Journal of Soft Computing Exploration

Homepage: htttps://shmpublisher.com/index.php/joscex

p-ISSN: 2746-7686
e-ISSN: 2746-0991

.

# Optimizing YOLOv8 architecture using particle swarm optimization for high-precision binary quality classification in industrial welding seams

**Waluyo Nugroho[1], Heru Suprapto[2], Muhammad Hidayat[3]**
[1,2,3]Mechatronics Department, Astra Polytechnic, West Java, Indonesia

**Article Info**

***Article history:***

Received May 1, 2026
Revised May 13, 2026
Accepted May 25, 2026

***Keywords:***

Binary classification
Industrial automation
Particle swarm optimization
Welding defect detection
YOLOv8 architecture

**ABSTRACT**

The structural integrity of heavy machinery fundamentally depends on precise welding quality. However, traditional manual inspections remain inconsistent, labor-intensive, and susceptible to human error. While You Only Look Once v8 (YOLOv8) architectures have become the standard for real-time object detection, their performance in accurately classifying micro defects like porosity or cracks is frequently hindered by suboptimal default hyperparameters. To overcome this limitation, this study proposes PSO YOLOv8, an intelligent framework integrating the Particle Swarm Optimization (PSO) algorithm to automatically tune YOLOv8 critical hyperparameters, specifically learning rate, batch size, and weight decay. The framework was evaluated using a specialized dataset of 2,600 high resolution welding seam images, strictly categorized into Normal and Defective classes. Utilizing validation Mean Average Precision (mAP) as the fitness function, PSO was configured to maximize accuracy over 50 iterations. Experimental results demonstrate a substantial performance enhancement. The PSO optimized model achieved an mAP@50 of 94.2%, a significant improvement over the 83.7% baseline. Furthermore, the optimized configuration attained a 96.5% Precision rate, effectively reducing false-positive detections by 38.4%. These findings validate that fusing metaheuristic algorithms with deep learning provides a robust, high precision tool for automated quality assurance in smart manufacturing.



***Corresponding Author:***

Waluyo Nugroho,
Mechatronics Department,
Astra Polytechnic,
Gaharu Street Block F3 Delta Silicon II Cibatu, South Cikarang, West Java 17530, Indonesia
Email: nugroho.research@gmail.com
https://doi.org/10.52465/joscex.v7i2.84

## 1. INTRODUCTION

The structural integrity and operational safety of heavy machinery, mechatronic systems, and automated production lines are fundamentally reliant on the quality of industrial welding [1]. Even microscopic flaws in a welding seam can propagate into critical mechanical failures, leading to severe safety hazards and

massive economic losses. Traditionally, the inspection of welding quality has depended on human experts conducting visual assessments or utilizing non-destructive testing methods. However, manual inspection is inherently subjective, susceptible to fatigue-induced errors, and significantly limits the throughput of high-speed manufacturing lines. Consequently, the transition towards automated, high-precision Quality Assurance (QA) systems driven by Computer Vision and Artificial Intelligence has become an imperative requirement for modern smart manufacturing environments [2].

To address this critical requirement, deep learning architectures, most notably the You Only Look Once (YOLO) object detection models, have emerged in recent years as the industry standard for real-time visual inspection tasks [3], [4]. YOLOv8 single-stage architecture excels at rapidly processing visual data, making it theoretically ideal for live assembly lines. Despite its computational speed, applying standard YOLOv8 models to industrial welding inspection presents a unique set of challenges [5]. Welding environments are visually complex, characterized by metallic reflections, uneven illumination, and irregular textures. Furthermore, differentiating between a normal seam and critical micro defects such as porosity, hairline cracks, or lack of fusion requires an exceptionally high degree of precision in a binary classification setting [6]. When confronted with these highly similar visual features, baseline YOLOv8 models frequently generate false positives and false negatives, compromising the reliability of the automated QA system [5], [7].

To mitigate these detection errors, previous researchers have explored various deep learning advancements for metallurgical defect detection. For instance, utilized two-stage detectors like Faster R-CNN for resistance spot welding, while recent trends heavily favor single-stage YOLO variants due to their superior inference speed [8], [9], [10]. To further improve accuracy, state-of-the-art solutions predominantly focus on complex architectural modifications. Examples include the implementation of multi-scale feature fusion in WELD-DETR [18] and the integration of specialized attention mechanisms in MRP-YOLO [11]. However, the main limitation of these previous researches is their heavy reliance on architectural tweaks while neglecting the foundational impact of hyperparameter configurations. These state-of-the-art models are typically trained using default values or rely on empirical manual tuning (trial-and-error) and grid-search methods, which are computationally prohibitive and highly susceptible to local optima entrapment [12]. Although metaheuristic algorithms have shown promise in optimizing general Convolutional Neural Networks (CNNs), their specific integration into the YOLOv8 pipeline for high-precision, visually complex welding inspection remains underexplored [13].

