

Article

Lightweight Transformer-Enhanced YOLOv11 for Real-Time Fabric Defect Detection: A Systematic Comparison with DETR-Based Architectures

Makara Mao ¹ and Min Hong ^{2,*} 

¹ Department of Information Technology Engineering, Faculty of Engineering, Royal University of Phnom Penh, Phnom Penh 12156, Cambodia; mao.makara@rupp.edu.kh

² Department of Computer Software Engineering, Soonchunhyang University, Asan-si 31538, Republic of Korea

* Correspondence: mhong@sch.ac.kr

Abstract

Fabric defect detection is a critical task in automated textile quality inspection, where both high localization accuracy and fast inference are required, especially for small and irregular defects. Although recent YOLO-based detectors offer a favorable balance of speed and accuracy, their convolutional architectures primarily capture local features and are less effective at detecting defects that require broader contextual understanding. Transformer-based detectors improve global context modeling but typically introduce higher computational complexity. To balance this trade-off, this study proposes a lightweight hybrid detector, YOLOv11-Transformer, which integrates multi-head self-attention (MHSA) modules into the backbone and neck of YOLOv11. The model was evaluated on a fabric defect dataset of 4375 images covering four defect categories: holes, cuts, thread errors, and metallic contamination. Comparative experiments were conducted against YOLOv5, YOLOv8, YOLOv9, YOLOv10, YOLOv11, YOLOv11-DETR, and RT-DETR under unified training conditions, with additional efficiency benchmarks against YOLOv6 and YOLOv7-Tiny. The proposed model achieved the best mAP₅₀₋₉₅ of 94.50%, outperforming the baseline YOLOv11 by 2.40 percentage points, while maintaining efficient inference at 14.2 ms per image (approximately 70 FPS) on an NVIDIA RTX 4060 GPU. These results indicate that lightweight attention integration can improve fine-grained defect localization while preserving a practical accuracy-efficiency trade-off for automated textile inspection.

Keywords: YOLO; transformer; RT-DETR; real-time; fabric defect detection



Academic Editors: Yuexiang Li and
Faqian Guan

Received: 16 June 2026

Revised: 13 July 2026

Accepted: 17 July 2026

Published: 23 July 2026

Copyright: © 2026 by the authors.
Licensee MDPI, Basel, Switzerland.
This article is an open access article
distributed under the terms and
conditions of the [Creative Commons
Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

The textile industry has undergone a rapid development over the past decade, with automation and digital manufacturing technologies transforming production capabilities [1]. However, this rapid growth has intensified the challenge of maintaining fabric quality, where even microscopic defects [2] can compromise product performance, aesthetic appeal, and market value. Traditional inspection methods [3], which predominantly rely on manual visual assessment, present significant limitations, including inconsistent detection rates, operator fatigue-induced errors, and an inability to meet the demands of high-speed production [4].

The inadequacy of conventional approaches to identifying subtle anomalies such as micro-cuts, pinhole defects, thread misalignments, and metallic contamination has created an urgent need for intelligent, automated detection systems that operate in real time with

high accuracy [5]. Recent advances in deep learning have catalyzed a paradigm shift in industrial inspection applications, with object detection algorithms demonstrating strong performance for defect identification [6]. The You Only Look Once (YOLO) architecture family has emerged as a leading solution, distinguished by its favorable balance between detection speed and accuracy, particularly for small-scale object detection scenarios critical in fabric inspection [7].

The architectural evolution from YOLOv5 through YOLOv11 has incorporated substantial enhancements, including refined network topologies, sophisticated loss functions, and advanced feature extraction methods, culminating in measurable improvements in both detection precision and computational efficiency [8]. The recent integration of Transformer-based attention mechanisms [9] represents a promising direction, enabling models to capture complex spatial relationships and long-range dependencies, which are essential for identifying intricate fabric defect patterns [10].

This paper presents a comparative analysis of state-of-the-art object detection architectures for fabric defect detection, including YOLOv5, YOLOv8, YOLOv9, YOLOv10, YOLOv11, RT-DETR, YOLOv11-DETR, and the proposed YOLOv11-Transformer. The evaluation is conducted on a fabric defect dataset containing four categories: cut, hole, metal contamination, and thread error.

Experimental results show progressive improvements across YOLO variants, with mAP50-95 increasing from 88.10% (YOLOv5) to 92.10% (YOLOv11). Transformer-enhanced models further improve performance, with YOLOv11-DETR achieving 93.30% and the proposed YOLOv11-Transformer achieving 94.50%.

Despite recent progress in YOLO-based and Transformer-based detection, two limitations remain for fabric defect inspection. First, many existing studies emphasize overall detection accuracy but provide limited analysis of localization quality under stricter IoU thresholds, which is essential for small and irregular defects. Second, fully Transformer-based detectors often incur substantial computational overhead, reducing their suitability for latency-sensitive industrial inspection. Motivated by these limitations, this study investigates whether lightweight self-attention can be selectively integrated into YOLOv11 to improve localization performance while preserving the efficiency of a single-stage detector. The main contributions of the study are summarized as follows:

- We propose a YOLOv11-Transformer architecture that integrates multi-head self-attention modules into both the backbone and neck to enhance global feature representation while maintaining the efficiency of a single-stage detector.
- We design a lightweight attention integration strategy that improves contextual modeling without incurring the high computational cost of full Transformer-based encoder-decoder architectures.
- We conduct a systematic evaluation of multiple object detection models under unified experimental settings to ensure consistent and fair comparison.
- We analyze the impact of the proposed architecture on detection performance, showing improved mAP50-95 for fine-grained defect localization while maintaining near-real-time inference performance.
- We examine the trade-off between accuracy, model size, and inference speed, demonstrating that the proposed model achieves competitive performance with relatively low computational overhead.

The paper is structured as follows: Section 2 examines existing literature on deep learning approaches to fabric defect detection, and Section 3 details our dataset development and experimental methodology. Section 4 presents comprehensive results and performance analysis. Section 5 provides concluding remarks and identifies promising directions for future research.

2. Related Work

Computer vision techniques have significantly advanced automated quality inspection in the textile industry, enabling rapid detection of defects that would otherwise be difficult to identify manually [11]. Early computer vision approaches relied on traditional image processing techniques, such as thresholding, edge detection, and texture analysis [12,13]. While these methods achieved moderate success, they were susceptible to variations in fabric texture, illumination, and noise, which limited their robustness in real-world manufacturing environments [14].

The emergence of convolutional neural networks (CNNs) has revolutionized the field by enabling the direct learning of hierarchical representations from raw data [15]. CNN-based models demonstrated higher generalization and accuracy in detecting complex fabric defects, such as broken yarns, pinholes, and irregular weaves [16,17]. Subsequently, deep object detection frameworks, including Faster R-CNN [18], SSD [19], and YOLO [20], have been widely adopted for defect detection tasks. Among these, YOLO has become a preferred choice due to its ability to perform real-time detection without compromising accuracy.

2.1. Evolution of YOLO for Defect Detection

The YOLO family of models has undergone several architectural refinements since its initial release, each iteration improving detection accuracy, speed, and robustness. YOLOv1 [21] introduced a single-stage detection paradigm that reframed object detection as a regression problem, significantly reducing inference time compared to region-based approaches. However, its limitations in detecting small objects prompted subsequent upgrades.

YOLOv2 [22] improved detection accuracy and recall by introducing anchor boxes, batch normalization, and a high-resolution classifier. It replaced the original backbone with Darknet-19, enabling faster computation while maintaining strong feature extraction capability [23]. Additionally, multi-scale training enhanced robustness to varying input sizes, and the use of dimension clusters improved bounding box prediction. These refinements significantly improved small-object detection and overall performance, making YOLOv2 more suitable for industrial defect detection tasks.

YOLOv3 and YOLOv4 [24,25] improved performance through multi-scale feature extraction, enhanced backbone networks (e.g., Darknet-53), and proposed loss functions, making them more suitable for fine-grained defect detection. YOLOv5, developed by Ultralytics, further optimized training pipelines and model sizes: Nano (N), Small (S), Medium (M), and Large (L) to achieve an optimal trade-off between accuracy and speed [26]. It became widely used for industrial tasks due to its ease of deployment and flexibility.

The recent versions, YOLOv8 to YOLOv11, introduced significant architectural advancements, including decoupled heads for classification and localization, adaptive anchor-free mechanisms, and improved feature pyramids [27–30]. These modifications enhanced the models' ability to detect small and subtle fabric defects while maintaining real-time performance, as shown in Figure 1.

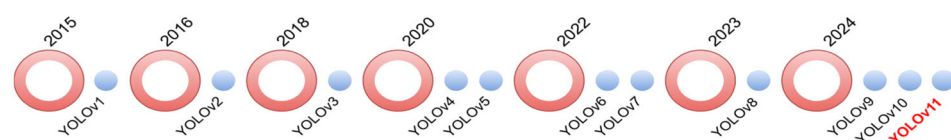


Figure 1. Timeline of YOLO architecture evolution from YOLOv1 to YOLOv11.

2.2. Transformer-Based Detection Models

While CNNs remain dominant in object detection, the integration of Transformer-based attention mechanisms has marked a significant shift in the paradigm. Vision Trans-

formers (ViTs) [31] capture global contextual relationships, which are especially advantageous for detecting complex and irregular patterns in fabrics. Models like DETR [32] and its variants have demonstrated competitive performance by leveraging self-attention to model long-range dependencies.

In this context, hybrid models that combine YOLO with Transformer modules have emerged. For example, YOLOv5-Transformer and YOLOv11-Transformer integrate attention layers into the neck or backbone to improve localization accuracy for small defects [33]. Similarly, RT-DETR [34], a real-time Transformer-based detection framework, eliminates reliance on hand-crafted anchors while maintaining high inference speeds, making it a strong candidate for industrial applications.

2.3. Fabric Defect Detection with YOLO

Numerous studies have applied YOLO to textile quality control, reporting higher detection performance than traditional CNNs or classical image processing techniques. For example, studies using YOLOv4 and YOLOv5 [35] have demonstrated over 90% detection accuracy for common defect types, such as broken yarns and stains. Recent research indicates that Transformer-enhanced YOLO variants achieve further gains by more effectively modeling intricate defect boundaries and irregular textures. These findings underscore the potential of combining the YOLO family with attention-based mechanisms to meet the stringent demands of modern textile inspection.

