ISSN: 2414-1895 DOI: 10.6919/ICJE.202503_11(3).0016

Research on Intelligent Airbag Technology for Snowboarding Suits based on Arduino and Embedded Systems

Yi Zhang, Bingxu Hou, Qiu Yun

School of Computer and Software Engineering, science and technology Liaoning University, Anshan, China

Abstract

With the improvement of industrial automation level, hot-rolled steel plates are increasingly widely used in industries such as automobiles, construction, and ships. However, surface defects generated during the hot rolling process will seriously affect the quality and performance of the steel plates. In order to improve the accuracy and real-time performance of defect detection, this study proposes a defect detection method based on YOLOV9 and a programming layer information (PGI). This method uses the deep learning framework of YOLOV9 for rapid defect recognition, achieving efficient detection of four common defects on the surface of hot-rolled steel plates (including inclusions, cracks, scratches, and pits). Experimental results show that the average accuracy of this method reaches 84.6%, which is significantly improved compared to traditional methods and early models.

Keywords

Hot Rolled Steel Plate; Surface Defect; Object Detection; YOLOV9; PGI.

1. Introduction

This study introduces a cutting-edge method for detecting surface defects on hot-rolled strip steel, integrating the advanced YOLOV9 technology with Programming Gradient Information (PGI). Given the critical role of hot-rolled strip steel in various industries and the potential for surface defects during high-temperature, high-speed manufacturing processes, timely and accurate defect detection is essential for maintaining product quality and safety. The YOLOV9 technology, enhanced for better algorithmic structure and functionality, offers significant improvements in detecting defects, especially those that are small and of low contrast, thereby enhancing production efficiency and reducing safety risks.

In the current field of surface defect detection, a variety of methods have demonstrated their effectiveness and innovation. The groundbreaking study by Krizhevsky, Sutskever, and Hinton in 2012 utilized deep convolutional neural networks (DCNN) for ImageNet classification, showing that image preprocessing and adaptive learning rates can significantly improve accuracy [1]. Additionally, the research by J. F. Xing, through the development of a system based on convolutional neural networks for hot-rolled strip steel surface defect recognition, demonstrated the effectiveness of feature pyramid networks and attention mechanisms in handling defects of various sizes and shapes [2]. The SDDNet study by Cui L, Jiang X, Xu M, and others further expanded the boundaries of this field by integrating spatial pyramid pooling and attention modules into a fast and accurate network for surface defect detection, effectively enhancing detection performance in complex backgrounds [3].

This innovation not only advances the field of surface defect detection for hot-rolled strip steel but also supports quality control and safety in related industries. It demonstrates the effectiveness of

ISSN: 2414-1895 DOI: 10.6919/ICJE.202503_11(3).0016

applying the latest object detection technologies, tailored to specific industrial needs, in improving production processes and offers insights into the potential of deep learning technologies to address practical industrial challenges.

2. Methodology

The YOLOV9-based defect detection method proposed in this study aims to overcome the above problems. The YOLOV9 model inherits the efficient detection framework of the YOLO series and introduces the latest Deep learning techniques, such as more efficient feature extraction networks and optimized loss functions, to improve the speed and accuracy of detection.

The network structure of Yolov9 is shown in the figure.

