LE-YOLO: A Lightweight and Enhanced Algorithm for Detecting Surface Defects on Particleboard

Chao He, Yongkang Kang, Anning Ding, Wei Jia, and Huaqiong Duo *

Current algorithms for surface defect detection in particleboard often encounter limitations such as high computational complexity and excessive parameter scale. To address these challenges, this study proposes the LE-YOLO model, which incorporates a normalized Wasserstein distance into the loss function to enhance the detection capability for minute surface defects. A dynamic mixed convolutional network module is introduced to construct a lightweight backbone architecture. Moreover, the Shared Dilated Feature Pyramid (SDFP) module is employed in the neck network, effectively reducing computational overhead while preserving detection accuracy. A lightweight detection head was further designed, integrating shared convolutional operations with a distribution-aware loss function, thereby substantially improving detection performance in complex textured environments. Experimental evaluations conducted on the Chipboardv1.0 particleboard surface defect dataset demonstrated that compared to the baseline YOLOv11n model, LE-YOLO achieved a 5% improvement in recall, a 1% increase in F1 score, a 4% enhancement in mAP@50, a 6% gain in mAP@50–95, a 12.69% acceleration in inference speed, and an 18.6% reduction in parameter count. Compared with other models, the proposed approach not only improved detection precision but also effectively reduced model complexity, achieving a lightweight and efficient detection framework.

DOI: 10.15376/biores.20.3.7179-7193

Keywords: Object detection; Lightweight architecture; YOLO; Particleboard surface defects; Deep learning; Feature fusion

Contact information: College of Materials Science and Art Design, Inner Mongolia Agricultural University, Hohhot 010018, Inner Mongolia Autonomous Region, China;
** Corresponding author: duohuaqiong@163.com*

INTRODUCTION

Traditional wood surface defect detection methods have predominantly relied on handcrafted feature extraction algorithms. For example, Ji *et al.* (2019) introduced a wavelet moment-based feature extraction algorithm that combines the advantages of wavelet energy and Hu invariant moments to classify and identify wood defect images. Compared to conventional Hu moments, their method significantly improved recognition accuracy. However, this approach suffers from high computational complexity, resulting in inefficiency when processing high-resolution or large-scale data and requiring substantial hardware resources. Additionally, parameter tuning is challenging, and the algorithm is vulnerable to noise interference, which compromises real-time application performance.

Pan *et al.* (2022) employed near-infrared spectroscopy combined with Extreme Learning Machine (ELM) feature extraction and a Whale Optimization Algorithm-based Support Vector Machine (WOA-SVM) to preprocess spectral data and identify wood regions. Experimental results demonstrated that the ELM-optimized approach effectively enhanced region recognition. In another study, Li *et al.* (2025) introduced a supervised learning-based image super-resolution method using Discrete Wavelet Transform (DWT) within a U-Net architecture, incorporating DWT sampling and channel attention residual modules. Ablation studies and comparative experiments validated the effectiveness of each component, showing superior performance in PSNR and SSIM metrics. While these traditional methods can improve detection accuracy and reduce noise interference to some extent, issues, such as high computational cost, complex parameter optimization, and limited real-time applicability, remain unresolved.

In contrast, deep learning methods offer the advantage of automatic feature extraction and end-to-end learning, enabling greater adaptability to complex data. Compared to traditional algorithms, deep learning-based approaches successfully overcome the aforementioned limitations. Contemporary object detection networks are generally categorized into two types: two-stage and one-stage detectors. In the two-stage detection domain, notable contributions include Girshick *et al.* (2014), who combined the region proposal algorithm Selective Search (Uijlings *et al.* 2013) with convolutional neural networks (CNNs) to develop the R-CNN model, achieving significant improvements in detection accuracy. Subsequently, Girshick *et al.* (2015) introduced the Fast R-CNN model based on the Spatial Pyramid Pooling Network (SPPnet), which further accelerated detection speed while maintaining high accuracy. More recently, Zou *et al.* (2025) enhanced the Faster R-CNN framework by integrating an improved ResNet-50 backbone, a focal loss function, and soft Non-Maximum Suppression (soft-NMS). Their model achieved a 6.76% improvement in mean Average Precision (mAP), reaching 67.80%, while also reducing detection time by 3.6%, thereby improving detection performance in wood surface defect scenarios. Typical one-stage detectors include SSD (Single Shot MultiBox Detector) and the YOLO (You Only Look Once) series. Ding *et al.* (2020) incorporated DenseNet into the SSD framework to improve deep feature extraction and multi-layer feature map fusion in wood imagery, yielding superior performance over the traditional SSD and meeting real-time demands of industrial wood processing. The evolution of YOLO from YOLOv1 to YOLOv3 has driven the development of end-to-end optimized detectors that perform inference in a single pass. Meng and Yuan (2023) proposed SGN-YOLO, an improved YOLOv5-based model incorporating a Semi-Global Network (SGN), an enhanced E-ELAN module, and an EIOU loss function. On a public wood defect dataset, the model achieved an mAP of 86.4%, representing a 3.1% improvement over the baseline, with ablation experiments validating the contribution of each enhancement. Nevertheless, challenges remain in detecting small defects and ensuring robust dataset generalization.

Wang *et al.* (2024) further improved the YOLOv7 architecture by replacing standard convolution in the ELAN module with Partial Convolution (PConv), forming the P-ELAN module. This modification reduced computational redundancy and memory usage while enhancing detection accuracy.

Despite the progress of deep learning in object detection, real-world applications in wood processing environments still face challenges due to lighting variations, dust interference, and the complexity of textures. Issues persist, such as missed detections and large model sizes. Consequently, there remains considerable room for improving the accuracy and efficiency of particleboard surface defect detection.

