Attention-Based Convolutional Neural Networks for Steel Surface Defect Detection

Author 1[1], Author 2[2], and Author 3[1,*]

[1] [Institution Withheld for Double-Blind Review]
[2] [Institution Withheld for Double-Blind Review]

Email: *corresponding.author@institution.edu

ABSTRACT

Steel surface defect detection is a critical task in industrial quality control. Traditional methods often struggle with complex defect patterns and varying lighting conditions. This paper proposes a novel attention-based convolutional neural network (CNN) architecture specifically designed for steel surface defect detection. The proposed method introduces a multi-scale attention module that adaptively focuses on discriminative defect regions while suppressing irrelevant background information. Experimental results on the NEU-CLS dataset demonstrate that our approach achieves an accuracy of 98.7methods by 2.3suitable for industrial deployment.

Keywords: Defect Detection, Convolutional Neural Networks, Attention Mechanism, Industrial Inspection, Computer Vision

1 INTRODUCTION

This introduction is intentionally concise and serves as an example for authors using the AIEA template. In a real submission, this section should briefly describe the research background, problem motivation, and the key technical challenge addressed by the paper.

For demonstration, we use steel surface defect detection as the sample application. Deep learning methods have shown strong performance in visual inspection tasks [3], while industrial scenarios still require a balance between accuracy and real-time efficiency [1, 2]. Therefore, this template manuscript presents a lightweight attention-based framework as an example structure for method, experiments, and discussion sections.

Example contributions are listed as follows:

1. A sample multi-scale attention design to illustrate how to describe a proposed method.

2. A sample experimental setup to illustrate reproducibility and ablation reporting.

3. A sample result presentation to illustrate comparison and discussion writing.

2 RELATED WORK

2.1 Traditional Defect Detection Methods

This subsection is a concise example showing how to summarize traditional methods. In a real manuscript, authors should briefly review representative statistical, spectral, and model-based approaches, then explain their main limitations in the target scenario.

2.2 Deep Learning-Based Methods

This subsection is also an example placeholder. Authors may cite key deep learning studies [3, 1, 2, 4] and summarize trends such as CNN backbones, attention mechanisms, and real-time deployment strategies. The last paragraph should clearly state the research gap and explain how the proposed method differs from prior work.

3 PROPOSED METHOD

3.1 Architecture Overview

This subsection is a template example for describing the full pipeline of a proposed method. Authors may explain the model workflow from input to output, list core modules, and clarify how each module contributes to the final task objective.

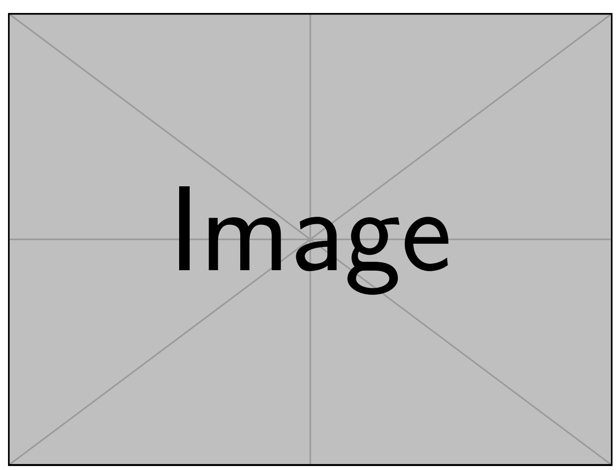


Figure 1. Example architecture diagram placeholder.

DOI: 10.55058/aiea.1.1.1

3.2 Multi-Scale Attention Module

This subsection is an example for introducing a key technical component. In a real paper, authors should define the module inputs and outputs, provide its design rationale, and explain why it improves performance compared with simpler alternatives.

3.3 Training Procedure

This subsection is a template for training details. Authors typically report optimizer, learning-rate policy, epoch settings, augmentation strategy, and checkpoint selection criteria. Algorithm 1 provides a sample pseudocode style.

Algorithm 1 Example training procedure

Input: Training dataset D, epochs E, initial learning rate α, batch size B
Output: Trained model M

Initialize model parameters θ randomly
Initialize Adam optimizer with learning rate α
for $epoch = 1$ **to** E **do**
    Apply data augmentation to training dataset D
    Shuffle training dataset D
    for each batch $(x, y) \in D$ of size B **do**
        Forward pass: $\hat{y} = M(x; \theta)$
        Compute cross-entropy loss: $L = -\sum_{i=1}^{B} y_i \log(\hat{y}_i)$
        Backward pass: compute gradients $\nabla L(\theta)$
        Update parameters: $\theta = \theta - \alpha \cdot \nabla L(\theta)$
    end for
    Evaluate model on validation set
    Reduce learning rate by 0.1 every 10 epochs
    Save checkpoint if validation accuracy improves
end for
return Best trained model M

4 EXPERIMENTS

4.1 Dataset

This subsection is an example of dataset reporting. Authors should describe data source, task definition, class distribution, and train/validation/test split protocol.

