Development of Automated Welding Defect Detection Based on YOLO Algorithm

Junwei Tan[1], Helin Xu[2*] and Xintong Zhang[3]

[1] School of Education, Beijing Institute of Technology, Beijing, China
[2] HDU-ITMO Joint Institute, Hangzhou Dianzi University, Zhejiang, China
[3] Department of Information Science and Engineering, Ocean University of China, Shandong, China
23320411@hdu.edu.cn

Abstract. The application of deep learning algorithms to welding defect detection has advanced rapidly in recent years within domestic and international research communities. This paper reviews the developmental trajectory of deep learning technologies and the evolutionary improvements of the You Only Look Once (YOLO) algorithm from versions v1 to v11. Focusing on the application of YOLO in welding defect detection, it provides a systematic exposition of architectural enhancements in the YOLOv3 to YOLOv8 series, particularly in five key aspects: noise robustness optimization, real-time performance optimization, adaptive feature selection, feature fusion, and lightweight design, while critically analyzing their innovative contributions. A comprehensive comparative analysis of these models reveals that, despite continuous efforts by global laboratories to develop novel YOLO variants for welding processes, the generalization capabilities of existing technologies under complex operational conditions still require further enhancement. Building on this conclusion, the paper prospects the application potential of multi-scale analysis and few-shot learning in welding defect detection and outlines future research directions in this field.

Keywords: Machine vision, deep learning, YOLO, Automated welding, defect detection.

1 Introduction

In the contemporary electronics industry, the demand for equipment precision and cost-effectiveness continues to escalate. As a fundamental process in industrial production, welding requires high-precision quality inspection to ensure subsequent manufacturing and application reliability. However, the diversity of welding defect types, structural complexity, and microscopic dimensions impose stringent requirements on detection methodologies.

Traditional welding defect detection methods, including visual inspection, penetrant testing, radiographic testing, and conventional machine vision techniques, exhibit notable limitations. Firstly, visual inspection heavily relies on subjective judgment and

J. Iqbal (ed.), *Proceedings of the 2025 2nd International Conference on Mechanics, Electronics Engineering and Automation (ICMEEA 2025)*, Advances in Engineering Research 272,
https://doi.org/10.2991/978-94-6463-821-9_100

struggles to identify internal defects. Secondly, while radiographic testing enables internal structure analysis, it suffers from high costs, operational complexity, and a lack of real-time capabilities. Additionally, traditional machine vision approaches, such as template matching or statistical classifiers, depend on manually engineered features, rendering them sensitive to image noise and illumination variations, thereby hindering their adaptability to automated detection of complex defects.

With advancements in computer hardware performance, researchers have increasingly explored sophisticated image processing algorithms. Early rule-based feature extraction methods have gradually been supplanted by deep learning-based end-to-end detection frameworks. By leveraging convolutional neural networks (CNNs) to autonomously learn multi-level feature representations, deep learning significantly enhances the robustness of defect recognition. Among these frameworks, the YOLO algorithm has emerged as a focal point in industrial inspection due to its real-time performance and balanced accuracy-speed trade-off. Through single-forward propagation to simultaneously achieve target localization and classification, YOLO satisfies the immediacy requirements of online welding defect detection. This paper aims to review recent advancements in YOLO-based deep learning algorithms for welding defect detection, systematically analyze improvement strategies such as lightweight design and noise robustness optimization, and compare the performance variations among different model variants. By synthesizing technological innovations and future directions, this work seeks to provide theoretical references for optimizing industrial inspection systems.

2 Deep Learning Overview

2.1 Convolutional Neural Network

The development of CNNs has greatly improved image processing methods in the field of deep learning. There are two major problems with traditional image processing techniques. First, the massive volume of data—for instance, processing a 1000 x 1000 pixel image with RGB three-channel requires handling over 3 million parameters. Second, the picture features cannot be successfully preserved by simple digitization. CNNs address these issues by simulating the functioning of the human visual system. The main concept is to use spatial down sampling, weight sharing, and local sensory fields to effectively extract hierarchical characteristics from data. CNNs typically consist of a convolutional layer, an activation function, a pooling layer, and a fully connected layer, as depicted in Figure 1.

In the architectural design of convolutional neural networks, the convolutional layer undertakes the important task of extracting local features of an image. Taking a photo with a size of 1000*1000 pixels as an example, the convolutional layer is able to process the photo with the help of 5*5 convolutional kernels, transforming it into a feature map of 996*996 pixels. By down sampling, the pooling layer, on the other hand, drastically shrinks the feature map's size, which lowers the model's parameter count. The collected features are subsequently mapped to the output layer via the fully connected layer. Each

neuron in the fully connected layer is coupled to every other neuron in the layer above, and weight and bias parameters are used to effectively integrate the data.

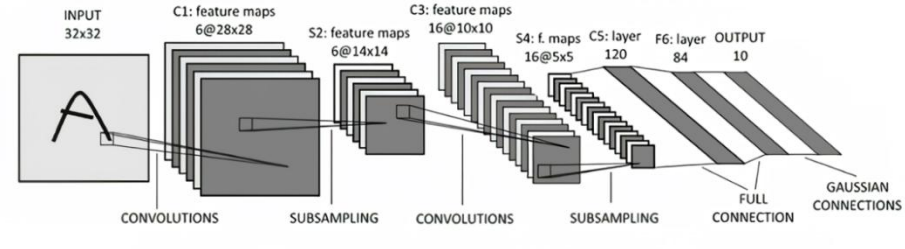

Fig. 1. CNNs Structure [1].

