MALATYA TURGUT OZAL UNIVERSITY JOURNAL of ENGINEERING and NATURAL SCIENCES Volume 5, Issue 2 (2024) 77-81

NATURENGS
E-ISSN 2717-8013
https://dergipark.org.tr/en/pub/naturengs

Weld Defect Detection with YOLOv10

Emine Cengil

Bitlis Eren University, Department of Computer Engineering, Bitlis Türkiye – 13100

ARTICLE INFO

Received 28.11.2024
Accepted 16.12.2024

Doi: 10.46572/naturengs.1592956

ABSTRACT

Welding is one of the important processes used in various industries with various applications. The change of weld defects has the feature of continuous critical monitoring of safety, quality control and cost-effectiveness in industrial production ranges. Although traditional high accuracy offers time-consuming, it depends on the product and operator experience. This study implements three-class detection of Bad Weld, Good Weld and defect with YOLOv10 object detection for automatic detection of weld defects. In the relevant data set, the model provides 0.939 Precision-Confidence and 0.91 Recall-Confidence values. The obtained results show that the model can detect defects. This study aims to reveal the potential of deep learning in the detection of weld defects, providing a faster, cost-effective and reliable solution.

Keywords: Artificial intelligence, YOLOv10, Defect Detection, Object Detection, Deep Learning.

1. Introduction

Weld defect detection has a critical application area in many fields ranging from industrial manufacturing processes to the automotive and aerospace industries. Weld defects can be structural defects, cracks, or faults that usually occur during the welding process [1]. Detection of such defects is very important in terms of improving production quality, ensuring product safety and preventing failures. Traditional methods such as visual inspection, ultrasonic testing, magnetic grounding, etc. for the detection of weld defects are often time-consuming and error-prone [2]. In recent years, artificial intelligence techniques have been increasingly used to solve such problems faster and more accurately. Deep learning methods for weld defect detection have been frequently used in the literature.

Image processing techniques are one of the most widely used deep learning approaches for weld defect detection. Visual data may include microstructures, cracks or other distortions in the weld zone. Convolutional Neural Networks (CNN) show high success in object recognition and classification by extracting features from such visual data [3].

S. Oh et al. [4] introduced an approach for the automated detection of weld defects utilizing Faster R-CNN grounded in deep learning, aiming to carry out both feature extraction and classification within a unified algorithm while achieving comprehensive automation. Algorithms were analyzed to learn the data and data augmentation method was used to artificially increase the limited data. Two integrated feature extractors of Faster R-CNN are chosen to effectively extract the features from the radiographic test image.

D. Palma-Ramírez et al. [5] introduced an innovative CNN model derived from ResNet50 to differentiate four categories of weld defects in radiographic images: crack, pore, non-penetration, and no defect. To enhance generalization and prevent overfitting, they employed layered cross-validation, data augmentation, and regularization techniques. The model was evaluated using three datasets, achieving accuracies of 98.75%, 90.255%, and 75.83% respectively.

Detecting source defects often requires an anomaly detection approach. Deep learning models learn what is normal in the training data and can be used to detect anomalies. Such approaches can be useful in scenarios where labeled data is limited [6].

G. Stemmer et al. [7] sought to utilize deep learning techniques for the real-time detection of welding defects by capturing the welding process with microphones and cameras. They compiled an extensive database consisting of over 4000 welding samples that included a variety of weld types, materials, and defect categories. Notably, their multi-modal strategy achieved an average Area Under the ROC Curve (AUC) score of 0.92 across all eleven defect categories represented in the dataset.

H. Engbers et al. [8] introduced a model selection technique for detecting multivariate anomalies in manufacturing systems by employing a meta-learning method based on multi-output regression. The

* Corresponding author. e-mail address: ecengil@beu.edu.tr
ORCID : 0000-0003-4313-8694

suggested approach leveraged the strengths of meta-learning to identify and understand intricate relationships within multivariate data, facilitating the selection of the most effective anomaly detection model.

