Intelligent Web Application with Computer Vision Algorithms for Detecting Flaws and Damage on Fabric in the Textile Industry

Gonzales Sebastían,[1] Sabino Rodrigo[2] and Ugarte Willy[3]

Abstract— **Manual inspection of textile defects is time-consuming, error-prone, and limits industrial scalability. The following paper presents the design and evaluation of an intelligent web-based system that automates fabric defect detection using deep learning and computer vision. The proposed solution integrates a YOLOv8-based model trained on a custom dataset comprising over 6,000 images, augmented from an initial 2,404, representing three common defect types: holes, lines, and stains. The system is deployed via a web application connected to a camera and a cloud hosted backend, enabling real-time detection and reporting. Eight YOLOv8 and YOLOv8-OBB models were trained and evaluated using multi-label and single-label strategies. Results show that single-label YOLOv8-OBB models achieve superior accuracy, with the best model reaching an mAP@0.5 of 0.954. The study highlights the benefits of rotation-aware detection and class-specific training in industrial contexts. Future work includes expanding the dataset, enabling real-time video processing, and deploying the solution in live production environments.**

I. Introduction

Manual inspection of fabric defects has long been a standard practice in the textile industry to ensure the quality of the product. However, these methods are limited by factors such as reduced attention span, human exhaustion, and high time consumption [14]. Inspectors have to work long shifts, finding faults, damages, writing reports, and responding accordingly. Furthermore, human inspection is not very precise because of human exhaustion. Human inspectors have an accuracy rate of approximately 60–70%, with performance declining as fatigue sets in [15].

Fabric defects can significantly reduce the commercial value of textile products. The presence of defects can reduce the price of fabrics by more than 50% [16]. Ensuring quality and physical integrity is a key priority for the industry to sell their textiles. To avoid waste of materials and resources is important to identify and correct any defects or production problems, which optimizes the production and maximizes the profitability [17]. Considering the limitations of manual inspections, industries face many challenges for detecting flaws and tear in textiles, which comes at the cost of effectiveness and increases waste of materials.

The complexity of fabric defects further complicates detection efforts. There are over 235 identified types of fabric defects, each varying in appearance and severity [18]. This vast variability poses challenges for both manual inspection and automated systems. Early computer vision approaches, such as analyzing gray-pixel-value distributions or histogram statistics, have struggled with accuracy and adaptability to real-world conditions [18]. Additionally, manual inspection of fabric flaws presents several significant challenges that affect the efficiency and reliability of quality control processes. Manual inspection speeds are limited, often less than 20 meters per minute, making it inefficient for high-volume production [19].

Although various automated inspection techniques have been proposed, many fail to generalize across real-world production environments. Earlier models often rely on handcrafted features or shallow classifiers that lack robustness and require frequent re-tuning. Deep learning models have emerged as promising alternatives, but many are not optimized for real-time use, demand excessive computing resources, or lack integration into scalable, user-friendly applications. Therefore, there remains a gap in deployable, accurate, and efficient systems designed specifically for textile inspection.

Furthermore, the textile industry holds significant relevance in the Peruvian economy. As of 2020, it contributed approximately 6.3% to the manufacturing GDP and 0.8% to the national GDP [20]. The sector is also a major source of employment, generating around 400,000 jobs directly [20]. Despite its economic importance, many textile companies in Peru still rely on traditional manual inspection methods, which limit scalability, reduce quality consistency, and hinder global competitiveness. Therefore, improving fabric inspection processes is essential not only for quality assurance but also for boosting industrial efficiency and supporting national economic growth.

Given the limitations of manual inspection and the high complexity of defect types, there is a growing need for intelligent, automated solutions that can enhance detection accuracy and operational efficiency. Advances in computer vision and deep learning offer promising tools for addressing these challenges. In this study, we propose the development of an intelligent web application that integrates the YOLOv8 object detection model, trained with a custom-labeled dataset of fabric defects. This system aims to provide fast, accurate, and scalable defect detection tailored to the needs of the textile industry. Our approach focuses on balancing detection performance with usability and deployment feasibility.

Our main contributions are as follows:

- We are developing an intelligent web application which detects flaws and damages on fabric using computer vision.
- We propose a combined dataset, using of public and manually captured images, containing over 6,000 images. The dataset includes the 3 most common types of

defects: holes, lines and stains.