3. Methodology

3.1. Object Detection

In this study, fabric defect detection is formulated as an object detection task, where the model predicts bounding boxes and class labels for four defect categories: hole, cut, thread error, and metal contamination. The main challenge is detecting small and irregular defects, as shown in Figure 2, whose visual appearance can be subtle and highly dependent on surrounding texture patterns.

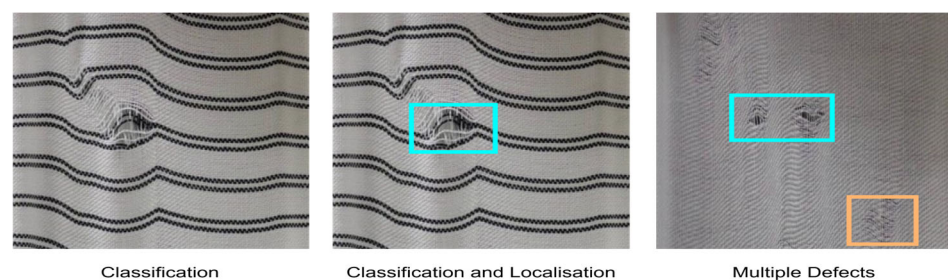


Figure 2. The single and multiple objects in an image: classification, localization, and multiple defects.

Modern deep learning-based detectors, particularly single-stage models such as the YOLO family, have revolutionized the field by offering a balanced trade-off between detection speed and accuracy. These models use CNNs to process images in a single forward pass, enabling real-time performance while maintaining high precision.

However, challenges such as detecting small, irregular defects, handling class imbalance, and maintaining robustness under varying lighting conditions remain key concerns. To address these limitations, recent advancements incorporate attention mechanisms, Transformer-based architectures, and improved feature pyramids to capture both local and global context, significantly enhancing the reliability of object detection in industrial applications.

3.2. Dataset

The dataset used in this study comprises 4375 images of fabric defects at a resolution of 608×608 pixels. It includes four defect categories: hole, cut, thread error, and metal contamination, with an approximately balanced sample distribution to reduce training bias. The dataset was collected in-house by our laboratory (Computer Graphics Lab) from production samples and material testing processes; representative examples of each defect type are illustrated in Figure 3. The dataset was divided into training (3500 images, 80%), validation (438 images, 10%), and test (437 images, 10%) sets using stratified sampling to maintain balanced class distributions; images from the same fabric sequence and production batch were assigned exclusively to a single subset to prevent data leakage. As illustrated in Figure 3, the remaining images were collected from additional production samples and material testing processes. The data were acquired under controlled imaging conditions using a material testing machine with consistent lighting to ensure stable visual quality.

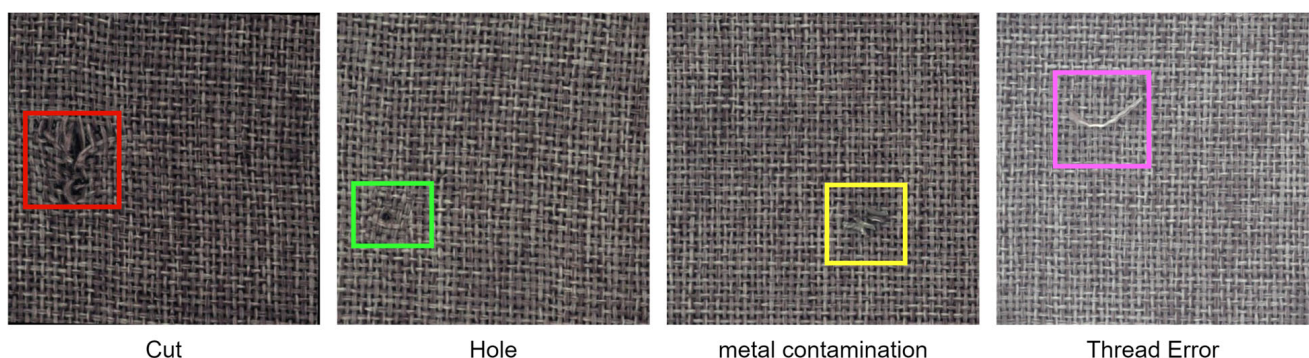


Figure 3. Representative defect type in the common fabric defect dataset.

3.3. Data Augmentation

This section describes the annotation and data augmentation pipeline used to prepare the fabric defect dataset for model training. All defect instances were manually annotated with LabelMe, an open-source graphical image annotation tool that supports bounding-box labeling for object detection. For each image, annotators drew axis-aligned bounding boxes around visible defect regions and assigned the corresponding class label (hole, cut, thread error, or metal contamination). LabelMe then exported the annotations as JSON files, which were converted to the YOLO-compatible label format with a custom Python script.

To improve model generalization and reduce the risk of overfitting, a set of augmentation transformations was applied offline to the annotated images using Python scripts integrated into the LabelMe workflow. As illustrated in Figure 4, these transformations include horizontal and vertical flips, rotations of $+45^\circ$ and -45° , brightness enhancement and contrast adjustments, scaling, cropping, and Gaussian noise injection. In particular, each annotated image was rotated by $\pm 45^\circ$ to account for arbitrary defect orientations, and the brightness was increased to simulate stronger illumination conditions on the inspection line. The augmentation parameters were carefully selected to simulate realistic variations encountered in industrial fabric inspection, including changes in fabric orientation, non-uniform illumination, and background texture variability. Corresponding bounding box coordinates were updated automatically for each transformation to preserve label accuracy. These strategies are particularly important for fabric defect detection, where defects such as micro-cuts and thread irregularities can appear at arbitrary locations, orientations, and scales within the image.

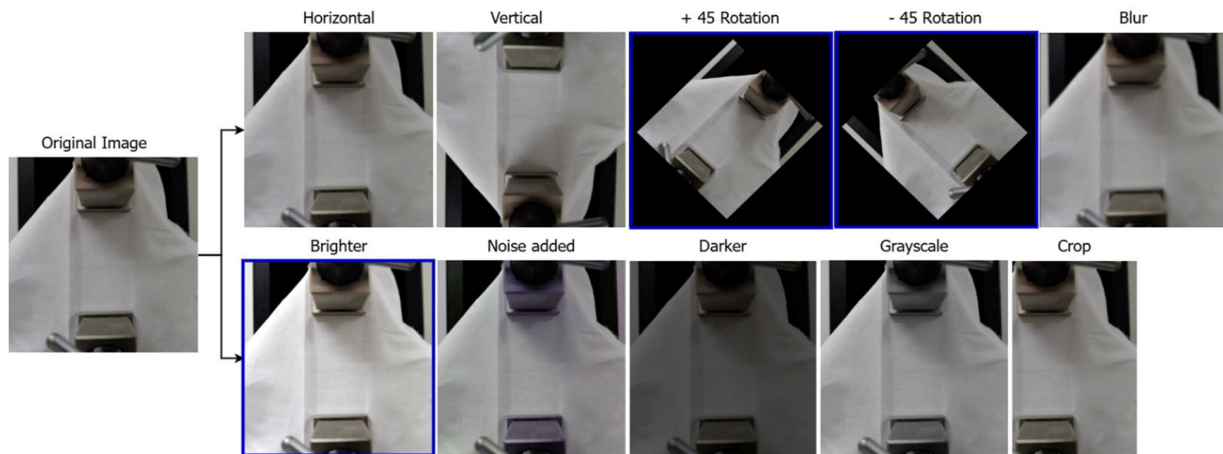


Figure 4. Data augmentation techniques applied during training include horizontal/vertical flips, rotation, brightness adjustment, scaling, cropping, and Gaussian noise.

By expanding the dataset through this augmentation pipeline, the effective training set size was substantially increased without additional data collection. The augmented images expose the model to a wider range of defect appearances, enabling more robust feature learning and improving detection reliability across the varied conditions typical of automated textile inspection.

3.4. Architecture of YOLO Variant, RT-DETR, and Transformer

To ensure a systematic and fair evaluation of fabric defect detection, we compare representative convolution-based and Transformer-based object detection frameworks, including YOLO variants (YOLOv5 to YOLOv11), RT-DETR, and the proposed YOLOv11-Transformer, as illustrated in Figures 5 and 6. These models were selected to analyze the impact of progressive architectural improvements and global attention mechanisms on detecting small and subtle defects.

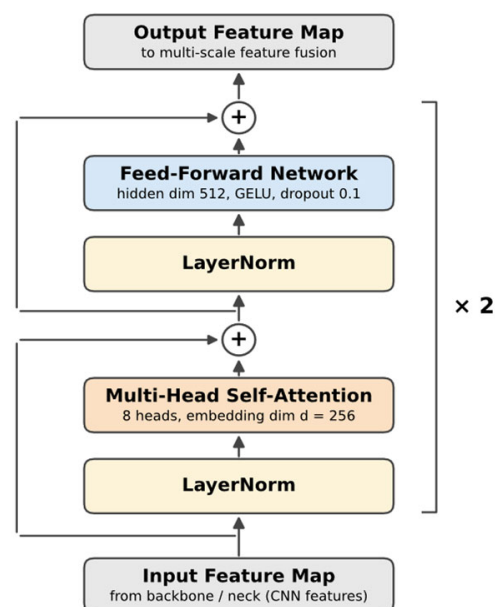


Figure 5. Standard Transformer encoder block used in YOLOv11-Transformer, consisting of multi-head self-attention (MHSA), layer normalization, and a feed-forward network (FFN) with residual connections.