The process-based nature of welding operations creates the need to model with time series data. Long Short Term Memory (LSTM) networks are effective for examining data that changes over time. LSTM can learn the dynamics of welding processes and detect any abnormal changes during the process [9].

GANs are another approach in deep learning techniques that has attracted attention in recent years. GANs can be used to generate data where data is limited and to diversify the existing dataset. This allows the model to generalize better for the detection of source defects [10]. R. Guo et al. [11] introduced a method for detecting weld defects by utilizing a generative adversarial network alongside transfer learning to address the issue of data imbalance in radiographic images, thereby enhancing the accuracy of defect detection. The defect detection model effectively identified five different types of weld defects: cracks, lack of fusion, penetration, porosity, and slag inclusion, achieving an F1 score of 90% and a recognition accuracy of 92.5%.

In this study, welding defects were detected using YOLOv10 (You Only Look Once version 10). In the experiments with the three-class dataset, 0.939 Precision-Confidence and 0.91 Recall-Confidence were obtained. This accuracy metric indicates that the suggested model is suitable for the task of detecting weld defects. In the first part of the organization of the paper, general information about the subject and related literature studies are given. The materials and methods used are analyzed in the second part of the paper. The dataset and the YOLOv10 model are presented in this section, followed by the implementation and the results obtained in the third section. The study is finalized with a conclusion section.

2. Material and Methods

The dataset and method used are explained in this part of the study.

2.1. Dataset

The weld detection dataset [12] was used to perform the study. The Object Detection dataset contains 3 classes to detect defects on weld surfaces: bad weld, good weld, and defect. This dataset is formatted for the object detection task in the YOLO annotation format. The images in this dataset are taken from various image collections and datasets. Some images belonging to the dataset are given in the figure. The images in the first row belong to the “Bad Weld” class. The following rows show images belonging to the “Good Weld” and “Defect” classes, respectively. The last line presents images of examples where different classes coexist.

Figure 1. Image samples from dataset [12].

The distribution of examples across each class in the dataset is illustrated in Figure 2.

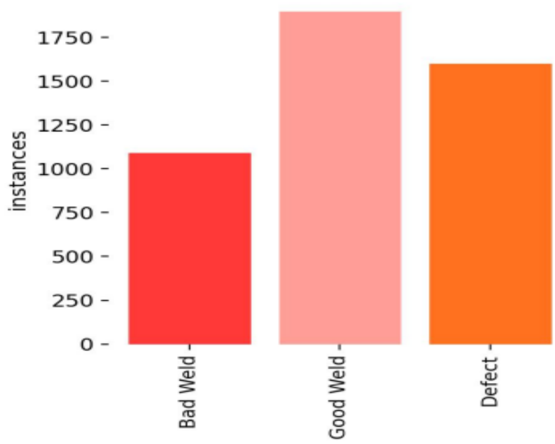

Figure 2. The distribution of examples across each class in the dataset.

2.2. YOLOv10

In recent years, YOLOs have emerged as the dominant paradigm in the field of real-time object detection due to their effective balance between computational cost and detection performance. Researchers have achieved significant progress by studying the architectural designs, optimization goals, data augmentation strategies, and others for YOLOs [13]. YOLOv10 is the tenth version of the YOLO model used in the field of object detection. This model is an extremely fast and efficient structure that performs the function of detecting and classifying objects in images. The YOLO series was first introduced in 2015 by Joseph Redmon et al., and each new version includes significant improvements in both accuracy and speed. YOLOv10 is specifically designed for real-time object detection and is widely used in industrial applications, security systems, autonomous vehicles, and robotics. YOLOv10 offers several technical innovations and optimizations compared to previous versions. These innovations enable the model to run faster, have lower latency, and provide more accurate results [14].