The role of activation functions in CNN. Activation functions can introduce non-linear features that allow convolutional neural networks to express complex functions. Nonlinear features fit the data better. This paper lists two common activation functions, Sigmoid and ReLU [1]. When combined, these activation functions improve the network's capacity for learning and representation, albeit each one has advantages in particular situations.

In neural networks, Sigmoid functions are frequently employed to solve bisection difficulties. Displays an S-shaped curve after mapping the input values to the interval (0, 1) [2]. When the input value is tiny, the output tends to zero, and when the input value is large, it tends to one (Figure 2).

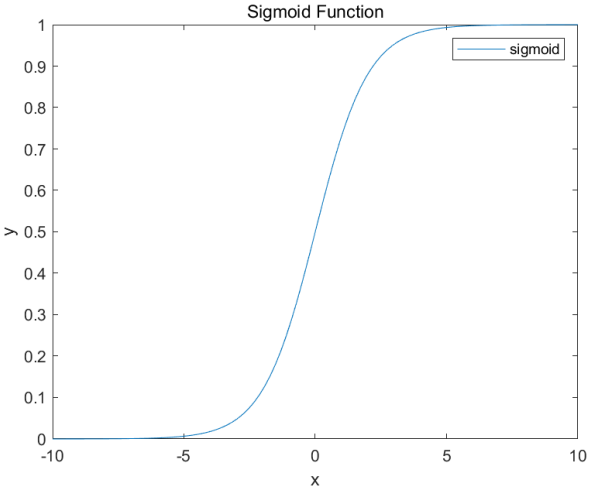

Fig. 2. Sigmoid Function [2].

The ReLU function is often used to correct linear units. When the input is greater than or equal to 0, the output is equal to the input (Figure 3) [3].

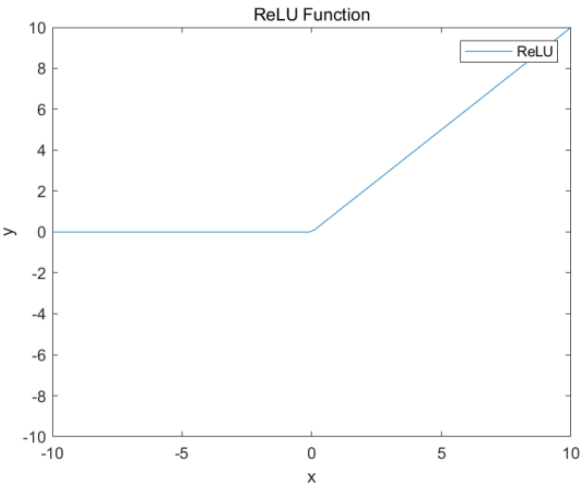

Fig. 3. ReLU Function [3].

Gradient Descent and Backpropagation. In deep learning, gradient descent is a popular optimization technique for minimizing the loss function. By computing the gradient of the loss function with respect to the model parameters, the model parameters are modified to minimize the loss. Gradient descent can be expressed mathematically as:

$$\theta_{\text{new}} = \theta_{old} - \eta \nabla_{\theta} L(\theta) \tag{1}$$

The gradient of the loss function with respect to the parameter is denoted by $\nabla_{\theta} L(\theta)$, where θ is the model parameter and η is the learning rate.

The backpropagation algorithm computes the gradient by means of a chain law, passing the error from the output layer to the previous layer in reverse order, updating the weights and biases of each layer to minimize the loss function. Specifically, the partial derivatives of the loss function with respect to the output of each neuron are computed according to the chain law to obtain the gradient of the loss function with respect to each weight and bias. The weights and biases of the network are then updated using optimization algorithms (e.g., gradient descent, stochastic gradient descent, Adam, etc.) based on the calculated gradients. The mathematical expression for backpropagation is given by.

$$\delta^{l} = ((W^{l+1})^{T} \delta^{l+1}) \odot \sigma'^{(z^{l})} \tag{2}$$

Loss Function. The difference or mistake between the model's expected and actual results is measured by the loss function. The model's performance improves with a smaller loss function value. Mean square error (MSE), cross-entropy loss (CE), and hinge loss are examples of common loss functions [4]. In regression issues, the squared difference between the true and predicted values is determined using the Mean Square Error (MSE). It can be expressed mathematically as:

$$L_{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (3)$$

In classification issues, the difference between the true and predicted probability distributions is calculated using CE. It can be expressed mathematically as:

$$L_{CE} = -\sum_{i=1}^{n} y_i \log(\hat{y}_i) \quad (4)$$

The model's performance and training are significantly impacted by the loss function selection. In actuality, the type of task usually determines which loss function is best.