- We propose an simple architecture involving one camera for the flaw detection on textiles

This paper is distributed in the following sections: In first place, we review the related works about the use of computer vision in flaw detection and its effectiveness in the automatization of quality control in Section II. In second place, we discuss about some preliminary concepts related to our research, describing its main contribution with more detail in Section III. In addition, the procedures and experiments that were carried out in this study will be explained in Section IV. In the end, we will analyze the results and show the main conclusions of the project, closing with some recommendations for future works in Section V.

II. Related works

The detection of defects in fabric using intelligent systems has been widely explored through various innovative approaches. The following works present diverse strategies, ranging from lightweight neural networks and composite reconstruction methods to phased noise enhancement and human–robot collaboration models. Each contributes unique techniques aimed at improving accuracy, efficiency, and adaptability in defect detection.

A. Paper 1

The authors propose GH-YOLOx, a lightweight network for comprehensive fabric anomaly detection that addresses the high computational resource consumption often associated with such tasks. They solved this problem by integrating ghost convolutions and a hierarchical backbone network (GHNetV2), together with the use of dynamic convolutions, feature fusion modules, and a shared group convolution head to efficiently detect anomalies across various scales. In addition, pruning techniques and knowledge distillation are applied to enhance model accuracy while reducing the number of parameters, achieving a balance between efficiency and effectiveness in real-time detection. The experimental results demonstrate that GH-YOLOx outperforms existing lightweight models in both detection rate and computational efficiency. Our approach aims to harness the full potential of YOLOv8 within a more extensive framework, ultimately aiming to improve detection accuracy and operational efficiency without undermining the lightweight advantages that models like GH-YOLOx provide.

B. Paper 2

The study proposes a diverse embedding-based composite reconstruction (DECR) method for unsupervised defect detection in color fabrics, which addresses the limitations of existing techniques that rely solely on pixel-level or feature-level reconstruction. To solve the problem of reconstruction noise and improve defect localization, DECR integrates a composite reconstruction (CR) strategy that simultaneously captures pixel-level details and feature-level semantics. Additionally, a multi-scale dual source feature aggregation (MDSFA) module is introduced to extract comprehensive feature information across different resolutions, while a multi-variate embedding extension (MEE) module generates diverse embeddings to improve data diversity. Experimental results demonstrate that DECR effectively reduces noise interference and achieves state-of-the-art performance in detecting and localizing fabric defects. Our approach using YOLOv8 aims to be simpler compared to DECR, which requires managing complex modules for diverse embedding-based composite reconstruction and extensive parameter tuning during training.

C. Paper 3

The authors propose a Phased Noise Enhanced Multiple Feature Discrimination Network (PNMFD) to enhance the detection of subtle fabric defects, which are often challenging to identify due to their complexity and rarity. To solve this problem, they utilized a phased noise enhancement strategy to simulate high-quality anomalies in the feature space, allowing better perception and reconstruction of defects. Additionally, the network incorporates a dual branch multi-feature discrimination module to improve the distinction of detailed textural features while employing a subsampling technique to reduce redundancy and ensure efficient inference speed. Extensive experiments on the AITEX and Kaggle Fabric datasets demonstrate that the proposed method significantly outperforms existing state-of-the-art approaches in terms of image and pixel-level accuracy. Our approach using YOLOv8 aims to be more simple compared to PNMFD which would require implementing phased noise enhancement and managing dual discriminators

D. Paper 4

In [4], the study proposes a defect segmentation model based on deep learning to enhance the detection of steel surface defects. Integrates ResNet50 for deep feature extraction, along with Residual Block, Residual Squeeze-and-Excitation Block, and Residual Refinement Module to improve segmentation accuracy. A custom dataset was created using images from a real production line, ensuring diverse defect representation. Additionally, Synthetic Defect Generation techniques were employed to augment the dataset, improving the robustness and effectiveness of the model in industrial applications. Our proposed solution uses a model that can detect any flaws in the fabric without the need to train it any further unless necessary. This helps the company save money while ensuring the quality of the fabric. In our solution, YOLOv8 is the model used to detect any imperfections in the fabric. This simplifies the detection process and can achieve the same accurate result, as long as the dataset provides enough information about the imperfections.