YOLOv10, similar to its previous versions, uses a convolutional neural network (CNN) architecture. However, in YOLOv10, the network depth has been increased and more parameters have been optimized. This improves the model's ability to detect more complex objects. YOLOv10 enables the model to use processor resources more efficiently thanks to weight sharing and efficient computational techniques. This provides a significant advantage especially in real-time applications.

YOLOv10 incorporates attention mechanisms into its architecture by using transformer-based layers, which have gained popularity in recent years. This enhancement leads to improved precision in object detection and localization.

YOLOv10 strikes an excellent balance between speed and accuracy. The speed issues seen in earlier iterations have been significantly minimized through optimization strategies and advancements in model architecture. YOLOv10 is capable of processing thousands of frames per second with impressive speed. It employs sophisticated data augmentation techniques to enhance data diversity. Additionally, it streamlines the training process by leveraging pre-trained weights through transfer learning methods.

YOLOv10 has the advantages of higher accuracy, real-time performance, and model size compared to other YOLO versions. For these reasons, YOLOv10 was preferred in the study.

2.3. Performance Metrics

Precision, recall, intersection over union, mean Average Precision, Frame Per Second,(FPS), latency are the basic metrics utilized to measure the performance of object detection algorithms [13, 15, 16].

Precision: Measures how many of the objects detected by the algorithm are correct. High precision means that false positives are low. The formula is as in equation (1).

$$Precision = \frac{TP}{TP+FP} \tag{1}$$

Recall: Measures how many real objects are correctly detected. High recall means fewer missed objects. The formula is as in equation (2).

$$Recall = \frac{TP}{TP+FN} \tag{2}$$

F1-Score: Used to measure the balance between precision and recall. The formula is as in equation (3).

$$F1 - Score = \frac{2*Precision*Recall}{(Precision+Recall)} \tag{3}$$

Intersection over Union (IoU): Measures how much the bounding box detected by the algorithm overlaps with the actual bounding box. IoU > 0.5 is generally accepted for the algorithm to be considered to have made an accurate detection. The formula is as in equation (4).

$$IoU = \frac{Intersection\ Area}{Union\ Area} \tag{4}$$

Mean Average Precision (mAP): It is one of the most widely used metrics. It represents the averaged average precision (AP) values for all classes and IoU threshold values. It is found by calculating the area under the average precision curve (Precision-Recall Curve).

FPS (Frame Per Second): Evaluates the real-time operability of the algorithm. Faster models are more valuable, especially in practical applications.

Latency: The time it takes for the algorithm to process a frame. Lower latencies are critical for real-time applications.

Confusion Matrix and Associated Metrics:

True Positive (TP): Objects that were accurately identified.

False Positive (FP): Objects that were incorrectly identified.

False Negative (FN): Objects that were overlooked.

True Negative (TN): Areas that are not relevant.These metrics are important for comparing both the accuracy and efficiency of algorithms and are evaluated with different priorities for different application domains.

3. Experimental Results

This study was conducted using the Python programming language. 1619 datasets were used for training and 283 for validation. The experiment requires a Windows 10 operating system, 16 GB of RAM, an NVIDIA GeForce 3050 Ti GPU, and an Intel(R) Core(TM) i7-11370H CPU.

YOLOv10 has six variants with various scales to meet different application needs. These are; YOLOv10-n, YOLOv10s, YOLOv10m, YOLOv10b, YOLOv10l and YOLOv10x [14]. In the study, YOLOv10n, which is the smallest scale, was preferred. Information on training parameters is given in Table 1.

Table 1. Hyperparameters.

Parameters	Values
Epoch	50
Batch Size	8
Momentum	0.937
Weight Decay	0.0005
Learning Rate	0.01

The complexity matrix obtained in the task of detecting defects in weld images as “Bad Weld”, “Good Weld”, and “Defect” is shown in Figure 3.