To address these challenges, this study proposes a lightweight and enhanced detection model based on YOLOv11 (Khanam and Hussain 2024), named LE-YOLO. Rather than merely aggregating existing architectural components, the model incorporates several targeted innovations. The primary contributions of this work are as follows:

1. Design of the Adaptive Multi-Kernel Depthwise Conv2d (AMDC) module. Integrated into the backbone network as a replacement for the C3K2 module in YOLOv11, AMDC leverages multiple shapes of 2D depthwise separable convolution kernels to adaptively extract features. This design reduces the parameter count while preserving model expressiveness.
2. Development of the Shared Dilated Feature Pyramid (SDFP) module. By concatenating feature maps derived from varying dilation rates with the original convolutional output along the channel dimension, the SDFP module facilitates multi-scale object recognition and improves detection performance across defect sizes.
3. Proposal of the Lightweight Detection Head (LWDetHead). This detection head utilizes shared convolutions and multi-scale feature fusion to enhance fine-grained representation. Additionally, it incorporates a distributed focal loss mechanism, enabling effective detection with minimal parameter overhead and computational cost.
4. Incorporation of the Normalized Wasserstein Distance (NWD) in the loss function. As introduced by Wang *et al.* (2022), NWD effectively measures distributional similarity with reduced sensitivity to object scale, making it particularly suitable for small object detection tasks, such as tiny surface defects.

EXPERIMENTAL

Wood Defects Dataset

The Chipboardv1.0 dataset, collected from Shandong Luli Wood Industry Co., Ltd.'s automated production line, contains three common particleboard surface defects: large shavings, sand leakage, and black spots, with a balanced class distribution of 33% black spots, 33% large shavings, and 34% sand leakage. To enhance model robustness and generalization, data augmentation techniques such as horizontal scaling and Gaussian noise were applied, improving adaptability to complex textures and varying lighting. All images were standardized to 640×640 pixels and captured under different lighting conditions, including normal, reflective, and low-light environments. A stratified sampling strategy was used to ensure class balance, with 3,000 images selected—2,400 for training, 300 for validation, and 300 for testing (8:1:1 split). Figure 1 shows representative examples of each defect type.

Fig. 1. Representative examples of the three surface defect types in particleboard

LE-YOLO

YOLOv11, developed by Ultralytics and released in late 2024, represents a next-generation object detection algorithm that builds on the strengths of its predecessors. It introduces notable advancements in both network architecture and training methodologies, thereby greatly enhancing feature extraction capabilities. These improvements enable more accurate detection of complex features, even under challenging environmental and textural conditions. To overcome the limitations of existing particleboard surface defect detection algorithms—specifically, high computational complexity and large model size—this study proposes a lightweight and enhanced detection model based on the YOLOv11 framework, referred to as LE-YOLO. The overall architecture of the proposed model is illustrated in Fig. 2. This section outlines the core components of LE-YOLO, including the Adaptive Multi-Kernel Depthwise Conv2d (AMDC) module, the multi-scale feature fusion module (Shared Dilated Feature Pyramid, SDFP), the LWDetHead, and the Integration of the NWD in the loss function.

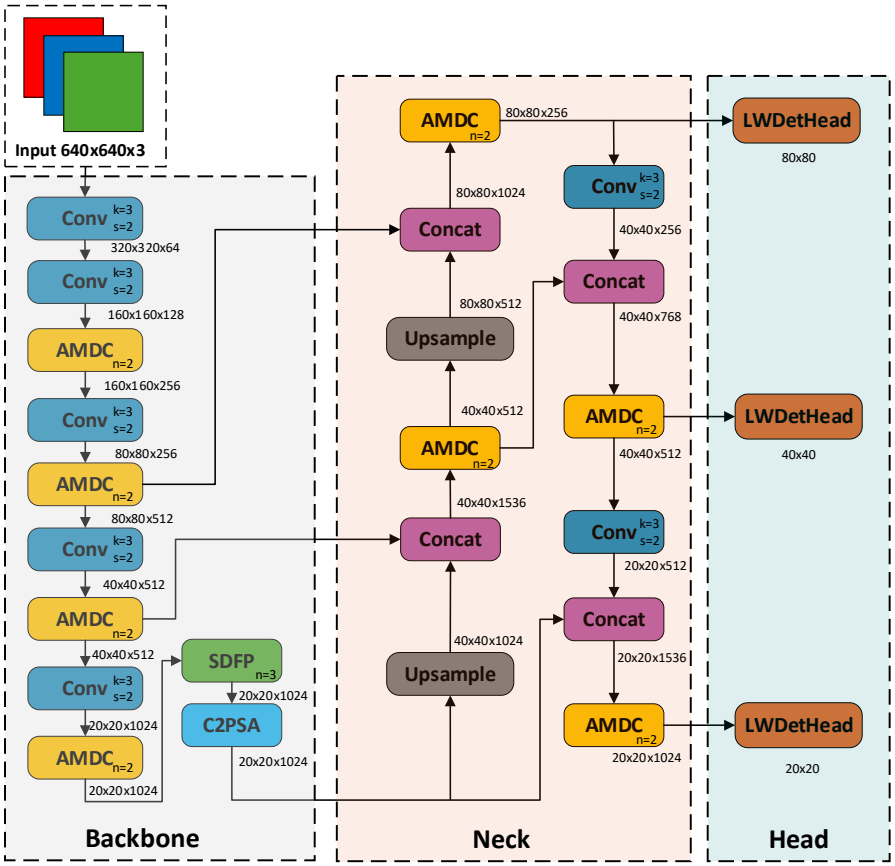

Fig. 2. LE-YOLO structure diagram The k and s in Conv blocks represent the kernel size and stride size. The n in AMDC and SDFP represents the number of Bottlenecks. 640×640×3 refers to the size of the input image, and subsequent numbers located below each block represent the dimension of feature maps.

Adaptive Multi-Kernel Depthwise Conv2d

The original feature extraction module C3K2 in YOLOv11 employs a multi-branch and multi-layer stacking strategy based on the C3k structure. While this design improves feature representation, it also introduces a large number of parameters and increases computational complexity, resulting in reduced inference speed. In addition, the reliance

on fixed-size convolution kernels limits the ability to capture features across diverse spatial scales, affecting the model's adaptability and robustness in complex scenarios.