E. Paper 5

Int [5], the authors propose an AI-powered fabric inspection system that enhances textile quality control through a safe human–robot collaboration model using deep convolutional neural networks (DCNNs). Their goal is to automate defect detection while allowing humans to verify results, improving accuracy and workplace safety. They used an anthropomorphic robot (UR5) equipped with a camera and photoelectric sensor, integrated with an enhanced YOLOv8n-based DCNN model trained on 13 defect types. The system incorporates transfer learning, data augmentation, and a real-time alert mechanism, achieving 97.49% mean Average Precision (mAP) while maintaining a low inference time and high recall and precision. Our solution is to use only a camera that records fabric defects, saving resources and achieving the same results.

III. Detecting Flaws on Fabric with Computer Vision

A. Preliminary Concepts

This sections presents the main concepts used in our work. We use computer vision to detect flaws and damage in fabric using an 8-megapixel camera.

Definition 1 (Machine Learning [6]): a branch of artificial intelligence that enables machines and computers to imitate human learning, allowing them to enhance their accuracy and performance through greater data exposure and experience.

Example 1 (Machine Learning [11]): in Figure 1, we can see a confusion matrix illustrating how well a machine learning model classifies the 10 most commonly used verbs and the 10 most common nouns, demonstrating its effectiveness in everyday applications.

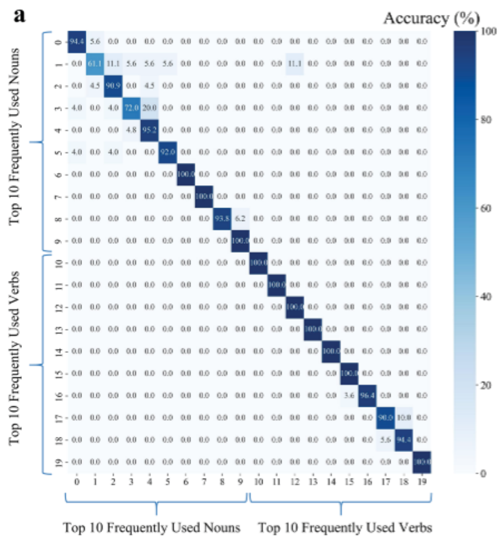

Fig. 1. Using machine learning to classify the most common used verbs and nouns [11]

Definition 2 (Computer Vision [7]): is a branch of artificial intelligence that uses machine learning and neural networks to enable computers and systems to extract meaningful insights from digital media and respond with recommendations or actions when defects or problems are detected.

Example 1 (Computer Vision [8]): In Figure 2, we can see the techniques of computer vision used to extract the color palette of clothing.

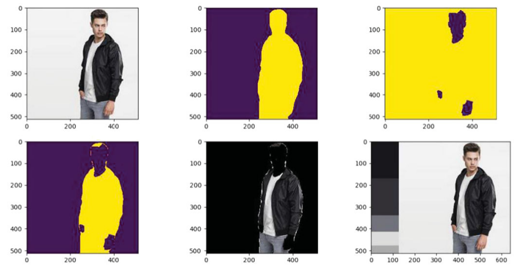

Fig. 2. Using Computer Vision techniques to extract color palette [8]

Definition 3 (Supabase [9]): an open source serverless back-end platform that serves as an alternative to Firebase. It enables developers to create high-performance applications with minimal setup, using a PostgreSQL database.

Example 1 (Supabase [12]): In Figure 3, we can see a bucket created to save avatar pictures, which are organized under folders

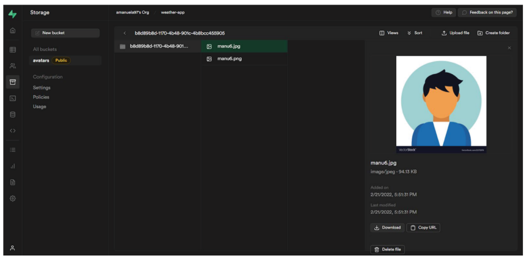

Fig. 3. Using Supabase for storing images [12]

Definition 4 (YOLOv8 [10]): a computer vision model developed by Ultralytics. It is designed for tasks such as object detection, image classification, and instance segmentation.