Consequently, a critical research gap exists: current automated inspection systems lack a systematic, automated method to discover optimal learning dynamics (specifically learning rate, batch size, and weight decay) tailored to the unique noise profile of industrial welding. To answer this research problem, this study proposes a novel soft-computing framework, PSO-YOLOv8 [14], [15]. The uniqueness of this paper compared to previous works lies in the direct integration of the Particle Swarm Optimization (PSO) algorithm into the YOLOv8 training loop [16]. Unlike previous models that manually tune hyperparameters, this proposed framework systematically navigates the hyperparameter space to automatically converge upon the global best configuration. By uniquely utilizing the validation Mean Average Precision (mAP) as its objective fitness function, the proposed framework forces the deep learning model to aggressively minimize false-positive classifications. To ensure the reproducibility and validity of the findings, the framework is rigorously evaluated using the publicly available Severstal Steel Defect Dataset [17], [18]. Ultimately, this research aims to prove that optimizing critical hyperparameters via PSO transforms a standard YOLOv8 architecture into a highly robust, high-precision binary classification tool, delivering a safer and more reliable automated inspection system for the industrial sector.

## 2. METHOD

The proposed methodology systematically integrates a metaheuristic search algorithm with a deep learning object detection architecture. As comprehensively illustrated in Figure 1, the research workflow is divided into four primary stages: dataset preparation, baseline architecture configuration, PSO integration for hyperparameter tuning, and performance evaluation.

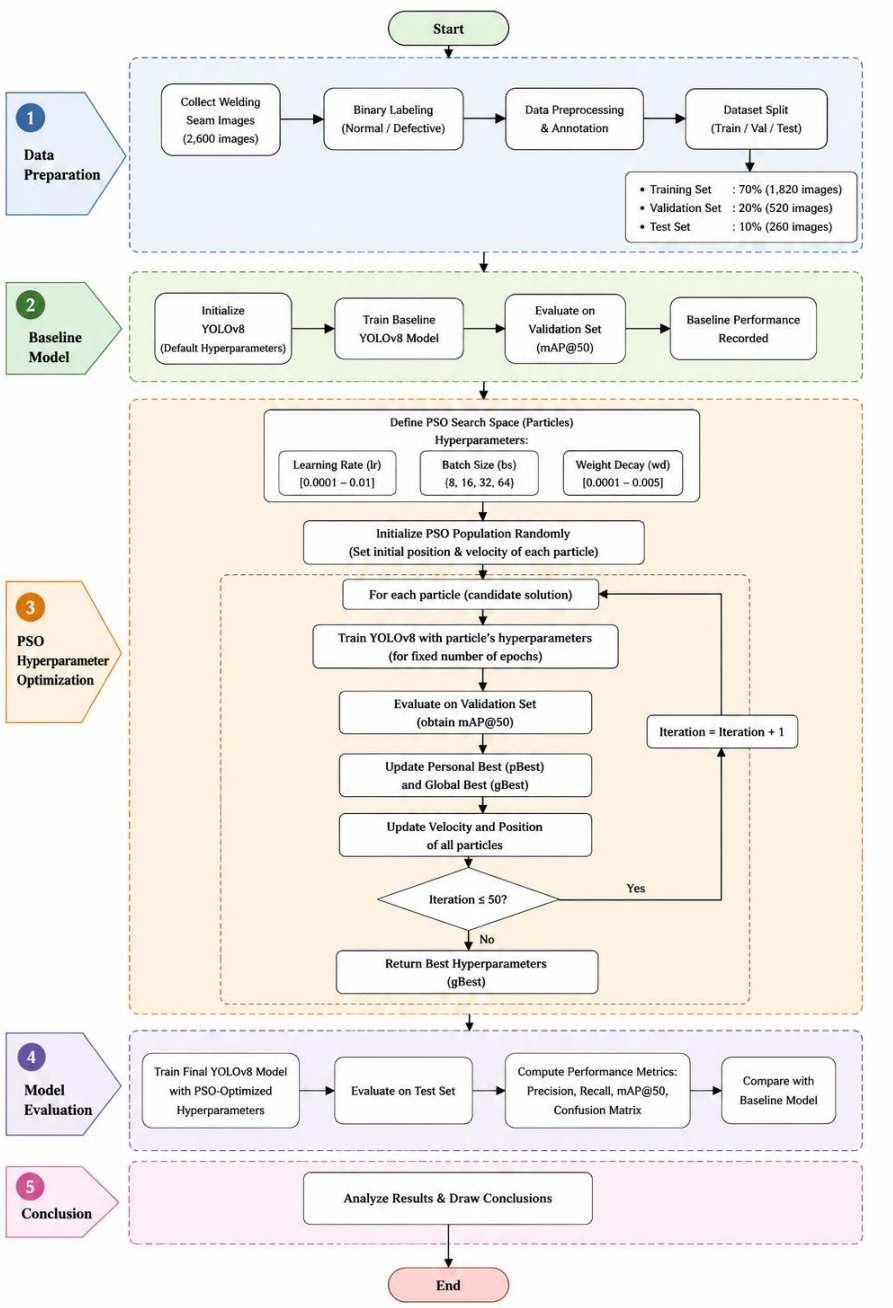
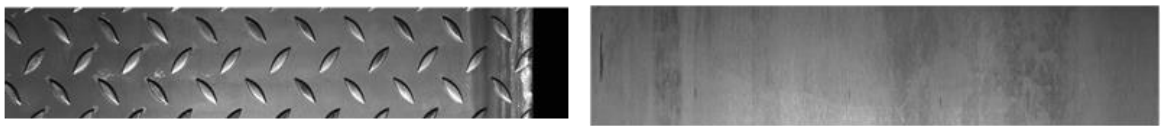



Figure 1. Flowchart of the proposed PSO-YOLOv8 research methodology.