To address the aforementioned limitations, this study proposes the Adaptive Multi-Kernel Depthwise Conv2d (AMDC) module (refer to Fig. 3 for the AMDC structure diagram). The AMDC module integrates dynamic kernel selection, cross-layer information fusion, and a computationally efficient structure, effectively enhancing multi-scale feature extraction and improving the inference efficiency of the model. The AMDC module processes the input through three parallel convolutional paths, each with a distinct kernel configuration to extract complementary features. The outputs of these paths are transformed *via* a shared weight matrix (W). Meanwhile, the original input feature map undergoes adaptive average pooling to generate a low-dimensional context vector. This vector is then passed through a 1 × 1 convolution and a SoftMax activation to produce a set of probability weights, which are subsequently applied to the corresponding outputs from the three convolutional branches. The weighted features are finally aggregated through summation, resulting in an output feature map with dimensions of C × H × W. Compared to the original module in YOLOv11, the AMDC offers enhanced flexibility and adaptability, while substantially improving computational efficiency. These improvements make it more suitable for real-time surface defect detection tasks under industrial conditions.

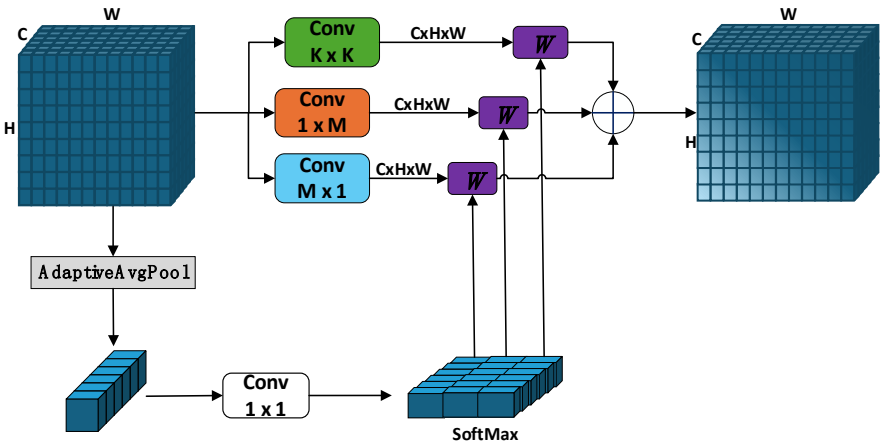

Fig. 3. AMDC structure diagram

Shared dilated feature pyramid

The Spatial Pyramid Pooling-Fast (SPPF) module uses fixed-size max pooling kernels to extract and fuse multi-scale contextual information. Although effective in aggregating features from various receptive fields, its dependence on fixed pooling sizes limits its adaptability to objects with significant scale variation. This constraint can reduce detection accuracy, particularly in complex scenes where object sizes vary widely.

To overcome this limitation, this study proposes the Shared Dilated Feature Pyramid (SDPF) module, designed to enhance multi-scale feature representation while maintaining computational efficiency. Specifically, the input feature map first passes through a 1×1 convolution to reduce the number of channels by half, thereby lowering computational cost. Next, three parallel max pooling operations with different kernel sizes are applied to produce feature maps at multiple scales. These outputs are concatenated with the original feature map along the channel dimension, followed by another 1×1 convolution to restore the desired output channel dimensions. Despite the advantages of max pooling

in enlarging the receptive field, it inherently introduces downsampling, which can result in the loss of fine-grained spatial information. Moreover, the repeated pooling operations impose additional computational burden, particularly in resource-constrained deployment environments. The use of fixed-size pooling kernels further limits the ability to adaptively capture features across scales. In contrast, the proposed SDPF module utilizes shared convolutional layers to minimize memory usage and computational overhead. Through incorporating dilated convolutions with varying dilation rates—smaller rates for capturing local structural details and larger rates for encoding broader contextual semantics—the module achieves more flexible and effective multi-scale feature extraction. Unlike max pooling, this convolution-based approach preserves spatial resolution and allows the network to retain fine-grained features, thereby enhancing detection performance across objects of different sizes. Figure 4 illustrates the architectural differences between the SPPF and SDPF modules, highlighting the underlying design principles and improvements introduced in SDPF.

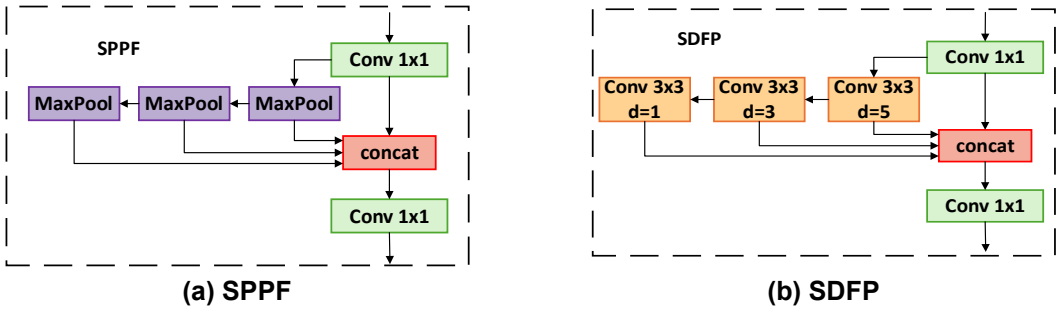

Fig. 4. (a) SPPF and (b) SDFP structure comparison diagram

Lightweight detection head

YOLOv11's detection head utilizes a dual-label assignment strategy, combining one-to-many assignments during training and one-to-one assignment during inference. This approach improves detection performance but introduces added computational complexity, particularly during the training and inference stages. The increased computational load may hinder real-time deployment in resource-constrained environments. To address this, a novel Lightweight Detection Head (LWDetHead) is proposed, designed to balance performance and efficiency for wood surface defect detection. As shown in Fig. 5, the LWDetHead architecture integrates multiple convolutional modules and scale layers to optimize both feature characterization and computational efficiency. A key innovation of LWDetHead is the Detail-Enhanced Convolution (DEConv), which improves feature representation by incorporating prior knowledge into standard convolution operations. During inference, DEConv is reparameterized into a standard convolution, avoiding additional parameters or computational overhead and ensuring compatibility with lightweight deployment scenarios. To further enhance localization and classification, the architecture includes a normalized convolutional layer (Conv_GN) with group normalization. This stabilizes training and reduces the number of learnable parameters. LWDetHead also addresses scale variation by employing shared convolutional layers (Conv_Reg) alongside scale layers. These components adaptively adjust feature map scales to enhance robustness against object size differences. The Conv_Reg layer improves regression tasks, while the dedicated Conv_Cls layer focuses on classification, ensuring consistent object identification across

feature levels. Through combining shared convolutions, scale-aware processing, and task-specific modules, LWDetHead reduces parameter count and computational complexity with minimal impact on detection performance, making it suitable for real-time deployment in resource-constrained environments.