Example 1 (YOLOv8 [13]): In Figuer 4, we can see a modified YOLOv8 model to detect road cracks with better accuracy

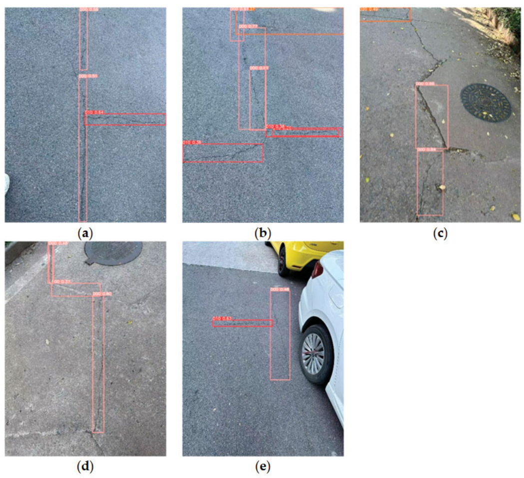

Fig. 4. Using a custom model of YOLOv8 to detect road defects [13]

B. Methods

1) Architecture: The contribution of this research is the use of computer vision and machine learning algorithms to detect defects and damage in fabric, then generate reports to ensure quality control and automation. In Figure 5, we can see the proposed physical architecture for the solution.

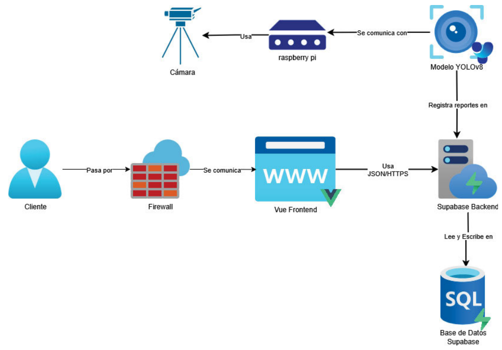

Fig. 5. Physical Architecture of the web application

- Firewall: The firewall filters users who are not authorized to access the web application.
- Camera: The camera is used by the model to detect flaws and damage in the textile.
- Raspberry Pi: This is used to connect the model to the camera and run model inference while also registering results in the backend.
- Supabase Backend: The backend provides information for the frontend to show.
- Supabase Database: Cloud database used to store website data and flaw reports generated by the model.
- Vue Frontend: The frontend of the web application uses the Vue framework to display all the information necessary for the user to facilitate quality control.

2) Flaw and Damage Dataset: The second contribution of this study is a collection of real photos taken on the fabric production line. Each one has been preprocessed and tagged accordingly for the training of the model.
3) Web Application: The third contribution of this study is a web application that presents all the flaws detected by the model. The application offers the essential functions for the user to manage the quality control of textiles:
   - Camera List: The camera list option offers a list of all installed cameras. This allows quality control staff to identify the camera, its location, and its state (active, inactive, or blurry).
   - Camera view: This function allows the user to select a camera and view the recording and detection process. Moreover, this function lists the last report registered by the model.
   - Report list: This function lists all the reports of the flaws and damages detected by the model. Each record lists the location and date of the detection.