2.2 YOLO Algorithm

Joseph Redmon and his colleagues initially introduced the YOLO algorithm in 2015 [5]. The YOLO algorithm based on a fully convolutional neural network recognizes multi-targeted, atypical welding defects. It is suitable for detecting defects in welded joints under standard working conditions. It predicts the bounding box and class probabilities straight from the picture pixels, treating the target identification problem as a single regression problem. Compared to the two-stage method, improved YOLO algorithms like YOLOV4 can achieve better real-time performance and higher accuracy.

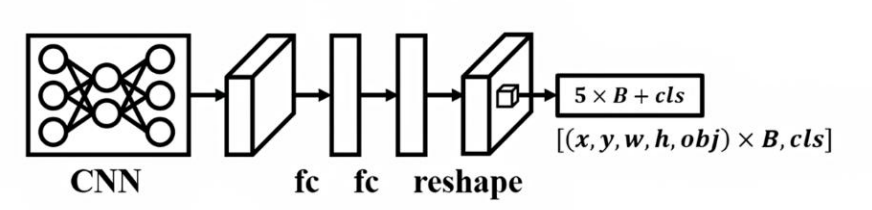

Fig. 4. Architecture of YOLOv1[5].

By segmenting the input image into grids and forecasting several bounding boxes along with their associated category probabilities and confidence scores in each grid, the YOLO algorithm detects targets in an image in real time. By turning the target identification work into a single regression problem, YOLO predicts bounding box and category probabilities straight from picture pixels, in contrast to two-stage techniques like RCNN. This method improves the detecting frame rate and real-time performance. An input layer, a convolutional layer, a pooling layer, a fully connected layer, and an output layer make up the structure of the YOLO algorithm. Receiving the input image data—typically a fixed-size image—is the responsibility of the input layer.

The network may learn intricate feature representations thanks to the convolutional layer's extraction of local features from the input data and the introduction of nonlinearities via the activation function.

2.3 Working Principle

Figure 4 illustrates the YOLOv1 architecture, in which a convolutional neural network (CNN) first extracts features from the input image. To acquire global contextual information, these extracted features are then fed into two successive fully connected layers. In order to enable prediction on a single grid basis, these global traits are then rearranged into a 2D structure.

Meshing. In YOLOv1, the input image was divided into 7×7 grids, each of which was responsible for predicting targets within that region. In subsequent versions, the number of grids increased to capture finer features. For example, YOLOv3 and YOLOv4 use smaller grid sizes such as 13×13, 26×26, and 52×52 to improve detection accuracy.

Predictive Bounding Box. The B bounding boxes that each grid forecasts are made up of the center coordinates (x, y), width (w), height (h), and confidence (c). The accuracy of the forecast and the likelihood that an object is inside the bounding box are shown by the confidence c. The accuracy of the forecast and the likelihood that an object is inside the bounding box are shown by the confidence c. The following formula determines the confidence level:

$$\mathrm{ci} = \Pr(\mathrm{object}) \times \mathrm{IOU} \tag{5}$$

where IOU indicates the intersection and concurrency ratio of the predicted bounding box to the actual bounding box, and Pr(object) is the likelihood that the bounding box contains an object.

Category Probability and Location. The predicted values for each bounding box include (x, y, w, h, c), and the category probabilities. The category probabilities are computed by the Softmax function and represent the probability that an object within that bounding box belongs to each category. The dimension of the output is S*S*(B*5+C), where S is the grid size, B is the number of bounding boxes predicted for each grid, 5 denotes the coordinates of the bounding box and the confidence level, and C is the number of categories.

NMS Remove Redundant Bounding Boxes. To remove duplicate results, YOLO uses the Non-Maximum Suppression (NMS) algorithm. The procedure is to first find the predictor with the highest confidence level, then calculate the IOU of this predictor with other predictors, and finally delete the predictors whose IOU reaches a certain threshold (usually 0.5).

The mathematical expression for NMS is:

$$\mathrm{NMS}(\mathrm{B}, \mathrm{C}, \mathrm{IOU}_{\mathrm{threshold}}) \tag{6}$$

where B is the set of predicted bounding boxes, C is the category probability, and IOU_threshold is the threshold of the IOU.

3 YOLO History

YOLOv1 has a fixed image size for input data, poor performance in small target detection and localization accuracy, and poor data processing ability for difficult samples. However, from the original YOLOv1 to the latest YOLOv11, each generation has introduced important innovations in feature extraction, bounding box prediction and optimization techniques. These improvements, especially in the backbone, neck and head components, have made YOLO the leading solution for real-time target detection. This paper focuses on analyzing YOLO algorithms that have been maturely applied, mainly related to the versions YOLOv5 and YOLOv8 [6].

In the YOLOv2 version a passthrough layer was added, which is a CNN network for stitching together photos of various sizes. The ability to detect targets of varying sizes can now be detected more easily (Figure 5).

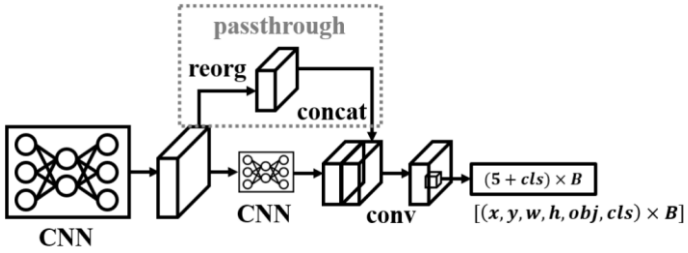

Fig. 5. Architecture of YOLOv2 [6].

