwardly in Tables 4–6 and Figure 8. Please refer to Table 7 for a detailed breakdown.

Table 7. Metrics for the proposed method using the decision tree C4.5.

Letter	Accuracy	Precisión	Recall	F1-Score	Support
A	0.99	0.98	1	0.99	82
B	0.99	0.98	1	0.99	47
C	0.99	1	1	1	54
D	0.99	1	0.95	0.98	42
E	0.99	1	0.99	0.99	70
F	0.99	0.94	0.97	0.96	34
G	0.99	1	1	1	67
H	0.99	1	0.97	0.99	39
I	0.99	1	1	1	62
L	0.99	1	1	1	80
M	0.99	0.98	0.94	0.96	52
N	1	0.97	1	0.99	71
O	0.99	0.97	0.98	0.98	62
P	1	1	1	1	47
R	0.99	1	0.96	0.98	50
S	0.99	0.97	0.94	0.96	35
T	1	1	1	1	54
U	1	0.98	0.97	0.97	60
V	0.99	0.95	0.98	0.97	64
W	1	1	1	1	36
Y	1	0.97	1	0.98	30
Weighted Avg	0.99	0.99	0.99	0.99	1138

Table 7 shows the high performance of the proposed method using DT C4.5 in classifying letters.

- **Accuracy:** This represents the proportion of correct predictions from the total predictions made for each class, providing a general measure of model performance. Values range from 0.99 to 1.0, indicating that the model is highly accurate for most classes;
- **Precision:** Indicates the proportion of instances correctly classified as positive out of all instances classified as positive. It measures the model's ability to avoid misclassifying a negative instance as positive. Values range from 0.94 to 1.0, indicating that the model has a low false-positive rate for most classes;
- **Recall:** Represents the proportion of positive instances correctly identified by the model out of all true-positive instances. It measures the model's ability to identify all relevant instances in a dataset. Values between 0.94 and 1.0 indicate that the model correctly identifies the most positive instances in each class;
- **F1 Score:** Measures model accuracy by considering both precision and recall. The harmonic mean of precision and recall balances the two metrics. Values range from 0.96 to 1.0, indicating a good balance between precision and recall for most classes;
- **Support:** values vary by class and represent the number of instances of each class in the test dataset.

4.4. Structure of Prediction in DT C4.5

A DT C4.5 is a graphical representation of a set of decision rules used to classify examples or predict outcomes, where each internal node represents a feature (attribute), each branch represents a decision based on that attribute, and each leaf represents the result of the decision. The following describes how decisions are made based on specific features by dividing the dataset into smaller groups at each internal node until a prediction is obtained. This leads to the classification of hand angles into letters of the alphabet. A decision tree is generated and visualized based on the given data, and the resulting image is saved. This process helps understand how DT C4.5 classifies different data and which features are most important for classification. See Figure 10.

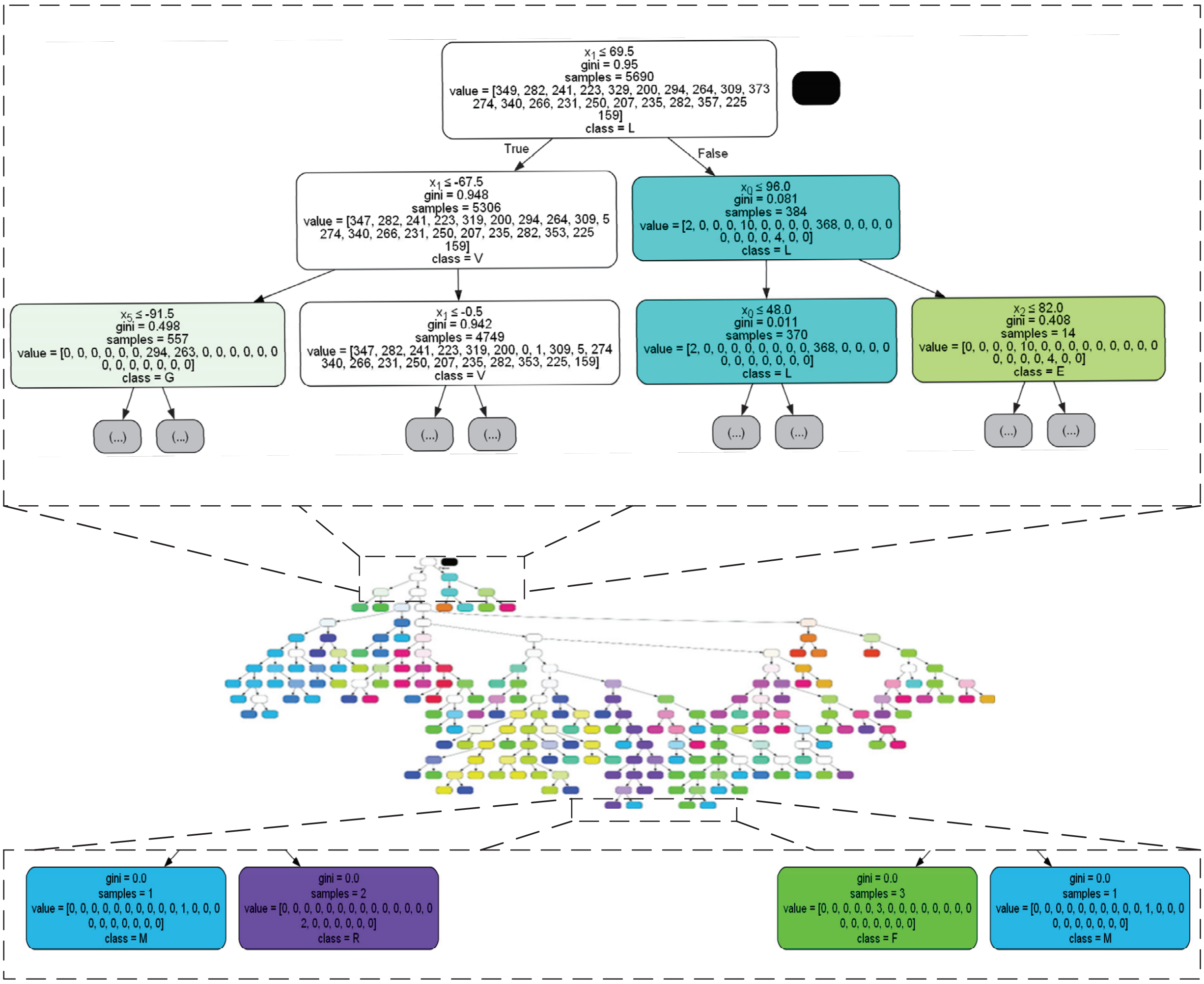

$x_1 \leq 69.5$
gini = 0.95
samples = 5690
value = [349, 282, 241, 223, 329, 200, 294, 264, 309, 373 274, 340, 266, 231, 250, 207, 235, 282, 357, 225 159]
class = L
True
False
$x_1 \leq -67.5$
gini = 0.948
samples = 5306
value = [347, 282, 241, 223, 319, 200, 294, 264, 309, 5 274, 340, 266, 231, 250, 207, 235, 282, 353, 225 159]
class = V
$x_0 \leq 96.0$
gini = 0.081
samples = 384
value = [2, 0, 0, 0, 10, 0, 0, 0, 0, 368, 0, 0, 0, 0 0, 0, 0, 0, 4, 0, 0]
class = L
$x_5 \leq -91.5$
gini = 0.498
samples = 557
value = [0, 0, 0, 0, 0, 0, 294, 263, 0, 0, 0, 0, 0, 0 0, 0, 0, 0, 0, 0, 0]
class = G
$x_1 \leq -0.5$
gini = 0.942
samples = 4749
value = [347, 282, 241, 223, 319, 200, 0, 1, 309, 5, 274 340, 266, 231, 250, 207, 235, 282, 353, 225, 159]
class = V
$x_0 \leq 48.0$
gini = 0.011
samples = 370
value = [2, 0, 0, 0, 0, 0, 0, 0, 0, 368, 0, 0, 0, 0 0, 0, 0, 0, 0, 0, 0]
class = L
$x_2 \leq 82.0$
gini = 0.408
samples = 14
value = [0, 0, 0, 0, 10, 0, 0, 0, 0, 0, 0, 0, 0, 0 0, 0, 0, 0, 4, 0, 0]
class = E
(...)
(...)
(...)
(...)
(...)
(...)
(...)
(...)
gini = 0.0
samples = 1
value = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0 0, 0, 0, 0, 0, 0, 0]
class = M
gini = 0.0
samples = 2
value = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 2, 0, 0, 0, 0, 0, 0]
class = R
gini = 0.0
samples = 3
value = [0, 0, 0, 0, 0, 3, 0, 0, 0, 0, 0, 0, 0, 0 0, 0, 0, 0, 0, 0, 0]
class = F
gini = 0.0
samples = 1
value = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0 0, 0, 0, 0, 0, 0, 0]
class = M

Figure 10. DT C4.5 is magnified twice to show (1) the computed features and arguments of the input tree and (2) the output of the already labeled leaves in the DT C4.5 classifier for predicting letters based on certain hand features detected by SRL. The meaning of each part of the tree is fully ex-plained.

4.5. The Results of the Characteristics of DT. C4.5

i. **Feature Names:** In the context of a C4.5 DT, these are the features used to make decisions at each tree node. It is almost always specified that default feature names be used;

ii. **Gini:** The Gini impurity measures how impure a node is. Determining which features and split values are best for dividing the dataset into smaller subsets is critical. In the decision-tree graph, nodes are split based on the Gini impurity value to minimize impurity in the resulting nodes;

iii. **Examples:** In the context of DT C4.5, this refers to the number of data instances that arrive at a particular node during the tree's training process. The number of samples arriving at each node can be shown in the decision-tree graph, providing information about the data distribution in the tree;

iv. **Value:** The value at a node represents the class distribution of the samples arriving at that node. In the decision-tree graph, the value of a node can be visualized as a list showing how many samples of each class are present at that node;

v. **Class:** The leaf nodes of the decision tree represent the predominant class of samples arriving at that node. In the decision-tree graph, the class of a leaf node can be displayed as a label indicating the predominant class at that node.

The leaf nodes in the tree indicate the predominant class of arriving samples, constituting the final classification of letters in sign language. These steps are essential for understanding and effectively applying the classification process using a DT C4.5 classifier in SLR.