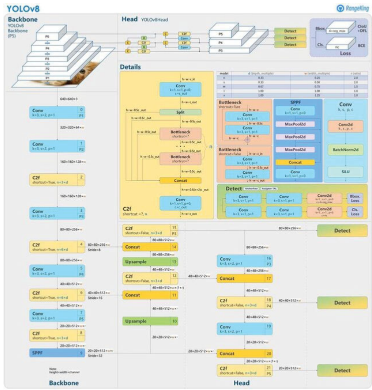

Fig. 6. YOLOv8 Architecture

4) Computer Vision Model Architecture: The architecture chosen for the project is YOLO v8. This architecture achieves a state-of-the-art balance between accuracy (mean Average Precision - mAP) and inference speed while boasting a mature and well-established ecosystem, efficient training processes, and relatively lower memory requirements compared to many other state-of-the-art models. YOLOv8 was selected over newer versions such as YOLOv9 and YOLOv11 due to its proven balance of accuracy, speed, and model simplicity, which make it well-suited for deployment in real-time or resource-constrained environments. While YOLOv9 and YOLOv11 introduce advanced features such as Neural Architecture Search (NAS), improved quantization techniques, or more complex attention mechanisms, these enhancements often come at the cost of increased computational overhead and reduced interpretability. YOLOv8, by contrast, offers a streamlined architecture with C2f modules, anchor-free detection, and built-in support for detection, segmentation, and classification—making it highly versatile and easy to train or fine-tune with custom datasets. Furthermore, its widespread community support, extensive documentation, and export options (e.g., ONNX, TensorRT) provide practical advantages for integration into a wide range of applications without sacrificing performance. The YOLOv8 architecture is composed of three main components: the backbone, the neck, and the head (as illustrated in Fig6) . The backbone is responsible for extracting visual features from the input image and utilizes C2f (Concatenate-to-Fuse) modules, which enhance gradient flow and computational efficiency. The neck, built using a combination of Path Aggregation Network (PAN) and Feature Pyramid Network (FPN), fuses multi-scale features to improve detection of objects at various sizes. Finally, the head is decoupled and handles the final predictions, including bounding boxes, objectness scores, class probabilities, and optional segmentation masks.

IV. EXPERIMENTS

This section presents the experiments conducted in our project, the requirements for replicating them, and an analysis of the results obtained after this process.

A. *Experimental Protocol*

All experiments were conducted using Google Colab Pro with the High RAM runtime, which provides access to a Tesla T4 GPU (16 GB VRAM), approximately 56 GB of RAM, and a shared virtualized 2-core Intel Xeon CPU. The Colab environment allowed the use of GPU acceleration, which was essential for training all eight YOLOv8-based models within reasonable time constraints. Training and inference were carried out using Ultralytics' YOLOv8 framework within Jupyter Notebooks. The yolov8n (nano) variant was selected for all models due to its reduced computational footprint, enabling fast prototyping and deployment on Colab without exceeding resource quotas.

The dataset used combines several public sources and manually curated data. It includes:

- Fabric Defects Dataset (Saleem, 2024), aggregating:
- Fabric Defect Dataset (Ranathunga, 2020)
- Fabric Stain Dataset (Pathirana, 2020)
- Aitex Fabric Dataset (Silvestre-Blanes et al., 2019)
- Dataset from Peng et al. (2020)
- FabricSpotDefect Dataset [Sumaya et al., 2024]
- 100 user-provided images, annotated manually.
- Annotations include both bounding boxes and polygon-based labels (for diagonal line defects). The images feature three main defect classes: holes, lines, and stains.

To improve model generalization and compensate for class imbalance, the original final dataset was extended from 2,404 to over 6,000 images through data augmentation. The following augmentation techniques were applied:

- **Horizontal flip**
- **Brightness adjustment** (ranging from -15% to $+15\%$)
- **Gaussian blur** (with a radius of up to 2.5 pixels)

These transformations were applied probabilistically during training to preserve the natural characteristics of fabric textures while enhancing defect variability. The resulting synthetic samples significantly enriched the dataset, particularly for underrepresented defect types such as *holes* and *stains*.

For training, the dataset was split into:

- 70% training (70 images)
- 20% validation (20 images)
- 10% test (10 images)

A public version of the dataset is available at: https://app.roboflow.com/fabric-defect-detection-zunat/my-first-project-bw4o9/3

A public version of the code is available at: https://github.com/SebGonS/tp1

The web app was developed on a laptop with an AMD Ryzen 7 7735HS CPU, and NVIDIA GeForce RTX 4060 GPU and 32 GB of RAM at 4800 Mhz. The web application was developed with VS Code, Vue.js version 3.5.16, Vuetify 3 and JavaScript.

The objective of these experiments was to evaluate the effect of two variables on fabric defect detection performance:

1) **Detection Mode**: Multi-label vs. Single-label training
2) **Model Architecture**: Standard YOLOv8 vs. YOLOv8-OBB (Oriented Bounding Boxes)

To this end, **eight models** were trained:

- **4 YOLOv8 models**:
  - 1 Multi-label (hole, stain, line)
  - 3 Single-label (one for each defect type)
- **4 YOLOv8-OBB models**:
  - 1 Multi-label
  - 3 Single-label

All models were trained using identical hyperparameters (epochs, batch size, augmentation probability) to ensure fairness. Evaluation metrics include:

- **mAP@0.5**
- **mAP@0.5:0.95**
- **Precision, Recall, and F1-Score**

YOLOv8 models use a *decoupled head* architecture with CSPDarknet backbones. The `YOLOv8-OBB` models extend this by adding rotation angle regression, enabling them to handle defects with non-axis-aligned geometries (e.g., diagonal lines). While this introduces a slight increase in model complexity and training time, it improves precision for elongated and rotated objects. In terms of computational cost, both remain efficient due to the use of the `yolov8n` variant.