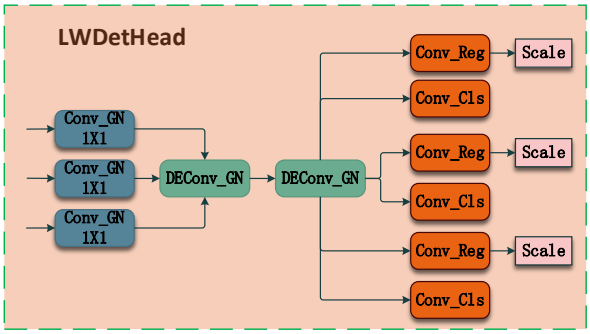

Fig. 5. LWDetHead structure diagram

NWD Loss

In the detection of wood surface defects, particularly for particleboard, the identification of minor defect categories often presents significant challenges due to their low pixel occupancy and high intra-class variability. These small defects typically occupy only a tiny fraction of the image, making them more susceptible to background noise and difficult to distinguish from one another. The CIOU (Complete-IoU) loss function used in YOLO-based algorithms has notable limitations when dealing with such small targets. Specifically, the aspect ratio of small objects is highly susceptible to image noise interference, which may cause the CIOU loss function to wrongly penalize otherwise reasonable predictions. Additionally, CIOU's reliance on a rectangular assumption, its sensitivity to scale variations, and its dependence on specific parameters can substantially constrain its performance in complex scenarios. To address these issues, especially the challenges associated with low-pixel-ratio defects and high intra-class divergence, this study incorporates the Normalized Wasserstein Distance (NWD) into the loss function, aiming to enhance overall model performance by reinforcing feature distribution alignment. This alignment helps to reduce regression deviation caused by inconsistent defect patterns, thereby improving the localization accuracy of small targets. To balance the contributions of the CIoU loss and NWD loss, a scale factor μ is introduced. The revised loss function is shown in Eq. 1, where loss_iou denotes the IoU-based loss, and nwd_loss represents the NWD-based loss:

$$\text{loss\_iou} = \mu \times \text{loss\_iou} + (1 - \mu) \times \text{nwd\_loss} \quad (1)$$

The formula for NWD is provided in Eq. 2,

$$L_{NWD} = \frac{W(P, Q)}{\text{Normalization Factor}} \quad (2)$$

where $W(P, Q)$ denotes the original Wasserstein distance between the predicted and ground truth boxes (as illustrated in Fig. 6), and the denominator is a normalization term to ensure scale invariance.

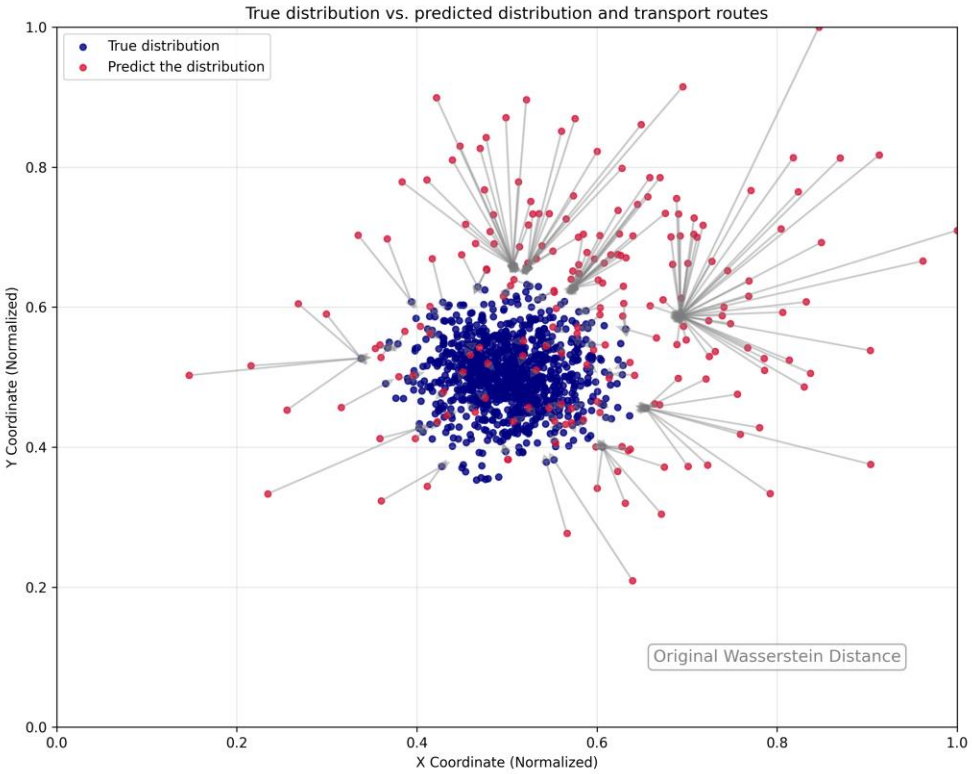

Fig. 6. Schematic diagram of Wasserstein distance

To determine the optimal value of μ, experiments were conducted by varying the scale factor in increments of 0.1 over the range [0, 0.9], resulting in ten different configurations. The detection results of the LE-YOLO model under each configuration are summarized in Table 1, with the best performance highlighted in bold. The results show that when μ = 0.5, the model achieves the highest detection accuracy, with an mAP@50 of 90% and an mAP@50:95 of 55%.

