Journal of Computer Science and Electrical Engineering
Print ISSN: 2663-1938
Online ISSN: 2663-1946
DOI: https://doi.org/10.61784/jcsee3132

THE ALGORITHM FOR DETECTING OIL STAINS AND FOREIGN OBJECTS ON THE BOTTOM OF EMU CARS BASED ON DEEP LEARNING

ZhiJian Wei[#], SongTao Zhang[#], ZiYi Xu, Hang Zhou[*]
Beijing Jiaotong University Weihai International College, Weihai 264200, Shandong, China.
[#]*These authors contributed equally to this work.*
[*]*Corresponding Author: Hang Zhou*

Abstract: The operation safety of EMU (Electric Multiple Unit) trains is the core guarantee of the high-speed railway transportation system. Currently, the widely deployed EMU operation fault dynamic image detection system (TEDS) in China mainly relies on manual image interpretation, resulting in low detection efficiency, prone to missed detections and false detections. To address the challenges in detecting oil stains under EMU car bottoms, such as scarce samples, insufficient accuracy of a single model, and difficulty in identifying reflective oil stains, this paper proposes an oil stain detection algorithm based on the YOLOv12 and DeepLabV3+ dual-model collaboration. In terms of data augmentation, to address the deficiency of only 356 original oil stain samples, this paper designs a three-stage data augmentation strategy. This strategy expands the training set to 6786 images through basic geometric transformations, noise addition and blurring processing, as well as a composite enhancement pipeline based on the Albumentations library, effectively enhancing the generalization ability of the model. This paper uses the YOLOv12 as the target detection model and trains an oil stain detector on the expanded dataset. Experimental results show that the YOLOv12 model achieves an accuracy of0.88 on the validation set, a recall rate of 0.70, and a recall rate of 0.79 for the oil stain category, effectively identifying most oil stain targets. The oil stain candidate regions detected by YOLOv12 are input into the DeepLabV3+ network, using MobileNetV2 as a lightweight backbone network, and training a pixel-level oil stain segmentation model. Experimental results show that the model achieves an average IoU of 0.4535 and an average Dice coefficient of 0.4872 on the validation set. The test results show that the model can effectively identify reflective oil stains that are difficult for humans to distinguish and reduces false alarms for dried traces. The joint detection framework combining target detection and semantic segmentation proposed in this paper integrates the characteristics of YOLOv12's rapid localization and DeepLabV3+'s fine segmentation. This model can adapt to the oil stain defect detection requirements in complex EMU operation environments and provide a reference path for research on railway image intelligent detection technologies.
Keywords: Multiple-unit train; Oil contamination detection; YOLOv12; DeepLabV3+; Data augmentation; Semantic segmentation

1 INTRODUCTION

1.1 Research Background and Significance

High-speed railways, as an important component of modern transportation systems, have developed rapidly in China. EMU (Electric Multiple Unit) trains operate at high speeds and with high operational density, and their safety directly affects people's lives and property as well as the order of railway transportation. Key components of EMUs (such as bogies, axles, gearboxes, couplings, etc.) may experience oil leakage due to seal failure, mechanical wear, and other reasons during long-term high-load operation. Oil stains not only affect the appearance of the equipment but may also mask cracks, accelerate component aging, and pose fire hazards. In severe cases, they can lead to train accidents. Therefore, timely and accurate detection of oil stains under the EMU is an important link in ensuring the safety of EMU operation.

Currently, China's railways have widely deployed the Trouble of Moving EMU Detection System (TEDS). This system installs high-speed cameras along the tracks to take real- time photos of the underside and sides of moving EMUs and transmits the images to a monitoring center, where professionals analyze the images to identify oil stains, foreign objects, missing bolts, and other faults [1]. However, the TEDS system still has significant problems in practical application. The volume of train image data is huge, with a single EMU train generating thousands of high-definition images. Relying solely on manual review leads to high labor intensity and low efficiency for inspectors, and is prone to missed detections and false alarms due to visual fatigue. Oil stain areas often have high visual similarity with rust, shadows, water stains, and other backgrounds, making it difficult for traditional image processing algorithms to effectively distinguish them. Existing intelligent detection algorithms mostly use a single model, which has limited detection accuracy for small targets and weak-textured oil stains, and insufficient real-time performance [2-3].

In recent years, deep learning technology has made break- throughs in the field of computer vision, especially in object detection and semantic segmentation algorithms, which have been widely applied in remote sensing image analysis, medical image processing, industrial defect detection, and other fields [4-5]. In the field of railway defect detection, some

researchers have attempted to apply deep learning to tasks such as bolt loss detection and contact network component defect recognition, achieving good results [6-7]. However, for the specific target of oil stains under EMUs, existing research still has deficiencies. Oil stain samples are scarce, making it difficult to train models with high generalization ability. Most methods only use a single object detection or segmentation model, which is difficult to balance real-time performance and precision. Existing studies also have insufficient recognition capabilities for complex forms of oil stains such as reflective and dried oil stains.