B. *Results*

Multi-label Model Performance:

TABLE I. MULTI-LABEL MODELS RESULTS

Model	mAP@0.5	mAP@0.5:0.95	Precision	Recall	F1-Score
YOLOv8-OBB	**0.803**	**0.621**	**0.720**	**0.697**	**0.708**
YOLOv8	0.650	0.444	0.689	0.599	0.641

Table I shows the main performance metrics for both multi-label models.

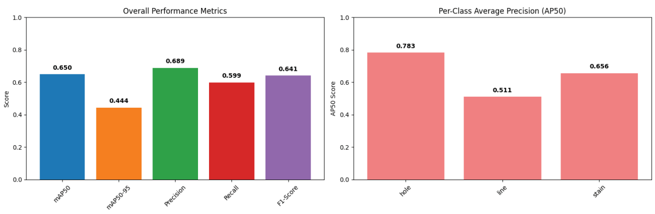

Fig. 7. YOLOv8 Model Evaluation

Fig. 7 allows a visualization of the performance metrics shown in Table 1 while also displaying the mAP@0.5 score for each of the trained labels of the YOLOv8 model.

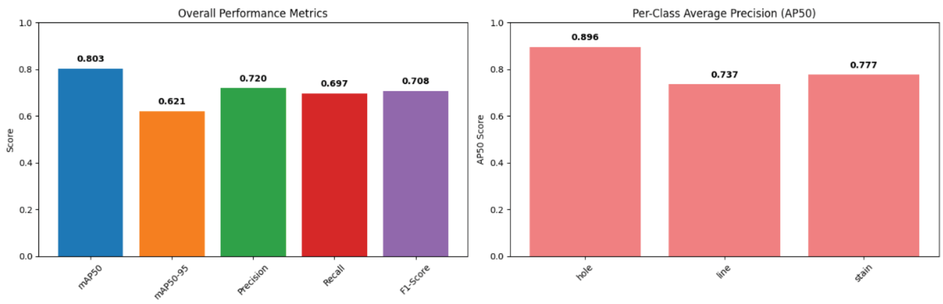

Fig. 8. YOLOv8-OBB Evaluation

Fig. 8 shows the same metrics and indicators of the previous image but for the YOLOv8-OBB model.

Single-Label Model Performance:

As seen in Table 1 and Table 2 single-label models outperformed their multi-label counterparts across all metrics. This supports the hypothesis that **dedicated training on isolated classes** reduces inter-class confusion and enables more accurate detection of subtle defect patterns. Notably, YOLOv8-OBB continued to outperform YOLOv8 in this setting as well, especially for *line* and *stain* defects.

C. Discussion

The experiments clearly demonstrate the following:

- **Orientation-aware detection** (YOLOv8-OBB) provides measurable performance gains for non-axis-aligned defects, especially *lines*.
- **Single-label training** results in significantly better class-specific performance than multi-label models, likely due to reduced label interference.
- **YOLOv8-OBB combined with class-specific training** achieved the best overall results in terms of both mAP and per-class precision/recall.

These findings indicate that, for applications such as textile quality control, leveraging **specialized, lightweight OBB detectors** per defect type is an effective strategy under computational constraints. While multi-label models offer convenience and generalized inference, they underperform when compared to tailored, defect-specific models. Future work may explore *model ensembling*, *real-time deployment*, and *active learning* strategies to further reduce annotation costs and improve system robustness.

V. CONCLUSIONS AND PERSPECTIVES

This project presented the design, development, and evaluation of a computer vision-based web application for automated fabric defect detection, tailored to the needs of the textile industry. The proposed solution integrated Ultralytics' YOLOv8 object detection models into a fully functional web application capable of real-time defect reporting. It addressed limitations of traditional manual inspection, such as low accuracy, slow throughput, and inconsistency, by automating the process using deep learning and cloud-based deployment.