In summary, the code uses a DT C4.5 classifier to predict letters based on the characteristics of the detected hands. The decision-tree graph provides a visualization of how these decisions are made, and terms such as Gini, impurity, samples, value, and class refer to different aspects of the tree construction process and data distribution in the tree. The images in Figure 11 show the manual sign results for the 21 non-movement letters that define MSL.

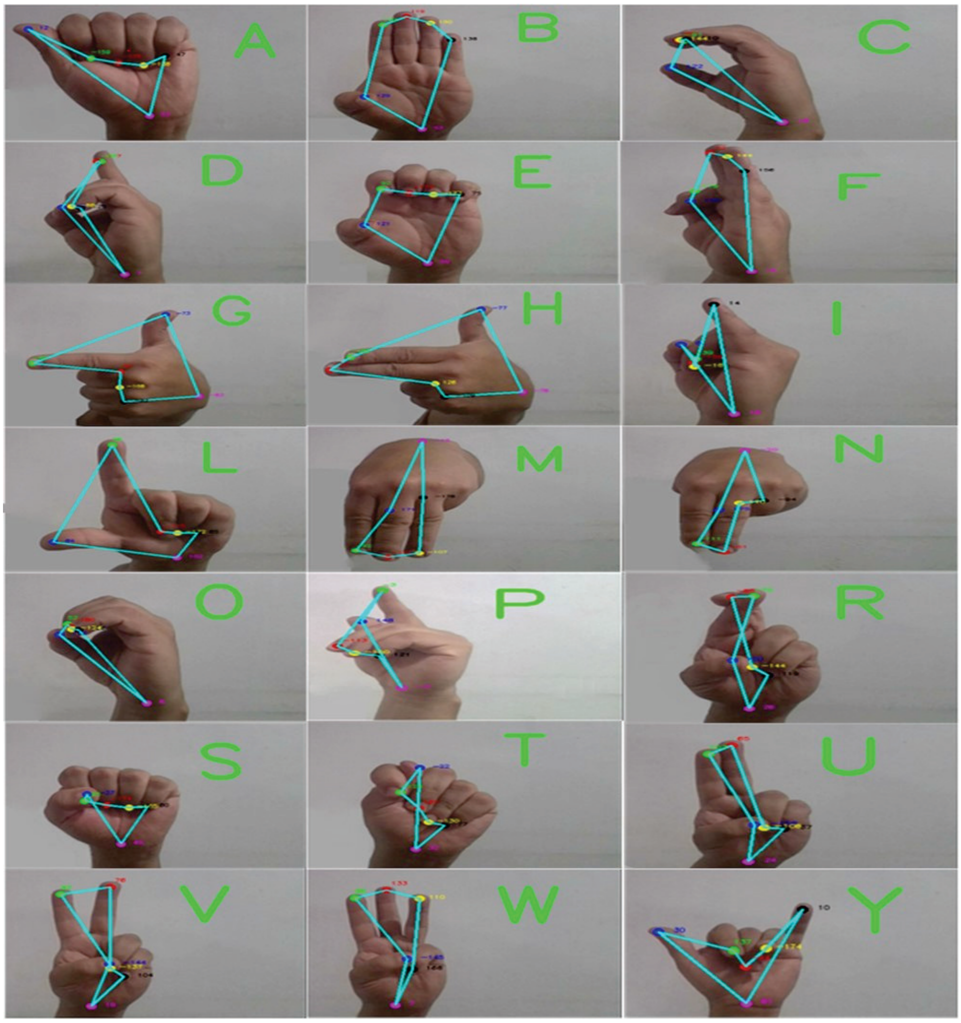


Figure 11. The outcomes were achieved using the suggested approach.

It is important to note that the results from the instantaneous completion of the process are obtained from real-time video, not photographs. As with all video systems, the two limitations of the method are the lighting quality and the lack of focus of the webcam. The images in Figure 11 show the 21 letters of the alphabet, excluding those with movement.

5. Discussion

This work is distinctive in the scientific literature because it analyzes six features that represent the angles between the distal phalanges and the palm. This approach minimizes the need for complex image processing. Additionally, there is limited research on MSL.

In recent MSL research published by Gonzalez et al. [12], MediaPipe was used as a descriptor of the face, body, and hands to create avatars. The system includes an easy-to-use graphical interface with modes to translate between MSL and Spanish in both directions. Users can enter characters or text and receive corresponding translations. The performance evaluation shows high accuracy, with the bidirectional neural network model achieving an accuracy of 98.8%. Like us, they reduce the dimensionality of the features in their work to 11 for the face and 5 for the body but keep the 21 critical points of the hand. In contrast, in our work, the dimensionality is reduced to six features, obtaining similar results in accuracy.

On the other hand, Sosa et al. [13] conducted a study using MSL. They proposed a system to recognize and animate signs related to general medical consultations with avatars in real time. This system facilitates dynamic and non-intrusive interaction between hearing doctors and deaf patients. The recognition module uses an MS Kinect sensor to capture sign trajectories and images processed in real time by hidden Markov models (HMMs). The study involved 22 participants and demonstrated the recognition of 82 different signs, achieving average accuracy rates and obtaining F1 scores of 99% and 88%, respectively. The work uses MSL, but the Kinect sensor requires two computers to program speech and train an avatar using motion caption (MoCap), which cannot track finger movements. Therefore, it needs to be adapted afterward. The researchers' contribution is valuable because it focuses on helping hearing-impaired people communicate in a medical context. Compared to our work, we used a low-cost camera with a medium-capacity computer, which processes finger images much faster. We do not need to train an avatar beforehand or the patient; communication is facilitated directly by the person in need, regardless of hand size or skin color.

According to the same methodology but without the incursion of MSL, we found a promising work by Subramanian et al., who employed a hand-feature descriptor integrating an optimized MediaPipe called gated recurrent units (MOPGRU) for ISL recognition, obtaining an average accuracy of 95% [26]. In contrast, our work did not require an optimized MediaPipe, and we obtained similar results. Just as Hussain [45] identified two alphabets, ASL and ISL-HS, using different kinds of ML, including random forest, DT C4.5, and naive Bayes, to classify hand gestures using a dataset with 28 gestures between letters and 2 signs. The random forest classifier was the best-performing classifier, showing an accuracy of 96.7% with ISL and 93.7% with ASL. However, in our study, the random forest showed more extended training and prediction. However, it is our second-best performance after DT C4.5. Our work has some limitations that make it imperfect; we only have 21 letters of the alphabet. It was not possible to include letters that imply movement, and it would be desirable to complete the MSL. Some classes, such as M, R, and U, show false positives or negatives. These classes are of utmost importance and require special attention to improve the accuracy of the model in classifying these letters.

6. Conclusions

This study presents a comprehensive analysis of various ML models applied to the classification of MSL. Our decision tree C4.5 algorithm demonstrated remarkable performance, achieving near-perfect precision, recall, and an F1 score of 99%. Compared to cutting-edge algorithms like random forest, XGBoost, LightGBM, CatBoost, and neural networks, the DT C4.5 algorithm stands out for its balance between computational efficiency and predictive accuracy. While models such as CatBoost and neural networks offer competitive accuracy, they require significantly longer training times, which may not be ideal for real-time or large-scale applications. CatBoost, in particular, exhibited excellent performance in accuracy and handling categorical data, but its training time was considerably longer compared to DT C4.5.

Although the neural network was adequate, with a precision and recall of 96%, it presented the most extended training and cross-validation times among all the models tested. This makes it less practical for scenarios requiring quick deployment and iteration. However, its ability to handle complex patterns in the data is noteworthy, suggesting its potential for future improvements where computational resources are less constrained.

Our findings highlight the importance of model selection based on the specific needs of the application, such as training speed, prediction time, and accuracy. DT C4.5 proved to be the most balanced option for our MSL predictor, offering robust performance without the drawbacks associated with more computationally demanding models. This study underscores the potential of simpler models like DT C4.5 to achieve high accuracy in specialized tasks where the advanced models' complexity and resource demands may not be justified.

Supplementary Materials: The following supporting information can be downloaded at https://www.mdpi.com/article/10.3390/technologies12090152/s1, https://github.com/gggvamp/pdi/blob/main/datosO.xlsx (accessed on 4 April 2024), database, https://github.com/gggvamp/MSL/blob/main/letras2%20(3).png (accessed on 4 April 2024), extended decision tree C4.5, and https://github.com/gggvamp/pdi/blob/main/videomsl.mp4 (accessed on 4 April 2024), Video.

Author Contributions: G.G.-G. conceptualized the work, edited the manuscript, and prepared the dataset. B.A.S.-T. conceptualized the work and analyzed the dataset using computer vision algorithms. G.d.C.L.-A. conceptualized the work, wrote parts of the manuscript, provided additional analysis, and revised the manuscript. J.J.S.-E. supervised, wrote parts of the manuscript, provided additional analysis, and revised the manuscript. A.N.R.-V. supervised and revised the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: Technical Industrial Teaching Center supported this research. The action program was under academic direction (research work PI-03-2023).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: This information has been detailed at Supplementary Materials.