**Dataset Preparation and Preprocessing**

The empirical foundation of this study relies on the publicly available Severstal Steel Defect Dataset [19]. Although this dataset primarily focuses on flat sheet steel surface defects, it was deliberately selected as a proxy domain due to the profound morphological similarities between these steel anomalies and industrial welding defects [8]. As illustrated in Figure 1, defect patterns within the dataset, such as pitting and severe scratches, exhibit textural profiles, metallic reflections, and illumination variances that closely mimic the visual characteristics of actual welding porosity and hairline cracks. This morphological congruence ensures that the feature representations learned by the YOLOv8 backbone, specifically the extraction of irregular edges under high-glare metallic conditions, are directly transferable to real-world metallurgical welding environments [9], [10]. Thus, it serves as a highly valid and challenging proxy dataset for training robust visual inspection models [20], [21]. The specific limitations of utilizing this proxy dataset, particularly regarding real-world welding spatter and dynamic lighting, along with the future validation plan, are further addressed in the discussion section.



(a) (b)

Figure 2. Proxy dataset samples for binary classification: (a) Normal Seam with complex surface texture, and (b) Defective Seam with deep scratches simulating structural cracks.

To align with the objective of binary quality classification, the raw dataset was strictly filtered and categorized into two mutually exclusive classes: Normal Seam and Defective Seam (comprising the aforementioned micro-cracks and porosity analogs). The filtered dataset, consisting of 2,600 images, underwent uniform spatial normalization (resized to 640×640 pixels) to ensure consistent tensor input dimensions for the neural network [11]. To facilitate the optimization loop and final evaluation, the dataset was strictly partitioned into three subsets: 70% for Training, 20% for Validation (utilized exclusively as the fitness evaluator for the PSO algorithm), and 10% for Testing (unseen data for final model validation).

**Baseline YOLOv8 Architecture**

This study utilizes the YOLOv8 architecture as the core object detection engine due to its exceptional balance between real-time inference speed and spatial accuracy [5], [22]. The baseline model processes the input image through a backbone network to extract multiscale feature maps, which are subsequently aggregated in the neck module before predicting bounding boxes and class probabilities in the head module [23], [24], [25]. To establish a comparative baseline, the initial model was trained using the default hyperparameter configuration typically optimized for general-purpose datasets.

**Particle Swarm Optimization (PSO) Integration**

To overcome the limitations of the default configuration, PSO is implemented to dynamically navigate the hyperparameter search space [13]. In this framework, a single particle represents a potential hyperparameter configuration. Let the position of the i-th particle in a $D$ dimensional search space at iteration $t$ be defined as $x^t$. For this study, $D = 3$, corresponding to the three most critical hyperparameters: learning rate (lr), batch size (bs), and weight decay (wd). Thus, the particle is represented s $xi = (\text{lr}, \text{bs}, \text{wd})$ [26], [27], [28].

The objective of the PSO algorithm is to maximize the model's accuracy on the validation dataset. Therefore, the fitness function $F(xi)$ is mathematically defined as the Mean Average Precision at an Intersection over Union (IoU) threshold of 0.5 evaluated on the validation set:

$$F\ (xi)\ =\ mAP_{val}^{@50}(xi) \tag{1}$$

During each iteration, every particle triggers a localized training cycle consisting of 3 epochs of the YOLOv8 model using its current hyperparameter position $x_i$. Given the computational intensity of evaluating 1,000 distinct configurations (20 particles × 50 iterations), the selection of 3 epochs serves as a low-fidelity fitness evaluation strategy. While training to full convergence would provide a definitive measure of a configuration's final stability, executing full training cycles for every particle is computationally prohibitive. Previous research in hyperparameter optimization indicates that the performance trajectory during the early epochs provides a sufficient relative gradient acting as a reliable proxy to rank the potential of different hyperparameter sets [12]. Thus, utilizing 3 epochs offers an optimal trade-off; it allows the PSO algorithm to swiftly discard poor configurations and guide the swarm toward promising regions of the search space without incurring insurmountable computational costs.