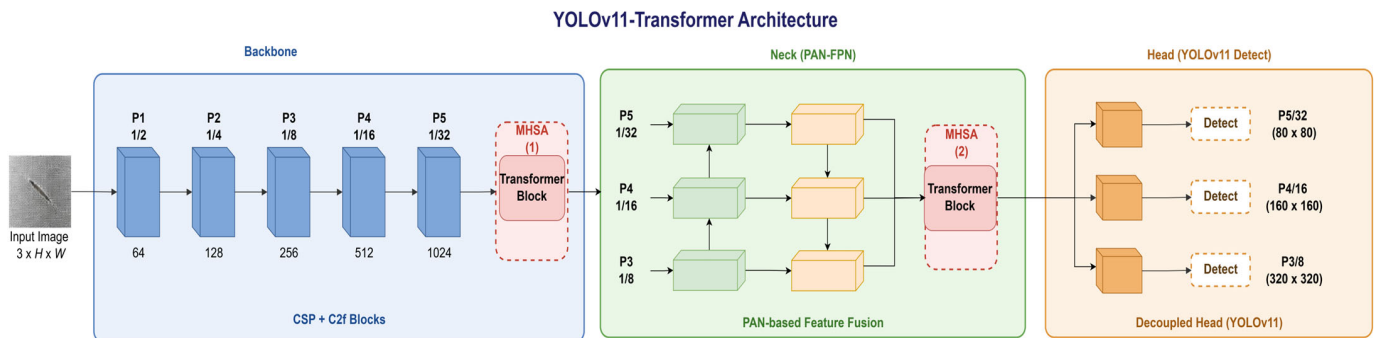


Figure 6. Proposed YOLOv11-Transformer architecture. Multi-head self-attention (MHSA) modules are inserted at two locations: (1) after the backbone’s final high-level feature extraction stage (P5), and (2) within the PAN-based neck before multi-scale feature fusion. Each MHSA block consists of layer normalization, multi-head self-attention (8 heads, embedding dimension 256), a feed-forward network (hidden dimension 512, GELU activation), and a residual connection with a dropout rate of 0.1. All other components remain identical to the baseline YOLOv11.

The YOLO family serves as a strong real-time detection baseline due to its efficient single-stage design. Recent versions (YOLOv8 to YOLOv11) incorporate key advancements, including decoupled detection heads, anchor-free prediction, and enhanced feature pyramid networks (FPN/PAN), which improve localization accuracy and multi-scale feature representation. Despite these improvements, convolutional operations inherently rely on local receptive fields, which limit their ability to capture the long-range spatial dependencies needed to detect fine-grained defects.

In contrast, RT-DETR adopts a Transformer-based paradigm, replacing anchor-based detection with a query-driven approach. It combines a CNN backbone with a multi-scale Transformer encoder–decoder to model global contextual relationships. While effective, this design introduces significant computational complexity and latency, which restrict its applicability in real-time industrial scenarios.

In addition to RT-DETR, we evaluate a hybrid variant, denoted YOLOv11-DETR, which retains the YOLOv11 backbone but replaces the original YOLO detection head with a DETR-style Transformer encoder–decoder head that uses object queries. This variant serves as an intermediate design point between the fully convolutional YOLOv11 and the fully query-based RT-DETR. It should be distinguished from the proposed YOLOv11-Transformer, which retains the complete YOLOv11 detection pipeline, including its anchor-free detection head, and only inserts lightweight encoder-only multi-head self-attention (MHSA) blocks after the backbone’s final feature-extraction stage and within the PAN-based neck. Consequently, YOLOv11-DETR changes the detection paradigm (set-based prediction with a decoder), whereas YOLOv11-Transformer changes only the feature representation, which explains their large difference in model size (32.0 M vs. 2.8 M parameters) and inference cost.

To address these limitations, we propose a lightweight hybrid architecture, YOLOv11-Transformer, as illustrated in Figure 7, which extends the YOLOv11 framework by integrating Transformer-based attention modules into both the backbone and neck. Specifically, Transformer blocks are inserted after the backbone’s final high-level feature-extraction stage and within the PAN-based neck, before multi-scale feature fusion. This placement enables the model to enhance global context modeling while preserving efficient multi-scale feature aggregation.

$$\text{Attention}(Q, K, V) = \text{softmax} \left(\frac{QK^T}{\sqrt{d_k}} \right) V \quad (1)$$

where Q , K , V are the query, key, and value matrices, respectively, and $d_k = 256$ is the key dimension.

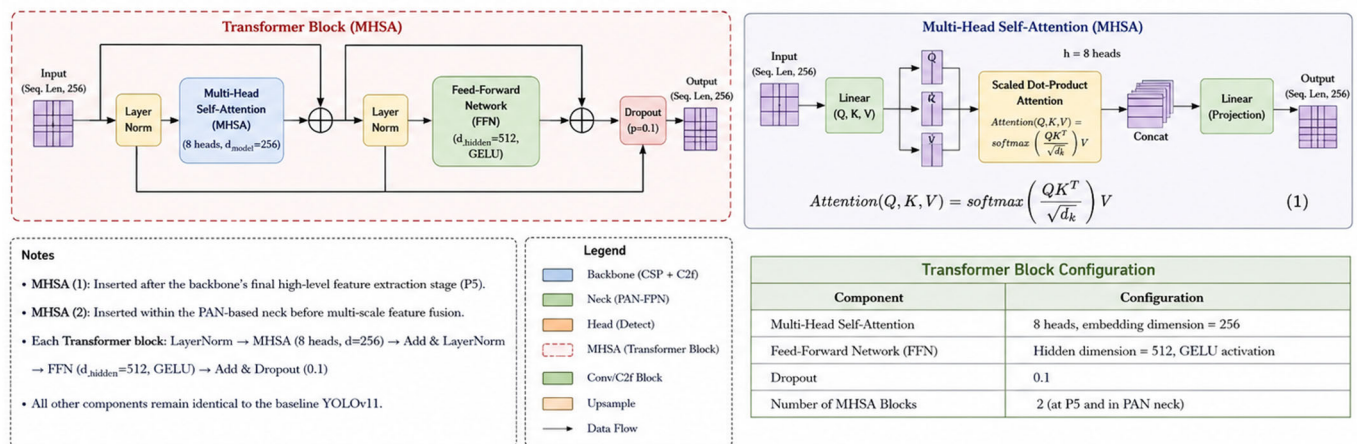


Figure 7. Each Transformer block consists of Multi-Head Self-Attention (MHSA), Layer Normalization, and a Feed-Forward Network (FFN) with residual connections. In this study, two MHSA layers are employed, each with eight attention heads and an embedding dimension of 256. The FFN uses a hidden dimension of 512, GELU activation, and a dropout rate of 0.1 to reduce overfitting. The attention mechanism is defined as (1).

The Transformer modules, as illustrated in Figure 8, are inserted after the final high-level feature-extraction stage in the backbone and within the PAN-based neck, before multi-scale feature fusion. This integration strategy enables the model to preserve local texture representation through convolutional operations while simultaneously capturing long-range spatial dependencies via self-attention.

This design is intended to improve sensitivity to fine-grained and small-scale fabric defects, such as micro-cuts and thread irregularities. Compared to the baseline YOLOv11, the proposed model introduces a moderate increase in parameter count due to the added attention layers; however, it avoids the heavy encoder–decoder structure characteristic of DETR-based detectors.

Consequently, the hybrid architecture achieves improved detection accuracy with minimal computational overhead, while maintaining near-real-time inference performance on an NVIDIA RTX 4060 GPU. This balanced design allows YOLOv11-Transformer to achieve stronger global contextual modeling than YOLOv11 while remaining significantly lighter and more computationally efficient than RT-DETR.

This hybrid design allows the model to retain strong local feature extraction through convolution while incorporating global contextual reasoning via self-attention. Compared to RT-DETR, the proposed approach avoids the heavy encoder–decoder structure, resulting in significantly lower computational overhead. Compared with YOLOv11, it improves sensitivity to small and irregular defects by enhancing long-range feature interactions. The proposed architecture is designed to enhance long-range feature interaction while preserving the lightweight single-stage design of YOLOv11.

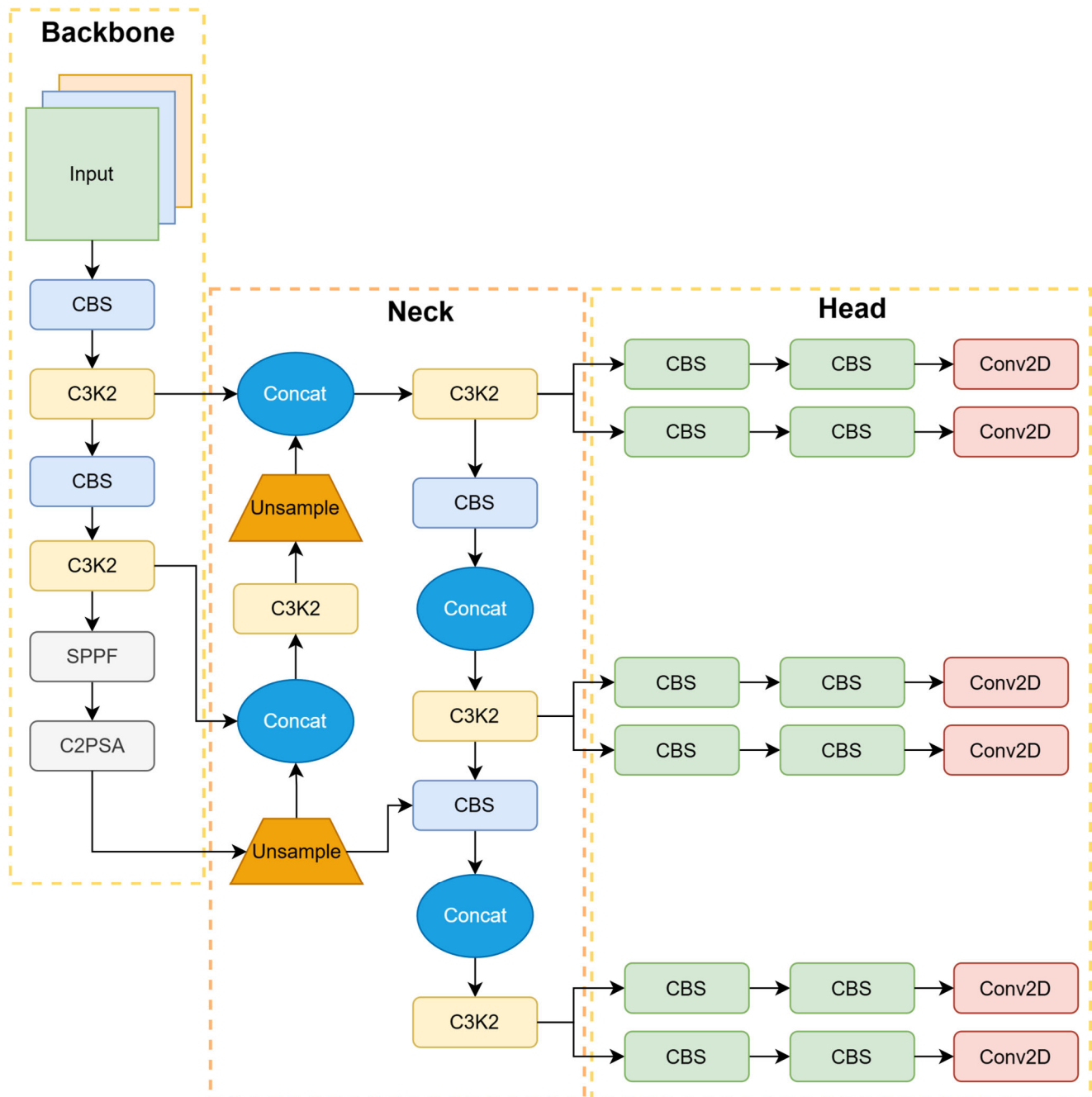


Figure 8. YOLOv11 Architecture.