YOLOv3 uses a darknet53 network instead of the original GoogLeNet-based CNN network, adds an FPN feature pyramid component, adds an SPP component, and improves multiscale detection by fusing features (Figure 6).

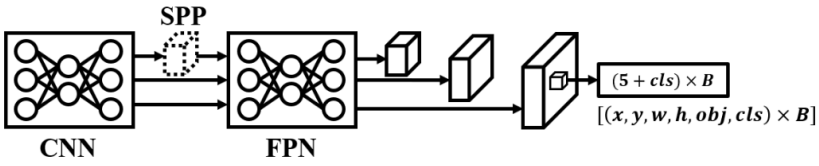

Fig. 6. Architecture of YOLOv3, YOLOv5 [7].

YOLOv4 uses the CSPDarknet53 network, an improved CNN network, and also introduces a more advanced loss function that introduces a PAN network. V4 is the first version of the algorithm to outperform two-stage algorithms such as FAST-CNN and R-CNN in terms of accuracy, and lays the groundwork for YOLO algorithms to flourish in the field of real-time detection.

YOLOv5 is based on the Pytorch platform for improved compatibility, benefiting from its dynamic computational graphs, ease of use, strong community support, well-distributed training capabilities, and continually improving deployment options. YOLOv5 is easier to use, debug, and scale, while being able to be effectively

implemented in actual production settings to satisfy the demands of various consumers. YOLOv10 proposes PSA space vectors, attention mechanisms that can enhance the algorithm's real-time performance and optimize the distribution of computational resources (Figure 7).

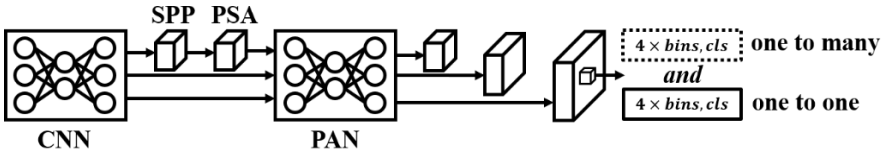

Fig. 7. Architecture of YOLOv10 [8].

This paper has compiled the development and progress of YOLO and its advantages and disadvantages over the generations, as shown in Table 1 and Table 2.

Table 1. Innovative Advances in Successive Generations of YOLOs [9][10].

Version	Advantage	Disadvantage
YOLOv2	The resolution of YOLOv2 is improved compared to YOLOv1, and the detection accuracy is improved by batch normalization. In the image training process, multi-scale training and dynamic resizing can be performed.	Multi-scale capability remains poor
YOLOv3	YOLOv3 introduces a multi-scale prediction mechanism to increase the accuracy of target detection, and adopts a feature pyramid network to further improve the detection performance.	Compared with v1,v2, training and inference are slower, and the recognition accuracy is lower, with poor recall.
YOLOv4	The model's performance is enhanced by using CSPDarknet53 as the backbone network; feature fusion is realized through technologies such as PANet and SAM to further enhance the detection accuracy	This model is more complex and requires greater computational resources
YOLOv5	Efficient target detection is achieved, and the introduction of adaptive model scaling improves the model's capacity for generalization.	Poor recognition accuracy and real-time performance
YOLOv8	Improved image feature extraction using innovative backbone network and neck structure. Mechanisms such as multi-scale feature fusion and adaptive anchor frame selection enhance the model's resilience and detection accuracy.	There are still deficiencies in the ability to detect very small sized objects. There is still room for improvement in areas such as occlusion handling and category generalization capabilities

Table 2. Key Improvements in YOLOv3-v8 Releases [11].

Version	Improve	Feature
YOLOv2	Batch normalization with high-resolution classifiers; use of anchor frame mechanism; dimensional clustering and spaghetti clustering; use of multi-scale training	Improved model robustness
YOLOv3	CSPDarknet53 More efficient backbone network to improve network inference speed and accuracy, introduction of CIoU loss function to improve bounding box regression performance	Good detection of small targets
YOLOv4	Proposing Bag of Freebies and Bag of Specials optimization strategies to improve model accuracy; CSPDarknet53 more efficient backbone network to improve network inference speed and accuracy; introducing CIoU loss function to improve bounding box regression performance	strikes a better balance between speed and accuracy.
YOLOv5	Based on PyTorch framework, the adaptive anchor box learning mechanism improves the detection efficiency; provides pre-trained models of various sizes to meet the needs of different scenarios.	A more user-friendly development and deployment experience.
YOLOv8	Enables users to grow in accordance with the requirements of the integrated multiple training and hyper-parameter optimization techniques to streamline the model tuning procedure. Combined tracking, segmentation, and detection capabilities	Performance and efficiency