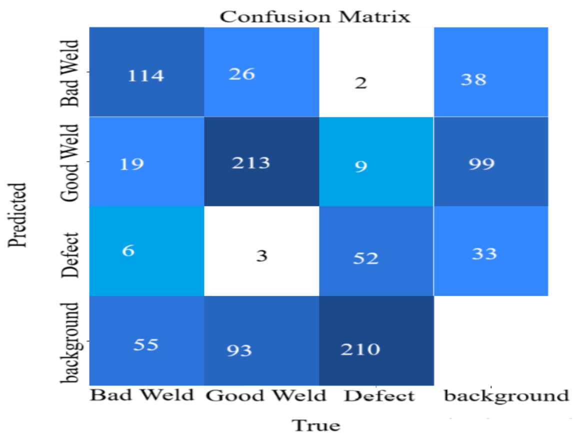

Figure 3. Confusion matrix of the model.

The two fundamental performance metrics utilized in the realm of object detection, namely the precision-confidence curve and the precision-recall curve, offer insights from various perspectives to assess a model's predictive effectiveness.

The Precision-Confidence Curve connects the accuracy of a model's predictions with the confidence threshold.

Confidence Score: An indication of how certain the model is about the presence of the object it forecasts.

Various confidence threshold values are established. Precision is computed for each threshold. At elevated confidence thresholds, the model produces fewer predictions, yet these are typically more precise, leading to an increase in precision. The Precision-Confidence curve helps to analyze the performance of object detection systems based on the model's confidence level in its predictions. The Precision-Confidence curve for the method is illustrated in Figure 4. A Precision-Confidence value of 0.939 is attained across all classes.

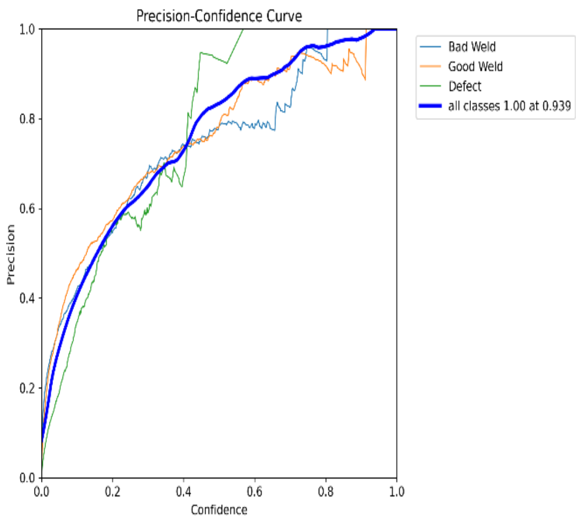

Figure 4. Precision-Confidence Curve of the model.

The precision-recall curve shows the relationship between the model's precision and recall. Precision and recall are calculated with different threshold values or various settings. The precision-recall curve shows how precision changes at higher recall levels. Precision and recall are usually inversely proportional: precision usually decreases for higher recall, and vice versa. It is used to evaluate the overall accuracy and coverage (recall) of the model's predictive performance. This curve can be especially informative in imbalanced data sets.

The precision-recall curve usually provides a more general performance evaluation, while the precision-confidence curve is more suitable for examining how confidence values are adjusted. The precision-recall curve of the model is given in Figure 5.

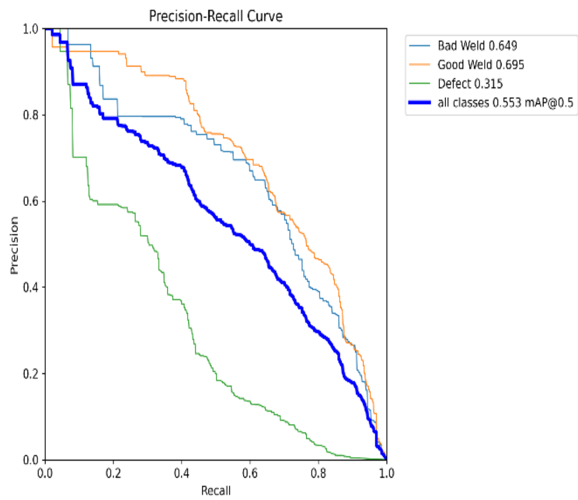

Figure 5. Precision-Recall curve of the model.