4. Experiment and Results

This section interprets the experimental findings in the context of the proposed architectural design and its implications for practical fabric inspection. The proposed architecture is designed to enhance long-range feature interactions while preserving YOLOv11's lightweight single-stage design.

4.1. Datasets Preparation and Discussion

The fabric defect dataset used in all experiments is described in detail in Section 3.2. In summary, it comprises 4375 images across four defect categories, partitioned into training, validation, and test sets. The complete data distribution and experimental parameters are provided in Table 1.

Table 1. Fabric dataset for training, validation, and test set splits.

Dataset Split	Number of Images	Covered Categories
Training set	3500	Four classes
Validation set	438	Four classes
Test set	437	Four classes

4.2. Experimental Setup

All experiments were implemented in Python 3 using the PyTorch framework. GPU acceleration was enabled with the CUDA toolkit to improve training efficiency. OpenCV was used for auxiliary computer vision operations, while data augmentation was performed using the LabelMe platform. The experimental environment is summarized in Table 2.

Table 2. Experiment environment.

Component	Specification
Operating system	Windows 10 (64-bit)
Python	3.12.4
PyTorch	2.3.1
CUDA	11.8
CuDNN	8.9.7
CPU	AMD Ryzen 9 5900X
GPU	NVIDIA GeForce RTX 4060
RAM	DDR4 128 GB
Dataset	4375 images

All models were trained and evaluated on the same fabric defect dataset to ensure a fair comparison. As shown in Table 3, the training process employed Stochastic Gradient Descent (SGD) with a learning rate of 0.0001 for 100 epochs. A batch size of 32 was selected to balance computational efficiency and model performance.

Table 3. Training Parameters.

Component	Specification
Batch size	32
Momentum	0.937
Weight decay	0.0005
Image size	608 × 608
Epochs	100
Learning rate	0.0001
Optimizer	SGD
Workers	8

Momentum was set to 0.937 to accelerate convergence, while a weight decay of 0.0005 was applied to reduce overfitting. Input images were resized to 608 × 608 pixels to preserve fine defect details while maintaining computational efficiency. Data loading was parallelized using eight workers to improve training throughput. The same optimizer, learning rate, and training schedule were deliberately applied to all models, including the

DETR-based architectures, to isolate the effect of architectural differences from optimizer- and schedule-specific tuning.

4.3. Evaluation Metric

To assess model performance, we employ the following standard object detection metrics:

- Precision evaluates the correctness of positive predictions by comparing true positives against all predicted positives, including false positives. It is defined as (2):

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

where TP indicates correctly predicted positives, whereas FP represents negative instances that are mistakenly classified as positive.

- Recall quantifies how effectively a model detects positive instances by comparing true positives against all actual positives, including false negatives. It is defined as (3):

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

- F1-Score is a widely used metric for evaluating classification performance, defined as the harmonic mean of precision and recall. It provides a balanced assessment by considering both false positives and false negatives simultaneously. This metric is particularly effective in imbalanced datasets, where accurate evaluation of minority classes is critical. The F1 score is defined as follows (4):

$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (4)$$

- Average Precision (AP) evaluates detection performance by summarizing the precision–recall relationship into a single scalar value. It is computed as the area under the precision–recall curve, with precision typically interpolated across recall levels to ensure a stable estimate. The AP is defined as follows (5):

$$AP = \sum_{n=1}^N (R_n - R_{n-1}) P_n \quad (5)$$

where N is the total number of thresholds or recall levels, R_n is the recall at the n -th threshold, R_{n-1} is the recall at the previous thresholds and P_n is the precision at n -th threshold.

Mean Average Precision (mAP) summarises detection performance across all object classes. mAP50 is the mean AP computed at a single IoU threshold of 0.50, measuring overall detection accuracy. mAP50-95 averages the AP across ten IoU thresholds from 0.50 to 0.95 in steps of 0.05, providing a stricter and more discriminative measure of localization quality. Because mAP50 values are nearly saturated across YOLO-based models in this study, mAP50-95 is used as the primary evaluation metric to differentiate bounding-box refinement performance.

4.4. Comparison Experiment

To evaluate the effectiveness of the proposed approach, we conducted a comprehensive comparison experiment that included representative YOLO variants and a Transformer-based detector, all trained and evaluated under identical conditions. Specifically, models ranging from YOLOv5 to YOLOv11, along with YOLOv11-DETR, RT-DETR, and the proposed YOLOv11-Transformer, were fine-tuned on the fabric defect dataset to assess their

suitability for detecting small, subtle defects commonly encountered in textile manufacturing processes.

Model performance was evaluated using standard object detection metrics, including mAP50 and mAP50-95, which measure detection accuracy at fixed and progressively stricter IoU thresholds, respectively. These metrics provide a comprehensive assessment of both classification confidence and localization precision. In addition, training time was recorded to analyze computational efficiency under identical hardware conditions. This metric reflects the total wall-clock duration (in seconds) of the full 100-epoch training run and is distinct from the per-image inference benchmarks reported.

Table 4 presents the comparative performance of all evaluated models. The proposed YOLOv11-Transformer achieves the highest mAP50-95 of 94.50%, outperforming the baseline YOLOv11 by 2.40 percentage points. While mAP50 values remain similar across YOLO-based models, the improvement in mAP50-95 indicates enhanced localization accuracy under stricter IoU thresholds. The high overall performance observed across all models is partly attributable to the controlled acquisition conditions and the dataset's consistent imaging setup.

Table 4. Detection performance and total wall-clock training time (s) on GPU.

Models	mAP50 (%)	mAP50-95 (%)	Recall (%)	F1-Score	Training Time (s)
YOLOv5	99.50	88.10	98.05	98.10	1123.2
YOLOv8	99.05	89.30	98.30	98.20	1047.6
YOLOv9	99.50	90.30	98.70	98.50	2282.4
RT-DETR	99.10	90.70	98.70	99.10	9031.1
YOLOv10	99.50	91.50	99.10	99.82	1627.2
YOLOv11	99.50	92.10	99.96	99.83	1263.6
YOLOv11-DETR	99.50	93.30	99.98	99.89	3412.8
YOLOv11-Transformer	99.50	94.50	99.97	99.83	1213.2

Transformer-based models such as YOLOv11-DETR and RT-DETR also demonstrate strong detection performance, achieving mAP50-95 values of 93.30% and 90.70%, respectively. However, these models require substantially more training time, reflecting the increased computational complexity of fully Transformer-based architectures.

In contrast, the proposed YOLOv11-Transformer maintains a relatively efficient training time while achieving improved localization performance. The small (~4%) training-time difference between YOLOv11 and YOLOv11-Transformer lies within the typical run-to-run variance of single-trial wall-clock measurements, which are affected by data-loader scheduling, background system load, and GPU thermal/clock state; parameter count therefore remain more reliable indicators of model complexity.

Unless otherwise noted, the results in Table 4, correspond to a single representative training run; multi-seed statistics for the baseline and proposed models are reported in Table 5. To verify that the observed improvement is not attributable to run-to-run training variance, we repeated training of the baseline YOLOv11 and the proposed YOLOv11-Transformer using three random seeds, while keeping all other settings in Table 3 fixed. Table 5 reports the mean $\pm$ standard deviation on the fixed test set. The proposed model achieves $94.50 \pm 0.12\%$ mAP50-95, compared with $92.10 \pm 0.18\%$ for the baseline; the 2.40 percentage-point improvement exceeds the observed run-to-run variation by an order of magnitude, indicating that the gain is not explained by training stochasticity.

Table 5. Multi-seed stability analysis (3 seeds, mean $\pm$ std).

Models	mAP50 (%)	mAP50-95 (%)
YOLOv11 (baseline)	99.50 $\pm$ 0.02	92.10 $\pm$ 0.18
YOLOv11-Transformer	99.50 $\pm$ 0.01	94.50 $\pm$ 0.12

Overall, the comparison results indicate that the proposed YOLOv11-Transformer yields the greatest improvement in mAP50-95, increasing performance from 92.10% to 94.50% relative to YOLOv11. Because mAP50 values are nearly saturated across most YOLO-based models on this dataset, mAP50-95 is a more discriminative metric for evaluating localization quality. The observed gain, therefore, suggests that lightweight attention integration primarily improves bounding-box refinement for subtle defects rather than coarse detection alone.

4.5. Training Stability and Comparative Performance

In this section, we analyze the comparative performance of different object detection models on the fabric defect dataset, focusing on detection accuracy and training behavior. The evaluated models include YOLOv5, YOLOv8, YOLOv9, YOLOv10, YOLOv11, RT-DETR, YOLOv11-DETR, and the proposed YOLOv11-Transformer.

Figure 9 presents a visual comparison of all evaluated models across key performance metrics, including Precision, Recall, mAP50, and mAP50-95. As shown, the YOLOv11-Transformer achieves the highest mAP50-95 among all models, indicating improved localization performance under stricter IoU thresholds. However, precision, recall, and F1-score remain comparable across YOLOv11-based variants, with YOLOv11-DETR achieving slightly higher values.