To address the above problems, this paper proposes a dual- model collaborative oil stain detection algorithm for EMU underbodies based on YOLOv12 and DeepLabV3+. This algorithm first uses the YOLOv12 object detection model to quickly and roughly locate the oil stain areas, and then employs the DeepLabV3+ semantic segmentation model to perform pixel-level fine segmentation on the candidate areas, thereby achieving accurate location and boundary delineation of the oil stain areas. At the same time, this paper designs a multi-stage data augmentation strategy to effectively expand the scale of oil stain samples and enhance the model's generalization ability. Experimental results show that this method achieves high detection accuracy and segmentation performance on the self-built dataset, providing technical support for the intelligent upgrade of the TEDS system.

1.2 Research Status

The research on intelligent railway image detection started relatively early. The "Doctor Yellow" detection system adopted by the Japanese Shinkansen integrates laser scanning and infrared imaging technologies. This algorithm can conduct comprehensive detection of the contact network and track conditions, but its specificity in identifying oil-soiled targets is limited. The Transportation Technology Center (TTC) in the United States conducted research on the detection of key components of the bogie based on Faster R-CNN, but the model has a large number of parameters, making it difficult to meet real-time requirements [7]. Some research institutions in Europe attempted to use traditional machine learning methods (such as support vector machines, random forests) to identify faults in railway components, but these methods rely on manually designed features, resulting in weak generalization ability [8].

In China, the TEDS system has been widely deployed, and related research mainly focuses on the improvement of fault detection algorithms based on deep learning. Fan Li et al. proposed a TEDS image-assisted recognition system for EMU based on the improved SOLOv2 network [1], reducing memory usage through deep separable convolution and achieving unsupervised anomaly detection by integrating the GAN network and feature matching. The improved recognition system has an average false alarm rate of 8.66% under zero false alarm conditions. Luo Hui improved the YOLOv4 and DeepLabV3+ models for the problems of high-speed rail bolt loss and oil-soil detection [2], and significantly improved the detection accuracy and speed by introducing attention mechanisms, K-means++ clustering, and lightweight backbone networks. Yang Zihua designed a two-stage joint detection model including a lightweight feature extraction network and a global attention module for contact network equipment defects [3], achieving effective identification of minor defects such as missing and loose fasteners. Yin Tengqing et al. applied DeepLabV3+ to the extraction of water body information in high-resolution remote sensing images [5], verifying the model's fine segmentation ability for target boundaries in complex backgrounds. Zhu Yongjun et al. improved the YOLOv11 model for the problem of small target detection in complex backgrounds of open-pit mines [6]. This research team introduced robust feature downsampling modules and PIoU loss functions to effectively alleviate the problem of feature attenuation of small targets.

These studies provide important technical references for the detection of oil stains on the underside of EMUs, but still have limitations. Most studies focus on a single model (object detection or segmentation), failing to fully utilize the complementary advantages of the two models. The problem of scarce oil-soil samples has not been systematically solved. The existing detection methods have limited ability to distinguish complex shapes such as reflective oil stains and dried traces. In terms of object detection algorithms, based on the different network structures, mainstream algorithms can be divided into two-stage and single-stage categories. Two-stage detection algorithms are represented by the R-CNN series, including Fast R-CNN and Faster R-CNN. This algorithm first generates candidate regions through the region proposal network and then classifies and regresses the candidate regions. Faster R- CNN has advantages in detection accuracy, but due to the need for two forward passes, the inference speed is slow and difficult to meet real-time detection requirements [9]. Single- stage detection algorithms are represented by the YOLO series, treating object detection as a regression problem and directly predicting bounding boxes and class probabilities on the im- age. YOLOv4 introduced CSPDarknet53 backbone network, PANet feature fusion, and Mosaic data augmentation, achieving a good balance between accuracy and speed [10]. YOLOv8 and YOLOv10 further optimized the network structure and training strategies, maintaining high speed while improving detection accuracy [11]. YOLOv11 further reduced the latency by introducing the C3K2 module and lightweight depthwise separable convolution [12].