Through a series of controlled experiments, we trained and evaluated eight models—four based on the standard YOLOv8 architecture and four on the oriented bounding box extension YOLOv8-OBB—using both multi-label datasets and class-specific (single-label) training. The results showed that models trained for a single class performed significantly better than multi-label models, and that YOLOv8-OBB consistently outperformed regular YOLOv8, especially in the detection of defects like lines, which often appear in rotated or irregular orientations.

The most accurate model, trained specifically on hole defects using YOLOv8-OBB, reached an mAP@0.5 of 0.954 and an mAP@0.5:0.95 of 0.728, far surpassing the multi-label baseline. These findings confirm the advantage of using rotation-aware models and targeted training strategies in complex industrial contexts. They also validate our initial hypothesis that defect-specific models can reduce class confusion and enhance detection precision in real-world settings.

Limitations

Despite promising outcomes, the current system has several limitations. First, although the dataset was augmented to over 6,000 images, the number of original images was relatively small, especially for underrepresented defect types. This reliance on augmentation could affect generalization in uncontrolled environments. Second, only three types of defects were considered—holes, stains, and lines—while real production environments may involve dozens of different defect categories with varying shapes, textures, and lighting sensitivities. Third, all evaluations were conducted on static images in a simulated environment. Real-world application will require testing on continuous fabric rolls with variable lighting and motion, introducing further complexity.

Unaddressed Problems and Open Questions

Several important aspects of real-world deployment remain outside the scope of this study. These include:

- **Temporal consistency**: How to ensure reliable detection over time when the same defect appears across multiple frames in video streams.
- **Environmental variability**: How well the system performs under changing lighting conditions, vibration, or fabric motion.
- **Adaptive learning**: Can the system continuously improve by learning from user corrections or post-inspection reviews?

TABLE II. SINGLE-LABEL MODELS RESULTS

Class	Model	mAP@0.5	mAP@0.5:0.95	Precision	Recall
Hole	YOLOv8-OBB	0.954	0.728	0.959	0.914
Hole	YOLOv8	0.844	0.544	0.772	0.776
Line	YOLOv8-OBB	0.919	0.688	0.882	0.816
Line	YOLOv8	0.807	0.629	0.779	0.714
Stain	YOLOv8-OBB	0.954	0.827	0.885	0.902
Stain	YOLOv8	0.932	0.733	0.867	0.844

- **Quality control thresholds**: What level of confidence is acceptable for triggering a fabric rejection, and how should this be calibrated?

These questions point to broader challenges in industrial automation that go beyond model accuracy—requiring considerations of reliability, calibration, and human-machine collaboration.

Future Perspectives

The results of this work suggest multiple directions for future exploration:

1) **Dataset Expansion**: Creating a larger and more diverse dataset covering additional defect types (e.g., creases, misalignment, color irregularities) from real production lines.
2) **Edge Deployment**: Deploying optimized versions of the models on embedded systems such as Raspberry Pi or Jetson Nano for on-site inference without reliance on cloud infrastructure.
3) **Ensemble Models**: Combining multi-label and single-label models to balance generalization with specialization, potentially using decision fusion or weighted confidence scores.
4) **Active Learning**: Implementing feedback loops where users confirm or reject detections to progressively improve model accuracy with minimal human intervention.
5) **Video Stream Integration**: Moving from image-by-image inference to real-time video processing with frame-to-frame consistency checks and defect tracking.
6) **Collaboration with Industry**: Validating the system in operational settings in partnership with textile manufacturers in Peru, to evaluate usability and real-world ROI.

Personal Interest and Continuity

This project has sparked a deeper academic and practical interest in the intersection between artificial intelligence and industrial automation. We are particularly interested in extending this work into a broader research effort on real-time computer vision systems for manufacturing quality assurance. We believe this topic holds significant social and economic value, particularly in countries like Peru, where automation can bridge the gap between traditional practices and global competitiveness. As such, we plan to pursue further development of the dataset, application and model, with a focus on optimizing it for industrial deployment, gathering additional field data, and publishing subsequent findings.

REFERENCES

[1] G. O. Young, “Synthetic structure of industrial plastics (Book style with paper title and editor),” in Plastics, 2nd ed. vol. 3, J. Peters, Ed. New York: McGraw-Hill, 1964, pp. 15–64.