Table 1. Performance Comparison Across Scaling Factors μ

μ	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
mAP@50	0.86	0.85	0.85	0.87	0.87	**0.90**	**0.90**	0.85	0.87	0.87
mAP@50:95	0.50	0.48	0.49	0.50	0.51	**0.55**	0.54	0.51	0.53	0.49

Experimental Details

The experiments reported in this study were conducted on a Windows 10 system equipped with a 12th Gen Intel(R) Core(TM) i5-12600KF 3.70 GHz CPU and 32 GB RAM. Graphics processing was handled by an NVIDIA 4060 Ti GPU with 16 GB of VRAM. The model was developed using Python 3.10.14, with CUDA 12.1 and PyTorch 2.2.2. No pre-trained weights were utilized during the experiments. The experimental settings were as follows: the input image resolution was set to 640 × 640 pixels; training was conducted for over 300 epochs with a batch size of 32. The initial learning rate was set to 0.01. The stochastic gradient descent (SGD) momentum was set to 0.937, and the weight decay coefficient was set to 0.0005. These hyperparameters, including the learning rate and momentum, were not arbitrarily selected nor directly inherited from previous studies;

instead, they were meticulously tuned through a series of preliminary experiments to ensure optimal training stability and model performance. All experiments were conducted using a fixed random seed (42) and consistent data splits to reduce variability. The results showed minimal fluctuations across repeated training runs, indicating stable performance.

Evaluation metrics

To comprehensively assess model performance, accuracy metrics were employed —Precision, Recall, F1 Score, and mean Average Precision (mAP)—together with efficiency metrics including model size, parameter count, and GFLOPs. Precision and Recall are defined as shown in Eq. 3 and Eq. 4, respectively,

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4}$$

where TP represents the number of true positives, FP refers to false positives, and FN denotes false negatives. The F1 Score, which representing the harmonic mean of precision and recall, is calculated as shown in Eq. 5:

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5}$$

Average Precision (AP) quantifies the area under the precision–recall curve for each class, while mean Average Precision (mAP) is the average across all N classes, as shown in Eq. 6:

$$\text{Mean Average Precision (mAP)} = \frac{1}{\text{N}} \sum_{\text{i}=1}^{\text{N}} \text{AP}_\text{i} \tag{6}$$

Both mAP@50 (with an IoU threshold of 0.5) and mAP@50–95 (averaged over thresholds ranging from 0.5 to 0.95 in 0.05 increments) are reported in this work to assess detection robustness. To evaluate computational complexity, GFLOPs is calculated as shown in Eq. 7:

$$\text{GFLOPs} = \frac{\text{H} \times \text{W} \times \text{C}_\text{in} \times \text{C}_\text{out} \times \text{k} \times \text{k}}{10^9} \tag{7}$$

where H and W represent the feature map's height and width, $\text{C}_\text{in}$and C_out are the input and output channel counts, and k is the kernel size. This evaluation framework offers a balanced view of detection performance and model efficiency.

RESULTS AND DISCUSSION

Module Comparative Experiments

To validate the effectiveness of the proposed Adaptive Multi-Kernel Depthwise Conv2d (AMDC) module, a series of comparative experiments were conducted using three classical feature extraction modules and a baseline model. Specifically, the original C3K2 module was replaced with C3K2-Faster (Chen *et al.* 2023), C3K2-Mambaout (Yu and Wang 2024), and C3K2-DBB (Ding *et al.* 2021), respectively. The experimental results are presented in Table 2. C3K2-Faster achieved reductions in model parameters and

computational cost, but its detection performance declined substantially. C3K2-Mambaout and C3K2-DBB improved detection accuracy yet it failed to reduce the model's complexity. In contrast, the proposed AMDC module not only improved detection performance but also reduced parameter count and computational overhead, demonstrating superior efficiency and robustness.

Table 2. Comparison of Feature Extraction Modules

Model	P	R	F1	mAP@50	mAP@50:95	GFLOPs (G)	Parameters (M)
baseline	**0.89**	0.79	**0.82**	0.86	0.49	6.3	2.60
C3k2-Faster	**0.89**	0.76	0.8	0.84	0.47	**5.8**	**2.30**
C3k2-mambaout	0.84	0.8	0.8	**0.87**	0.49	6.9	2.50
C3k2-DBB	0.78	**0.82**	0.8	**0.87**	**0.50**	6.3	2.58
AMDC	0.85	0.79	0.81	**0.87**	**0.50**	**5.8**	**2.30**

To further assess the impact of attention mechanisms on the performance of the shared detection head, comparative experiments were conducted with several state-of-the-art attention-based designs, including DyHead (Dai *et al.* 2021), EfficientHead (Tan *et al.* 2020), and SEAM Head (Yu *et al.* 2022), which incorporates occlusion-aware attention. A baseline model without attention mechanisms in the shared convolution was used for comparison. The results are summarized in Table 3. DyHead introduced only limited performance gains while substantially increasing model size and computational overhead. Although EfficientHead and SEAM Head offered a better trade-off between accuracy and efficiency, only the proposed LWDetHead achieved the highest detection performance while substantially reducing parameters and maintaining a low computational cost.