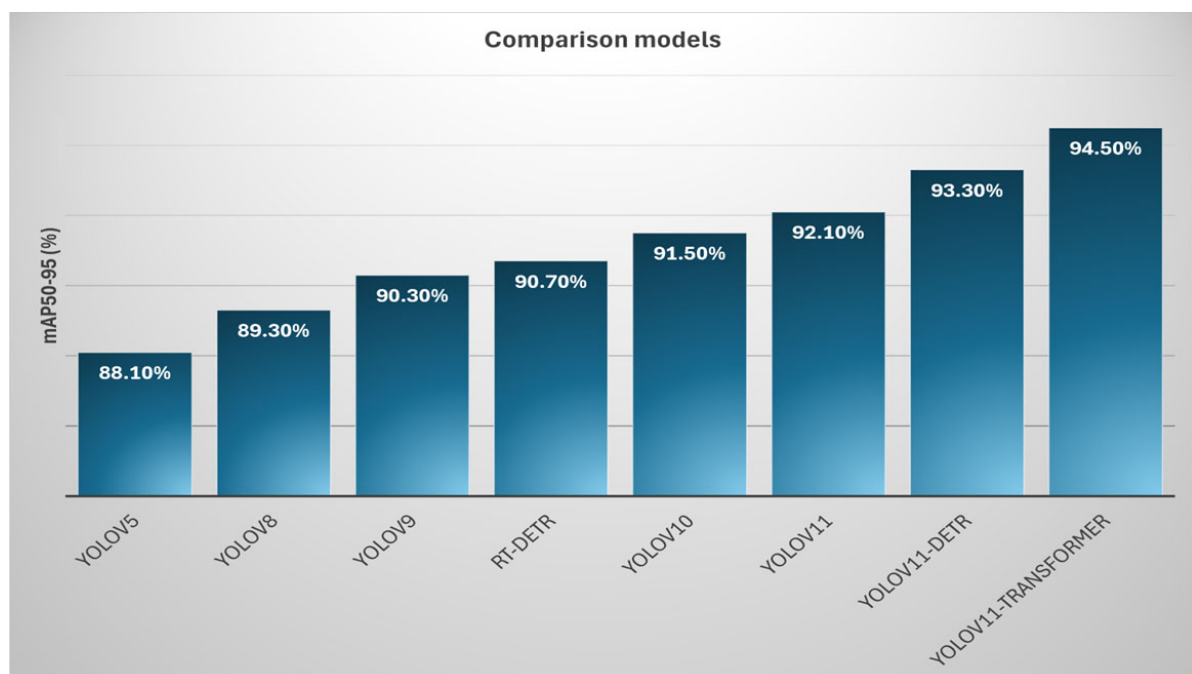
**Figure 9.** Bar chart comparing mAP50-95 across all evaluated models from YOLOv5 to YOLOv11-Transformer.

Table 6 shows that the proposed YOLOv11-Transformer achieves the highest mAP50-95 of 94.50%, surpassing YOLOv11-DETR by 1.20 percentage points and YOLOv11 by 2.40 percentage points. By contrast, the differences in precision, recall, and F1-score are marginal

across YOLOv11-based variants. This result indicates that the principal advantage of the proposed model lies in improved localization under stricter IoU criteria, rather than in substantial changes to coarse-detection sensitivity.

Table 6. Experimental results for YOLOv11 and its variants.

Models	mAP50 (%)	mAP50-95 (%)	Recall (%)	F1-Score (%)
YOLOv11	99.50	92.10	99.96	99.83
YOLOv11-DETR	99.50	93.30	99.98	99.89
YOLOv11-Transformer	99.50	94.50	99.97	99.83

All models were trained and evaluated under identical experimental settings, including the same dataset, number of epochs, optimizer configuration, and input resolution, ensuring a consistent basis for comparison.

To further analyze training behavior, we examine the training and validation curves shown in Figures 10 and 11. Both models exhibit stable convergence, with smooth loss reduction and consistent improvement in validation metrics. The YOLOv11-Transformer exhibits a convergence pattern similar to the baseline YOLOv11, indicating that integrating Transformer modules does not adversely affect training stability.

As illustrated in Figure 12, the YOLOv11-Transformer maintains competitive performance throughout training, compared with other models. The improvement in mAP50-95, together with stable convergence behavior, suggests that incorporating lightweight self-attention modules enhances feature representation while preserving efficient optimization dynamics.

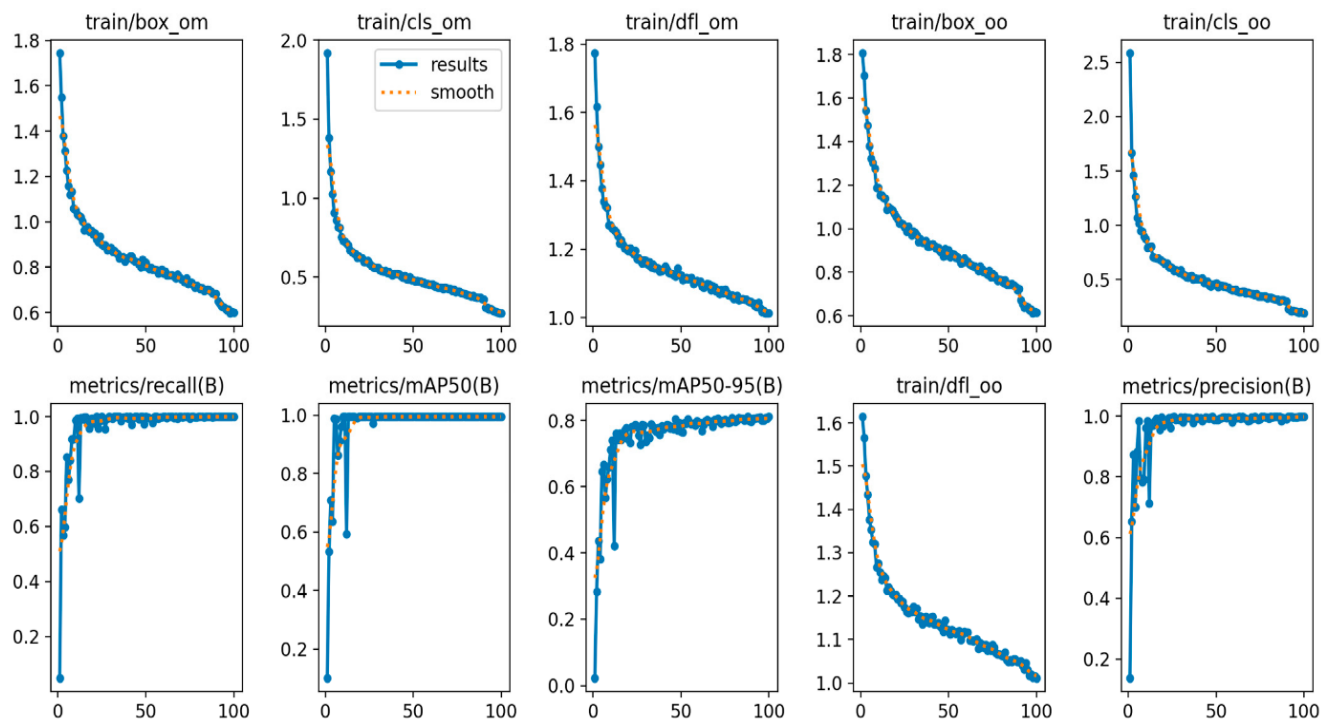


Figure 10. Performance of the original YOLOv11 model.

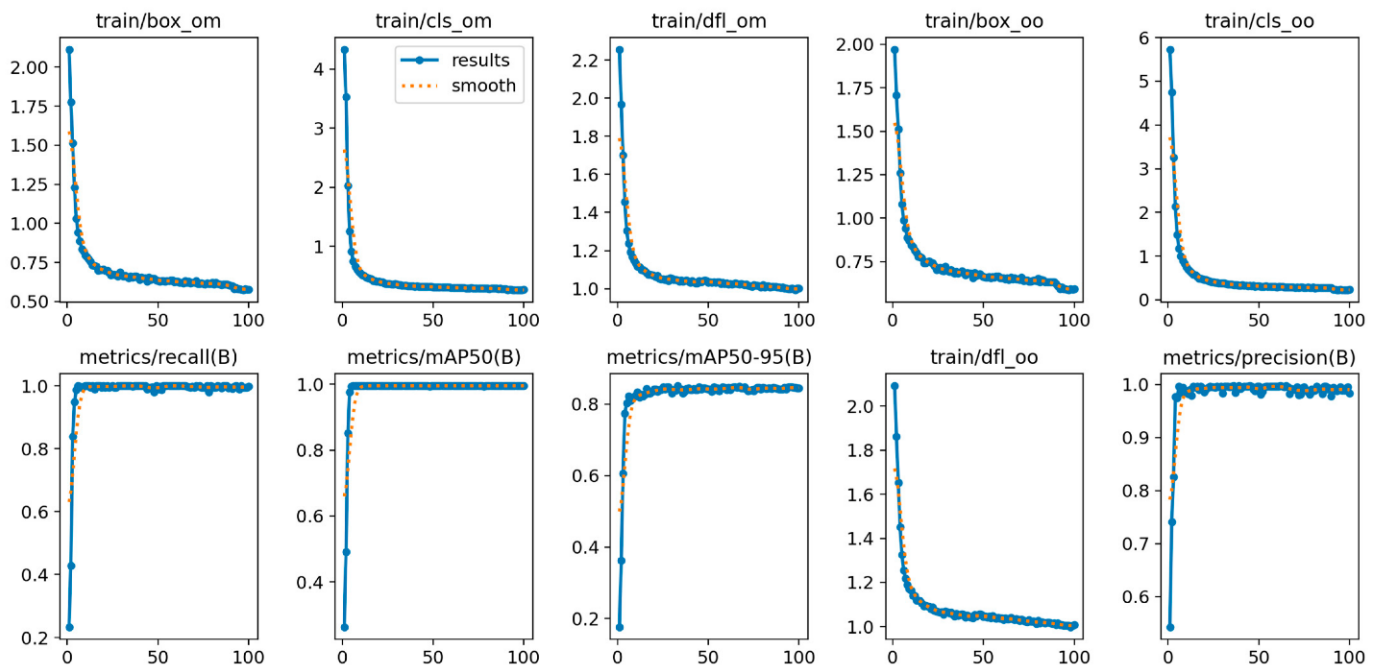


Figure 11. Performance of the original YOLOv11-Transformer model.