The recall-confidence curve is a performance evaluation tool that visualizes how the recall value of a model changes as the confidence threshold changes. This curve helps understand how comprehensively the model can predict at different confidence levels. The recall-confidence curve focuses on comprehensiveness (recall) rather than a single measure of accuracy like the precision-confidence curve. Therefore, it is especially preferred when it is desired to understand how well the model can detect all true positives. Figure 6 presents the model's Recall-Confidence curve. A Recall-Confidence value of 0.91 was achieved in all classes.

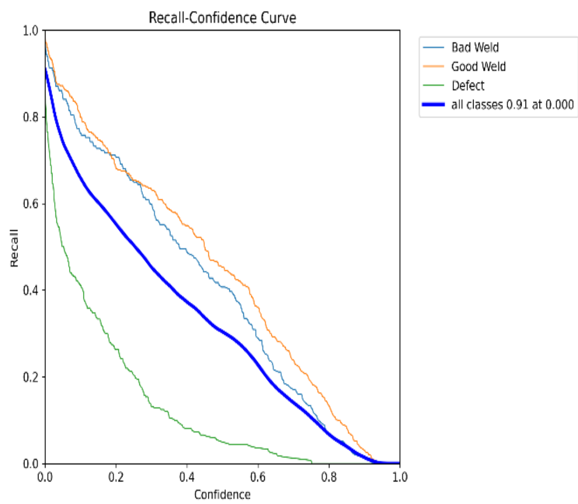

Figure 6. Recall-Confidence curve of the model.

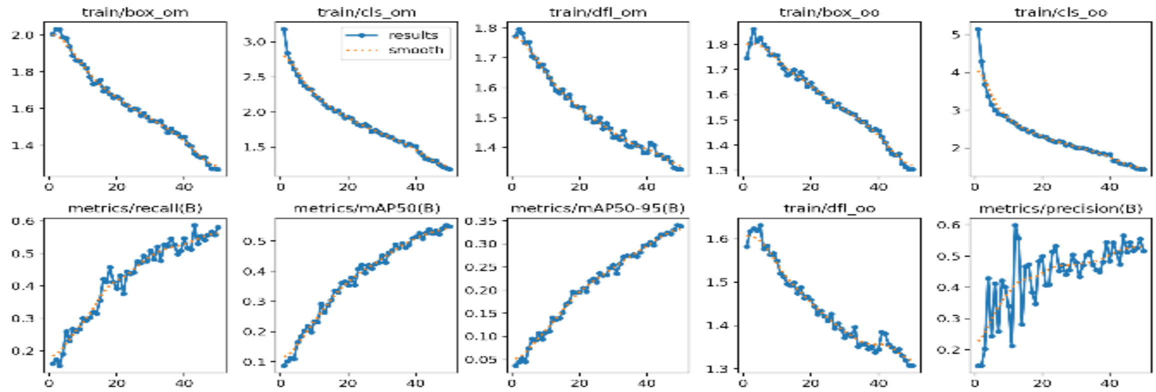

Figure 7. Results of the model.

Other results obtained from the model are as given in Figure 7. YOLOv10 has achieved successful results in the detection of defects in the source in the relevant dataset.

4. Conclusions

In this study, the YOLOv10n model was used for the detection of weld defects and showed strong performance and efficiency. Among the six variants of the YOLOv10 family, YOLOv10n, the smallest scale variant, was preferred due to its suitability for the dataset and computational efficiency. The model achieved impressive results by correctly classifying into "Bad Weld," "Good Weld" and "Defect" categories. Precision-confidence, precision-recall and recall-confidence curves, which are among the performance metrics, confirmed the model's reliability. These findings reveal the effectiveness of the YOLOv10n model in the task of detecting weld defects, especially within the scope of the used dataset. This study shows that the YOLOv10n model provides a robust and effective solution for automatic defect detection in industrial applications. Future research shows that this approach can be applied to larger, more diverse datasets. The promising results obtained here provide a basis for the development of non-destructive testing methods using deep learning methods.