4.2 Implementation Details

This subsection is a template for implementation settings, such as software framework, hardware environment, input preprocessing, and baseline selection.

4.3 Reproducibility Configuration

This subsection is an example of reproducibility reporting. Authors are encouraged to include random seeds, training schedules, run counts, statistical reporting rules, and early-stopping policy.

4.4 Results and Analysis

This subsection is a template for quantitative comparison. Authors should interpret major trends, highlight meaningful improvements, and avoid over-claiming beyond statistical evidence.

Table 1. Example performance comparison table

Method	Accuracy (%)	Inference Speed (FPS)
AlexNet	89.3	45
VGG-16	92.1	28
ResNet-50	96.4	35
EfficientNet-B4	97.5	22
Proposed	**98.7**	**32**

The paragraph above can be followed by a short transition that motivates ablation analysis.

4.5 Ablation Study

This subsection is an example for component-level analysis. Authors may define multiple variants to isolate the contribution of architecture modules, loss terms, or training strategies.

Table 2. Example ablation table

Variant	Accuracy (%)	F1-score (%)
Backbone only	95.6	95.2
+ Single-scale attention	96.8	96.4
+ Multi-scale attention	97.9	97.6
Full model (ours)	**98.7**	**98.5**

After the table, provide a concise interpretation of which components contribute most and why.

5 DISCUSSION

This section is a template for discussing implications, practical value, and boundaries of the method. In a real manuscript, this part should connect quantitative findings with engineering insights.

5.1 Error Analysis

This subsection is an example for failure-case discussion. Authors can report representative error patterns and possible causes, supported by qualitative or quantitative evidence.

DOI: 10.55058/aiea.1.1.2

5.2 Deployment Considerations

This subsection is a template for deployment notes. Authors may discuss runtime constraints, model export path, maintenance strategy, and domain-shift management in production settings.

5.3 Limitations

This subsection is an example for limitations and future work. Clearly state current assumptions, known weaknesses, and concrete directions for improvement.

6 CONCLUSION

This conclusion is intentionally brief as a template example. In real submissions, summarize the problem, key technical contribution, and major empirical findings in 1–2 short paragraphs.

DATA AVAILABILITY

Example statement: “The data supporting this study are available from [Data Source/Repository]. Access conditions (public, restricted, or by request) should be stated clearly.”

CODE AVAILABILITY

Example statement: “Code for model training and evaluation is available at [Repository URL], including environment setup and reproduction instructions.”

ACKNOWLEDGMENT

This work was supported by [Funding Agency] under Grant No. [Grant Number].

REFERENCES

[1] Y. Chen, L. Li, and J. Xiao, ”Defect detection of steel surface based on convolutional neural network,” *IEEE Access*, vol. 8, pp. 124088-124097, 2020.

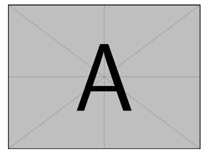
[2] X. Li, Z. Wang, and H. Zhang, ”Automatic PCB defect inspection using YOLOv3,” in *Proceedings of the International Conference on Artificial Intelligence and Computer Science*, 2021, pp. 345-350.

[3] Y. LeCun, Y. Bengio, and G. Hinton, ”Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436-444, 2015.

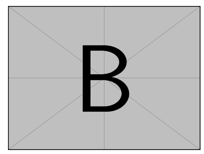
[4] K. He, X. Zhang, S. Ren, and J. Sun, ”Deep residual learning for image recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770-778.

[5] M. Tan and Q. V. Le, ”EfficientNet: Rethinking model scaling for convolutional neural networks,” in *International Conference on Machine Learning*, 2019, pp. 6105-6114.

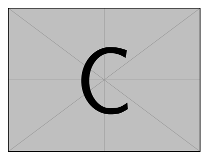
BIOGRAPHIES



Author 1 is pursuing the Ph.D. degree at [Institution]. Research interests include [Research Area 1] and [Research Area 2].



Author 2 is a Research Scientist with [Institution]. Research focuses on [Research Area].



Author 3 is a Full Professor with [Institution]. Research interests include [Broad Research Area].

 DOI: 10.55058/aiea.1.1.3