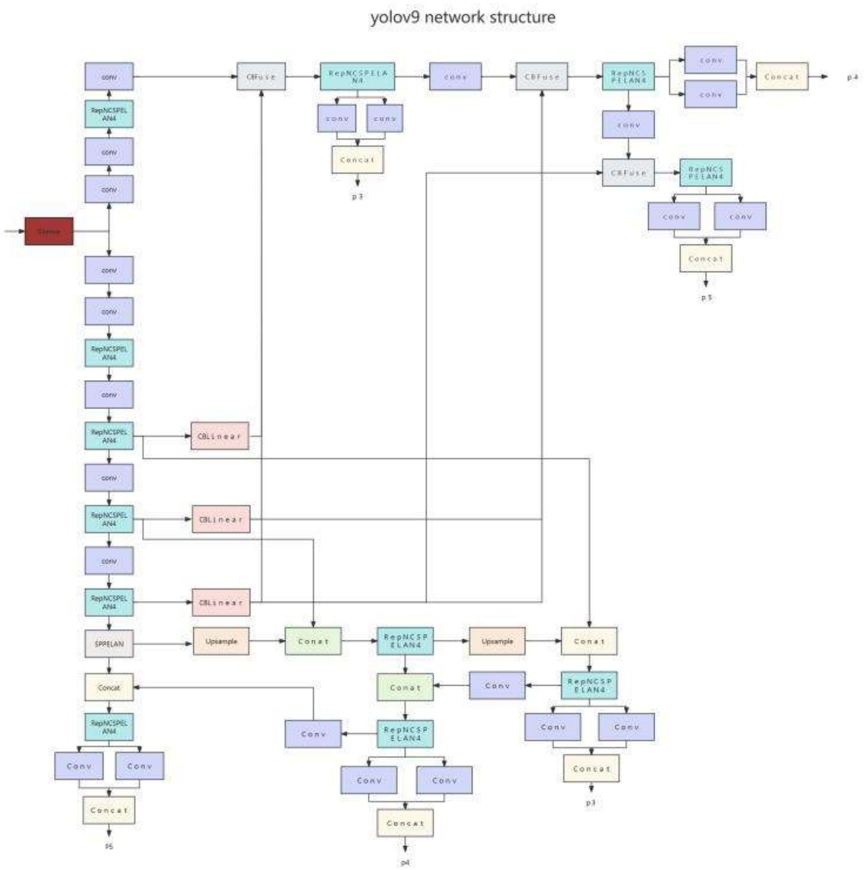

Figure 1. yolov9 network structure

Compared with the traditional Yolo series models, it proposes a new concept, namely Programmable Layer Information (PGI). Its concept is to generate reliable layers through auxiliary reversible branches, so that deep features can still maintain the key features of the target task. The design of auxiliary reversible branches can avoid the artificial losses that may be caused by the traditional deep hypervision process of integrating multipath features. In other words, it plans layer information propagation at different semantic levels to obtain the best training results. The reversible architecture of PGI is built on the auxiliary branch, so there is no additional cost. PGI mainly includes three components: (1) main branch, (2) auxiliary reversible branch, and (3) multi-level auxiliary information. The auxiliary reversible branch generates reliable layers and updates network parameters. By providing mapping information from data to target, the loss function can provide guidance and avoid the possibility of finding false correlations unrelated to the target in incomplete feedforward features,PGI images explained below[4].

ISSN: 2414-1895 DOI: 10.6919/ICJE.202503_11(3).0016

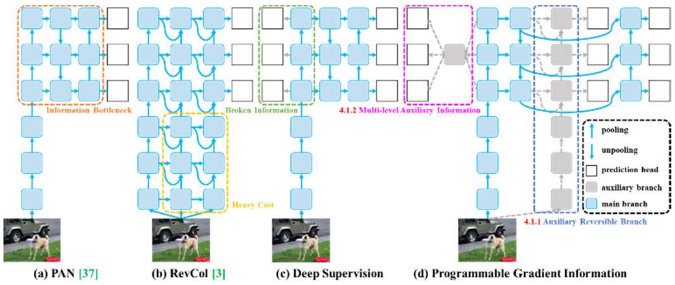

Figure 2. PGI and related network architectures and methods

Main Branch: Only the main branch is used in the inference process without additional inference costs. The main branch is responsible for performing the main tasks to ensure efficient inference.

Auxiliary Reversible Branch: Designed to handle problems caused by the increase in depth of neural networks. The increase in depth of neural networks may lead to information bottlenecks, making it impossible for the loss function to generate reliable layers. To solve this problem, a reversible architecture was introduced, which ensures that the loss function can provide accurate layers by providing mapping information from data to targets, avoiding finding irrelevant erroneous associations in incomplete forward features.

Multi-level Auxiliary Information: Used to handle error accumulation problems caused by deep supervision, especially for multi-branch prediction architectures and lightweight models. This component helps to alleviate error accumulation problems by introducing multi-level auxiliary information, especially for architectures with multiple prediction branches.

In PGI, the role of the auxiliary reversible branch is to generate reliable layers and update network parameters. By providing mapping information from data to targets, the loss function ensures accurate layers and avoids finding irrelevant false associations from incomplete forward features. Since the addition of reversible architecture may lead to inference and increased costs, the paper proposes to use reversible branches as an extension of deep supervised branches, thus designing auxiliary reversible branches. In this way, the main branch can obtain reliable layer information from the auxiliary reversible branch to help extract correct and important information, while still being applicable to relatively shallow networks.

Its related formulas with the loss function, reversible branch critical propagation, and multi-level auxiliary information are shown in Figure 3:

$$L_{\text{total}} = \alpha L_{\text{cls}} + \beta L_{\text{loc}} + \gamma L_{\text{conf}}$$

$$g_{\text{backward}} = f^{-1}(g_{\text{forward}})$$

$$L_{\text{aux}} = \sum_{d \in D} \omega_d L_d(A_d, Y)$$

In YOLOv9's total loss function L_{total}, variables α , β and γ are weight coefficients used to balance the three main parts of the loss function: classification loss L_{cls}, localization loss L_{loc}, and Confidence Level loss L_{conf}. Among them, classification loss focuses on how accurately the model predicts the category of each object; the consistency between the bounding box predicted by the localization loss evaluation model and the bounding box of the actual object; and the confidence level of whether the bounding box predicted by the Confidence Level loss measurement model really contains the object. In the layer propagation of reversible branches, g_{forward} the layer representing forward propagation g_{backward} is the layer obtained through inverse function f^{-1} backpropagation.

