A Deep Learning System for Automated Weld Seam and Surface Defect Inspection

João A. G. Neves[1], Davidson Nunes de Oliveira[1], Helton Maia[1]

[1]Escola de Ciências e Tecnologia da Universidade Federal do Rio Grande do Norte (ECT/UFRN) Caixa Postal 1524, 59.078-970, Natal, RN, Brazil

joao.galdino.089@ufrn.edu.br, davidson.oliveira.175@ufrn.edu.br, helton.maia@ufrn.br

Abstract. *This work presents a deep learning-based system for automated weld seam and surface defect inspection. A REST API with NestJS sends uploaded images to a secure FastAPI service running a YOLOv11 model, which returns predictions with class, confidence, and bounding boxes. The system delivers high accuracy, reduces human error, and supports deployment on various devices, including smartphones, for flexible industrial integration.*

1. Introduction

Weld inspection is vital for ensuring the structural integrity of metallic components in industries like construction, oil and gas, and automotive manufacturing. However, manual methods suffer from subjectivity and inconsistency due to inspector fatigue and environmental factors [1, 2].

To overcome these issues, this study presents an AI-based system for automated weld bead detection using deep learning models trained with Roboflow-annotated datasets. A RESTful API with NestJS handles image uploads, which are processed by a FastAPI inference service secured with an API key. This architecture improves accuracy and efficiency while reducing reliance on manual inspection [3].

2. Methodology

2.1. Dataset and Training Process

The dataset annotations were created using the Roboflow platform. Bounding boxes and labels were employed to annotate the data. A custom API was developed to support machine training. All model training was conducted on an Nvidia GTX 1660 Ti graphics card. The dataset was divided into 87% for training (2577 images), 9% for validation (265 images), and 5% for testing (135 images). Preprocessing and data augmentation techniques were also applied to improve model generalization. The model architecture used was based on YOLOv11.

3. Results and Discussion

3.1. Training Performance and Model Convergence

The YOLOv11 model was trained for 130 epochs, showing consistent improvement in all key metrics. Figure 1 illustrates the evolution of losses and performance during training.

Table 1 summarizes the loss evolution, with significant reductions in classification loss (79.6% improvement), indicating effective learning of weld seam patterns.

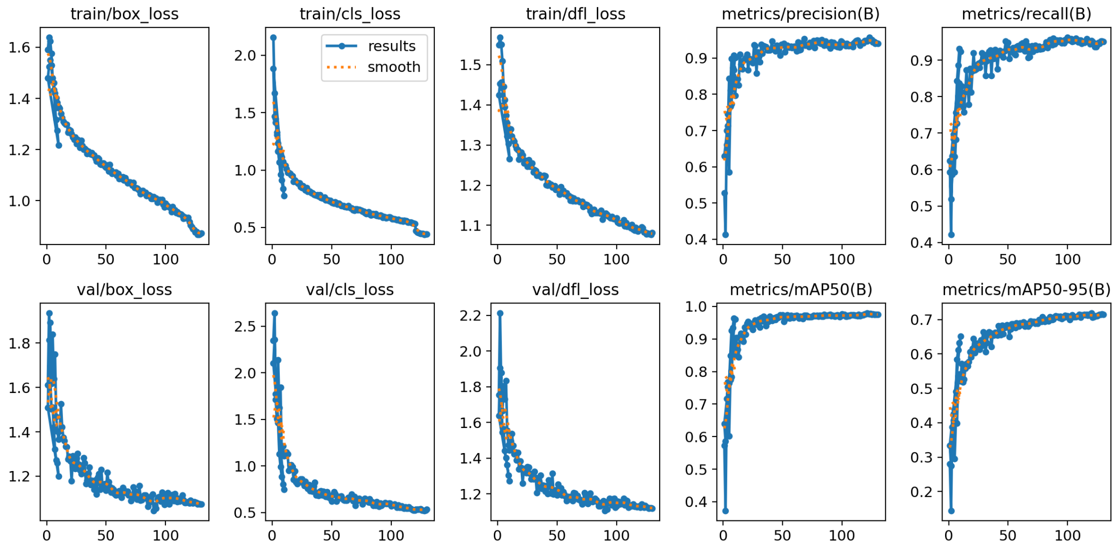

Figure 1. Training and validation curves of the YOLOv11 model over 130 epochs. The plots show reductions in loss functions and improvements in precision, recall, and mAP metrics, indicating effective learning and generalization.