The YOLOv12 proposed in 2025 is the first real-time object detector centered on the attention mechanism [13]. To address the issue of low computational efficiency of the traditional attention mechanism, YOLOv12 introduces a regional attention module, dividing the feature map into several regions for local attention calculation, significantly reducing the computational complexity. YOLOv12 is also equipped with a residual efficient layer aggregation network to solve the optimization problem of the attention model and introduces FlashAttention to accelerate memory access. Experiments show that YOLOv12 achieves higher detection accuracy on the COCO dataset at a speed comparable to YOLOv11, especially excelling in the detection of small targets [13]. Given the characteristics of the oil stain targets on the underside of high-speed trains, which are small in size, variable in shape, and have complex backgrounds, this paper selects

YOLOv12 as the basic model for oil stain target detection.

In the aspect of semantic segmentation algorithms, the fully convolutional network is the first to apply convolutional neural networks to pixel-level classification, achieving end-to- end semantic segmentation [14]. U-Net adopts a symmetric encoder-decoder structure, fusing shallow details and deep semantics through skip connections, achieving great success in medical image segmentation [15]. The DeepLab series is another important branch in the field of semantic segmentation. DeepLabV1 introduces dilated convolution to expand the receptive field without increasing the parameter quantity; DeepLabV2 adds a dilated spatial pyramid pooling module to capture context information of different scales through multi- scale dilated convolution; DeepLabV3 improves the ASPP structure, not relying on random fields; DeepLabV3+ intro- duces an encoder-decoder structure, using DeepLabV3 as the encoder and adding a simple decoder module to restore the target boundary details [16-17]. DeepLabV3+ has achieved excellent segmentation accuracy on multiple public datasets, especially suitable for target segmentation tasks requiring fine boundaries. In the field of railway oil stain detection, Luo Hui applied DeepLabV3+ to detect oil stains on the underside of high-speed trains [2], replacing the Xception backbone network with MobileNetV2, maintaining the segmentation accuracy while increasing the inference speed by 3.4 times, and using Focal Loss to solve the imbalance problem of positive and negative samples. The research of Yin Tengqing et al. also shows that DeepLabV3+ can fully utilize the spectral and spatial texture features of high-resolution remote sensing images to achieve high-precision water body extraction in complex environments [5].

A single model often has limitations in complex detection tasks: object detection models are good at rapid positioning but have rough boundaries, while semantic segmentation models have high accuracy but large computational costs. In recent years, some researchers have begun to explore the dual-model collaborative architecture that combines object detection and semantic segmentation. For example, Heckhel and Helali fused MRI and DTI multimodal images [18], using YOLOv11 to achieve early detection and four-stage classification of Alzheimer's disease, leveraging the complementarity of multimodal data to improve diagnostic accuracy. Inspired by this, this paper proposes to combine YOLOv12 with DeepLabV3+, constructing a hierarchical detection framework of "global coarse positioning - local fine segmentation", aiming to im- prove the precision of oil stain segmentation while ensuring real-time performance.

1.3 Main Research Contents of This Paper

In response to the problems such as scarce samples, insufficient accuracy of a single model, and difficulty in identifying reflective oil stains in the inspection of the underside of EMU (Electric Multiple Unit) vehicles, this paper conducts the following research. In terms of data augmentation, considering the shortage of only 356 original oil stain samples and 294 training samples, a three-stage data augmentation strategy is designed. The strategy includes basic geometric transformations, noise addition and blurring processing, as well as a composite enhancement pipeline based on the Albumentations library. Through operations such as brightness contrast adjustment, affine transformation, and CoarseDropout, the training set is expanded to 6786 samples, effectively enhancing the generalization ability of the model. In terms of object detection, using YOLOv12 as the baseline model, an oil stain detector is trained on the expanded dataset, and the convergence of the loss function, precision, recall rate, mAP50 and mAP50-95 during the training process are analyzed to evaluate the model's ability to locate the oil stain area. In terms of semantic segmentation, the oil stain candidate regions detected by YOLOv12 are input into the DeepLabV3+ network, using MobileNetV2 as a lightweight backbone network, and a pixel-level oil stain segmentation model is trained. The segmentation accuracy is evaluated through metrics such as mIoU, Dice coefficient, and pixel accuracy, and the model's ability to distinguish reflective oil stains from dry marks is analyzed. Finally, a joint detection framework of "object detection + semantic segmentation" is designed, using the rapid localization ability of YOLOv12 to filter candidate regions, and then conducting fine segmentation by DeepLabV3+ to achieve high-precision extraction of the oil stain area.