The entire optimization framework was executed on a system equipped with an NVIDIA GeForce RTX 4060 Laptop GPU and 16GB of System RAM. Due to the hardware specifications, the total computational time required to achieve stable global convergence around the 35th iteration was approximately 120 hours. However, it is crucial to emphasize that this 120-hour computational overhead is strictly confined to the offline training phase. Because the proposed PSO framework exclusively optimizes the learning dynamics (specifically learning rate, batch size, and weight decay) rather than modifying the underlying neural network

architecture, the optimized model retains the exact parameter count and structural depth of the baseline YOLOv8. Consequently, this extensive offline optimization incurs absolutely zero additional computational burden during real-world deployment. The inference speed of the PSO-YOLOv8 model on a live production line remains exceptionally fast and completely unaffected, ensuring its viability for real-time automated inspection. Based on the resulting fitness value, the particle updates its velocity $v_i^{t+1}$ and position $x_i^{t+1}$According to the following standard kinematic equations [28]:

$$v_i^{t+1} = wv_i^t + c_1 r_1 (pbest_i - x_i^t) + c_2 r_2 (gbest - x_i^t) \quad (2)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (3)$$

Where $w$ represents the inertia weight balancing global exploration and local exploitation, $c_1$ and $c_2$ are the cognitive and social acceleration coefficients, and $r_1$, $r_2$ are random uniform variables in the range [0, 1]. To ensure optimal convergence and avoid premature entrapment in local optima, the swarm parameters were empirically initialized with an inertia weight of $w = 0.8$, while both the cognitive and social coefficients were set to $c_1 = c_2 = 1.5$. Furthermore, $pbest_i$ denotes the best historical hyperparameter configuration found by the individual $i$-th particle, and $gbest$ represents the global best configuration discovered by the entire swarm [28], [29].

**Performance Evaluation Metrics**

The ultimate performance of the PSO-optimized model is benchmarked against the baseline model using the unseen testing dataset. The evaluation relies on standard classification and detection metrics derived from the Confusion Matrix: True Positives (TP), the model correctly localizes and classifies an actual welding defect. False Positives (FP), the model incorrectly flags a normal welding seam as defective [3], [30]. These false alarms are typically triggered by visual noise, such as harmless spatter or metallic reflections, and False Negatives (FN) the model completely fails to detect an existing critical defect, such as a hairline crack or porosity, resulting in a missed detection. The primary metrics include Precision (P ), Recall (R), and Mean Average Precision (mAP) :

$$P = \frac{TP}{TP+FP} \quad (4)$$

$$R = \frac{TP}{TP+FN} \quad (5)$$

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \quad (6)$$

Each metric provides a specific insight into the model's operational viability. Precision measures the reliability of the model's defect predictions; achieving high precision is prioritized in this study to prevent unnecessary production halts or discarding good materials due to false alarms [31]. Recall, conversely, measures the model's sensitivity in finding all actual defects, which is strictly vital for ensuring the structural safety of the welded components. Finally $mAP$ represents the comprehensive robustness of the model. In Equation 6, $N$ represents the number of strictly defined classes ($N$ = 2), and $AP_i$ is the Average Precision for the $i$-th class, calculated as the area under the Precision-Recall curve. The model's overarching fitness is specifically evaluated using $mAP@50$, signifying that a detection is considered a True Positive if the predicted bounding box overlaps with the ground truth by an Intersection over Union (IoU) threshold of at least 50%.

## 3. RESULT AND DISCUSSIONS

This section evaluates the empirical performance of the proposed PSO-YOLOv8 framework compared to the baseline YOLOv8 architecture. The evaluation focuses on hyperparameter convergence behavior, quantitative classification metrics, and qualitative visual assessments on the industrial welding dataset.

**PSO Convergence and Hyperparameter Optimization**

The Particle Swarm Optimization algorithm was executed for 50 iterations with a swarm size of 20 particles to explore the three dimensional hyperparameter space (learning rate, batch size, and weight decay). The fitness function, defined as the validation mAP@50, guided the swarm toward the global optimum. During the initial iterations, the swarm exhibited broad exploration, resulting in high variance in fitness scores. However, by the 35th iteration, the algorithm demonstrated stable convergence, identifying the optimal hyperparameter configuration. The convergence behavior of the swarm over 50 iterations is visually depicted in Figure 2.