Table 1. Loss Function Evolution

Loss	Epoch 1	Epoch 130	Improvement (%)
Box Loss (Train)	1.59	0.87	45.3
Classification Loss (Train)	2.16	0.44	79.6
Distribution Focal Loss (Train)	1.55	1.08	30.3
Box Loss (Val)	1.61	1.07	33.5
Classification Loss (Val)	2.35	0.53	77.4
Distribution Focal Loss (Val)	1.75	1.12	36.0

Validation losses followed similar trends, indicating strong generalization and no overfitting. The learning rate schedule (starting at 6.63×10^{-4} and annealing to 3.52×10^{-5}) facilitated smooth convergence, consistent with findings reported by Jegham et al. [4].

3.2. Model Performance

The YOLOv11 model shows outstanding weld seam detection with 94.0% precision, 95.1% recall, and 97.5% mAP@0.5 (Table 2). These high metrics confirm the model's accuracy in identifying weld seams and minimizing errors.

Table 2. Final Model Performance Metrics for Weld Seam Detection

Metric	Value	Interpretation
Precision	0.940	High accuracy in positive weld seam predictions
Recall	0.951	Excellent detection of actual weld seams
mAP@0.5	0.975	Superior localization accuracy of weld seams
mAP@0.5:0.95	0.715	Robust across various Intersection over Union thresholds

Training curves exhibited rapid improvement and gradual refinement, indicating effective transfer learning from YOLOv11 weights [4, 5]. With 60-70 second epochs, the model is industrially feasible. Figure 2 visually confirms its robust, accurate weld seam detection in diverse real-world images.

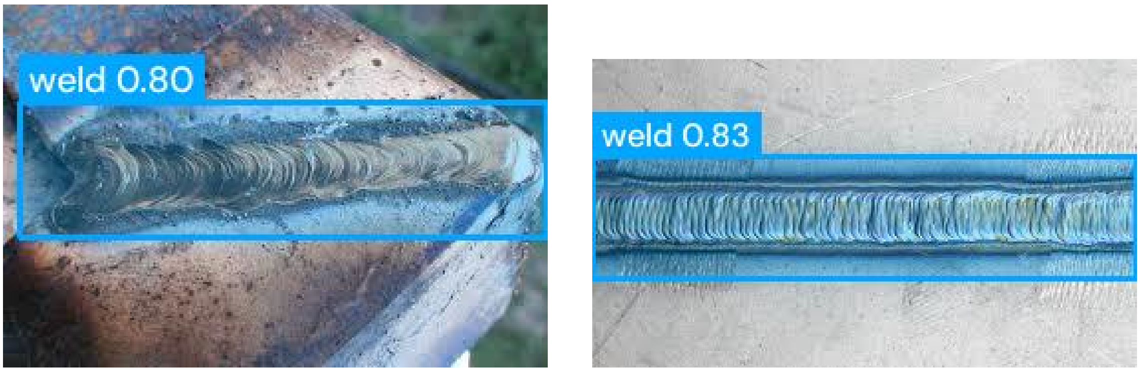

Figure 2. Examples of weld seam detection using the YOLOv11-based model. (a) Curved weld seam with a confidence score of 0.80. (b) Linear weld seam with a confidence score of 0.83.

3.3. System Implementation and User Experience

Our web application streamlines weld seam inspection with features like secure authentication, image uploads, real-time analysis, and report export [6, 7]. It uses NestJS for the main API and FastAPI for optimized inference, enabling efficient processing and returning JSON results with class, confidence, and bounding box data.