1.4 Organization Structure of the Paper

This paper is divided into five chapters. Chapter 1 is the Introduction, which introduces the research background and significance, the current domestic and international research status, the main research contents, and the structure of the paper. Chapter 2 is the Related Theoretical Foundation, which systematically introduces the basic concepts of deep learning, the YOLOv12 object detection algorithm, the DeepLabV3+ semantic segmentation algorithm, and the evaluation metrics adopted in this paper. Chapter 3 is the YOLOv12 Oil Stain Object Detection Module, which elaborates on the data set preparation and multi-stage data augmentation strategy, model training configuration, loss function convergence analysis, and detection accuracy evaluation. Chapter 4 is the DeepLabV3+ Semantic Segmentation Module, which introduces the data set format conversion, model training parameter settings, seg- mentation result analysis, and test set verification. Chapter 5 is the Conclusion, which summarizes the work of the entire paper, analyzes the existing shortcomings, and looks forward to future research directions.

2 THEORETICAL BASIS RELATED TO THIS TOPIC

2.1 Foundation of Deep Learning

Deep learning is a branch of machine learning. By con- structing neural networks with multiple hidden layers, it automatically learns hierarchical feature representations from data. In the field of computer vision, the convolutional neural network (Convolutional Neural Network, CNN) is the core deep learning model. The basic structure of CNN

includes convolutional layers, pooling layers, and fully connected layers. The convolutional layer slides a learnable convolution kernel over the input image to extract local features; the pooling layer downsamples the feature map to reduce dimensions while enhancing translational invariance; the fully connected layer maps the extracted high-level features to the sample label space to complete classification or regression tasks. The weight sharing mechanism of CNN significantly reduces the network parameters, enabling it to efficiently process high-dimensional image data (As shown in Figure 1) [19].

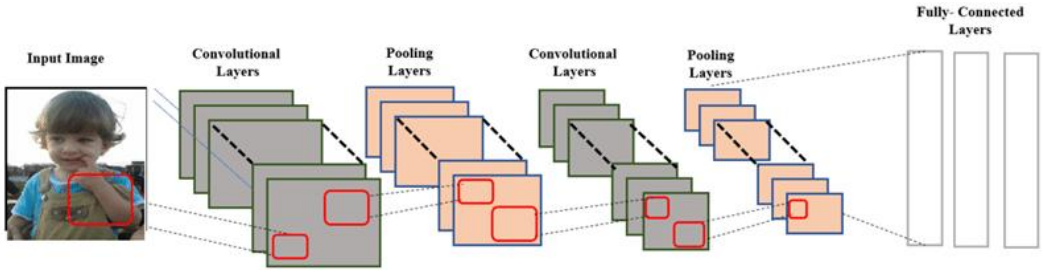


Figure 1 The Typical Architecture of Convolutional Neural Networks (CNNs)

During the training process of deep learning, the loss function is used to measure the difference between the model's predictions and the true labels. Common loss functions include cross-entropy loss and mean squared error loss. Optimization algorithms (such as Stochastic Gradient Descent SGD, Adam) calculate gradients through backpropagation and update the network parameters to gradually minimize the loss function. To prevent overfitting, regularization techniques (such as weight decay), Dropout, and early stopping are often employed. Transfer learning is an important technique in deep learning, referring to the transfer of pre-trained model parameters from a large dataset (such as ImageNet) to the target task for fine-tuning. This method can significantly reduce the training data requirements for the target task, accelerate convergence, and improve the model's generalization ability [20]. In the training of DeepLabV3+, this paper uses the MobileNetV2 pre-trained on ImageNet as the initialization weights for the backbone network.

2.2 YOLOv12 Object Detection Algorithm

The YOLO series is a representative of single-stage object detection algorithms. YOLOv1 divides the input image into an S×S grid, and each grid is responsible for predicting B bounding boxes and class probabilities, converting the detection problem into a regression problem, achieving extremely high inference speed [21]. YOLOv2 introduced anchor boxes, batch normalization, and multi-scale training, improving detection accuracy. YOLOv3 uses a residual network as the backbone and introduces a feature pyramid to achieve multi- scale detection. YOLOv4 has made numerous improvements in data augmentation, network structure, and training strategies, becoming one of the best models in terms of balance between accuracy and speed at that time [10]. Subsequent versions such as YOLOv5, YOLOv8, and YOLOv10 have further optimized the network architecture and training process (As shown in Figure 2) [13-14].