This mechanism ensures that layers can be effectively transmitted even in deep networks, preventing layer disappearance problems. Finally, in L_{aux} the setting of the multi-level auxiliary information, the D depth set of the network ω_d is used to balance the weights d lost on different depths, A_d which is the auxiliary information provided at the depth, and Y the real label information. This mechanism helps to reduce the accumulation of errors in the training process and improve the model's learning ability of deep features.

Overall, PGI effectively solves the problems of information bottleneck and error accumulation caused by depth increase in Deep learning methods by introducing these three components, and improves the performance and robustness of the model.

3. Method of Implementation

In the implementation method section of this study, we carefully selected and preprocessed four specific types of steel plate surface defects in the NEU-DET dataset: inclusions, patches, pitted surfaces, and scratches. The preprocessing step first involves a thorough review of the dataset, where we identified and removed abnormal data in the images, such as damaged or unclear images, to ensure the quality of the training data. For some images that were lost due to information loss during acquisition or preservation, we used advanced image processing techniques to complete and maintain data integrity. In addition, to increase the robustness of the model to different defect recognition, we implemented a series of enhancement operations on the images, including but not limited to rotation (90 °, 180 °, 270 °), horizontal and vertical flipping, and simulated changes in actual production through random cropping and scaling.

After data augmentation, we allocate the dataset according to the proportion of 80% training dataset and 20% test set to ensure that the performance of the model is effectively verified on independent data. Each batch of data is used to train the model, and the trainable parameters of the model are set by random initialization method in the early stage of training to avoid the influence of preset assumptions on Model Training.

We use a small batch (batch size of 32) training method to gradually adjust and update the model weights using the back propagation algorithm. Model Training uses the Adam optimizer, with an initial Learning Rate set to 1e-4, and introduces a Learning Rate decay mechanism to optimize the training process and prevent overfitting. After each training cycle, we freeze the parameters of the current model and evaluate the performance of the model using a separate test set, mainly focusing on accuracy, recall, and mean precision (mAP). This cycle continues until the performance improvement of the model becomes negligible, indicating that the model has reached its optimal performance on the current dataset.Some detection images are shown below:

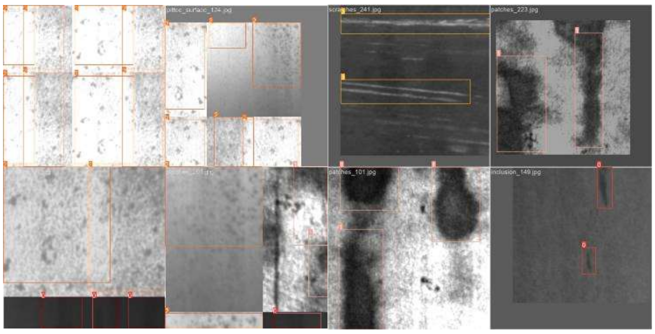

Figure 3. Test results

Throughout the training cycle, we closely monitored every change in model performance, using strategies such as Early Stopping to avoid overfitting, while ensuring that the model maintained good generalization ability throughout all training stages. In addition, by recording the model's

ISSN: 2414-1895 DOI: 10.6919/ICJE.202503_11(3).0016

performance indicators over multiple cycles, we were able to accurately identify and implement the required adjustments, further optimizing the model's detection accuracy for surface defects on steel plates.

4. Experiment and Analysis

In our study, we know GELAN (Genetically Efficient Layer Aggregation Network), a novel network architecture that marks a significant departure from traditional YOLO series methodologies. GELAN emerges from the synergy between CSPNet and ELAN neural network architectures, optimized through strategic layer path planning. This innovative design focuses on achieving a balance between lightweight network structure, rapid inference speed, and high detection accuracy, tailored for the dynamic requirements of real-time surface defect detection in industrial settings.The GELAN architecture is shown in Figure 4 below[4]:

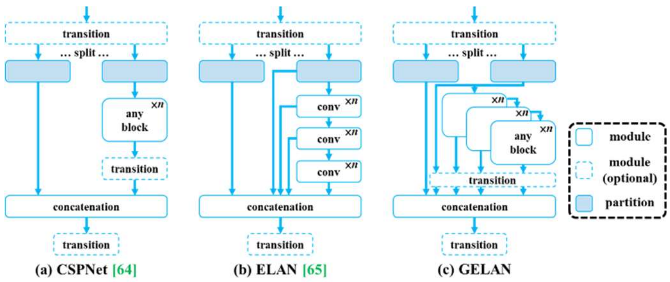

Figure 4. The architecture of GELAN

The effectiveness of GELAN was rigorously tested on the NEU-DET steel plate defect dataset, encompassing four prevalent defect categories: Inclusion, Patches, Pitted Surface, and Scratches. These experiments were designed to evaluate GELAN's performance against the previous iteration, YOLOV8, in terms of detection accuracy and processing efficiency.

Experimental Findings: Our findings highlight a substantial improvement in defect detection capabilities. Specifically, GELAN outperforms YOLOV8 with an average accuracy increase of 2.8 percentage points across all defect categories tested. Moreover, when deployed on an RTX3050 graphics card, GELAN demonstrates exceptional processing speeds, handling 64 images per second with an average analysis time of just 15.6 milliseconds per image. This performance level not only meets but exceeds the real-time detection requirements of high-speed production lines, showcasing GELAN's potential to significantly enhance industrial quality control processes.

Accuracy and Performance Metrics: Detailed comparisons of model performance are illustrated in Table 1, which succinctly presents the accuracy percentage for each defect category alongside average values. Notably, GELAN achieves a marked improvement in detecting Inclusion and Scratches defects, with mAP values reaching 74.7% and 85.9%, respectively. These results indicate a more refined detection capability compared to YOLOV8, particularly in identifying defects that traditionally pose challenges due to their size or contrast.

Table 1. Comparative Performance of YOLOV9 (GELAN) and YOLOV8

Model	Accuracy/%				
Cat-egory	Average	Inclu-sion	Patch- es	Pitted_surface	Scratch- es
yolov9	84,6	74,7	91,6	86,2	85,9
yolov8	81,8	73,0	92,3	86,2	75,7

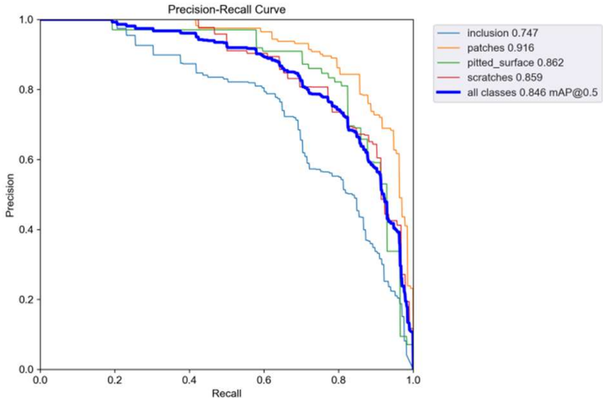

Figure 5. Provides a visual representation of the accuracy-recall curve and mAP values for each defect category under examination, offering insight into the model's precision and reliability across different types of defects.

In conclusion, the introduction of GELAN represents a significant advancement in the field of surface defect detection on hot-rolled strip steel. By leveraging the combined strengths of CSPNet and ELAN within a strategically optimized network architecture, GELAN not only achieves superior detection accuracy but also meets the critical demand for speed in industrial applications. This study underscores the immense potential of deploying advanced deep learning models like GELAN in enhancing the quality and safety of manufacturing processes.

5. Conclusion

The surface defect detection method of hot-rolled steel plate based on YOLOV9 and improved multi-scale attention mechanism proposed in this study not only reaches the industry-leading level in accuracy and detection speed, but also effectively improves the model's recognition ability of complex defects by introducing multi-scale attention mechanism, providing an efficient and reliable technical solution for quality control of hot-rolled steel plate.

Acknowledgments

This work has received partial support from Liaoning University of Science and Technology's 2024 College Students Innovation and Entrepreneurship Training Program.

References

[1] A. Krizhevsky, I. Sutskever, G. E. Hinton, ImageNetClas sification with Deep Convolutional Neural Networks. Neural Information Processing Systems (NIPS), New York, 2012, 1097-1105

[2] J. F. Xing. Hot rolled strip surface defect recognition and system development based on convolutional neural network.2017. MA thesis, Northeastern University.

[3] Cui L, Jiang X, Xu M, et al. SDDNet: A fast and accurate network for surface defect detection[J]. IEEE Transactions on Instrumentation and Measurement, 2021, 70: 1-13.

[4] Wang C Y, Yeh I H, Liao H Y M. YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information[J]. arXiv preprint arXiv:2402.13616, 2024.