Table 3. Comparative Analysis of Detection Heads

Model	P	R	F1	mAP@50	mAP@50:95	GFLOPs (G)	Parameters (M)
baseline	**0.89**	0.79	0.82	0.86	0.49	6.3	2.60
dyhead	0.80	0.80	0.79	0.85	0.47	7.4	3.10
EfficientHead	0.88	**0.83**	**0.83**	0.86	0.51	**5.1**	2.32
SEAMHead	0.85	0.82	0.82	**0.88**	**0.52**	5.8	2.50
LWDetHead	0.85	0.82	**0.83**	**0.88**	**0.52**	6.0	**2.26**

Ablation Studies

To evaluate the impact of the proposed enhancement modules on the model's performance for chipboard surface defect detection, a series of ablation experiments were conducted based on the YOLOv11n framework. The core evaluation metrics included the F1 score, mAP@50, mAP@50:95, GFLOPs, and the number of parameters. The experimental results are presented in Table 4. The assessment of the proposed LE-YOLO model involved eight experimental configurations: Experiment 1: Baseline YOLOv11n model; Experiment 2: Incorporation of the SDFP module into the baseline; Experiment 3: Replacement of the C3k2 module in the YOLOv11 backbone with the AMDC module; Experiment 4: Redesign of the detection head using the proposed LWDetHead; Experiment 5: Integration of AMDC on top of the Experiment 2 configuration; Experiment 6: Integration of LWDetHead on top of the Experiment 2 configuration; Experiment 7:

Integration of LWDetHead on top of the Experiment 3 configuration; Experiment 8: Comprehensive application of all proposed modules, representing the final LE-YOLO architecture.

Table 4. Ablation Experiment

Method	F1	mAP@50	mAP@50:95	GFLOPs(G)	Parameters (M)
method(1)	0.82	0.86	0.49	6.3	2.58
method(2)	0.83	0.87	0.50	6.3	2.63
method(3)	0.81	0.87	0.49	5.8	2.30
method(4)	0.82	0.88	0.52	6.0	2.26
method(5)	0.83	0.84	0.50	5.8	2.47
method(6)	0.83	0.88	0.50	6.0	2.31
method(7)	0.83	0.89	0.51	**5.5**	**1.99**
method(8)	**0.84**	**0.90**	**0.55**	**5.5**	2.10

The results show that, except for the SDFP module, all additional enhancement modules contributed to a reduction in model parameters. Although the inclusion of SDFP resulted in a slight increase of approximately 50K parameters compared to the baseline, it led to a 1% improvement in F1 score, mAP@50, and mAP@50:95. This highlights the SDFP module's effectiveness in enhancing detection accuracy through multi-scale feature fusion under a lightweight design constraint. The integration of the AMDC module yielded a 1% decrease in F1 score but improved mAP@50 1%, increased inference speed 7.9%, and reduced the parameter count by 280K compared to the baseline. These results demonstrate the robustness and computational efficiency of AMDC. Replacing the original detection head with the proposed LWDetHead led to a 2% increase in mAP@50 and a 3% increase in mAP@50:95, along with a 4% increase in inference speed and a reduction of 320K parameters, emphasizing the critical role of LWDetHead in achieving model compression without sacrificing accuracy. Overall, the results validate that the proposed LE-YOLO model not only enhances detection precision but also achieves significant model compression and computational efficiency.

Visual Analytics

In addition to these quantitative results, the visual analysis of the ablation study in Fig. 7 further demonstrates the improvements in detection performance and efficiency, providing a clearer depiction of how each enhancement module contributes to the overall capabilities of the model. Overall, the results validate that the proposed LE-YOLO model not only enhances detection accuracy but also achieves substantial model compression and optimization in computational efficiency.

To both intuitively and quantitatively assess the performance differences among the proposed LE-YOLO model, the baseline YOLOv11n, and the latest model in the series, YOLOv12n (Tian *et al.* 2025), a heatmap-based visual analysis was conducted for chipboard surface defect detection under three lighting conditions: normal illumination, strong lighting, and low-light environments. As shown in Fig. 8, the heatmaps visualize pixel-level response intensities, effectively visualizing the regions of focus for defect feature extraction.