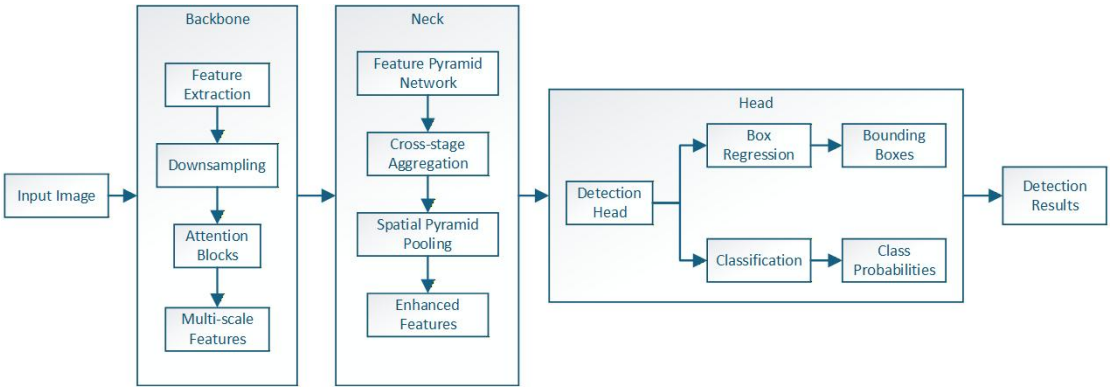


Figure 2 The Architecture of YOLOv12

YOLOv12 is the latest generation of the YOLO model proposed in 2025. Its core innovation lies in the first successful application of the attention mechanism to the real-time object detection framework [15]. Although traditional attention mechanisms (such as the Transformer) have strong modeling capabilities, they are difficult to meet real-time requirements due to their high computational complexity and inefficient memory access. YOLOv12 solves this problem through three key designs. The first design is the regional attention, which divides the feature map into l regions (default l=4) along the horizontal or vertical direction, and independently calculates self-attention in each region. Compared with sliding windows or cross-attention, regional attention only requires simple reshape operations, avoiding complex window division and inversion processes, and significantly improving computational efficiency. Although the computational complexity is still $O(n^2)$, since n is fixed as the input resolution (such as 640×640), regional attention can meet real-time requirements while maintaining a large receptive field [15]. The second design is the residual efficient layer aggregation

network. YOLOv12 addresses the optimization challenges introduced by the attention mechanism, on the basis of ELAN, adds a residual shortcut from the input to the output, and introduces a scaling factor (default 0.01), effectively solving the training instability problem of large models. R-ELAN also improves the feature aggregation method, changing the original segmentation-processing-joining structure to a bottle- neck structure, reducing the computational load and param- eters [15]. The third design is the architecture optimization. YOLOv12 adopts a hierarchical design (retaining the first two stages of the backbone network), removes the three stacked modules in the final stage, and retains a single R-ELAN block; adjusts the MLP ratio (from 4 to 1.2 or 2), allocating more computing resources to the attention mechanism; replaces Linear+LN with Conv2d+BN to fully utilize the efficiency of the convolution operator; removes the position encoding, introduces a large kernel depth separable convolution (7×7) as a position-aware layer; and integrates FlashAttention to optimize memory access [15].

YOLOv12 provides five model scales: N, S, M, L, and X. Experiments on the COCO dataset show that YOLOv12-N achieves 40.6% mAP with a delay of 1.64ms, outperforming YOLOv10-N and YOLOv11-N [15]. Given the characteristics of the oil stain target on the train car bottom, which is small in size and has weak texture, this paper selects YOLOv12 as the basic model for oil stain detection. In terms of the loss function, the loss function of YOLOv12 consists of three parts: bounding box regression loss, classification loss, and confidence loss. The bounding box regression loss typically uses the CIoU loss, whose calculation formula is

$$L_C IoU = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v. \quad (1)$$

where IoU is the intersection-over-union of the predicted box and the real box, ρ is the Euclidean distance, c is the diagonal length of the minimum enclosing rectangle, and v is the aspect ratio consistency parameter. The CIoU loss considers overlapping area, center point distance, and aspect ratio, guiding the predicted box to approach the real box more quickly. However, the CIoU loss has the problem of anchor box expansion, that is, during training, anchor boxes may expand in size rather than precisely locate to reduce the loss [10].

2.3 DeepLabV3+ Semantic Segmentation Algorithm

In the semantic segmentation task, to obtain a larger receptive field to capture contextual information, traditional CNNs usually adopt the strategy of stacking convolutional layers and pooling layers. However, the pooling operation reduces the resolution of the feature map, resulting in the loss of spatial details. Dilated convolution expands the receptive field exponentially without increasing the number of parameters by inserting zero values between the elements of the convolution kernel [18]. For a convolution kernel of size k, the equivalent convolution kernel size under the dilation rate r is

$$K = k + (k-1)(r-1) \quad (2)$$

When r = 1, dilated convolution degenerates into standard con- volution; when r >1, the receptive field expands significantly. Dilated convolution enables the network to obtain multi-scale context information while maintaining high-resolution feature maps, which is particularly beneficial for segmenting small targets and fine boundaries (As shown in Figure 3).