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

References

1. Amrutha, K.; Prabu, P. ML Based Sign Language Recognition System. In Proceedings of the 2021 International Conference on Innovative Trends in Information Technology (ICITIIT), Kottayam, India, 11–12 February 2021; pp. 1–6. [CrossRef]
2. Hekmat, A.; Abbas, H.; Shahadi, H. Sign Language Recognition and Hand Gestures Review. *Kerbala J. Eng. Sci.* **2022**, *2*, 209–234.
3. Younas, F.; Nadir, J.; Usman, M.; Khan, M.A.; Khan, S.A.; Kadry, S.; Nam, Y. An Artificial Intelligence Approach for Word Semantic Similarity Measure of Hindi Language. *KSII Trans. Internet Inf. Syst.* **2021**, *15*, 2049–2068.
4. Mahesh, B. Machine Learning Algorithms—A Review. *Int. J. Sci. Res.* **2019**, *9*, 381–386. [CrossRef]
5. Napier, J. Sign Language Interpreter Training, Testing, and Accreditation: An International Comparison. *Am. Ann. Deaf* **2004**, *149*, 350–359. [CrossRef] [PubMed]
6. Alaghband, M.; Maghroor, H.R.; Garibay, I. A survey on sign language literature. *Mach. Learn. Appl.* **2023**, *14*, 100504. [CrossRef]
7. Escobar, L. Gestualidad y lengua en la lengua de seÃ±as mexicana. *Lingüíst. Mex. Nueva Época* **2019**, *1*, 141–166. [CrossRef]
8. Valli, C.; Lucas, C. *Linguistics of American Sign Language: An Introduction*, 3rd ed.; Gallaudet University Press: Washington, DC, USA, 2000; ISBN 978-1-56368-097-7.
9. Lugaresi, C.; Tang, J.; Nash, H.; McClanahan, C.; Uboweja, E.; Hays, M.; Zhang, F.; Chang, C.-L.; Yong, M.G.; Lee, J.; et al. MediaPipe: A Framework for Building Perception Pipelines. *arXiv* **2019**, arXiv:1906.08172.
10. Zhang, F.; Bazarevsky, V.; Vakunov, A.; Tkachenka, A.; Sung, G.; Chang, C.-L.; Grundmann, M. MediaPipe Hands: On-device Real-time Hand Tracking. *arXiv* **2020**, arXiv:2006.10214. [CrossRef]
11. Zelinsky, A. Learning OpenCV—Computer Vision with the OpenCV Library (Bradski, G.R. et al.; 2008) [On the Shelf]. *IEEE Robot. Autom. Mag. IEEE Robot Autom.* **2009**, *16*, 100. [CrossRef]
12. González-Rodríguez, J.-R.; Córdova-Esparza, D.-M.; Terven, J.; Romero-González, J.-A. Towards a Bidirectional Mexican Sign Language–Spanish Translation System: A Deep Learning Approach. *Technologies* **2024**, *12*, 7. [CrossRef]
13. Sosa-Jimenez, C.O.; Rios-Figueroa, H.V.; Solis-Gonzalez-Cosio, A.L. A Prototype for Mexican Sign Language Recognition and Synthesis in Support of a Primary Care Physician. *IEEE Access* **2022**, *10*, 127620–127635. [CrossRef]

14. Li, Z.; Liu, F.; Yang, W.; Peng, S.; Zhou, J. A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *33*, 6999–7019. [CrossRef] [PubMed]
15. Wangchuk, K.; Riyamongkol, P.; Waranusast, R. Real-time Bhutanese Sign Language digits recognition system using Convolutional Neural Network. *ICT Express* **2021**, *7*, 215–220. [CrossRef]
16. Kasapbaşi, A.; Elbushra, A.; Al-Hardanee, O.; Yilmaz, A. DeepASLR: A CNN based Human Computer Interface for American Sign Language Recognition for Hearing-Impaired Individuals. *Comput. Methods Programs Biomed. Update* **2022**, *2*, 100048. [CrossRef]
17. Arooj, S.; Altaf, S.; Ahmad, S.; Mahmoud, H.; Mohamed, A.S.N. Enhancing sign language recognition using CNN and SIFT: A case study on Pakistan sign language. *J. King Saud Univ. Comput. Inf. Sci.* **2024**, *36*, 101934. [CrossRef]
18. Ameen, S.; Vadera, S. A convolutional neural network to classify American Sign Language fingerspelling from depth and colour images. *Expert Syst.* **2017**, *34*, e12197. [CrossRef]
19. Thongtawee, A.; Pinsanoh, O.; Kitjaidure, Y. A Novel Feature Extraction for American Sign Language Recognition Using Webcam. In Proceedings of the 2018 11th Biomedical Engineering International Conference (BMEiCON), Chiang Mai, Thailand, 21–24 November 2018; pp. 1–5. [CrossRef]
20. Rastgoo, R.; Kiani, K.; Escalera, S. Video-based isolated hand sign language recognition using a deep cascaded model. *Multimed. Tools Appl.* **2020**, *79*, 22965–22987. [CrossRef]
21. Rastgoo, R.; Kiani, K.; Escalera, S. Real-time isolated hand sign language recognition using deep networks and SVD. *J. Ambient Intell. Humaniz. Comput.* **2022**, *13*, 591–611. [CrossRef]
22. Sharma, P.; Anand, R.S. A comprehensive evaluation of deep models and optimizers for Indian sign language recognition. *Graph. Vis. Comput.* **2021**, *5*, 200032. [CrossRef]
23. Katoch, S.; Singh, V.; Tiwary, U.S. Indian Sign Language recognition system using SURF with SVM and CNN. *Array* **2022**, *14*, 100141. [CrossRef]
24. Tripathi, S.; Singh, S.K.; Kuan, L.H. Bag of Visual Words (BoVW) with Deep Features—Patch Classification Model for Limited Dataset of Breast Tumours. *arXiv* **2022**, arXiv:2202.10701. [CrossRef]
25. Tian, Y.; Shi, Y.; Liu, X. Recent advances on support vector machines research. *Technol. Econ. Dev. Econ.* **2012**, *18*, 5–33. [CrossRef]
26. Subramanian, B.; Olimov, B.; Naik, S.M.; Kim, S.; Park, K.-H.; Kim, J. An integrated mediapipe-optimized GRU model for Indian sign language recognition. *Sci. Rep.* **2022**, *12*, 11964. [CrossRef]
27. Sak, H.; Senior, A.; Beaufays, F. Long Short-Term Memory Based Recurrent Neural Network Architectures for Large Vocabulary Speech Recognition. *arXiv* **2014**, arXiv:1402.1128. [CrossRef]
28. Sundar, B.; Bagyammal, T. American Sign Language Recognition for Alphabets Using MediaPipe and LSTM. *Procedia Comput. Sci.* **2022**, *215*, 642–651. [CrossRef]
29. Pathan, R.K.; Biswas, M.; Yasmin, S.; Khandaker, M.U.; Salman, M.; Youssef, A.A.F. Sign language recognition using the fusion of image and hand landmarks through multi-headed convolutional neural network. *Sci. Rep.* **2023**, *13*, 16975. [CrossRef]
30. Ruiz, D.S.; Olvera-López, J.A.; Olmos-Pineda, I. Word Level Sign Language Recognition via Handcrafted Features. *IEEE Lat. Am. Trans.* **2023**, *21*, 839–848. [CrossRef]
31. Mohsin, S.; Salim, B.W.; Mohamedsaeed, A.K.; Ibrahim, B.F.; Zeebaree, S.R.M. American Sign Language Recognition Based on Transfer Learning Algorithms. *Int. J. Intell. Syst. Appl. Eng.* **2024**, *12*, 390–399.
32. Amangeldy, N.; Ukenova, A.; Bekmanova, G.; Razakhova, B.; Milosz, M.; Kudubayeva, S. Continuous Sign Language Recognition and Its Translation into Intonation-Colored Speech. *Sensors* **2023**, *23*, 6383. [CrossRef]
33. Wali, A.; Shariq, R.; Shoaib, S.; Amir, S.; Farhan, A.A. Recent progress in sign language recognition: A review. *Mach. Vis. Appl.* **2023**, *34*, 127. [CrossRef]
34. Farooq, U.; Rahim, M.S.M.; Sabir, N.; Hussain, A.; Abid, A. Advances in machine translation for sign language: Approaches, limitations, and challenges. *Neural Comput. Appl.* **2021**, *33*, 14357–14399. [CrossRef]
35. Zhang, M.-L.; Zhou, Z.-H. ML-KNN: A lazy learning approach to multi-label learning. *Pattern Recognit.* **2007**, *40*, 2038–2048. [CrossRef]
36. Louppe, G. Understanding Random Forests: From Theory to Practice. *arXiv* **2014**, arXiv:1407.7502. [CrossRef]
37. Hinton, G.E.; Osindero, S.; Teh, Y.-W. A Fast Learning Algorithm for Deep Belief Nets. *Neural Comput.* **2006**, *18*, 1527–1554. [CrossRef]
38. Bajaj, Y.; Malhotra, P. American Sign Language Identification Using Hand Trackpoint Analysis. *arXiv* **2020**, arXiv:2010.10590. [CrossRef]
39. Sahoo, A.K.; Mishra, G.S.; Ravulakollu, K.K. Sign Language Recognition: State of the Art. 2014.
40. Maebatake, M.; Suzuki, I.; Nishida, M.; Horiuchi, Y.; Kuroiwa, S. Sign Language Recognition Based on Position and Movement Using Multi-Stream HMM. In Proceedings of the 2008 Second International Symposium on Universal Communication, Osaka, Japan, 15–16 December 2008; pp. 478–481. [CrossRef]
41. Athitsos, V.; Alon, J.; Sclaroff, S.; Kollios, G. BoostMap: An Embedding Method for Efficient Nearest Neighbor Retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *30*, 89–104. [CrossRef] [PubMed]
42. Rahim, A.; Hossain, N.; Wahid, T.; Azam, S. Face Recognition using Local Binary Patterns (LBP). *Glob. J. Comput. Sci. Technol.* **2013**, *13*, 1–8.
43. Huang, C.; Huang, J. A Fast HOG Descriptor Using Lookup Table and Integral Image. *arXiv* **2017**, arXiv:1703.06256. [CrossRef]