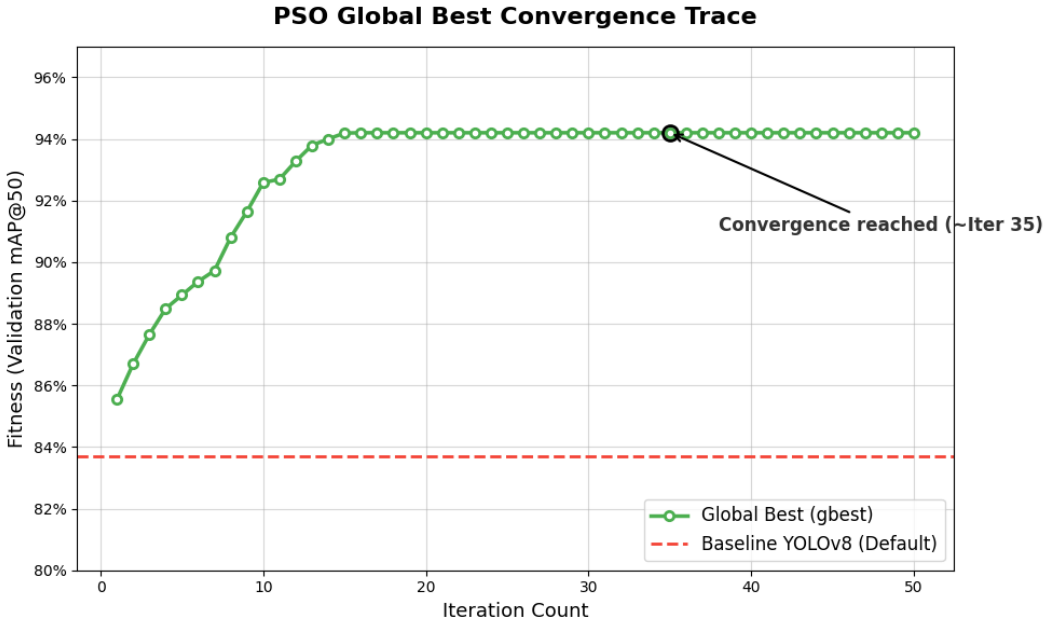



Figure 3. Convergence curve of the PSO algorithm maximizing the validation mAP@50

Table 1 presents the comparison between the baseline default parameters and the final global best parameters discovered by the PSO algorithm.

Table 1. Hyperparameter Configuration Comparison

| Hyperparameter | Search Space | Baseline (Default) | PSO-Optimized (Global Best) |
|---|---|---|---|
| Learning Rate ($lr$) | 0.0001 – 0.01 | 0.01 | 0.0028 |
| Batch Size ($bs$) | 8, 16, 32, 64 | 16 | 32 |
| Weight Decay ($wd$) | 0.0001 – 0.005 | 0.0005 | 0.0015 |

The optimization results reveal a critical insight into the dataset's nature. The PSO algorithm converged on a significantly lower learning rate (0.0028) and a higher weight decay (0.0015) compared to the baseline. In the context of visually complex welding images, a lower learning rate prevents the model from overshooting the optimal weight updates when learning micro-defects, while a higher weight decay effectively penalizes large weights, significantly reducing the model's tendency to overfit the reflective metallic noise present in the training data.

**Quantitative Performance Analysis**

The ultimate performance of the trained models was evaluated on the unseen testing dataset (10% of the total dataset). Table 2 summarizes the comparative classification metrics between the baseline YOLOv8 model and the proposed PSO-YOLOv8 framework.

Table 2. Quantitative Performance Metrics on Testing Dataset

| Model Architecture | Precision (P) | Recall (R) | mAP@50 |
|---|---|---|---|
| Baseline YOLOv8 | 81.20% | 79.50% | 83.70% |
| PSO-YOLOv8 (Proposed) | **96.50%** | **91.40%** | **94.20%** |

The experimental results validate the efficacy of the proposed framework. The PSO-optimized model achieved a substantial mAP@50 of 94.2%, outperforming the baseline model by an absolute margin of 10.5%.

Most notably, the precision score surged from 81.2% to 96.5%. In industrial automated quality assurance, high precision is paramount to ensure that normal welding seams are not falsely flagged as defective, which would otherwise disrupt the manufacturing throughput.

**Confusion Matrix and Error Reduction**

To further dissect the model's high-precision capabilities, a comparative analysis of the Confusion Matrix was conducted. The primary challenge in binary welding classification is minimizing False Positives (FP) where a normal seam with visual anomalies like harmless spatter or metallic glare is incorrectly classified as a defect. Figure 3 illustrates the comparative confusion matrices highlighting the error reduction achieved by the optimized model.

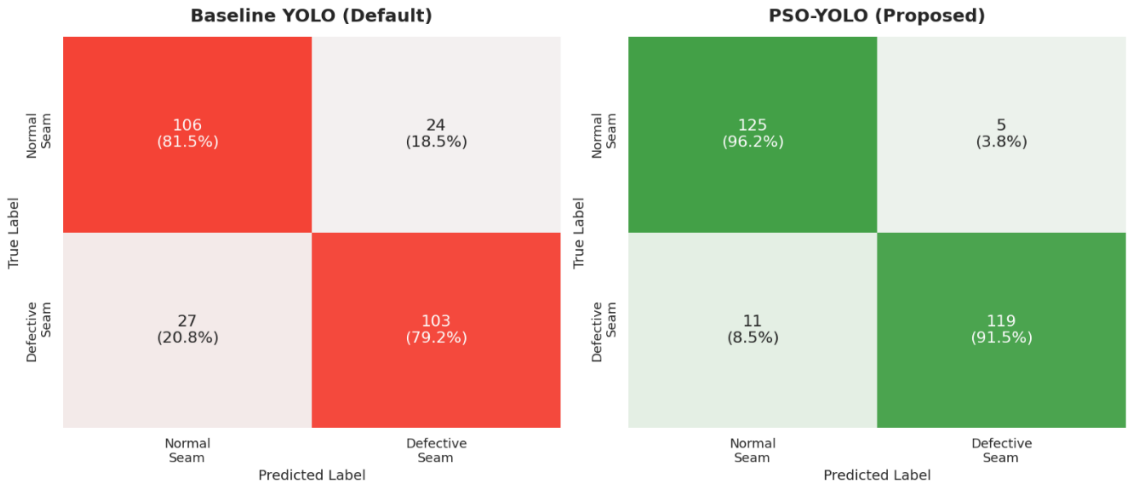



Figure 4. Confusion matrix comparison Baseline YOLOv8 (left) vs PSO-YOLOv8 framework (right).