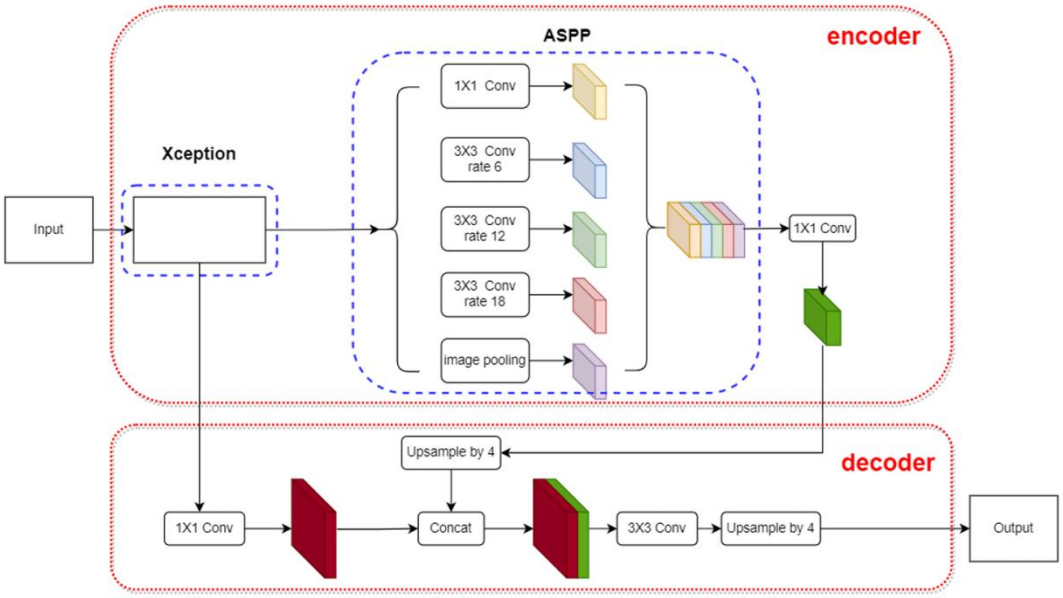


Figure 3 The Network of DeepLabV3+ with Mobilenetv2 as the Backbone (Source: [18])

DeepLabV3+ is the latest version of the DeepLab series, and its core contribution lies in integrating the encoder-decoder structure with dilated convolution and ASPP module [19]. The encoder part is based on DeepLabV3, first extracting features using a backbone network (such as Xception or MobileNetV2), and then capturing multi-scale context

information through the ASPP module. The ASPP module uses multiple dilated convolutions with different dilation rates (such as rates of 6, 12, 18, and 24) to parallelly concatenate the output features and then reduce the dimension through a 1×1 convolution, thereby integrating features of different scales. The decoder part is responsible for restoring spatial resolution. Specifically, the features from the encoder are upsampled by a factor of 4; at the same time, shallow features (such as high-resolution feature maps) are extracted from the backbone network and reduced in channel number through a 1×1 convolution to reduce computational load; then, the upsampled deep features are concatenated with the shallow features; finally, after a 3×3 convolution and upsampled by a factor of 4, the original input image size is restored, and the pixel-wise classification results are output. The encoder-decoder structure of DeepLabV3+ not only utilizes the multi-scale context information of ASPP but also restores the fine boundaries of the target through feature fusion in the decoder, achieving leading performance on multiple semantic segmentation benchmark datasets [19].

To reduce the computational complexity of DeepLabV3+ and improve the inference speed, a lightweight backbone network can be used to replace the original Xception. Mo- bileNetV2 is an efficient CNN architecture proposed by Google, whose core is the inverted residual structure and linear bottleneck [22]. The inverted residual structure first expands the channel dimension through a 1×1 convolution, then extracts features using depthwise separable convolution, and finally compresses the channels through a 1×1 con- volution. Depthwise separable convolution decomposes the standard convolution into depthwise convolution (per-channel convolution) and pointwise convolution (1×1 convolution), with parameters and computational cost approximately one- third of the standard convolution [22]. Luo Hui applied MobileNetV2 to the oil spill detection task of DeepLabV3+ [2], and in the process of maintaining the segmentation accuracy basically unchanged, the inference speed was increased by 3.4 times. This paper also uses MobileNetV2 as the backbone network of DeepLabV3+.

2.4 Evaluation Indicators

In order to evaluate the performance of the YOLOv12 oil spill detection model, the following evaluation metrics are adopted in this paper (as shown in Table 1).