44. Verma, R.; Kaur, M.R. Enhanced Character Recognition Using Surf Feature and Neural Network Technique. 2014. Available online: https://www.semanticscholar.org/paper/Enhanced-Character-Recognition-Using-Surf-Feature-Verma-Kaur/49f3939df922881dd857faac71aa5c7b873a606a (accessed on 18 May 2024).
45. Hussain, M.; Shaoor, A.; Alsuhibany, S.; Ghadi, Y.; Shloul, T.; Jalal, A.; Park, J. Intelligent Sign Language Recognition System for E-Learning Context. *Comput. Mater. Contin.* **2022**, *72*, 5327–5343. [CrossRef]
46. Zhang, H. The Optimality of Naive Bayes. In Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference (FLAIRS), Fredericton, NB, Canada, 1 January 2004.
47. 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, San Francisco, CA, USA, 13–17 August 2016; ACM: New York, NY, USA, 2006; pp. 785–794. [CrossRef]
48. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.Y. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: New York, HY, USA, 2017. Available online: https://proceedings.neurips.cc/paper_files/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html (accessed on 18 May 2024).
49. Lewis, R. An Introduction to Classification and Regression Tree (CART) Analysis. In Proceedings of the Annual Meeting of the Society for Academic Emergency Medicine, San Francisco, CA, USA, 22–25 May 2000.
50. 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. *Mach. Learn. Python* **2011**, *12*, 2825–2830.
51. Hubert, M.; Van der Veeken, S. Outlier detection for skewed data. *J. Chemom.* **2008**, *22*, 235–246. [CrossRef]
52. Butcher, B.; Smith, B.J. Feature Engineering and Selection: A Practical Approach for Predictive Models. In *The American Statistician*; Kuhn, M., Johnson, K., Eds.; Chapman & Hall/CRC Press: Boca Raton, FL, USA, 2020; Volume 74, pp. 308–309, ISBN 978-1-13-807922-9. [CrossRef]
53. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning*; Springer Series in Statistics; Springer: New York, NY, USA, 2009. [CrossRef]
54. Shin, J.; Matsuoka, A.; Hasan, M.A.M.; Srizon, A.Y. American Sign Language Alphabet Recognition by Extracting Feature from Hand Pose Estimation. *Sensors* **2021**, *21*, 5856. [CrossRef]
55. Obi, Y.; Claudio, K.S.; Budiman, V.M.; Achmad, S.; Kurniawan, A. Sign language recognition system for communicating to people with disabilities. *Procedia Comput. Sci.* **2023**, *216*, 13–20. [CrossRef]
56. Joksimoski, B.; Zdravevski, E.; Lameski, P.; Pires, I.M.; Melero, F.J.; Martinez, T.P.; Garcia, N.M.; Mihajlov, M.; Chorbev, I.; Trajkovik, V. Technological Solutions for Sign Language Recognition: A Scoping Review of Research Trends, Challenges, and Opportunities. *IEEE Access* **2022**, *10*, 40979–40998. [CrossRef]
57. Fang, G.; Gao, W.; Zhao, D. Large Vocabulary Sign Language Recognition Based on Fuzzy Decision Trees. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **2004**, *34*, 305–314. [CrossRef]
58. Quinlan, J.R. Induction of decision trees. *Mach. Learn.* **1986**, *1*, 81–106. [CrossRef]
59. Quinlan, J.R. Improved Use of Continuous Attributes in C4.5. *J. Artif. Intell. Res.* **1996**, *4*, 77–90. [CrossRef]
60. Li, X.; Yi, S.; Cundy, A.B.; Chen, W. Sustainable decision-making for contaminated site risk management: A decision tree model using machine learning algorithms. *J. Clean. Prod.* **2022**, *371*, 133612. [CrossRef]
61. Lyu, Y.; Huang, X. Road Segmentation Using CNN with GRU. *arXiv* **2018**, arXiv:1804.05164.

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.

technologies

Article

The Measurement of Contrast Sensitivity in Near Vision: The Use of a Digital System vs. a Conventional Printed Test

Kevin J. Mena-Guevara [1], David P. Piñero [1,2,*], María José Luque [3] and Dolores de Fez [1]

[1] Department of Optics, Pharmacology and Anatomy, University of Alicante, 03690 San Vicente del Raspeig, Spain; kevin.mena@ua.es (K.J.M.-G.); dolores.fez@ua.es (D.d.F.)
[2] Advanced Clinic Optometry Unit, Department of Ophthalmology, Medimar International Hospital, 03016 Alicante, Spain
[3] Department of Optics, Optometry and Vision Sciences, University of Valencia, Burjassot, 46100 Valencia, Spain; maria.j.luque@uv.es
* Correspondence: david.pinyero@ua.es; Tel.: +34-965-903-400

Abstract: In recent years, there has been intense development of digital diagnostic tests for vision. All of these tests must be validated for clinical use. The current study enrolled 51 healthy individuals (age 19–72 years) in which achromatic contrast sensitivity function (CSF) in near vision was measured with the printed Vistech VCTS test (Stereo Optical Co., Inc., Chicago, IL, USA) and the Optopad-CSF (developed by our research group to be used on an iPad). Likewise, chromatic CSF was evaluated with a digital test. Statistically significant differences between tests were only found for the two higher spatial frequencies evaluated ($p = 0.012$ and <0.001, respectively). The mean achromatic index of contrast sensitivity (ICS) was 0.02 ± 1.07 and -0.76 ± 1.63 for the Vistech VCTS and Optopad tests, respectively ($p < 0.001$). The ranges of agreement between tests were 0.55, 0.76, 0.78, and 0.69 log units for the spatial frequencies of 1.5, 3, 6, and 12 cpd, respectively. The mean chromatic ICS values were -20.56 ± 0.96 and -0.16 ± 0.99 for the CSF-T and CSF-D plates, respectively ($p < 0.001$). Furthermore, better achromatic, red–green, and blue–yellow CSF values were found in the youngest groups. The digital test allows the fast measurement of near-achromatic and chromatic CSF using a colorimetrically calibrated iPad, but the achromatic measures cannot be used interchangeably with those obtained with a conventional printed test.

Keywords: achromatic contrast sensitivity; chromatic contrast sensitivity; iPad; Optopad-CSF; Vistech VCTS; contrast sensitivity in near vision

1. Introduction

The use of modern technologies has modified our daily habits [1,2], especially following the COVID-19 pandemic, with the wide introduction of digital tools in academic [3] and work environments [4,5] and an exponential increase in their use. In the health context, many applications have been developed and released in digital stores (App Store and Google Play) for various purposes such as screening or diagnosis of distinct pathological conditions [6]. However, these tools require rigorous scientific studies to support and ensure their correct use in a clinical setting [6]. Among the applications/platforms that have been scientifically validated for clinical use, some evaluate visual acuity (VA) [7], stereopsis [8–11], achromatic contrast sensitivity function (CSF) [12], visual performance with a multifocal lens [13], and color vision [14].

As mentioned, several digital applications have been developed for the specific measurement of contrast sensitivity (CS), which is the ability of the visual system to distinguish objects against a set background. This type of clinical measurement is especially useful for characterizing visual performance in elderly patients, but it is not usually incorporated into routine optometric/ophthalmological examinations. Furthermore, the measurement of contrast sensitivity function over short distances is especially relevant when evaluating the outcomes of any procedure to compensate for presbyopia, such as multifocal contact lenses or intraocular lenses [15,16]. However, this parameter is not commonly used (only in very few studies), and researchers prefer to evaluate distance contrast sensitivity, which is expected to be affected less. One reason for that may be that this test can be considered time-consuming or difficult to perform, but this can be overcome by using the previously mentioned digital tests displayed on calibrated tablets.

Very few studies have tried to compare the results of a digital evaluation of CSF with those obtained with conventional tests [12,17–20]. Furthermore, most studies are focused on the evaluation of distance CSF. Bühren et al. [20] carried out a comparative study of achromatic CS under varying light conditions using three different instruments (FF-CATS at 4 m, FACT at optical infinity, and Pelli–Robson at 1 m) in three types of populations: healthy individuals under 50 years of age, healthy individuals over 50 years of age, and patients over 50 years of age with cataracts. In this study, the authors concluded that, under the different light conditions established, the results obtained with the evaluated instruments were not interchangeable. Rodriguez-Vallejo et al. [12] compared a new application (ClinicCSF) to measure CSF (at 2 m) with the iPad Retina against the Functional Acuity Contrast Test (FACT), confirming that there were no significant differences between tests when the same contrast sensitivity steps were used. However, no comparative analyses have been performed between printed and digital tests for measuring contrast sensitivity over short distances.

The Optopad digital tool is one of these new technologies for evaluating visual performance, and it was developed in collaboration between the Universities of Alicante and Valencia (Spain) [21]. This digital test has not yet been commercialized; thus, its use is restricted, but it has been validated for the detection of chromatic deficiencies (Optopad-Color) [14,22] and has been shown to be useful in characterizing achromatic and chromatic contrast sensitivity function (Optopad-CSF) over short distances [22,23]. This system was developed in an attempt to obtain a low-cost, portable digital tool for the evaluation of CSF and color vision in clinical settings, allowing a fast measurement procedure. Aside from the detection of chromatic anomalies, the intention was also to use advances in digital technology to create an easy-to-use test for measuring achromatic and chromatic CSF over short distances.

The evaluation of chromatic contrast sensitivity over short distances is a procedure that is not commonly performed in routine clinical practice, but its potential usefulness should be investigated further. To date, there have been a number of studies investigating the response of the visual system to chromatic contrast [24–27]. Our research group [28] conducted a comparative pilot study with 10 young subjects in which the effects produced by seven filters (three gray filters, four chromatic filters, and two low-vision lenses) on CSF were compared. This study concluded that, compared to gray filters of the same luminance, yellow filters may be useful when low achromatic contrasts are to be improved, although overall decreases in brightness may occur. Kim et al. [26] conducted a pilot study with 13 patients to record the differences obtained in the measurement of achromatic and chromatic CSF in near and far vision after varying luminance. They concluded that luminance causes a drop in the measure of contrast sensitivity, but it does not affect the shape of the CSF. After this, the same group conducted another study to establish normative values for achromatic and chromatic CSF measures [25]. However, it should be considered that the measurements were obtained at an intermediate distance (58 cm). These authors found higher sensitivity to the contrast of the L/M cone compared to the S cone and the achromatic responses. Wuerger et al. [24] evaluated the effect of the variation in luminance at two different distances, 91 cm (distance vision) vs. 45.5 cm (near vision), obtaining a luminance-dependent computational model predicted by the CSF for achromatic and chromatic stimuli of arbitrary size. Bodduluri et al. [29] conducted a comparative study of sensitivity to chromatic contrast (30 cm) in near vision with an application that operated on an iPad versus the results obtained with the Cambridge Color Test (CCT). The sample size was 100 healthy individuals. The authors concluded that, except for a game used to evaluate the blue–yellow contrast sensitivity, the CCT and tablet computer-based games showed similar repeatability, with comparable 95% limits of agreement. Wong et al. [27] performed color CS testing of each eye using Chromatest in a sample of 150 eyes of diabetic patients. This non-comparative study did not achieve results to justify use of Chromatest for screening, but it reinforced the changes seen in tritan color vision in diabetic retinopathy. The Optopad-CSF test also allows the evaluation of near chromatic CSF, but its clinical usefulness has not been investigated in detail. Only this test has been shown to be capable of detecting chromatic contrast sensitivity alterations in patients suffering from COVID-19 compared to age-matched healthy controls.

As previously mentioned, one critical aspect when a new digital test is developed is to confirm its clinical validity [6]. The validity of the measurements of near chromatic and achromatic CSF measurements obtained with the Optopad-CSF test is yet to be analyzed, and its clinical performance has not been compared with the performance of other tests for measuring near CSF. The aim of the current study was to analyze the performance of the Optopad-CSF test in a healthy population and to compare the data obtained with those provided by a conventional printed test that is considered as the gold standard.