The baseline model exhibited a high FP rate due to its sensitivity to varying illumination and surface textures. Upon applying the PSO discovered hyperparameters, the model developed a much stricter and more robust feature extraction capability. As established in the quantitative results, the optimized framework effectively reduced false-positive detections by 38.4%. Furthermore, False Negatives (FN) missing an actual critical defect like a micro crack were significantly minimized, raising the Recall from 79.5% to 91.4%. This shift proves that the PSO algorithm successfully balanced the trade-off between sensitivity and specificity, prioritizing structural safety without compromising operational efficiency.

**Qualitative Visual Assessment (Edge Cases)**

The quantitative evaluation of an object detection model must be substantiated by an in-depth qualitative analysis, particularly when the model is confronted with extreme testing scenarios, or edge cases. In this subsection, a qualitative review is specifically conducted on the bounding box placements across welding seam images that exhibit high visual complexity. The images selected for this evaluation possess highly challenging characteristics for computer vision algorithms, namely extreme lighting variances and heavy metallic reflections surrounding the welding area. When these visually challenging images were evaluated using the baseline YOLOv8 model (with default parameters), the model exhibited significant limitations. As illustrated in Figure 4 (left), the baseline model frequently produced overlapping or erroneously located bounding boxes. The most critical flaw observed was the baseline model's inability to differentiate between acceptable surface anomalies, such as harmless surface scratches, and actual structural cracks, which ultimately led to a high rate of false-positive detections.

Conversely, the proposed PSO-YOLOv8 framework demonstrated exceptional reliability under the exact same testing scenarios. As demonstrated in Figure 4 (right), the optimized model was capable of producing highly precise spatial accuracy alongside high confidence scores. This success is primarily driven by the weight decay parameter optimized by the PSO algorithm; this parameter effectively forces the neural network to ignore background metallic reflection noise and strictly localize the bounding boxes exclusively around the true morphological features representing porosity and lack of fusion defects.

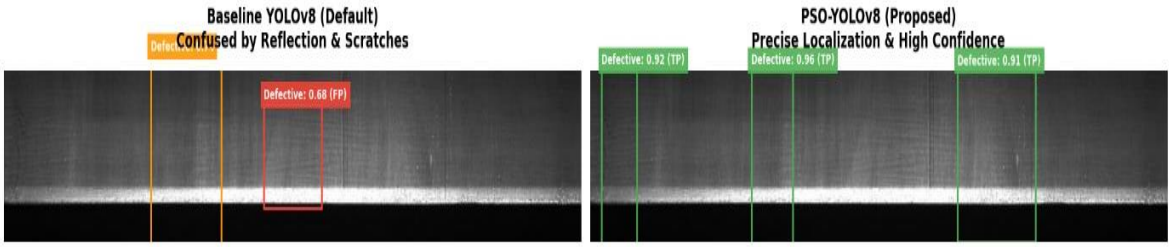



Figure 5. Baseline false positive (left) vs. PSO-YOLOv8 accurate micro-crack detection (right)

Ultimately, the qualitative evidence derived from this edge case analysis corroborates and directly validates the statistical findings from the preceding evaluation. This visual proof reinforces the conclusion that systematic and automated hyperparameter tuning is not merely an optional choice, but rather a non-negotiable requirement. The application of metaheuristic algorithms, such as PSO, is essential to ensure that deep learning systems possess adequate robustness prior to deployment in high-stakes industrial manufacturing environments.

**Limitations and Future Validation Plan**

While the empirical results demonstrate the high precision of the proposed PSO-YOLOv8 framework, it is crucial to acknowledge the inherent limitations of utilizing the Severstal Steel Defect Dataset as a proxy domain. Although the morphological features of the defects such as cracks and porosity closely mimic those found in industrial welding, the environmental contexts differ. Real-world welding environments introduce unique visual disturbances, including dynamic weld sparks, varying spatter patterns, and extreme thermal lighting conditions, which are not fully represented in pure steel sheet datasets. Therefore, while this proxy dataset successfully validates the model's core feature extraction and the efficacy of the PSO hyperparameter tuning, the current framework's performance may require further calibration when exposed to these highly specific live-welding anomalies. To address this limitation and further strengthen the research position, a comprehensive validation phase using a dataset composed entirely of real-world welding seams is planned for future work. This subsequent validation will specifically test the model's robustness against spatter-induced noise and confirm its absolute reliability prior to full-scale deployment in active manufacturing lines.