References

[1] Madhvacharyula, A. S., Pavan, A. V. S., Gorthi, S., Chitral, S., Venkaiah, N., & Kiran, D. V. (2022). In situ detection of welding defects: A review. Welding in the World, 66(4), 611-628.

[2] Mohandas, R., Mongan, P., & Hayes, M. (2024). Ultrasonic Weld Quality Inspection Involving Strength Prediction and Defect Detection in Data-Constrained Training Environments. Sensors, 24(20), 6553.

[3] Cengil, E., & Çınar, A. (2016). A new approach for image classification: convolutional neural network. European Journal of Technique (EJT), 6(2), 96-103.

[4] Oh, S. J., Jung, M. J., Lim, C., & Shin, S. C. (2020). Automatic detection of welding defects using faster R-CNN. Applied Sciences, 10(23), 8629.

[5] Palma-Ramírez, D., Ross-Veitía, B. D., Font-Ariosa, P., Espinel-Hernández, A., Sanchez-Roca, A., Carvajal-Fals, H., ... & Hernández-Herrera, H. (2024). Deep convolutional neural network for weld defect classification in radiographic images. Heliyon, 10(9).

[6] Pang, G., Shen, C., Cao, L., & Hengel, A. V. D. (2021). Deep learning for anomaly detection: A review. ACM computing surveys (CSUR), 54(2), 1-38.

[7] Stemmer, G., Lopez, J. A., Ontiveros, J. A., Raju, A., Thimmanaik, T., & Biswas, S. (2024). Unsupervised Welding Defect Detection Using Audio And Video. arXiv preprint arXiv:2409.02290.

[8] Engbers, H., & Freitag, M. (2024). Automated model selection for multivariate anomaly detection in manufacturing systems. Journal of Intelligent Manufacturing, 1-19.

[9] Özbay, E., Çinar, A., & Özbay, F. A. (2021). 3D Human Activity Classification with 3D Zernike Moment Based Convolutional, LSTM-Deep Neural Networks. Traitement du Signal, 38(2), 269-280.

[10] Sajeeda, A., & Hossain, B. M. (2022). Exploring generative adversarial networks and adversarial training. International Journal of Cognitive Computing in Engineering, 3, 78-89.

[11] Guo, R., Liu, H., Xie, G., & Zhang, Y. (2021). Weld defect detection from imbalanced radiographic images based on contrast enhancement conditional generative adversarial network and transfer learning. IEEE Sensors Journal, 21(9), 10844-10853.

[12] Welding Defect – Object Detection, Link address: https://www.kaggle.com/datasets/sukmaadhiwijaya/welding-defect-object-detection.

[13] Cengil, E., Çinar, A., & Yildirim, M. (2021, September). A case study: Cat-dog face detector based on YOLOv5. In 2021 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT) (pp. 149-153). IEEE.

[14] Wang, A., Chen, H., Liu, L., Chen, K., Lin, Z., Han, J., & Ding, G. (2024). Yolov10: Real-time end-to-end object detection. arXiv preprint arXiv:2405.14458.

[15] Kutlu, F., Ayaz, İ., & Garg, H. (2024). Integrating fuzzy metrics and negation operator in FCM algorithm via genetic algorithm for MRI image segmentation. Neural Computing and Applications, 1-21.

[16] Ayaz, I., Kutlu, F., & Cömert, Z. (2024). DeepMaizeNet: A novel hybrid approach based on CBAM for implementing the doubled haploid technique. Agronomy Journal, 116(3), 861-870.