2. Materials and Methods

2.1. Patients

This was a prospective cross-sectional clinical study that enrolled a total of 51 patients who underwent a complete visual examination at the Optometric Clinic of the University of Alicante. The inclusion criteria for the study were patients 18 years old or older and patients with no active ocular or systemic pathology compromising their visual function. Exclusion criteria included children, previous amblyopia, strabismus, and patients with any type of previous ocular surgery. The study received the approval of the Ethics Committee of the University of Alicante (Date: 26 February 2021. Exp. UA-2021-02-17) and was conducted following the standards of Good Clinical Practice and the international ethical principles applicable to research on humans (Declaration of Helsinki in its latest revision). All patients were informed about the nature of the study before their inclusion and provided signed consent to participate in it.

2.2. Clinical Tests

All patients had a complete eye examination including measurement of uncorrected and corrected distance and near visual acuity, manifest refraction, evaluation of ocular alignment with cover test, slit lamp biomicroscopy, measurement of stereopsis, and evaluation of near contrast sensitivity function at 40 cm with two different tests: the Vistech VCTS (Stereo Optical Co., Inc., Chicago, IL, USA) and the Optopad-CSF tests. With both tests, patches showing sinusoidal gratings with different spatial frequencies are presented and the subject must detect the orientation of the light–dark gratings. All measurements were performed in the right or left eye; the eye was selected randomly. Measurements were performed once after a clear explanation of the task. It should be considered that most of the tests performed were psychophysical measurements that could be very tiring for the patient. All these clinical examinations were performed by the same experienced optometrist (K.J.M.G.). Contrast sensitivity measurements were performed with the most optimal spectacle refractive correction in a darkened examination room.

The measurement with the Vistech VCTS test was performed in an iso-illuminated cabinet (a period of adaptation to experimental light conditions of 3 min was required before measurement) to prevent non-controlled illumination of the test or the presence of light reflexes that could interfere with the measurement procedure. A total of 5 spatial frequencies are evaluated with this test (F_A = 1.5 cpd, F_B = 3 cpd, F_C = 6 cpd, F_D = 12 cpd, and F_E = 18 cpd), each one corresponding to each horizontal line containing the text. Along the line, a total of 8 contrast levels (Michelson contrast formula) are presented to determine the threshold for the spatial frequency evaluated. In our study, subjects were asked about the orientation of the gratings (left, right, center), starting from the easiest task (low frequency, high contrast) to the most difficult (high frequency, low contrast). The last correctly seen contrast in each line (each spatial frequency) was considered as the threshold, and the contrast sensitivity value was obtained (inverse of the discrimination threshold value).

A similar procedure was followed with the Optopad-CSF digital test, with a constant (maximum) illumination of the screen. Subjects were also asked about the orientation of the gratings (left, right, up), starting from the easiest task to the most difficult. The threshold contrast value was considered to be the average value between the last correctly seen stimulus and the first unseen stimulus. It should be considered that Optopad-CSF explores spatial CSF in achromatic and chromatic mechanisms. The test contains plates for measuring achromatic (CSF-A) (F_{1A} = 1.5 cpd, F_{2A} = 3 cpd, F_{3A} = 6 cpd, F_{4A} = 12 cpd, and F_{5A} = 24 cpd), red–green (CSF-T), and blue–yellow (CSF-D) (F_{1C} = 1 cpd, F_{2C} = 2 cpd, F_{3C} = 4 cpd, F_{4C} = 8 cpd, and F_{5C} = 12 cpd) spatial contrast sensitivity along the cardinal directions of DKL space [30]. Each CSF is measured by 5 plates, 1 for each spatial frequency evaluated. Each plate contains a series of sinusoidal gratings of achromatic or chromatic decreasing contrast (cone contrast formula) in 2-degree circular windows, arranged in a 4 × 4 grid on an achromatic background with the maximum generable luminance of the device. The orientation of the grid is randomly chosen from 3 possibilities ($-15°$, 0°, and 15°). The grille surround sound is, again, the achromatic stimulus of the device with 60 cd/m^2. To minimize intrusions of the achromatic mechanism, chromatic grids include random achromatic noise.

The Optopad-CSF 1.0 is a fast and non-invasive method to characterize CSF at near distance on a portable electronic display device emitting polarized light (Apple iPad 6th Gen A1893). The iPad Retina screen used had a display size of 2048 × 1536 pixels at 267 pixels per inch, with a screen size of 9.7 in and 8-bit per-channel color resolution. To correctly reproduce the spatial and colorimetric characteristics of the designed stimuli, the device was previously colorimetrically characterized using the 3DLUT method [31].

2.3. Statistical Analysis

Statistical analysis was performed using the SPSS statistical software version 28.0.0 for Windows (IBM SPSS Inc., Chicago, IL, USA). Data distributions were not normally distributed according to the Kolmogorov–Smirnov test; as a result, non-parametric statistics were used. Differences between the CS measurements corresponding to the different spatial frequencies obtained with the digital and conventional procedures were assessed by using the Friedman test, with post hoc analysis with the Wilcoxon test and Bonferroni correction. The Bland–Altman method was used to analyze the level of clinical interchangeability between digital and conventional measures [32]. Spearman's Rho correlation coefficient was calculated to investigate the relationship between different variables.

Only in the case of the comparison of the Optopad achromatic CSF with the VCTS test, both contrasts were calculated using the Michelson contrast formula. The comparison between the results of the two devices used to measure the CSF must be approached with care due to the different design and measurement characteristics of each device. In the current study, this problem was addressed by calculating the value of the index of contrast sensitivity (ICS), based on the study by Koefoed [17]. The normalized value of ICS is obtained by calculating the residuals, with respect to the median of the normal population, for each frequency. Differences were weighted according to the presumed clinical importance of each frequency. Thus, 6 cpd was assigned the highest power (factor 3), whereas the frequencies 3 and 12 cpd received factor 2, and the remaining test frequencies were not weighted.

3. Results

The sample included in this study was composed of 51 patients (51 eyes measured randomly), with a mean age of 36.3 ± 14.0 years (range: 19 to 72 years). There was a higher percentage of women than men (76.5% vs. 23.5%, respectively). The distribution of the eyes evaluated as a function of the refractive status was as follows: 39.2%, myopia; 17.6%, hyperopia; 17.6%, emmetropia; 11.8%, presbyopia; and 13.7%, a combination of myopia and presbyopia.

3.1. Comparative Analysis of Achromatic CSF Measured with Vistech VCTS and Optopad-CSF

Table 1 summarizes the results obtained with the printed and digital test in the sample evaluated. The contrast sensitivity values for each test were calculated using the Michelson contrast formula. No statistically significant differences were found between tests for the spatial frequencies of 1.5 ($p = 0.161$), 3 ($p = 0.138$), and 6 cpd ($p = 0.378$). However, significant differences were found in the contrast sensitivity (CS) measured for 12 cpd with both tests ($p = 0.012$). It should be considered that the highest spatial frequency in the Vistech VCTS test was 18 cpd, whereas it was 24 cpd in the Optopad-CSF test. Mean achromatic ICS values were 0.02 (standard deviation, SD: 1.07; median: 0.17; range: −2.37 to 1.90) and −0.76 (SD: 1.63; median: −1.24; range: −4.30 to 1.69) with the Vistech VCTS and Optopad tests, respectively. The difference was statistically significant ($p < 0.001$).

Table 1. Achromatic contrast sensitivities for each spatial frequency (SF), with the mean value on a logarithmic scale and its corresponding standard deviation, measured with the printed (left) and digital (right) tests. The contrast sensitivity values were calculated with the Michelson contrast formula.

Vistech VCTS			**Optopad-CSF**			***p*-Values**
SF (cpd)	**Mean Value (log)**	**Standard Deviation**	**SF (cpd)**	**Mean Value (log)**	**Standard Deviation**	
1.5	1.62	0.19	1.5	1.68	0.22	0.161
3	1.91	0.23	3	1.83	0.35	0.138
6	1.86	0.29	6	1.82	0.45	0.378
12	1.66	0.36	12	1.51	0.43	0.012
18	1.20	0.26	---	---	---	---
---	---	---	24	0.86	0.42	---

Bland–Altman plots were used to evaluate the level of interchangeability between tests in the near CS measured for the spatial frequencies of 1.5, 3, 6, and 12 cpd (Figure 1). Logarithmic values of CS were used in this comparison. As displayed in Figure 1, the limits of agreement were large and represented potential differences between systems for log CS corresponding to all the spatial frequencies evaluated with clinical relevance. The ranges of agreement, defined as 1.98 times the standard deviations of the differences, were 0.55, 0.76, 0.78, and 0.69 for the spatial frequencies of 1.5, 3, 6, and 12 cpd, respectively.

Table 2 shows the correlation of near CS values measured for each spatial frequency with each test with age. As shown, significant inverse correlations were found between CS and age for the highest spatial frequencies evaluated, although correlations between age and CS were weaker when using the Vistech VCTS test. Figure 2 shows the CSFs measured with Vistech and Optopad tests in groups of subjects defined according to age. As expected, higher CSFs were found in the younger groups with the two tests evaluated, although the Vistech VCTS showed nearly no difference between the first three decades.