4 Application of YOLO in Welding Defect Detection

4.1 Lightweight Model-Based Welding Inspection

Yue et al. suggested the YOLO-DEFW model, an improved version of YOLOv8, to overcome the shortcomings of the current YOLO models in real-time welding defect detection, such as their low accuracy and inadequate lightweight construction [12]. By lowering the number of parameters and computational complexity, this model enhances lightweight performance. In particular, the authors used Distribution Shift Convolution (DSConv), which breaks down typical convolution kernels into two parts: distribution shift and variable quantized kernel (VQK), in place of classic convolutional neural networks (CNNs). DSConv maintains output consistency across kernel distribution shifts while achieving reduced memory consumption and increased computational speed by keeping only kernel values in VQK. Additionally, the team improved the recognition capabilities for multi-scale weld flaws, especially tiny features, by using an EMA multi-scale attention mechanism that uses parallel subnetworks to simultaneously record channel and spatial information. Additionally, YOLOv8's original Focal Loss was combined with Weighted Cross-Entropy (WCE) to create a hybrid FWCE Loss function that increases the accuracy of detection for datasets that are unbalanced.

According to experimental data, the YOLO-DEFW model outperforms the baseline YOLOv8 in terms of precision, recall, and mean average precision (mAP), with small-feature identification accuracy rising by more than 12%. Additionally, the model

achieves a processing time of 3.9 ms per image by drastically reducing the number of parameters and GFLOPs. The model is appropriate for intelligent vision-based online inspection systems since these developments allow for the accurate and instantaneous detection of welding flaws.

A lightweight YOLOv5 variant was proposed by Long et al. in another work that aimed to identify weld defects using X-rays in a way that was both accurate and lightweight [13]. Through simple architectural changes, the model maintains detection accuracy while lowering processing requirements and parameter counts. To accomplish dual optimization in lightweight design, the team used cascaded backbone networks like MobileNet, GhostNet, and ShuffleNet, as well as depthwise separable convolutions and grouped convolutions. A Copy-Pasting data augmentation technique was presented to overcome the challenges of small-target and few-sample. This strategy synthesizes both faulty and defect-free images to enlarge the dataset, which improves the model's capacity to detect real-world flaws with high precision.

Significant parameter reduction is demonstrated by the experimental results: ShuffleNet decreased parameters by more than 82%, while MobileNet-based solutions reached 1/5 of the initial parameter count. These improvements open the door for effective implementation in industrial applications by making the lightweight YOLO models extremely compatible with embedded devices with limited resources.

4.2 Noise Robustness-Optimized Welding Inspection

To address performance bottlenecks of conventional YOLO models under extreme welding noise conditions, Song et al. [14]and Gao et al. [15] proposed the Light-YOLO-Welding and YOLO-Weld models, respectively.

By adjusting the backbone network, Song et al. created the Light-YOLO-Welding model based on YOLOv4 to produce a lightweight detector designed for intricate high-noise situations. In order to improve detection speed and lower network parameters, the team implemented MobileNetv3, which improved the model's suitability for identifying weld feature spots in the presence of high noise. In order to increase detection accuracy, robustness, and efficiency, they also substituted the Bidirectional Feature Pyramid Network (BiFPN), which was first suggested in EfficientDet, for the Path Aggregation Network (PANet). Although it addressed notable object size fluctuations across contexts, the original PANet in YOLOv4 was still not ideal. As shown in Figure 8, Song et al. modified BiFPN adaptively by adding bidirectional cross-scale connections to shorten the low-level information's propagation path to high-level layers.

Gao et al. proposed YOLO-Weld, a YOLOv5s-based welding feature detection network, to tackle extreme welding noise, as illustrated in Figure 9.

The model enhanced its architecture by including Reparameterized Convolutional Neural Network (RepVGG) blocks, boosting detection speed. A Normalization-based Attention Module (NAM) was implanted to amplify feature point perception. A lightweight decoupling head (RD-Head) was also created by the researchers to increase the precision of regression and classification. To improve model robustness in challenging noise situations, they also created a welding noise generating technique. The final result, which is filtered using box information to improve predictions, uses

the center point of the predicted bounding box as the welding feature point, as shown in Figure 6. Other configurations (such the loss function) are still consistent with YOLOv5s despite these architectural improvements.

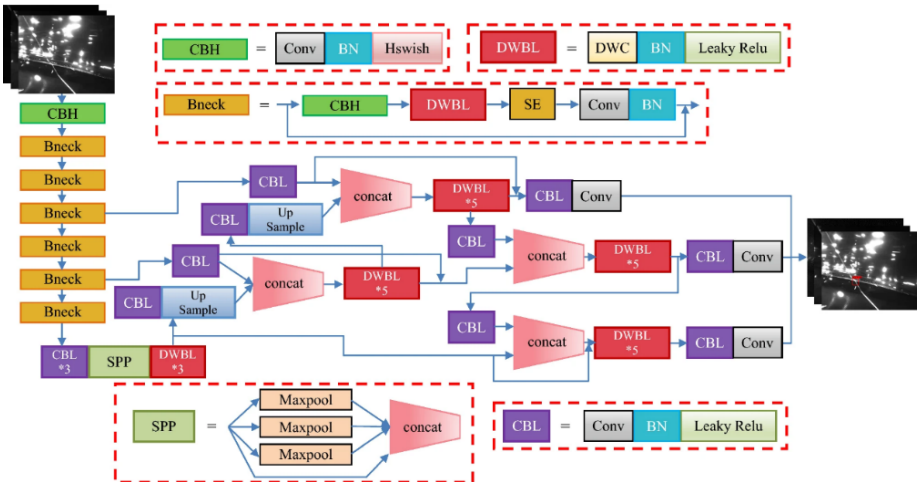

Fig. 8. The Net Structure of LiYW [14].