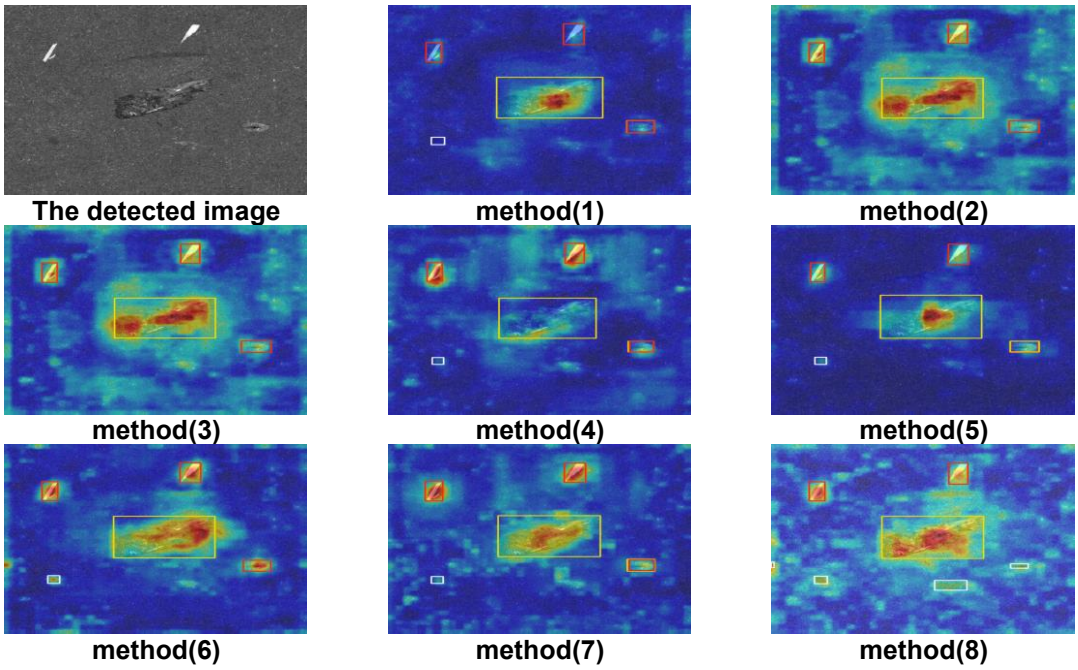

Fig. 7. The visual analysis of the ablation study

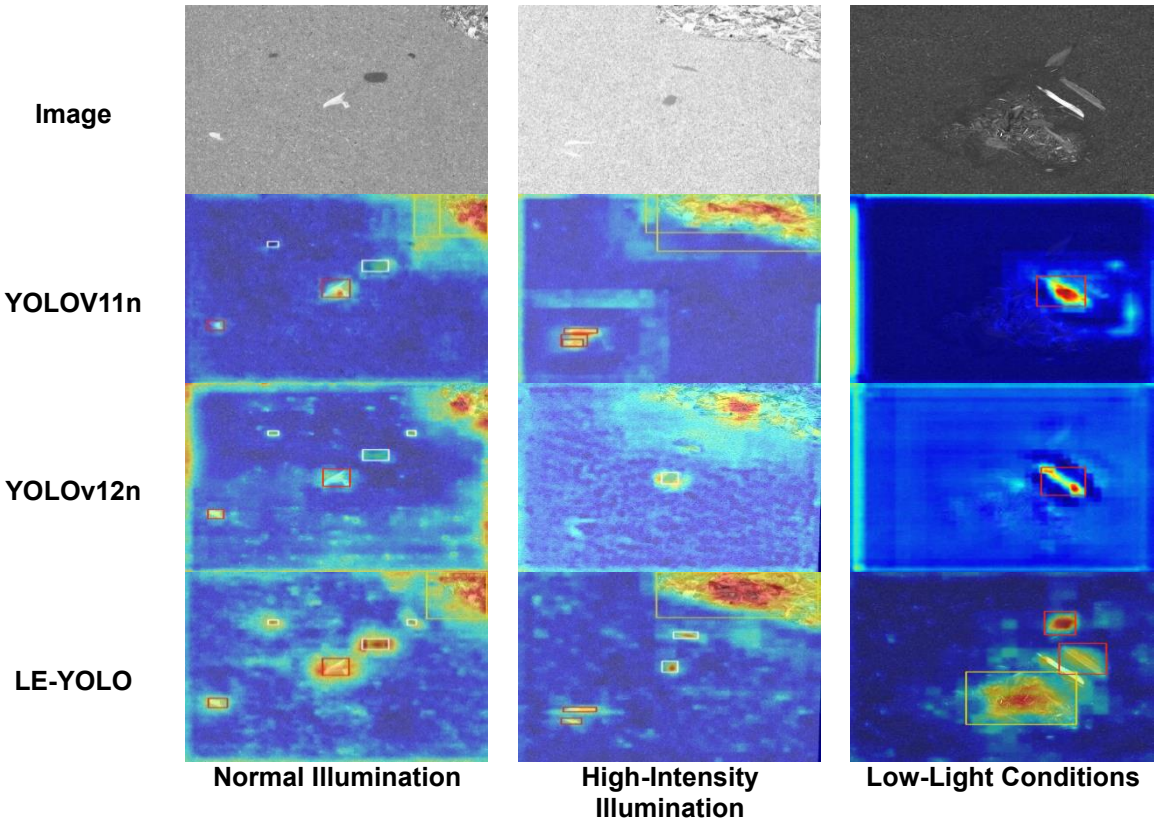

Fig. 8. Heatmap comparison

Experimental results show that both YOLOv11n and YOLOv12n produce scattered heatmap responses, which are notably influenced by background textures, leading to redundant and inaccurate feature representations. In contrast, the LE-YOLO model, with its enhanced feature perception mechanism, effectively focuses on defect regions, achieving comprehensive detection of fine-grained and complex surface flaws. It demonstrates superior performance in both coverage and focus. This comparative analysis not only highlights the substantial performance improvements of LE-YOLO but also emphasizes its increased robustness and resilience to interference, achieved through improved background suppression and refined feature filtering capabilities.

Performance Comparison with Mainstream Algorithms

To further verify the effectiveness of the proposed method, the authors compared LE-YOLO with eight state-of-the-art object detection algorithms: YOLOv5n, YOLOv7-tiny, YOLOv8n, YOLOv9n, YOLOv10n, YOLOv11n, YOLOv12n, and RT-DETR. All experiments were conducted on the unified Chipboardv1.0 dataset to ensure fair and consistent evaluation. The results, presented in Table 5, demonstrate that the proposed method achieves outstanding detection accuracy. Notably, LE-YOLO's performance also surpasses RT-DETR, except for precision, LE-YOLO outperforms all comparison models in recall, F1-score, mAP@50, and mAP@50:95. Notably, it substantially surpasses YOLOv12n, the latest model in the YOLO series, further confirming the superiority of LE-YOLO in detection precision. In terms of model lightweighting, the proposed method exhibits lower computational complexity and fewer parameters than all comparison algorithms except YOLOv5n. Compared with YOLOv5n, LE-YOLO improves recall 4%, F1-score by 1%, mAP@50 by 4%, and mAP@50:95 by 8%. Although the precision decreases 2%, this is accompanied by only a slight increase in resource consumption. These findings demonstrate that the proposed algorithm achieves high-precision detection while effectively balancing performance with lightweight design, exhibiting substantial comprehensive advantages.

Table 5. Performance Comparison with Mainstream Algorithms

Model	P	R	F1	mAP@50	mAP@50:95	GFLOPs (G)	Parameters (M)
YOLOv5n	0.88	0.81	0.84	0.86	0.47	**4.1**	**1.76**
YOLOv7-tiny	**0.89**	0.81	0.84	0.89	0.49	13.0	6.01
YOLOv8n	0.84	0.79	0.82	0.86	0.51	8.1	3.00
YOLO10n	0.83	0.81	0.82	0.87	0.50	8.2	2.70
YOLOv12n	0.78	0.79	0.78	0.84	0.48	5.8	2.51
YOLOv11n	**0.89**	0.80	0.84	0.86	0.49	6.3	2.58
RT-DETR	**0.89**	0.73	0.79	0.80	0.44	56.9	19.9
LE-YOLO	0.86	**0.85**	**0.85**	**0.90**	**0.55**	5.5	2.10

CONCLUSIONS

1. This study proposed LE-YOLO, a lightweight and enhanced object detection model based on YOLOv11, designed specifically for detecting surface defects on particleboard. Through integrating the Adaptive Multi-Kernel Depthwise Conv2d

(AMDC), Shared Dilated Feature Pyramid (SDFP), and a Lightweight Detection Head (LWDetHead), and introducing the Normalized Wasserstein Distance (NWD) into the loss function, the model effectively improves detection accuracy while reducing computational cost.