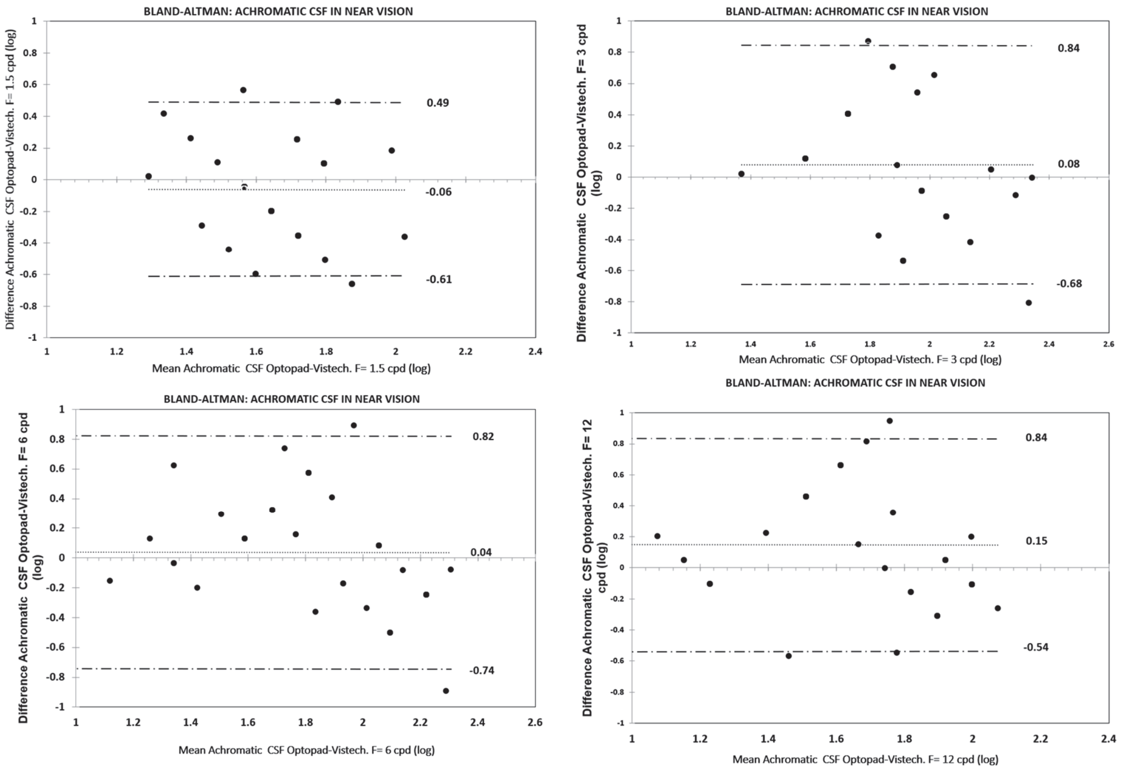


Figure 1. Bland–Altman analysis for near achromatic contrast sensitivity: Vistech VCTS vs. Optopad-CSF (digital test). From left to right, top to bottom: F = 1.5 cpd, F = 3 cpd, F = 6 cpd, and F = 12 cpd, respectively.

Table 2. Correlations (Spearman's Rho: ρ) of achromatic CSF measures obtained with the two tests evaluated and age, with their corresponding statistical significance represented by the *p*-value (*p*) for each frequency.

Optopad-CSF				
1.5	3	6	12	24
ρ = −0.568 $p < 0.001$	ρ = −0.564 $p < 0.001$	ρ = −0.656 $p < 0.001$	ρ = −0.615 $p < 0.001$	ρ = −0.343 $p = 0.014$
VISTECH VCTS				
1.5	3	6	12	18
ρ = −0.117 $p = 0.413$	ρ = −0.123 $p = 0.389$	ρ = −0.406 $p = 0.003$	ρ = −0.403 $p = 0.003$	ρ = −0.303 $p = 0.031$

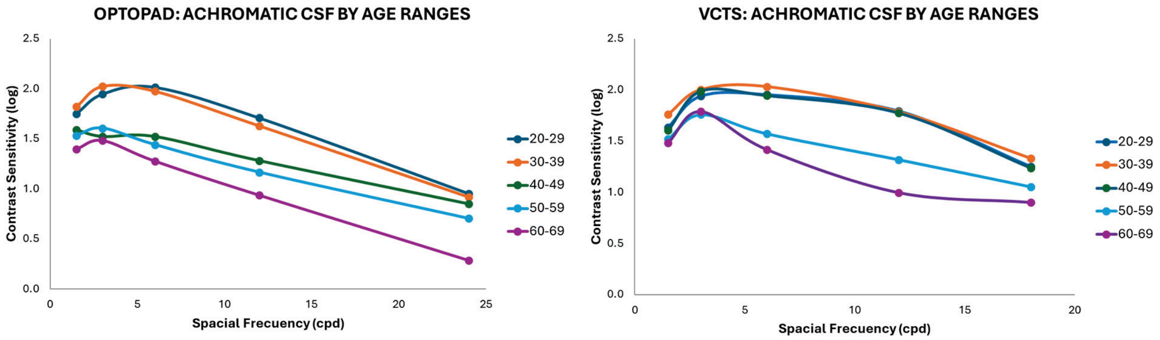


Figure 2. Near achromatic contrast sensitivity functions (CSFs) by age ranges measured with both instruments: Vistech VCTS (**right**) and Optopad-CSF (**left**).

3.2. Analysis of Chromatic CSF Measured with Optopad-CSF

Table 3 summarizes the results obtained with the digital test in terms of red–green (CSF-T) and blue–yellow (CSF-D) spatial contrast sensitivity in the sample evaluated. The contrast sensitivity values for each test were calculated using the cone contrast formula. The mean chromatic ICS values were −20.56 (SD: 0.96; median: −20.49; range: −22.20 to −17.94) and −0.16 (SD: 0.99; median: −0.25; range: −1.87 to 3.70) for the CSF-T and CSF-D plates, respectively. This difference was statistically significant ($p < 0.001$). Weak correlations were found between red–green and blue–yellow CS values for each spatial frequency evaluated. Likewise, as happened with the achromatic CSF, better red–green and blue–yellow CS values were found in the youngest groups of the sample evaluated (Figure 3).

Table 3. Chromatic contrast sensitivities for each spatial frequency (SF), with the mean value on a linear scale and its corresponding standard deviation measured with the digital test.

Optopad-CSF-T			Optopad-CSF-D		
SF (cpd)	**Mean Value (log)**	**Standard Deviation**	**SF (cpd)**	**Mean Value (log)**	**Standard Deviation**
1	2.24	0.03	1	0.86	0.04
2	2.20	0.03	2	0.75	0.04
4	1.93	0.03	4	0.42	0.03
8	1.82	0.05	8	0.31	0.04
12	1.72	0.04	12	0.27	0.04

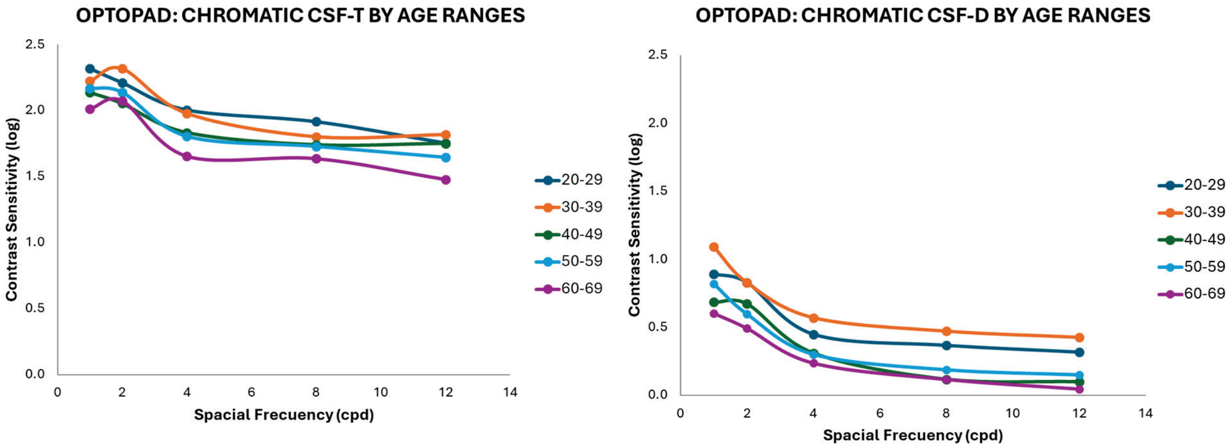


Figure 3. Chromatic contrast sensitivity by age ranges measured with the Optopad-CSF test for the red–green (CSF-T) and blue–yellow (CSF-D) mechanisms.

4. Discussion

As previously mentioned, this study aimed to evaluate and compare two different methodologies for measuring near achromatic CSF: a printed conventional test (Vistech VCTS) (considered as the classic gold standard) and a digital test to be used in previously colorimetrically characterized tablets (Optopad-CSF). Thus, the interchangeability of a classic test and a new digital test for measuring near achromatic CSF in clinical practice is analyzed and can be confirmed. Furthermore, as the digital test permits the measurement of chromatic CSF, an analysis of red–green and blue–yellow CSF in the healthy population evaluated was performed to provide a characterization of these new aspects of the visual function for their use in future studies and clinical practice. To date, the Optopad-CSF test is the only currently available portable device for characterizing near chromatic CSF.

4.1. Comparison of Near Achromatic CSF Measured with Conventional and Digital Tests

Our comparative analysis revealed that there were significant differences between Vistech and Optopad tests in terms of near CSF only for higher spatial frequencies (12 cpd). However, differences between tests for the remainder of the spatial frequencies measured by both tests did not reach statistical significance. This is consistent with the results of Rodriguez-Vallejo et al. [12], who compared a new digital application to measure CSF at 2 m with an iPad Retina against the Functional Acuity Contrast Test (FACT), confirming that differences between tests were not significant when the same contrast sensitivity steps were used. However, despite the absence of significant differences between the Vistech and Optopad tests in the CSF for most of the spatial frequencies evaluated, both tests cannot be used interchangeably. Bland–Altman analysis revealed a significant scatter of the differences between tests for all spatial frequencies evaluated, with ranges of agreement over the inherent variability of the measurement obtained with both systems [22,23,33]. In addition, a slight trend of increased CS values with the Vistech VCTS test was observed, especially for the intermediate frequencies. These findings are coherent, as it should be considered that both tests were designed differently, presenting different contrast steps. Indeed, the higher number of contrast steps provided by the Optopad test may have allowed a more accurate determination of the contrast threshold in some cases.

In any case, both tests showed a correlation of the CS measures obtained with them and age, which confirms the capability of both tests for detecting age-related differences in CSF. It should be mentioned that not all correlations between CS measures obtained with the Vistech VCTS test and age reached statistical significance. Furthermore, in the comparison of CS values according to age, the digital application detected a decrease in the precise discrimination of CS with increasing age [34]. Thus, for all the spatial frequencies, age groups over 40 years of age showed such CS decrease. However, this decrease was only observed for patients over the age of 50 when measuring CSF with the Vistech VCTS test.