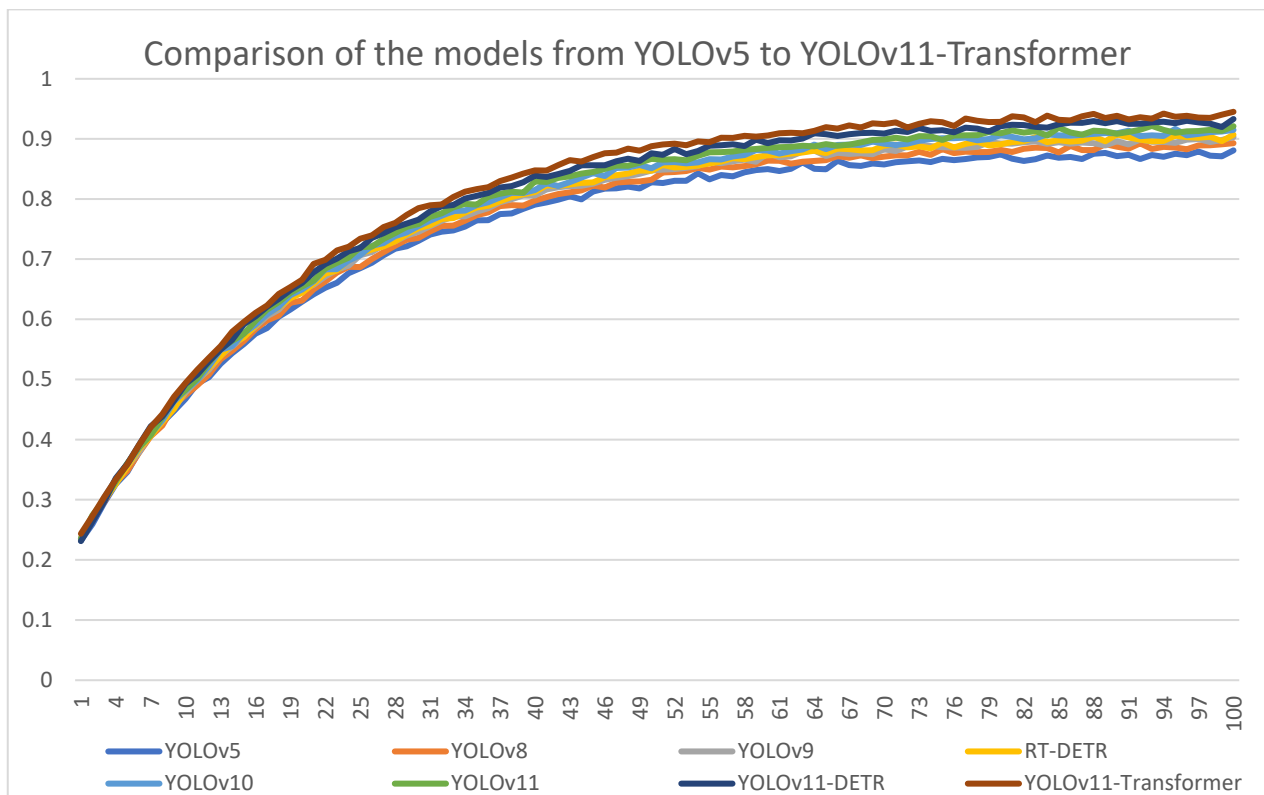


Figure 12. Comparison of the models from YOLOv5 to YOLOv11-Transformer.

Overall, the results indicate that the proposed YOLOv11-Transformer improves localization performance for subtle fabric defects while maintaining stable training behavior and comparable detection performance across standard metrics.

4.6. Comparative Qualitative Evaluation

To evaluate the practical impact of the architectural modification, we qualitatively compared the detection outputs of the baseline YOLOv11 and the proposed YOLOv11-Transformer on representative fabric samples. Visual inspection suggests that the proposed model detects several subtle defects more comprehensively than the baseline model, particularly for small or irregular patterns. As illustrated in Figure 13, the optimized network appears to provide improved sensitivity to fine-grained defects. These qualitative observations are consistent with the quantitative improvement in mAP50-95 reported in Tables 4 and 5.

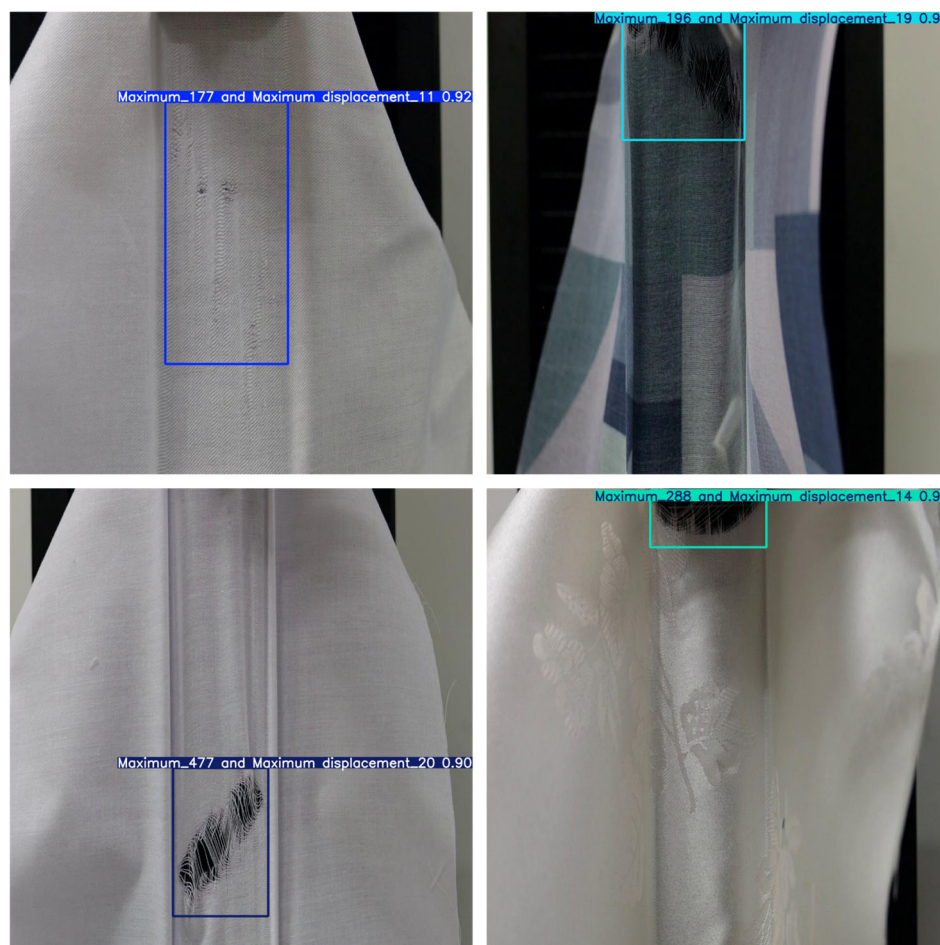


Figure 13. Qualitative detection results of YOLOv11 (baseline) and YOLOv11-Transformer on representative fabric samples, showing improved sensitivity to fine-grained defects.

4.7. Ablation Study of Transformer Integration

To better understand the contribution of each architectural component, we conducted an ablation study based on the YOLOv11 framework. Specifically, we evaluated the impact of integrating multi-head self-attention (MHSA) modules into the backbone and neck, both independently and in combination. All experiments were performed under identical training settings to isolate the effect of architectural modifications.

As shown in Table 7, introducing MHSA into either the backbone or the neck improves detection performance compared to the baseline YOLOv11 model. In particular, neck integration yields slightly higher gains, suggesting that enhancing multi-scale feature fusion is important for detecting fine-grained defects. The combined model achieves the highest performance, improving mAP50-95 from 92.10% to 94.50%, indicating that incorporating attention mechanisms at multiple stages is beneficial for this task.

Table 7. Ablation study.

Models Variant	Backbone	Neck	mAP50-95	Precision	Recall
YOLOv11 (Baseline)	×	×	92.10	99.70	99.96
+ Backbone MHSA	✓	×	93.20	99.72	99.96
+ Neck MHSA	×	✓	93.60	99.75	99.98
+ Backbone + Neck (Proposed)	✓	✓	94.50	99.70	99.97

Table 8 summarizes the inference efficiency of the evaluated models in terms of parameter count, inference time, and frames per second (FPS). All models were tested under the same hardware conditions (NVIDIA RTX 4060 GPU) with identical input resolution to ensure a consistent comparison.

Table 8. Efficiency comparison.

Models	Parameters (M)	FLOPs (G)	Inference Time (ms)	FPS
YOLOv5	2.6	7.7	14.8	68
YOLOv6	4.7	11.4	16.2	62
YOLOv7-Tiny	6.2	13.8	18.5	54
YOLOv8	3.2	8.7	13.4	75
YOLOv9	2.0	7.7	12.8	78
YOLOv10	2.3	6.7	11.9	84
YOLOv11	2.5	8.0	12.5	80
YOLOv11-DETR	32.0	65.0	28.0	35
RT-DETR	41.0	80.0	31.6	32
YOLOv11-Transformer	2.8	9.6	14.2	70

As shown in Table 8, the proposed YOLOv11-Transformer maintains a lightweight architecture with only 2.8 million parameters and 9.6 GFLOPs, representing a modest increase over the baseline YOLOv11 (2.5 M parameters and 8.0 GFLOPs) due to the incorporation of the Transformer module. Inference times were measured on an NVIDIA RTX 4060 GPU in FP32 precision at 608×608 input resolution during validation, with per-image times including NMS post-processing. The proposed model achieves an inference time of 14.2 ms per image (approximately 70 FPS), compared with 12.5 ms (80 FPS) for the baseline YOLOv11, indicating a small reduction in inference speed in exchange for enhanced feature representation and detection capability.