2. Based on an extensive evaluation on the Chipboardv1.0 dataset, the LE-YOLO outperformed the baseline YOLOv11n model in detecting particleboard surface defects, especially small and intricate flaws. Specifically, LE-YOLO achieved a 4% increase in mAP@50, a 6% improvement in mAP@50:95, and demonstrated enhanced robustness in challenging lighting conditions, such as low-light and high-reflection environments. These results highlight the model's ability to handle complex real-world industrial scenarios. Moreover, LE-YOLO showed superior detection accuracy with a 2% increase in recall and a 1% boost in F1-score, confirming its overall effectiveness in detecting defects. The model also exhibited a 7.9% increase in inference speed, while maintaining a low number of parameters, ensuring its suitability for real-time deployment with limited computational resources.

ACKNOWLEDGMENTS

The authors are grateful for the financial support from the Inner Mongolia Natural Science Foundation for the project "Construction of an Intelligent Inspection Model for Surface Defects of Particleboard Made from Psammophytic Shrubs" (Project No. 2024LHMS03063).

REFERENCES CITED

Chen, J., Kao, S.-h., He, H., Zhuo, W., Wen, S., Lee, C.-H., and Chan, S.-H. G. (2023). "Run, don't walk: Chasing higher FLOPS for faster neural networks," *arXiv* preprint, article ID 2303.03667v3. DOI: 10.48550/arXiv.2303.03667v3

Dai, X., Chen, Y., Xiao, B., Chen, D., Liu, M., Yuan, L., and Zhang, L. (2021). "Dynamic head: Unifying object detection heads with attentions," *arXiv* preprint, article ID 2106.08322v1. DOI: 10.48550/arXiv.2106.08322v1

Ding, F., Zhuang, Z., Liu, Y., Jiiang, D., Yan, X., and Wang, Z. (2020). "Detecting defects on solid wood panels based on an improved SSD algorithm," *Sensors* 20(18), article E5315. DOI: 10.3390/s20185315

Ding, X., Zhang, X., Han, J., and Ding, G. (2021). "Diverse branch block: Building a convolution as an inception-like unit," *arXiv* preprint, article ID 2103.13425v2. DOI: 10.48550/arXiv.2103.13425v2

Girshick, R. (2015). "Fast r-cnn," in: *Proceedings of the IEEE International Conference on Computer Vision*, Santiago, Chile, pp. 1440-1448.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). "Rich feature hierarchies for accurate object detection and semantic segmentation," in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, pp. 580-587.

Ji, X., Guo, H., and Hu, M. (2019). "Features extraction and classification of wood defect based on hu invariant moment and wavelet moment and BP neural network," in:

VINCI '19: Proceedings of the 12th International Symposium on Visual Information Communication and Interaction, Shanghai China, pp. 1-5.

Khanam, R., and Hussain, M. (2024). "YOLOV11: An overview of the key architectural enhancements," *arXiv* Preprint, article ID 2410.17725.

Li, C., Yao, H., and Jian, B. (2025). "Discrete wavelet transform sampling for image super resolution," *Applied Artificial Intelligence* 39(1), article ID 2449296. DOI: 10.1080/08839514.2024.2449296

Meng, W., and Yuan, Y. (2023). "SGN-YOLO: Detecting wood defects with improved YOLOv5 based on semi-global network," *Sensors* 23(21), article 8705. DOI: 10.3390/s23218705

Pan, S., Wang, K., Chen, J., and Zhang, Y. (2022). "Larch wood defect definition and microscopic inversion analysis using the ELM near-infrared spectrum optimization along with WOA–SVM," *BioResources* 17(1), 682-698. DOI: 10.15376/biores.17.1.682-698

Tan, M., Pang, R., and Le, Q. V. (2020). "EfficientDet: Scalable and efficient object detection," *arXiv* Preprint, Article ID 1911.09070v7. DOI: 10.48550/arXiv.1911.09070v7

Tian, Y., Ye, Q., and Doermann, D. (2025). "YOLOv12: Attention-centric real-time object detectors," *arXiv* Preprint, article ID 2502.12524v1. DOI: 10.48550/arXiv.2502.12524v1

Uijlings, J., VandeSande, K., Gevers, T., and Smeulders, A. (2013). "Selective search for object recognition," *International Journal of Computer Vision* 104(2), 154-171. DOI: 10.1007/s11263-013-0620-5

Wang, J., Xu, C., Yang, W., and Yu, L. (2022). "A normalized Gaussian Wasserstein distance for tiny object detection," *arXiv* Preprint, article ID 2110.13389v2. DOI: 10.48550/arXiv.2110.13389v2

Wang, R., Chen, Y., Liang, F., Wang, B., Mou, X., and Zhang, G. (2024). "BPN-YOLO: A novel method for wood defect detection based on YOLOv7," *Forests* 15(7), article 1096. DOI: 10.3390/f15071096

Yu, W., and Wang, X. (2024). "MambaOut: Do we really need mamba for vision?," *arXiv* preprint article ID 2405.07992v3. DOI: 10.48550/arXiv.2405.07992v3

Yu, Z., Huang, H., Chen, W., Su, Y., Liu, Y., and Wang, X. (2022). "YOLO-FaceV2: A scale and occlusion aware face detector," *arXiv* preprint, article ID 2208.02019v2. DOI: 10.48550/arXiv.2208.02019v2

Zou, X., Wu, C., Liu, H., Yu, Z., and Kuang, X. (2025). "An accurate object detection of wood defects using an improved Faster R-CNN model," *Wood Material Science and Engineering* 20(2), 413-419. DOI: 10.1080/17480272.2024.2352605

Article submitted: April 27, 2025; Peer review completed: May 18, 2025; Revised version received and accepted: July 2, 2025; Published: July 11, 2025.
DOI: 10.15376/biores.20.3.7179-7193