To the best of our knowledge, no comparative analysis between near CSF tests (including digital tests) has been performed to this date. Therefore, there is no possibility, according to the scientific evidence, of providing information about which near CSF tests should be used and which differences can be expected to be found with them. This is surprising considering that the measurement of near CSF is crucial when evaluating the efficacy of different methods of compensation for presbyopia [15,16,23,35,36]. However, very few studies used this variable to confirm how the optical correction method applied affects near visual quality. Possibly, an explanation for this could be the limited number of tests available for such purpose. However, more research should be performed on this to obtain information about how to better evaluate near visual performance with presbyopia correction options.

4.2. Measurement of Near Chromatic CSF

Although there are some studies investigating the response of the visual system to chromatic contrast [24–29], comparisons of our results with those from previous studies must be made very carefully, since, generally, the stimuli, lighting conditions, maximum luminance, reproduction devices, and color space can differ. In the case of sensitivity to achromatic contrast, this problem can be minimized by using the same definition of contrast (Michelson) and by comparing data using the ICS parameter [17]. In the case of sensitivity to chromatic contrast, the procedure to follow is not obvious. The color space and the chromatic characteristics of the stimuli can present a very wide variability among studies; this variability can determine the results.

As expected, considering differences in design between tests used for measuring near chromatic CSF in previous studies, our chromatic CSF results, in numerical terms, are very different from those obtained by previous authors, but some trends are shared. Specifically, our results show that when evaluating CSF in the three mechanisms using the cone contrast formula, the RG mechanism presents greater sensitivity than the achromatic mechanism and, in both cases, greater sensitivity than the BY mechanism, as has been shown in other studies [25,34,37]. Kim et al. [25] used a different cone contrast space than that used in our study and obtained the three contrast sensitivity functions using the qCSF approach. They found higher sensitivity measured at 58 cm to the contrast of the L/M cone compared to the S cone and the achromatic responses. Likewise, these authors found correlations between the two chromatic CSFs, but they concluded that these could be attributed to the narrow age range of the patients evaluated. Xu et al. [37] also proposed different conditions in their experiment compared to ours, with stimuli in the directions of cone contrast space but with white, red, yellow, and green backgrounds. Although the results of these two studies are not strictly comparable with ours, it was found that the shapes (band pass for A and low pass for RG and BY) and the relationships between the curves are similar. In our results, as previously mentioned, weak correlations were found between the red–green and blue–yellow CS values for each spatial frequency evaluated.

4.3. Correlation of Age and CS Measured with Optopad-CSF

Ashraf et al. [34] found that the decrease in sensitivity with age was more noticeable in the achromatic CSF for high spatial frequencies and in the chromatic CSFs for low spatial frequencies. These results did not exactly agree with those found in the study. The behavior of the curves was similar for the same lighting level used in our study, but the dependence on age seemed to be more remarkable for medium spatial frequencies for the three channels. This behavior was more evident if sensitivity values were expressed in dB. This difference may be due to the fact that Ashraf and colleagues [34] only evaluated two age groups, whereas five age groups were examined in our study.

Regarding the influence of age, Pearson et al. [38] studied the variation of achromatic and chromatic CS for two stimulus sizes and found that they agreed with the loss of retinal ganglion cells. These authors worked with contrast-sensitivity-modulated stimuli along the luminance, equiluminant L-cone, and equiluminant S-cone axes. The authors indicated that their results showing a decrease in CS with age (0.4–0.7 dB per decade) were consistent with those from other similar studies. Although the characteristics of the stimuli used in the test evaluated here were different, the CSF variation with age (in dB) for each spatial frequency and for the three measured channels was also analyzed. In CSF-A, the greatest decrease in sensitivity occurred between the first two decades (20 to 39) and the rest (40 to 69), especially for low and medium spatial frequencies. For the highest spatial frequencies, the greatest decrease occurred in the last decade. The variations were in the range between 0.1 and 0.5 dB per decade. In the CSF-T and D, the decrease varied between 0.1 and 0.2 dB per decade, a much more stable behavior than in the case of achromatic measurement. In both cases, the variations were smaller than those reported by Pearson et al. [38].

4.4. Limitations of the Study

This study has some limitations that should be acknowledged. First, the selection of a printed test as the gold standard for comparison can be considered as a potential limitation of the study, because this test only allows a very gross measurement of the CSF. It should be considered that our aim was to evaluate the clinical performance of a new low-cost, portable device to test near CSF, the Optopad test, that uses a Retina screen driven by an 8-bit graphic card. A graphic card with greater resolution would have been ideal for measuring the CSF more precisely but would have required a more expensive set-up not easily accessible for clinicians and may have only been useful for research purposes. The panel design of the Optopad test takes the limitations of the actual graphic card into account. In the first place, the low average luminance of the stimulus and the achromatic noise masking the chromatic stimuli and the high luminance of the surrounding area help to reduce the subject's sensitivity: the lowest generable contrast is not visible by any of the subjects used in the design stage, and the lowest contrast in the panel is greater than this value. On the other hand, with the panel design, the subject's threshold is not determined precisely, but it is determined by which fixed class or category this threshold belongs (the range between the last stimuli seen and the first stimuli not seen). In this line, it makes sense to compare our device with another with a similar measurement strategy, and for this reason, the VCTS test was chosen for comparison. The sample size can be considered as a second limitation of the study. The sample size could not be increased due to logistic problems. However, in the near future, we plan to enlarge it. Therefore, this study can be considered as the preliminary evidence of the potential usefulness of the Optopad test, which should be investigated further in future studies. Specifically, more analysis is required in studies with larger samples to better characterize the age-related changes in the measurements evaluated, as well as the influence of other factors such as sex or ethnicity.

Another aspect that may be considered as a potential limitation is that it is known that children and adults tend to have neural biases to gratings of different orientations [39]. As the test consists of the discrimination of the orientation of gratings, this potential preference may constitute a source of bias. However, this is especially present where there is an uncorrected refractive error, and in our sample, all subjects were evaluated with spectacle correction. Furthermore, the orientations of the gratings in the Optopad-CSF test are similar to those used in the VCTS test in order to avoid the well-known oblique effect that occurs in the human visual system when presenting gratings at 45°. Therefore, the potential contribution of this factor seems to be residual. In addition, none of the patients tested in our sample provided a response demonstrating a clear preference over a specific orientation, and consequently, this phenomenon was not a limiting factor.

5. Conclusions

In conclusion, the Optopad-CSF digital test allows rapid measurement of near achromatic and chromatic contrast sensitivity using a colorimetrically calibrated iPad. The chromatic and achromatic measurements provided by this device varied with age following a decreasing pattern compatible with those reported for other tests evaluating near contrast sensitivity. In terms of achromatic contrast sensitivity, the measurements obtained with the digital test cannot be used interchangeably with those provided by a conventional printed test (Vistech VCTS test). This lack of interchangeability may be mainly attributed to the different contrast steps used in each test. The digital test evaluated comprises more contrast step options; therefore, it presents a greater potential for providing accurate measurement of the contrast threshold. The measurements of the CSF in the three mechanisms using the cone contrast formula with the Optopad-CSF test are compatible with those obtained in previous experiments in terms of the shapes of the CSF curves (band pass achromatic CSF and low pass CSF-T and CSF-D) and the relationships between them. Considering that, in addition, similar age-related patterns were found in comparison to previous experiences with other tests, the Optopad-CSF provides a potentially useful measurement of near chromatic CSF. All this preliminary evidence must be investigated further in future studies with larger samples in order to define consistent ranges of normality for all CSF variables measured with this digital test. With this normality data, additional studies should be conducted with the Optopad-CSF test to confirm whether it can detect alterations in different types of conditions in which chromatic discrimination must be clearly affected.

Author Contributions: Conceptualization, D.d.F., D.P.P. and M.J.L.; methodology, D.d.F., D.P.P., M.J.L. and K.J.M.-G.; software, D.d.F. and M.J.L.; validation, D.d.F., D.P.P., M.J.L. and K.J.M.-G.; formal analysis, D.d.F., D.P.P. and K.J.M.-G.; investigation, D.d.F., D.P.P., M.J.L. and K.J.M.-G.; resources, D.d.F. and D.P.P.; data curation, K.J.M.-G.; writing—original draft preparation, K.J.M.-G.; writing—review and editing, D.d.F., D.P.P. and M.J.L.; visualization, D.d.F., D.P.P., M.J.L. and K.J.M.-G.; supervision, D.d.F. and D.P.P.; project administration, D.d.F. and D.P.P.; funding acquisition, D.d.F. and D.P.P. All authors have read and agreed to the published version of the manuscript.

Funding: Research action financed by Conselleria d'Innovació, Universitats, Ciència i Societat Digital of Generalitat Valenciana through the AICO/2021/130 project within the framework of grants to consolidated research groups AICO 2021.

Institutional Review Board Statement: This study was approved by the Ethics Committee of the University of Alicante (Date: 26 February 2021. Exp. UA-2021-02-17) and was carried out in accordance with the 1964 Helsinki declaration and its later amendments or comparable ethical standards.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Data are available upon reasonable request to the authors.

Conflicts of Interest: María José Luque and Dolores de Fez have the intellectual property of the Optopad-CSF test, which currently is not commercially available. All authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest (such as honoraria; educational grants; participation in speakers' bureaus; membership, employment, consultancies, stock ownership, or other equity interest; and expert testimony or patent-licensing arrangements).