3.4. Comparative Performance and Deployment Considerations

The model surpasses prior methods (Table 3) in precision and mAP, aligning with recent studies on YOLOv11's superior detection performance in infrastructure inspection [4, 8].

Table 3. Performance Comparison

Method	Precision (%)	Recall (%)	mAP@0.5 (%)
Traditional CV	75–85	70–80	65–75
Early CNN	85–90	80–88	80–85
Recent YOLO	88–92	85–90	88–92
This Work	**94.0**	**95.1**	**97.5**

Training stability, efficiency, and scalability confirm the model's readiness for real-world deployment with consistent accuracy across diverse scenarios [5].

3.5. Industrial Adoption and Hardware Requirements

The deployment of the proposed system in industrial settings depends on hardware capable of efficiently running the YOLOv11 inference model and API services. For the FastAPI-based inference server, the minimum recommended specifications are:

- **GPU:** NVIDIA with at least 6 GB of VRAM (e.g., GTX 1660 Ti); newer GPUs may improve performance.

- **CPU:** Multi-core processors like Intel Core i5 (8th gen) or AMD Ryzen.
- **RAM:** Minimum of 16 GB to ensure smooth execution and image processing.

The NestJS API server requires only a standard CPU and 4 GB RAM. Centralized inference reduces hardware needs at inspection sites and supports image uploads from desktops or smartphones with internet access. This boosts portability and usability in the field. Optimizations such as quantization or pruning can further reduce hardware requirements for edge applications.

4. Conclusions

This work presented a scalable system for automated weld seam detection using deep learning and a cloud-based API infrastructure. The YOLOv11-based model showed strong performance, with 94.0% precision, 95.1% recall, and 97.5% mAP@0.5, surpassing traditional and earlier deep learning methods. The architecture streamlines inspection by reducing manual effort and supports deployment on devices like smartphones. Future work will focus on classifying specific weld defects and adapting the model for edge devices to broaden its industrial use.

5. References

[1] Y. Wang and R. X. Gao, "Condition monitoring and fault diagnosis of welding processes: A review," *Journal of Manufacturing Processes*, vol. 59, pp. 101–117, 2020.

[2] D. Jung, J. Lee, and J. Kim, "Automated welding defect detection system using deep learning in manufacturing," in *2018 IEEE International Conference on Industrial Technology (ICIT)*, 2018, pp. 819–824.

[3] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. X. Gao, "Deep learning and its applications to machine health monitoring," *Mechanical Systems and Signal Processing*, vol. 115, pp. 213–237, 2019.

[4] N. Jegham, C. Y. Koh, M. Abdelatti, and A. Hendawi, "Evaluating the evolution of yolo (you only look once) models: A comprehensive benchmark study of yolo11 and its predecessors," *arXiv preprint arXiv:2411.00201*, 2024. [Online]. Available: https://arxiv.org/abs/2411.00201

[5] X. Tao, Z. Zeng, J. Zeng, T. Zhao, and Q. Dai, "Pavement crack detection and identification based on improved yolov11," *International Journal of Cognitive Informatics and Natural Intelligence*, vol. 18, no. 1, pp. 1–20, 2024.

[6] M. Mazni, A. R. Husain, M. I. Shapiai, I. S. Ibrahim, R. Zulkifli, and D. W. Anggara, "Identification of concrete cracks using deep learning models: A systematic review," *Applications of Modelling and Simulation*, vol. 1, no. 1, pp. 1–14, 2023. [Online]. Available: https://arqiipubl.com/ojs/index.php/AMS_Journal/article/view/547

[7] Q. Yuan, Y. Shi, and M. Li, "A review of computer vision-based crack detection methods in civil infrastructure: Progress and challenges," *Remote Sensing*, vol. 16, no. 16, p. 2910, 2024.

[8] Y. Zhang, W. Wu, J. Huang, and L. Cong, "Detection method of surface damage of concrete bridge based on improved yolov11," in *Proceedings of SPIE*, vol. 13107, 2024, p. 131071K.