[2] W.-K. Chen, Linear Networks and Systems (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.

[3] H. Poor, An Introduction to Signal Detection and Estimation. New York: Springer-Verlag, 1985, ch. 4.

[4] Guclu, E., Aydin, I. and Akin, E. (26 February 2025). Enhanced defect detection on steel surfaces using integrated residual refinement module with synthetic data augmentation. *Measurement, Volume 250*. https://doi.org/10.1016/j.measurement.2025.117136

[5] Hassan, S., Beliatis, M., Radziwon, A., Menciassi, A. and Oddo, C. (2 November 2024). Textile Fabric Defect Detection Using Enhanced Deep Convolutional Neural Network with Safe Human–Robot Collaborative Interaction. *Electronics 2024, Volume 13(21)*. https://doi.org/10.3390/electronics13214314

[6] IBM (22 September 2021). What is machine learning? https://www.ibm.com/think/topics/machine-learning

[7] IBM (27 July 2021). What is computer vision? https://www.ibm.com/think/topics/computer-vision

[8] Papachristou, E., Chrysopoulos, A. & Bilalis, N. (6 October 2020). Machine learning for clothing manufacture as a mean to respond quicker and better to the demands of clothing brands: a Greek case study. *The International Journal of Advanced Manufacturing Technology, Volume 115*, p. 691–702. https://doi.org/10.1007/s00170-020-06157-1

[9] Melvin Kosisochukwu (21 January 2025). What is Supabase: A Review of Serverless Database Features. https://bejamas.com/hub/serverless-database/supabase

[10] Solawetz, J. & Francesco (23 October 2024). What is YOLOv8? A Complete Guide. https://blog.roboflow.com/what-is-yolov8/

[11] Tang, C., Xu, M., Yi, W. et al. (07 May 2024). Ultrasensitive textile strain sensors redefine wearable silent speech interfaces with high machine learning efficiency. *Flex Electron 8, Volume 27*. https://doi.org/10.1038/s41528-024-00315-1

[12] Ayezabu, A. Z. (20 May 2022). Supabase vs Firebase: Evaluation of performance and development of Progressive Web Apps. https://ieeexplore.ieee.org/abstract/document/10652474

[13] Wang, X., Gao, H., Jia, Z., & Li, Z. (10 October 2023). BL-YOLOv8: An Improved Road Defect Detection Model Based on YOLOv8. *Sensors, Volume 23(20)*, p. 8361. https://doi.org/10.3390/s23208361

[14] Deageon, K. & Dongoun, L. (24 November 2023). Fabric Defect Detection Based on Artificial Intelligence. *International Journal of Intelligent Systems and Applications in Engineering, Volume 12*. https://ijisae.org/index.php/IJISAE/article/view/3814

[15] Yashika (31 March 2025). How AI is Reshaping the Technical Textile Industry’s Quality Control. *Robro Systems*. https://www.robrosystems.com/blogs/post/how-ai-is-reshaping-the-technical-textile-industry-s-quality-control

[16] Xiang, J., Pan, R., & Gao, W. (2022). Online Detection of Fabric Defects Based on Improved CenterNet with Deformable Convolution. *Sensors, Volume 22(13)*, p. 4718. https://doi.org/10.3390/s22134718

[17] Emmetex (29 February 2024). The importance of quality control in the textile industry. https://www.emmetex.com/en/importance-quality-control-textile-industry-emmetex/

[18] Carrilho, R., Hambarde, K. A., & Proença, H. (19 June 2024). A Novel Dataset for Fabric Defect Detection: Bridging Gaps in Anomaly Detection. *Applied Sciences, Volume 14(12)*, p. 5298. https://doi.org/10.3390/app14125298

[19] Jin, R. & Niu, Q. (30 September 2021) Automatic Fabric Defect Detection Based on an Improved YOLOv5. *Mathematical Problems in Engineering, Volume 2021(1)*. https://doi.org/10.1155/2021/7321394

[20] Ministerio de la Producción del Perú. (2020). Estudio de investigación sectorial: Sector textil y confecciones 2020. https://ogeiee.produce.gob.pe/index.php/en/shortcode/oee-documentos-publicaciones/publicaciones-anuales/item/1065-estudio-de-investigacion-sectorial-sector-textil-y-confecciones-2020