Compared with earlier YOLO variants, the proposed model incurs a slightly higher latency than the fastest baselines (YOLOv9–YOLOv11, 12.5–12.8 ms) but remains firmly within the real-time regime and is the only model in this speed class to exceed 94% mAP50-95. In contrast, DETR-based models, including YOLOv11-DETR and RT-DETR, require substantially greater computational resources, with 32.0 M–41.0 M parameters, 65.0–80.0 GFLOPs, and inference times of 28 ms or more per image (32–36 FPS). Overall, these results demonstrate that the proposed YOLOv11-Transformer achieves a favorable balance between computational efficiency and detection performance, making it well suited for real-time fabric defect inspection applications.

This trade-off is visualized in Figure 14, where the proposed model occupies the most favorable region of the accuracy–efficiency space, achieving the highest mAP50-95 among all evaluated detectors while remaining within the compact, real-time regime of the YOLO family.

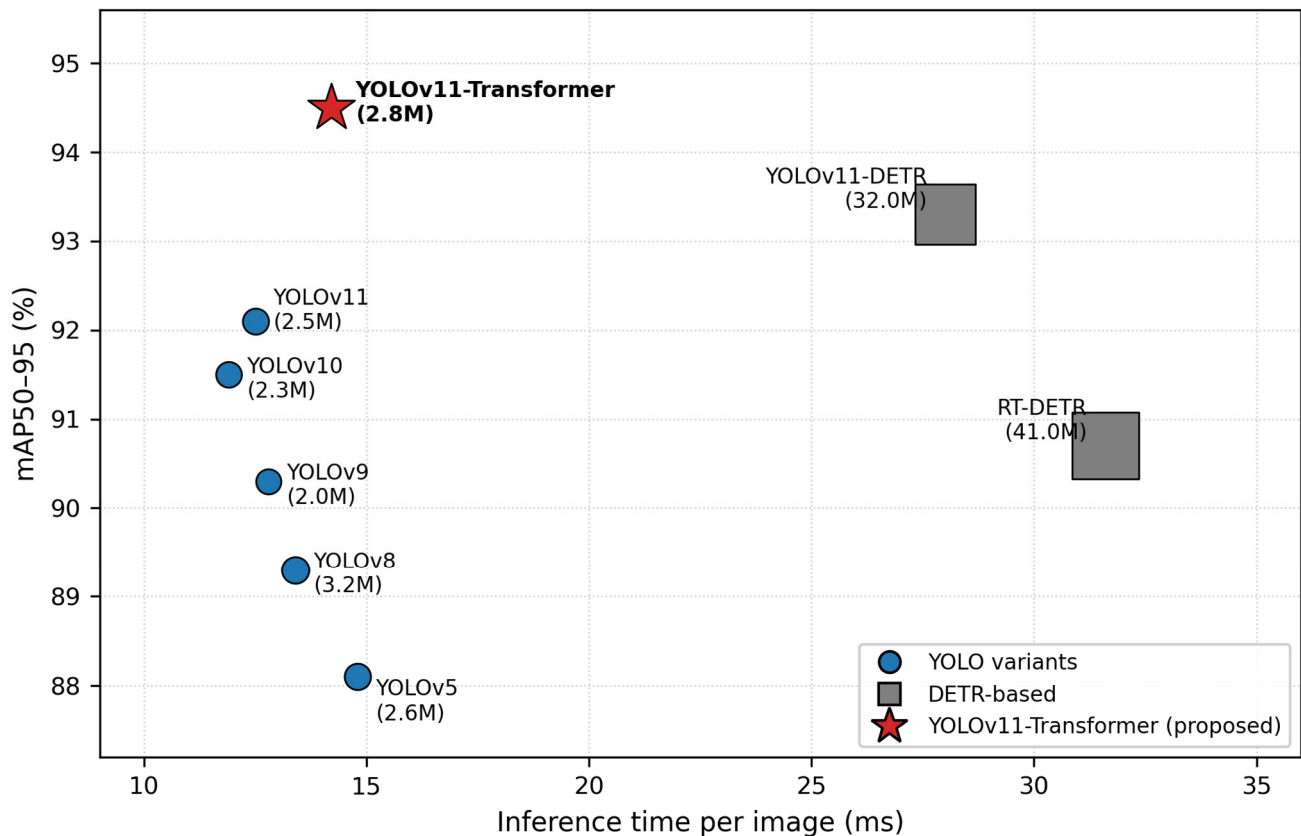


Figure 14. Accuracy–efficiency trade-off of the evaluated models, plotting mAP50-95 against per-image inference time on the NVIDIA RTX 4060 GPU. Marker size is proportional to parameter count (in millions).

4.8. Discussion

The experimental results indicate that the proposed YOLOv11-Transformer improves detection performance primarily in terms of mAP50-95, increasing from 92.10% (YOLOv11) to 94.50%. This improvement reflects higher localization accuracy at stricter IoU thresholds, particularly important for detecting small, subtle fabric defects. In contrast, precision, recall, and F1-score remain comparable across YOLOv11-based variants, suggesting that the main benefit of the proposed architecture lies in improved bounding-box refinement rather than in coarse detection.

The observed performance gain can be attributed to the integration of lightweight self-attention modules, which enable the model to capture long-range spatial dependencies and contextual relationships across the image. This is especially beneficial for fabric defect detection, where subtle defects often depend on surrounding texture patterns that may not be effectively captured by local convolutional operations alone.

Compared with fully Transformer-based models such as RT-DETR and YOLOv11-DETR, the proposed model achieves competitive accuracy while incurring lower computational overhead. While DETR-based architectures benefit from global attention mechanisms, their encoder–decoder design introduces a higher computational cost. In contrast, YOLOv11-Transformer selectively integrates attention within the backbone and

neck, improving feature representation while maintaining relatively efficient training and inference performance.

A limitation of this comparison is that all models were trained under a single unified protocol based on SGD. In contrast, DETR-based detectors are commonly trained with AdamW and longer training schedules. Consequently, the results reported for YOLOv11-DETR and RT-DETR should be interpreted as their performance under a controlled, common protocol rather than their best achievable performance, and their accuracy could improve with per-model hyperparameter optimization. Nevertheless, both DETR-based models converged stably and remained competitive under the unified settings, with YOLOv11-DETR attaining the second-highest mAP50-95. A systematic study with per-model tuned training configurations is left for future work.

The ablation study further supports this observation, showing that incorporating attention modules in both the backbone and neck yields the best performance. This suggests that global contextual modeling is beneficial at multiple stages of feature extraction and feature fusion.

The superior gain from neck MHSA (+1.50 pp) over backbone MHSA (+1.10 pp) can be attributed to the functional role of each stage in the detection pipeline. In the backbone, convolutional operations progressively encode local texture and structural patterns at increasingly abstract levels. While self-attention at this stage broadens the receptive field, the features remain single-scale and spatially compressed, limiting the scope of contextual interactions. In contrast, the neck aggregates multi-scale feature maps from different backbone stages through the PAN structure before fusing them into the detection outputs. Inserting MHSA at this point allows the model to capture long-range spatial dependencies across the spatially richer, semantically diverse feature representations produced during fusion.

For fabric defect detection specifically, where subtle defects such as micro-cuts and thread irregularities are defined by their relationship to surrounding texture context rather than by isolated local patterns, this cross-scale global reasoning is particularly valuable. Consequently, attention applied during multi-scale fusion provides a more direct benefit to bounding-box refinement, which is precisely what the mAP50-95 metric measures, than attention applied earlier in the pipeline.

Despite these advantages, several limitations should be acknowledged. First, the dataset was collected under controlled imaging conditions, which may limit generalization to more complex real-world environments with varying illumination, motion blur, or noise. Second, the study covers only four defect categories: holes, cuts, thread errors, and metallic contamination, whereas real textile production environments typically involve a broader, more diverse range of defect types; therefore, the model's performance on unseen defect classes remains unknown.

Third, no cross-dataset validation was performed, meaning that generalizability to other fabric types, weave structures, or imaging setups has not been confirmed; future work should evaluate the proposed architecture on publicly available benchmarks such as AITEX or NEU-DET to assess transferability. Fourth, the evaluation was conducted exclusively on a desktop GPU (NVIDIA RTX 4060), and performance on edge devices or resource-constrained platforms such as NVIDIA Jetson or mobile inference hardware remains to be explored.

Overall, the results suggest that integrating lightweight Transformer-based attention into a convolutional detection framework can improve localization performance for fine-grained defects while maintaining practical computational efficiency. This makes the proposed approach a promising direction for real-time industrial inspection systems.

5. Conclusions

This paper presents a lightweight Transformer-enhanced YOLOv11 framework for fabric defect detection. Experimental results indicate that the proposed model improves mAP₅₀₋₉₅ while maintaining real-time inference capability, suggesting its practical potential for automated textile inspection. To address the limitations of conventional manual inspection and previous deep learning frameworks, we constructed a comprehensive fabric defect dataset comprising four representative categories: cuts, holes, thread misalignments, and metallic contamination. Our systematic comparative analysis of state-of-the-art object detection models, including YOLOv5 through YOLOv11 and RT-DETR, confirms a progressive enhancement in baseline detection capabilities.

Most notably, integrating global attention mechanisms into the proposed YOLOv11-Transformer yields a higher mAP₅₀₋₉₅ of 94.50%. By effectively combining convolutional feature extraction with Transformer-based modeling of spatial dependencies, the architecture successfully identifies microscopic and irregular fabric defects that traditional convolutional baselines struggle to capture. Furthermore, the model maintains a highly efficient parameter footprint of approximately 2.8 million, significantly outperforming the computationally intensive RT-DETR while preserving real-time inference speeds. This research provides a practical, high-precision solution for modern manufacturing environments, demonstrating that lightweight hybrid architectures can successfully meet the stringent demands of high-speed industrial fabric inspection.