Table 1 Evaluation Metrics for YOLOv12 Model

Metric Category	Metric Name	Symbol	Meaning
Detection Accuracy	Precision	metrics/precision(B)	Proportion of samples predicted as oil stain that are actually oil stain,$P=\frac{TP}{TP+FP}$
Detection Accuracy	Recall	metrics/recall(B)	Proportion of actual oil stain samples correctly detected,$R=\frac{TP}{TP+FN}$
Detection Accuracy	Average Precision (IoU=0.5)	metrics/mAP50(B)	Mean average precision when IoU threshold is 0.5
Detection Accuracy	Average Precision (IoU=0.5:0.95)	metrics/mAP50-95(B)	Average over IoU thresholds from 0.5 to 0.95 step 0.05
Loss Function	Bounding Box Regression Loss	box_loss	CIoU loss for localization error
Loss Function	Classification Loss	cls_loss	Cross-entropy loss for classification error
Loss Function	Distribution Focal Loss	dfl_loss	Prediction accuracy of bounding box distribution

In order to evaluate the oil stain segmentation performance of DeepLabV3+, the following evaluation metrics are adopted in this paper (as shown in Table 2).

Table 2 Evaluation Metrics for DeepLabV3+ Model

Metric Category	Metric Name	Symbol	Meaning
Segmentation Accuracy	Mean Intersection over Union	mIoU	Measures the overlap between predicted segmentation region and ground truth mask.$mIoU=\frac{1}{n}\sum_{i=1}^{n}\frac{TP_i}{TP_i+FP_i+FN_i}$ .Higher value indicates better segmentation
Segmentation Accuracy	Dice Coefficient	Dice	Measures the similarity between predicted region and ground truth, equivalent to F1 score.$Dice=\frac{2\cdot TP}{2\cdot TP+FP+FN}$ .Robust to class imbalance.
Segmentation Accuracy	Pixel Accuracy	PA	Measure the classification accuracy of the model for all pixels in the image,$PA=\frac{\sum_{i=1}^{n}TP_i}{\sum_{i=1}^{n}(TP_i+FP_i+FN_i)}$, Reflects the overall performance of pixel-by-pixel classification

2.5 Summary of This Chapter

This chapter systematically introduces the deep learning theoretical basis related to the detection of oil stains on the underside of EMU (Electric Multiple Unit) vehicles. Firstly, it provides an overview of the basic structure of convolutional

neural networks and the principle of transfer learning. Then, it elaborates on the core innovations of the YOLOv12 object detection algorithm, including region attention, residual efficient layer aggregation network, and architecture optimization. Next, it introduces the DeepLabV3+ semantic segmentation algorithm, focusing on dilated convolution, ASPP module, and decoder-encoder structure, as well as the lightweight backbone network MobileNetV2. Finally, it presents the evaluation metrics adopted in this paper. These theories provide a foundation for the model design, experimental analysis, and result discussion in the subsequent chapters (As shown in Figure 4).