References

1. Soler, F.; Sánchez-García, A.; Molina-Martín, A.; Fez Saiz, D.D.; Díaz, V.; Piñero, D.P. Analysis of the characteristics of electronic equipment usage distance for common users. *Guoji Yanke Zazhi (Int. Eye Sci.)* **2021**, *21*, 1508–1514.
2. Soler, F.; Sánchez-García, A.; Molina-Martin, A.; de Fez, D.; Díaz, V.; Piñero, D.P. Differences in visual working and mobile phone usage distance according to the job profile. *Curr. Eye Res.* **2021**, *46*, 1240–1246. [CrossRef] [PubMed]
3. Hernández Yáñez, L. *Jornada «Aprendizaje Eficaz con TIC en la UCM»*; Ediciones Complutense: Madrid, Spain, 2022.
4. Castro-Sánchez, A. El Uso de las TICs en la Prevención de Riesgos Laborales. Ph.D. Thesis, University of Valladolid, Valladolid, Spain, 2022. Available online: https://uvadoc.uva.es/handle/10324/55018 (accessed on 15 March 2023).
5. Rono, H.; Bastawrous, A.; Macleod, D.; Wanjala, E.; Gichuhi, S.; Burton, M. Peek Community Eye Health—mHealth system to increase access and efficiency of eye health services in Trans Nzoia County, Kenya: Study protocol for a cluster randomised controlled trial. *Trials* **2019**, *20*, 502. [CrossRef] [PubMed]
6. Mena-Guevara, K.J.; Piñero, D.P.; de Fez, D. Validation of digital applications for evaluation of visual parameters: A narrative review. *Vision* **2021**, *5*, 58. [CrossRef]
7. Rodríguez-Vallejo, M.; Llorens-Quintana, C.; Furlan, W.D.; Monsoriu, J.A. Visual acuity and contrast sensitivity screening with a new iPad application. *Displays* **2016**, *44*, 15–20. [CrossRef]
8. Mena-Guevara, K.J.; de Fez, D.; Molina-Martín, A.; Piñero, D.P. Binocular vision measurements with a new online digital platform: Comparison with conventional clinical measures. *Clin. Exp. Optom.* **2023**. *ahead of print*. [CrossRef]
9. Rodríguez-Vallejo, M.; Vicente Ferrando, D.M.; Monsoriu, J.A.; Furlan, W.D. Stereopsis assessment at multiple distances with an iPad application. *Displays* **2017**, *50*, 35–40. [CrossRef]
10. Budai, A.; Czigler, A.; Mikó-Baráth, E.; Nemes, V.A.; Horváth, G.; Pusztai, Á.; Piñero, D.P.; Jandó, G. Validation of dynamic random dot stereotests in pediatric vision screening. *Graefes Arch. Clin. Exp. Ophthalmol.* **2019**, *257*, 413–423. [CrossRef]
11. Portela-Camino, J.A.; Martín-González, S.; Ruiz-Alcocer, J.; Illarramendi-Mendicute, I.; Garrido-Mercado, R. An evaluation of the agreement between a computerized stereoscopic game test and the TNO stereoacuity test. *Clin. Optom.* **2021**, *13*, 181–190. [CrossRef]
12. Rodríguez-Vallejo, M.; Remón, L.; Monsoriu, J.A.; Furlan, W.D. Designing a new test for contrast sensitivity function measurement with iPad. *J. Optom.* **2015**, *8*, 101–108. [CrossRef]
13. Fernández, J.; Rodríguez-Vallejo, M.; Tauste, A.; Albarrán, C.; Basterra, I.; Piñero, D. Fast measure of visual acuity and contrast sensitivity defocus curves with an iPad application. *Open Ophthalmol. J.* **2019**, *13*, 15–22. [CrossRef]
14. de Fez, D.; Luque, M.J.; Matea, L.; Piñero, D.P.; Camps, V.J. New iPAD-based test for the detection of color vision deficiencies. *Graefes Arch. Clin. Exp. Ophthalmol.* **2018**, *256*, 2349–2360. [CrossRef] [PubMed]
15. Gil, M.A.; Varón, C.; Cardona, G.; Vega, F.; Buil, J.A. Comparison of far and near contrast sensitivity in patients symmetrically implanted with multifocal and monofocal IOLs. *Eur. J. Ophthalmol.* **2014**, *24*, 44–52. [CrossRef]
16. García-Lázaro, S.; Albarrán-Diego, C.; Ferrer-Blasco, T.; Radhakrishnan, H.; Montés-Micó, R. Visual performance comparison between contact lens-based pinhole and simultaneous vision contact lenses. *Clin. Exp. Optom.* **2013**, *96*, 46–52. [CrossRef] [PubMed]
17. Koefoed, V.F.; Baste, V.; Roumes, C.; Høvding, G. Contrast sensitivity measured by two different test methods in healthy, young adults with normal visual acuity. *Acta Ophthalmol.* **2015**, *93*, 154–161. [CrossRef] [PubMed]
18. Karampatakis, V.; Papadopoulou, E.P.; Almpanidou, S.; Karamitopoulos, L.; Almaliotis, D. Evaluation of contrast sensitivity in visually impaired individuals using K-CS test. A novel smartphone-based contrast sensitivity test-Design and validation. *PLoS ONE* **2024**, *19*, e0288512. [CrossRef]
19. Hegde, P.G.; Rao, D.; Prudhvi, B.; Hegde, N. Digital contrast sensitivity chart with varying visual acuity: Development and validation. *Oman J. Ophthalmol.* **2023**, *16*, 467–471. [CrossRef]
20. Bühren, J.; Terzi, E.; Bach, M.; Wesemann, W.; Kohnen, T. Measuring contrast sensitivity under different lighting conditions: Comparison of three tests. *Optom. Vis. Sci.* **2006**, *83*, 290–298. [CrossRef] [PubMed]
21. de Fez, M.D.; Luque, M.J. The Optopad Project. 2023. Available online: https://web.ua.es/en/optopad/optopad-project.html (accessed on 15 March 2023).

22. Coco-Martín, M.B.; Leal-Vega, L.; Alcoceba-Herrero, I.; Molina-Martín, A.; de-Fez, D.; Luque, M.J.; Dueñas-Gutiérrez, C.; Arenillas-Lara, J.F.; Piñero, D.P. Visual perception alterations in COVID-19: A preliminary study. *Int. J. Ophthalmol.* **2023**, *16*, 1–9. [CrossRef]
23. Piñero, D.P.; Molina-Martin, A.; Ramón, M.L.; Rincón, J.L.; Fernández, C.; de Fez, D.; Arenillas, J.F.; Leal-Vega, L.; Coco-Martín, M.B.; Maldonado, M.J. Preliminary evaluation of the clinical benefit of a novel visual rehabilitation program in patients implanted with trifocal diffractive intraocular lenses: A blinded randomized placebo-controlled clinical trial. *Brain Sci.* **2021**, *11*, 1181. [CrossRef]
24. Wuerger, S.; Ashraf, M.; Kim, M.; Martinovic, J.; Pérez-Ortiz, M.; Mantiuk, R.K. Spatio-chromatic contrast sensitivity under mesopic and photopic light levels. *J. Vis.* **2020**, *20*, 23. [CrossRef] [PubMed]
25. Kim, Y.J.; Reynaud, A.; Hess, R.F.; Mullen, K.T. A Normative Data Set for the Clinical Assessment of Achromatic and Chromatic Contrast Sensitivity Using a qCSF Approach. *Investig. Ophthalmol. Vis. Sci.* **2017**, *58*, 3628–3636. [CrossRef]
26. Kim, K.J.; Mantiuk, R.; Lee, K.H. Measurements of achromatic and chromatic contrast sensitivity functions for an extended range of adaptation luminance. *Proc. SPIE* **2013**, *8651*, 86511A.
27. Wong, R.; Khan, J.; Adewoyin, T.; Sivaprasad, S.; Arden, G.B.; Chong, V. The ChromaTest, a digital color contrast sensitivity analyzer, for diabetic maculopathy: A pilot study. *BMC Ophthalmol.* **2008**, *8*, 15. [CrossRef] [PubMed]
28. de Fez, M.D.; Luque, M.J.; Viqueira, V. Enhancement of contrast sensitivity and losses of chromatic discrimination with tinted lenses. *Optom. Vis. Sci.* **2002**, *79*, 590–597. [CrossRef] [PubMed]
29. Bodduluri, L.; Boon, M.Y.; Ryan, M.; Dain, S.J. Normative values for a tablet computer-based application to assess chromatic contrast sensitivity. *Behav. Res. Methods* **2018**, *50*, 673–683. [CrossRef] [PubMed]
30. Smith, V.C.; Pokorny, J. The design and use of a cone chromaticity space: A tutorial. *Color Res. Appl.* **1996**, *21*, 375–383. [CrossRef]
31. de Fez, D.; Luque, M.J.; García-Domene, M.C.; Camps, V.; Piñero, D. Colorimetric characterization of mobile devices for vision applications. *Optom. Vis. Sci.* **2016**, *93*, 85–93. [CrossRef] [PubMed]
32. Bland, J.M.; Altman, D.G. Statistical methods for assessing agreement between two methods of clinical measurement. *Lancet* **1986**, *1*, 307–310. [CrossRef]
33. Scialfa, C.T.; Tyrrell, R.A.; Garvey, P.M.; Deering, L.M.; Leibowitz, H.W.; Goebel, C.C. Age differences in Vistech near contrast sensitivity. *Am. J. Optom. Physiol. Opt.* **1988**, *65*, 951–956. [CrossRef]
34. Ashraf, M.; Wuerger, S.; Kim, M.; Saunderson, H.; Martinović, J.; Mantiuk, R. Spatio-chromatic contrast sensitivity across the life span: Interactions between age and light level in high dynamic range. *J. Vis.* **2020**, *20*, 1286. [CrossRef]
35. Gil, M.Á.; Varón, C.; Cardona, G.; Buil, J.A. Far and Near Contrast Sensitivity and Quality of Vision with Six Presbyopia Correcting Intraocular Lenses. *J. Clin. Med.* **2022**, *11*, 4150. [CrossRef] [PubMed]
36. Łabuz, G.; Auffarth, G.U.; Özen, A.; van den Berg, T.J.T.P.; Yildirim, T.M.; Son, H.S.; Khoramnia, R. The Effect of a Spectral Filter on Visual Quality in Patients with an Extended-Depth-Of-Focus Intraocular Lens. *Am. J. Ophthalmol.* **2019**, *208*, 56–63. [CrossRef] [PubMed]
37. Xu, Q.; Ye, Q.; Mantiuk, R.; Luo, M. A Study of spatial chromatic contrast sensitivity based on different colour background. *CIC* **2022**, *30*, 236–240. [CrossRef]
38. Pearson, P.M.; Schmidt, L.A.; Ly-Schroeder, E.; Swanson, W.H. Ganglion cell loss and age-related visual loss: A cortical pooling analysis. *Optom. Vis. Sci.* **2006**, *83*, 444–454. [CrossRef]
39. Yap, T.P.; Luu, C.D.; Suttle, C.M.; Chia, A.; Boon, M.Y. The development of meridional anisotropies in neurotypical children with and without astigmatism: Electrophysiological and psychophysical findings. *Vision Res.* **2024**, *222*, 108439. [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

Technologies Editorial Office
E-mail: technologies@mdpi.com
www.mdpi.com/journal/technologies

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.

MDPI

Academic Open
Access Publishing

mdpi.com

ISBN 978-3-7258-6094-4