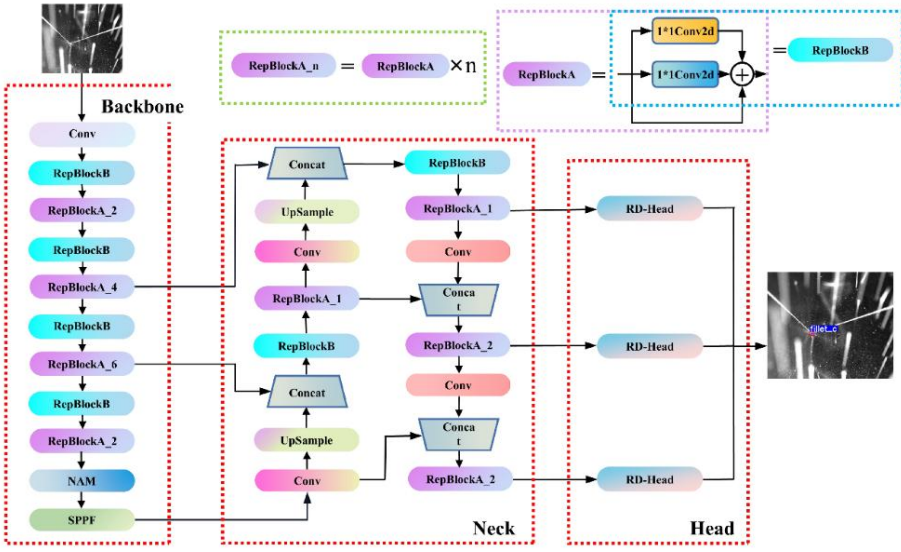

Fig. 9. Structure Diagram of the YOLO-Weld Neural Network [15].

4.3 Adaptive Feature Selection Welding Inspection

In response to the poor performance of the existing YOLO model in detecting complex, diverse, and minute welding defects, some research teams have adopted the

improvement strategy of adaptive feature selection, which improves the accuracy and efficiency of the model by introducing an adaptive computational method that autonomously selects and focuses on certain portions or features of the input according to the different characteristics of the input data.

By incorporating an attention mechanism and a feature pyramid network, Wang et al.'s [16] YOLO-AFK model, which is an enhanced adaptive feature selection model, can automatically choose the best features for weld defect detection, improving the model's recognition ability for complex defects. Among these, the FANet module uses the spatial attention mechanism and local perception capability to adaptively adjust the receptive field and feature weights, which improves the model's ability to detect small defects; the AKConv module uses a variable convolution kernel and non-uniform sampling to adaptively adjust the convolution kernel's shape and the sampling position, which allows the model to extract features more flexibly and improves its ability to detect irregular shape defects; and the C2f module adaptively selects and fuses various levels of feature information through multi-scale feature fusion and feature complementarity, which improves the model's ability to perceive subtle features.

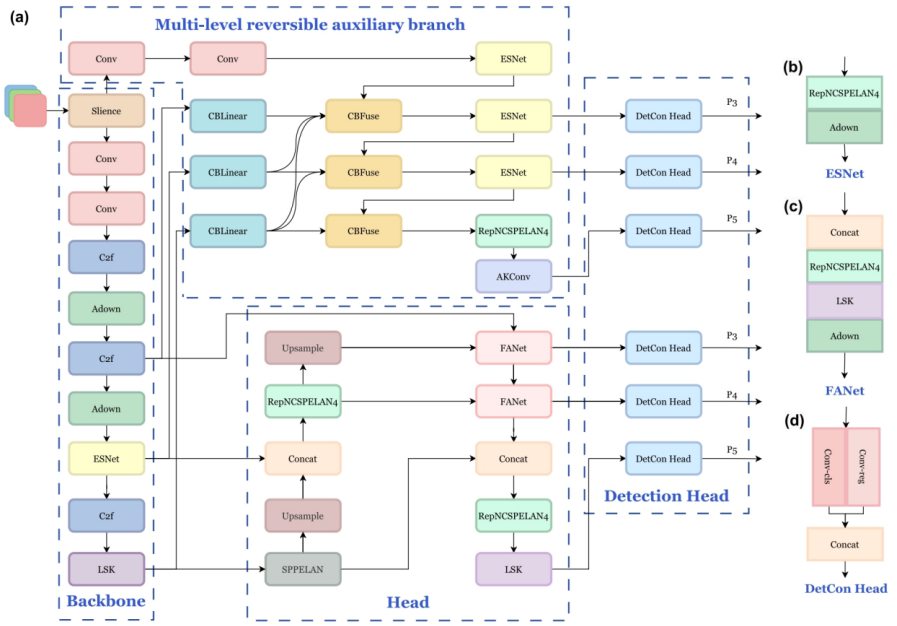

Fig. 10. Overall Architecture of YOLO-AFK [16].