Author Contributions: M.H. provided conceptualization, project administration, and edited and reviewed the manuscript. M.M.: format analysis, methodology, software, preparing original draft, and writing—review and editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education (NRF-2022R1I1A3069371), was funded by BK21 FOUR (Fostering Outstanding Universities for Research) (No.5199990914048), and was supported by the Soonchunhyang University Research Fund.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available because the fabric defect image dataset was collected in-house and is proprietary to the authors' laboratory.

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

Abbreviations

The following abbreviations are used in this manuscript:

CNNs	Convolutional neural networks
Conv	Convolutions
DETR	Detection Transformer
FPN	Feature Pyramid Network
GELU	Gaussian Error Linear Unit
GPU	Graphics Processing Unit
IoU	Intersection Over Union
L	Large
M	Medium
MHSA	Multi-Head Self-Attention
N	Nano

RT-DETR	Real-Time Detection Transformer
S	Small
mAP	Mean Average Precision
PAN	Path Aggregation Network
SGD	Stochastic Gradient Descent
ViTs	Vision Transformers
YOLO	You Only Look Once

References

- Nayak, R.; Truong Quang, H.; Pal, R. Recent Trends in the Role of Digital Technologies in Fashion and Textile Manufacturing. In *Use of Digital and Advanced Technologies in the Fashion Supply Chain*; Springer: Singapore, 2025; pp. 1–23.
- Teng, S.; Kim, J.Y.; Jeon, S.; Gil, H.W.; Lyu, J.; Chung, E.H.; Nam, Y. Analyzing Optimal Wearable Motion Sensor Placement for Accurate Classification of Fall Directions. *Sensors* **2024**, *24*, 6432. [\[CrossRef\]](#) [\[PubMed\]](#)
- Ozek, A.; Seckin, M.; Demircioglu, P.; Bogrekci, I. Artificial Intelligence Is Driving Innovation in Textile Defect Detection. *Textiles* **2025**, *5*, 12. [\[CrossRef\]](#)
- Zhang, S.; Gong, H.; Li, R. Visual Fault Inspection Framework for Closed-Loop Recycling in Textile Production. *Comput. Ind. Eng.* **2025**, *206*, 111216. [\[CrossRef\]](#)
- Mahmud, T.; Sikder, J.; Chakma, R.J.; Fardoush, J. Fabric Defect Detection System. In *Intelligent Computing and Optimization*; Vasant, P., Zelinka, I., Weber, G.W., Eds.; Springer: Cham, Switzerland, 2021; Volume 1324, pp. 1–10.
- Liu, Q.; Wang, C.; Li, Y.; Gao, M.; Li, J. A Fabric Defect Detection Method Based on Deep Learning. *IEEE Access* **2022**, *10*, 4284–4296. [\[CrossRef\]](#)
- Shi, J.; Fan, W.; Yang, Z.; Li, J.; Li, M. Improving Small Object Detection in Power Line Inspection with Graph-Enhanced YOLO. *Microsyst. Technol.* **2026**, *32*, 18. [\[CrossRef\]](#)
- Sapkota, R.; Flores-Calero, M.; Qureshi, R.; Badgujar, C.; Nepal, U.; Poullose, A.; Zeno, P.; Vaddevolu, U.B.P.; Khan, S.; Shoman, M.; et al. YOLO Advances to Its Genesis: A Decadal and Comprehensive Review of the You Only Look Once (YOLO) Series. *Artif. Intell. Rev.* **2025**, *58*, 274. [\[CrossRef\]](#)
- Nimma, D.; Al-Omari, O.; Pradhan, R.; Ulmas, Z.; Krishna, R.; El-Ebiary, T.Y.A.; Rao, V.S. Object Detection in Real-Time Video Surveillance Using Attention-Based Transformer-YOLOv8 Model. *Alex. Eng. J.* **2025**, *118*, 482–495. [\[CrossRef\]](#)
- Mao, Y.; Wang, G.; Ma, Y.; Gui, X. Enhancing Fabric Defect Detection with Attention Mechanisms and Optimized YOLOv8 Framework. *IEEE Access* **2025**, *10*, 96767–96781. [\[CrossRef\]](#)
- Gao, C.; Chen, X.; Zhou, J.; Wang, J.; Shen, L. Open-Set Fabric Defect Detection with Defect Generation and Transfer. *IEEE Trans. Instrum. Meas.* **2025**, *74*, 1–10. [\[CrossRef\]](#)
- Li, Y.; Zhang, D. Toward Efficient Edge Detection: A Novel Optimization Method Based on Integral Image Technology and Canny Edge Detection. *Processes* **2025**, *13*, 293. [\[CrossRef\]](#)
- He, F.; Xin, B.; Zhan, Z.; Yang, W.; Liu, J. A Review of Fabric Defect Detection in Textile Manufacturing. *J. Text. Inst.* **2026**, *117*, 327–345.
- Lee, E.; Yoo, H. FMCNN: Raw-Data Type Identification Using Feature Matrix and CNN. *IEEE Access* **2025**, *13*, 141886–141896. [\[CrossRef\]](#)
- Zhou, L.; Ma, B.; Dong, Y.; Yin, Z.; Lu, F. DCFE-YOLO: A Novel Fabric Defect Detection Method. *PLoS ONE* **2025**, *20*, e0314525. [\[CrossRef\]](#) [\[PubMed\]](#)
- Moon, H.; Rahman, M.M.; Shin, E.; Hong, I.; Choi, S.; Kim, H.; Park, C.K.; Koh, J. Lightweight Deep Learning for Defect Localization in Air-Jet Textured Yarn. *Fibers Polym.* **2025**, *26*, 1019–1030. [\[CrossRef\]](#)
- Mao, M.; Hong, M. YOLO Object Detection for Real-Time Fabric Defect Inspection in the Textile Industry: A Review of YOLOv1 to YOLOv11. *Sensors* **2025**, *25*, 2270. [\[CrossRef\]](#) [\[PubMed\]](#)
- Zhao, H.; Zhang, T. Fabric Surface Defect Detection Using SE-SSDNet. *Symmetry* **2022**, *14*, 2373. [\[CrossRef\]](#)
- Kotthapalli, M.; Ravipati, D.; Bhatia, R. YOLOv1 to YOLOv11: A Comprehensive Survey of Real-Time Object Detection Innovations and Challenges. *arXiv* **2025**, arXiv:2508.02067.
- Song, I.; Tam, P.; Kang, S.; Ros, S.; Kim, S. DRL-Based Backbone SDN Control Methods in UAV-Assisted Networks for Computational Resource Efficiency. *Electronics* **2023**, *12*, 2984. [\[CrossRef\]](#)
- Manickaraj, D.; Mann, P.S. HFR-YOLOv3 Model for Multi-Object Detection and Tracking. *Int. J. Intell. Robot. Appl.* **2025**, *9*, 992–1010. [\[CrossRef\]](#)
- Geetha, A.S. YOLOv4: A Breakthrough in Real-Time Object Detection. *arXiv* **2025**, arXiv:2502.04161.
- Alhassan, M.A.M.; Yilmaz, E. Evaluating YOLOv4 and YOLOv5 for UAV-Based Surveillance. *Processes* **2025**, *13*, 254. [\[CrossRef\]](#)
- Rasheed, A.F.; Zarkoosh, M. Optimized YOLOv8 for Multi-Scale Object Detection. *J. Real-Time Image Process.* **2025**, *22*, 6.

25. Song, H.; Wang, J.; Zhang, Y. Detection of Abandoned Objects Based on YOLOv9. *Signal Image Video Process.* **2025**, *19*, 54.
26. Sun, H.; Yao, G.; Zhu, S.; Zhang, L.; Xu, H.; Kong, J. SOD-YOLOv10: Small Object Detection in Remote Sensing Images. *IEEE Geosci. Remote Sens. Lett.* **2025**, *22*, 1–5. [[CrossRef](#)]
27. He, L.H.; Zhou, Y.Z.; Liu, L.; Cao, W.; Ma, J.H. Research on object detection and recognition in remote sensing images based on YOLOv11. *Sci. Rep.* **2025**, *15*, 14032. [[CrossRef](#)] [[PubMed](#)]
28. Song, H.; Xia, H.; Wang, W.; Zhou, Y.; Liu, W.; Liu, Q.; Liu, J. QAGA-Net: Enhanced vision transformer-based object detection for remote sensing images. *Int. J. Intell. Comput. Cybern.* **2025**, *18*, 133–152. [[CrossRef](#)]
29. Shehzadi, T.; Hashmi, K.A.; Liwicki, M.; Stricker, D.; Afzal, M.Z. Object Detection with Transformers: A Review. *Sensors* **2025**, *25*, 6025. [[CrossRef](#)] [[PubMed](#)]
30. Subramanian, P. Enhanced Automatic License Plate Detection and Recognition Using YOLOv11. In *Proceedings of the 10th International Conference on Applying New Technology in Green Buildings (ATiGB)*; IEEE: New York, NY, USA, 2025.
31. He, W.; Zhang, Y.; Xu, T.; An, T.; Liang, Y.; Zhang, B. Object Detection for Medical Image Analysis Using RT-DETR. In *Proceedings of the International Conference on Artificial Intelligence and Computational Intelligence*; Association for Computing Machinery: New York, NY, USA, 2025.
32. Magdy, A.; Moustafa, M.S.; Ebied, H.M.; Tolba, M.F. Lightweight Faster R-CNN for Object Detection. *Sci. Rep.* **2025**, *15*, 16163. [[CrossRef](#)] [[PubMed](#)]
33. Neha, F.; Bhati, D.; Shukla, D.K.; Amiruzzaman, M. From Classical Techniques to Convolution-Based Models: A Review of Object Detection Algorithms. In *Proceedings of the IEEE International Conference on Image Processing Applications and Systems (IPAS)*; IEEE: New York, NY, USA, 2025.
34. Allmendinger, A.; Saltik, A.O.; Peteinatos, G.G.; Stein, A.; Gerhards, R. Assessing the Capability of YOLO- and Transformer-Based Object Detectors for Real-Time Weed Detection. *Precis. Agric.* **2025**, *26*, 52. [[CrossRef](#)]
35. Le, H.F.; Zhang, L.J.; Liu, Y.X. Surface Defect Detection of Industrial Parts Based on YOLOv5. *IEEE Access* **2022**, *10*, 130784–130794. [[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.