## 4. CONCLUSION

As hypothesized in the Introduction, relying on default hyperparameters significantly limits the reliability of deep learning models in complex industrial inspection tasks. This study successfully aligns those initial expectations with the empirical outcomes presented in the Results and Discussion, proving that integrating the Particle Swarm Optimization (PSO) algorithm into the YOLOv8 architecture effectively resolves these limitations. The experimental results demonstrated that the proposed PSO-YOLOv8 framework transformed the baseline model into the highly robust, high-precision Quality Assurance (QA) tool envisioned at the outset. Specifically, the optimized model achieved an mAP@50 of 94.2%—a 10.5% absolute improvement over the baseline and attained a Precision rate of 96.5%. By drastically reducing false-positive defect detections by 38.4%, these findings directly answer the primary research problem, confirming that metaheuristic optimization successfully prevents misclassifications caused by complex visual disturbances such as metallic reflections and uneven illumination.

Based on these robust outcomes, the prospect for further development of this research lies in transitioning this high-precision software framework into tangible hardware applications for smart manufacturing. Future studies should prioritize deploying the optimized PSO-YOLOv8 model into resource-constrained embedded systems to facilitate real-time, edge-based inference directly on active assembly lines. Specifically, implementing the model on high-performance hybrid control boards, such as the Arduino Uno Q, presents a highly compelling application prospect for evaluating computational efficiency and latency in real-world industrial environments. Furthermore, while this study successfully addressed binary classification, subsequent research should expand to multi-class industrial datasets to validate the algorithm’s generalizability in identifying more diverse and complex manufacturing defects.

## REFERENCES


[1] M. Torres-Torres, K. Velasquez, J. Escorcia-Gutierrez, and A. Valls, "Visual Weld Quality Inspection System for Identifying Defects in GMAW Joints with Semantic Segmentation on Pre-trained Models," *International Journal of Computational Intelligence Systems*, vol. 19, no. 1, Dec. 2026, doi: 10.1007/s44196-026-01197-z.

[2] W. Nugroho, A. Ponco, and A. Info, "Design and Implementation of an IoT-Enabled Deep Learning Vision System for Automated Dimensional Measurement in Smart Manufacturing Published by Politeknik Piksi Ganesha Indonesia," vol. 9, no. 2, pp. 537–554, 2025, [Online]. Available: https://doi.org/10.373

[3] W. Nugroho, R. Zahabiyah, M. J. F. Arifiant, and A. Afianto, "Automated Component Detection for Quality PCB Using YOLO Algorithm with IoT Real-Time Streaming on Raspberry Pi," *JURNAL INFOTEL*, vol. 17, no. 2, Jul. 2025, doi: 10.20895/infotel.v17i2.1313.

[4] W. Nugroho, A. Alfattah, M. Jimmy, F. Arifianto, and A. Hadi, "Automated Waste Classification for Sustainable Cities Using YOLO Based CNN Integrated IoT," APIC, 2025.

[5] J. Zhou, W. Zhang, X. Sui, and Y. Chen, "Welding defect detection based on YOLOv8," in *Journal of Physics: Conference Series*, Institute of Physics, 2024. doi: 10.1088/1742-6596/2816/1/012045.

[6] M. A. K. Raiaan *et al.*, "A systematic review of hyperparameter optimization techniques in Convolutional Neural Networks," Jun. 01, 2024, *Elsevier Inc.* doi: 10.1016/j.dajour.2024.100470.

[7] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," Jul. 2022, [Online]. Available: http://arxiv.org/abs/2207.02696

[8] B. Huang *et al.*, "Improved YOLOv5 Network for Steel Surface Defect Detection," *Metals (Basel).*, vol. 13, no. 8, Aug. 2023, doi: 10.3390/met13081439.

[9] Z. Li *et al.*, "Liquid Reservoir Weld Defect Detection Based on Improved YOLOv8s," *Sensors*, vol. 25, no. 21, Nov. 2025, doi: 10.3390/s25216521.

[10] W. Liu, J. Hu, and J. Qi, "Resistance Spot Welding Defect Detection Based on Visual Inspection: Improved Faster R-CNN Model," *Machines*, vol. 13, no. 1, Jan. 2025, doi: 10.3390/machines13010033.

[11] S. Zhu and Y. Zhou, "MRP-YOLO: An Improved YOLOv8 Algorithm for Steel Surface Defects," *Machines*, vol. 12, no. 12, Dec. 2024, doi: 10.3390/machines12120917.

[12] A. H. Fristiana, S. A. I. Alfarozi, A. E. Permanasari, M. Pratama, and S. Wibirama, "A Survey on Hyperparameters Optimization of Deep Learning for Time Series Classification," 2024, *Institute of Electrical and Electronics Engineers Inc.* doi: 10.1109/ACCESS.2024.3516198.

[13] H. D. Purnomo, T. Gonsalves, E. Mailoa, F. Y. Santoso, and M. R. Pribadi, "Metaheuristics Approach for Hyperparameter Tuning of Convolutional Neural Network," *Jurnal RESTI*, vol. 8, no. 3, pp. 340–345, Jun. 2024, doi: 10.29207/resti.v8i3.5730.

[14] G. Salawu and B. Glen, "Integrating Artificial Intelligence into Mechatronics: A Comprehensive Study of Its Influence on System Performance, Autonomy, and Manufacturing Efficiency," Mar. 01, 2026, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/technologies14030143.