Subsequent trials demonstrate that YOLO-AFK performs better than other advanced networks in a number of parameters, including a 23% improvement in frame rate (FPS), a 10.1% increase in precision, and a 5.6% rise in mean accuracy (mAP). YOLO-AFK exhibits significant promise and usefulness in industrial production settings, particularly in the identification of flaws in intricate solder junctions.

4.4 Welding Detection Based on Feature Fusion

Of course, some teams are not willing to stick to the one-sided enhancement of the YOLO model, and they advocate a more comprehensive optimization, i.e., to make it lightweight while at the same time completing the upgrade of the traditional weld defect detection algorithm.

As Liu et al. refined the LF-YOLO algorithm model based on CNN-based weld flaw detection techniques, among other things, their group discovered that the original CNN model's basic structure made it ineffective at integrating multi-scale information [17]. Numerous feature fusion techniques were put out to better leverage local and global contexts in order to overcome this problem. In order to facilitate both parameter-based and parameter-free multi-scale information extraction processes, they created an RMF module.

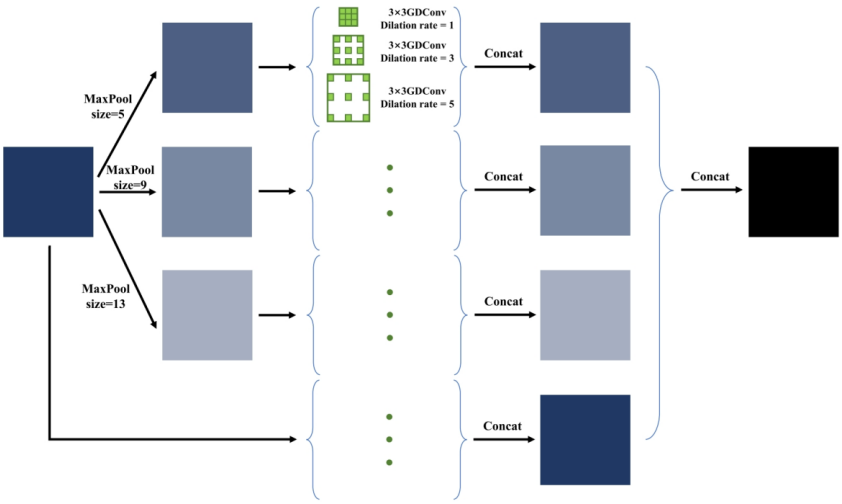

Fig. 11. Schematic Diagram of RMF Module [17].

As shown in the Figure 11, to improve the model's ability to represent complex features, the RMF module first performs parameter-free multiscale operation by building a multiscale basis using Spatial Pyramid Pooling (SPP) [18], then mines implicit information from various sensory fields using GDConv, and finally fuses the features of the two phases. This allows the extracted feature maps to represent richer information. Meanwhile, the Liu et al. team suggested an effective feature extraction (EFE) module to boost the detection network's performance. The EFE module is based on the "split-transform-merge" theory, which improves the ability to recognize complicated flaws by fusing features at multiple levels by dividing the feature map and keeping some of the channels to reuse the input features.

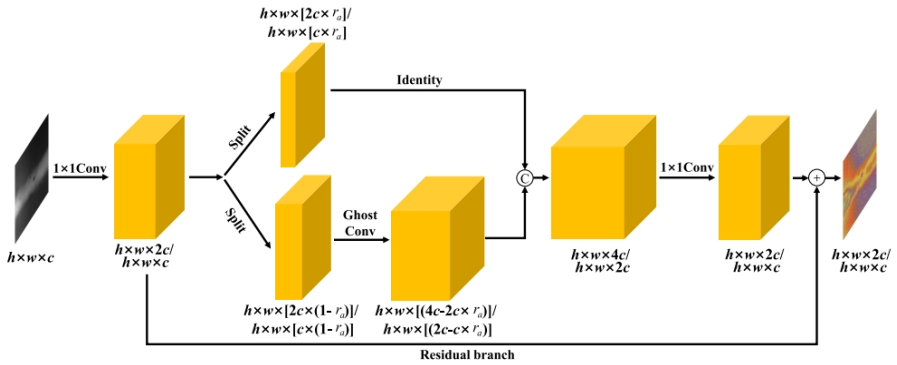

Fig. 12. Schematic Diagram of EFE Module [17].

The EFE module enables the model to be expanded with weld features in channel space to incorporate more implicit information, as seen in Figure 12 above. Ghost Conv is used to simplify modules, and the concept of "split-transform-merge" is added to reuse inputs.

5 Future Research Directions

Both YOLO-DEFW and LF-YOLO mentioned above have multiscale mechanisms, with YOLO-DEFW being very suitable for small samples and small features recognition with a single processing time of four milliseconds in real time. The LF-YOLO weld defect detection network strikes a satisfactory balance between performance and consumption, and achieves an average accuracy of 92.9 at 61.5 frames/second(mAP50). The YOLO algorithm-based weld defect detection system has demonstrated significant performance advantages, but the complex welding environments in industrial scenarios (e.g., small defects, dynamic lighting, multimodal interference, etc.) require continuous innovation in terms of algorithmic architecture and engineering optimization.