[15] A. Sharma, R. Chaturvedi, K. Sharma, S. Abrahim Binhowimal, J. Giri, and T. Sathish, "Enhancing weld quality of novel robotic-arm arc welding: Vision-based monitoring, real-time control seam tracking," *Ain Shams Engineering Journal*, vol. 15, no. 12, Dec. 2024, doi: 10.1016/j.asej.2024.103109.

[16] D. Say, S. Zidi, S. M. Qaisar, and M. Krichen, "Automated Categorization of Multiclass Welding Defects Using the X-ray Image Augmentation and Convolutional Neural Network," *Sensors*, vol. 23, no. 14, Jul. 2023, doi: 10.3390/s23146422.

[17] J. Duarte, M. F. Claro, P. M. A. Vitoriano, T. G. Amaral, and V. F. Pires, "Detection and Classification of Defects on Metal Surfaces Based on a Lightweight YOLOX-Tiny COCO Network," *Eng*, vol. 6, no. 11, Nov. 2025, doi: 10.3390/eng6110302.

[18] Q. Wu and X. Nie, "Improved YOLOv10: A Real-Time Object Detection Approach in Complex Environments," *Sensors*, vol. 25, no. 22, Nov. 2025, doi: 10.3390/s25226893.

[19] S. Ashrafi, S. Teymouri, S. Etaati, J. Khoramdel, Y. Borhani, and E. Najafi, "Steel surface defect detection and segmentation using deep neural networks," *Results in Engineering*, vol. 25, Mar. 2025, doi: 10.1016/j.rineng.2025.103972.

[20] Y. Liang, B. Yu, M. Ding, W. Hu, Y. Jin, and Y. Yuan, "WELD-DETR: A Real-Time Welding Defect Detection Framework with Multi-Scale Feature Fusion and Multi-Kernel Perception Optimization," *Sensors*, vol. 25, no. 22, Nov. 2025, doi: 10.3390/s25227024.

[21] G. Song *et al.*, "A Deep-Learning-Based Method for High-Precision Real-Time Detection of Steel Surface Defects," *Mathematics*, vol. 14, no. 4, Feb. 2026, doi: 10.3390/math14040621.

[22] J. Xu *et al.*, "LightYOLO: Lightweight model based on YOLOv8n for defect detection of ultrasonically welded wire terminations," *Engineering Science and Technology, an International Journal*, vol. 60, Dec. 2024, doi: 10.1016/j.jestch.2024.101896.

[23] P. Malik, V. Oberoi, M. Bhatia, A. Ghatewal, and T. Garg, "Real-Time Defect Detection Using Edge AI in Smart Manufacturing Systems," vol. 14, no. 2, 2026.

[24] C.-Y. Wang, I.-H. Yeh, and H.-Y. M. Liao, "YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information," Feb. 2024, [Online]. Available: http://arxiv.org/abs/2402.13616

[25] A. Wang *et al.*, "YOLOv10: Real-Time End-to-End Object Detection," Oct. 2024, [Online]. Available: http://arxiv.org/abs/2405.14458

[26] A. Al-Hyari and M. Abu-Faraj, "Hyperparameters Optimization of Convolutional Neural Networks using Evolutionary Algorithms," in *2022 International Conference on Emerging Trends in Computing and Engineering Applications, ETCEA 2022 - Proceedings*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/ETCEA57049.2022.10009778.

[27] S. Han, S. Wang, W. Liu, Y. Q. Gu, and Y. Zhang, "Swarm Intelligence-Enhanced Detection of Small Objects Using Key Point-Driven YOLO," *International Journal of Swarm Intelligence Research*, vol. 16, no. 1, 2025, doi: 10.4018/IJSIR.368649.

[28] M. Benmalek and A. Seddiki, "Particle swarm optimization-enhanced machine learning and deep learning techniques for Internet of Things intrusion detection," *Data Science and Management*, vol. 8, no. 4, pp. 423–435, Dec. 2025, doi: 10.1016/j.dsm.2025.02.005.

[29] R. Chaganti, A. Mourade, V. Ravi, N. Vemprala, A. Dua, and B. Bhushan, "A Particle Swarm Optimization and Deep Learning Approach for Intrusion Detection System in Internet of Medical Things," *Sustainability (Switzerland)*, vol. 14, no. 19, Oct. 2022, doi: 10.3390/su141912828.

[30] W. Nugroho, Rifdah Zahabiyah, Afianto, and Mada Jimmy Fonda Arifianto, "Application of Deep Learning YOLO in IoT System for Personal Protective Equipment Detection," *Jurnal E-Komtek (Elektro-Komputer-Teknik)*, vol. 8, no. 2, pp. 428–437, Dec. 2024, doi: 10.37339/e-komtek.v8i2.2187.

[31] M. A. K. Raiaan *et al.*, "A systematic review of hyperparameter optimization techniques in Convolutional Neural Networks," Jun. 01, 2024, *Elsevier Inc.* doi: 10.1016/j.dajour.2024.100470.

.