The future development direction of YOLO algorithm in welding defect detection needs to focus on the four core requirements of "accurate, real-time, lightweight, generalized", through multi-scale dynamic optimization, attention mechanism enhancement, cross-modal feature fusion and hardware co-design, to promote industrial inspection system to higher intelligence and practicality. With the deep integration of Transformer, NAS and other cutting-edge technologies, YOLO algorithm is expected to realize wider application in complex industrial scenarios.

6 Conclusions

This study examines recent advances in welding defect detection based on the YOLO algorithm. It begins with an overview of the historical development of CNNs and then analyzes the improvements from YOLOv1 to YOLOv11. The study critically evaluates

the persistent technical limitations inherent in standard YOLO implementations for metallurgical inspection scenarios. The evaluation concludes with an overview of five YOLO improvement models, including the noise-resistant YOLO-Weld variant, the computationally efficient and operationally simplified YOLO-DEFW, and the radiation imaging-optimized LF-YOLO configuration. To bridge the gap between current YOLO adaptations and real-world production requirements. Researchers may quickly grasp the current status of YOLO-based welding fault detection with this thorough review, which covers traditional approaches, improvement plans, related datasets, and evaluation measures. This work provides insights into future research areas by pointing out the shortcomings of previous studies, such as their reliance on extensive annotated datasets and their inability to generalize sufficiently under complicated operational situations. Developing multi-scale analysis, including few-shot learning paradigms, and improving real-time adaptation for industrial deployment are some possible directions. By bridging the gap between theoretical developments and real-world implementations, these initiatives hope to develop reliable and effective intelligent inspection systems for the welding sector.

Acknowledgments (Authors Contribution). All the authors contributed equally and their names were listed in alphabetical order.

References

1. Sheng Qian, Hua Liu, Cheng Liu, Si Wu, Hau San Wong, Adaptive activation functions in convolutional neural networks, Neurocomputing, **272**, pp.204-212, ISSN 0925-2312 (2018).
2. Vivek Singh Bawa, Vinay Kumar, Linearized sigmoidal activation: A novel activation function with tractable non-linear characteristics to boost representation capability, Expert Systems with Applications, **120**, pp. 346-356, ISSN 0957-4174 (2019).
3. Varshney, M., Singh, P. Optimizing nonlinear activation function for convolutional neural networks. SIViP **15**, 1323–1330 (2021).
4. Siyuan Zhang, Linbo Xie, Leader learning loss function in neural network classification, Neurocomputing, **557**, 126735, ISSN 0925-2312[5] YOLOv1 to YOLOv10: The fastest and most accurate real-time object detection systems (2023).
5. Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2017. ImageNet classification with deep convolutional neural networks. Commun. ACM **60**(6), pp.84–90 (2017).
6. Nunsong, W., & Woraratpanya, K. A ROBUST-TEXTURE CONVOLUTIONAL NEURAL NETWORK. Malaysian Journal of Computer Science, 157–171 (2019).
7. Karen Simonyan, Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition (2015).
8. Ding, Ing-Jr, Zheng, Nai-Wei, and Hsieh, Meng-Chuan. 'Hand Gesture Intention-based Identity Recognition Using Various Recognition Strategies Incorporated with VGG Convolution Neural Network-extracted Deep Learning Features'. 7775 – 7788 (2021).
9. Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich. Going deeper with convolutions. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), (2015).
10. R. Girshick, "Fast R-CNN," 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, pp. 1440-1448 (2015).

11. Liu, W., Hu, J., & Qi, J. Resistance Spot Welding Defect Detection Based on Visual Inspection: Improved Faster R-CNN Model. Machines, **13**(1), 33 (2025).
12. Yue Jianfeng, Li Weiming, Ning Lihua, et al. Research on real-time detection algorithm of weld defects based on YOLO-DEFW [J/OL]. China Laser, 1-22 [2025-02-09].
13. Long Ling, Chen Hao, Liang Hao, et al. Real-time X-ray weld defect detection based on lightweight YOLO network [J]. Network New Media Technology, **12**(02): 30-38 (2023).
14. Song, L., Kang, J., Zhang, Q. et al. A weld feature points detection method based on improved YOLO for welding robots in strong noise environment . SIViP **17**, 1801–1809 (2023).
15. Gao, A.; Fan, Z.; Li, A.; Le, Q.; Wu, D.; Du, F. YOLO-Weld: A Modified YOLOv5-Based Weld Feature Detection Network for Extreme Weld Noise. Sensors, **23**, 5640 (2023).
16. X.Wang, Y. Xuan, X. Huang and Q. Yan, "YOLO-AFK: Advanced Fine-Grained Object Detection for Complex Solder Joints Defect," in IEEE Access, **12**, pp. 166238-166252 (2024).
17. M.Liu, Y. Chen, J. Xie, L. He and Y. Zhang, "LF-YOLO: A Lighter and Faster YOLO for Weld Defect Detection of X-Ray Image," in IEEE Sensors Journal, **23**(7), pp. 7430-7439, 1, (2023).
18. A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal speed and accuracy of object detection,”, arXiv:2004.10934 (2020).