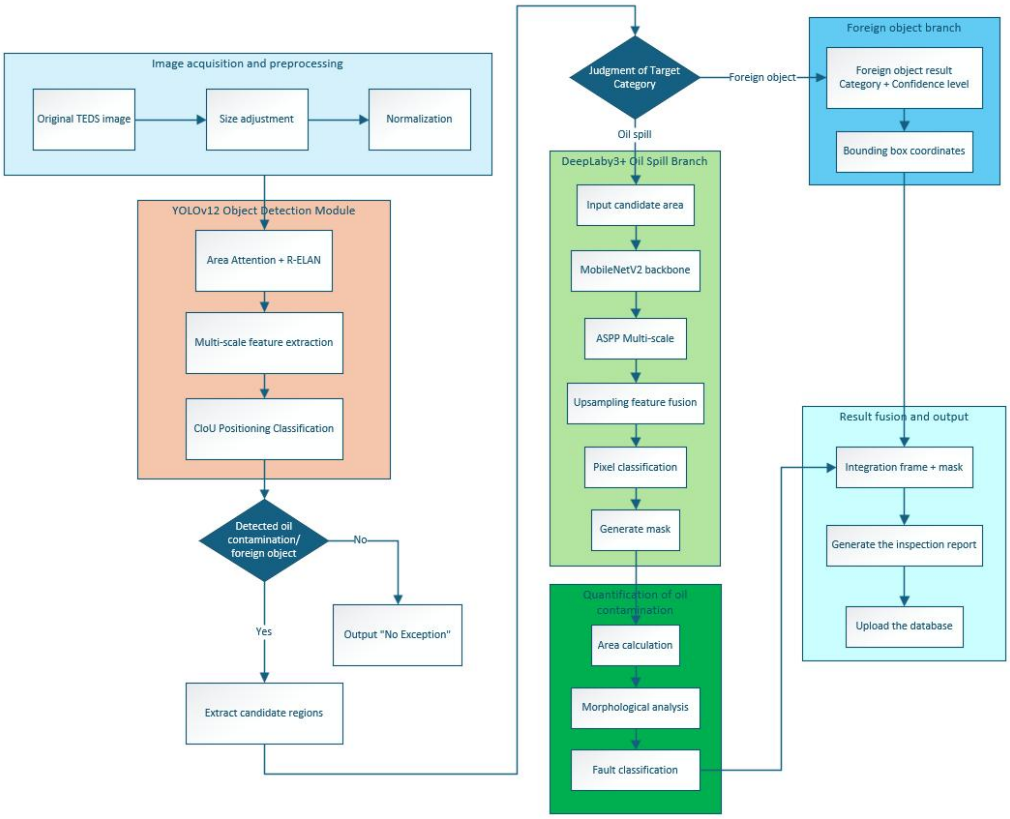


Figure 4 Flowchart of Dual-Model Collaborative Detection for Oil Stains and Foreign Objects under the EMU Carriage

3 YOLOV12 OIL SPILL TARGET DETECTION MODULE

3.1 Dataset Preparation and Enhancement Strategies

3.1.1 Construction of the original dataset

The original image data studied in this paper is derived from the real operation images collected by the TEDS system of a certain railway locomotive depot, focusing on the key parts prone to oil leakage such as gearboxes and couplings. This paper marks the clearly reflective liquid oil stains for subsequent identification. The LabelMe software is used to label the oil stain areas in the original images, with the annotation format being Pascal VOC, and then converted to YOLO format. There are a total of 356 valid images in the initial annotation results, with 294 in the training set and 62 in the validation set. The number of photos containing oil stains collected from the actual locomotive bottom is relatively small, making it difficult to complete the model training. The following figure shows some images from the original dataset (As shown in Figure 5).

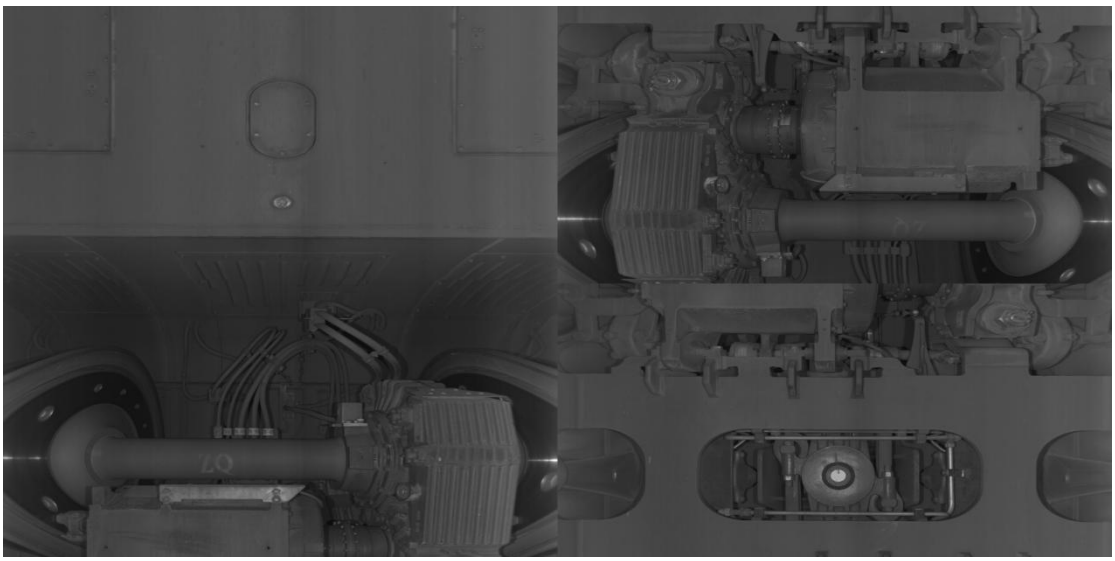


Figure 5 Original Dataset Images of the Bottom of some Bullet Trains

3.1.2 Multi-stage data augmentation strategy

To address the issue of insufficient training data, this paper designs the following multi-stage data augmentation strategy. In the basic augmentation stage, the original images are horizontally flipped, slightly adjusted in brightness, and rotated within a small range to achieve preliminary data expansion. The second stage employs random cropping, Gaussian noise addition, and blurring processing to enhance the diversity of the data. The third stage is the reinforcement augmentation stage, where a composite augmentation pipeline is constructed through a program script and by calling the Albumentations library. The augmentation in this stage expands the original dataset from multiple dimensions. Through training and fine-tuning, the specific strategy configuration for the third stage is as follows in the Table 3.

Table 3 Data Augmentation Configuration in the Intensive Enhancement Stage

Augmentation Operation	Parameter Con-	Augmentation Operation
Random Rotation 90°	—	0.6
Horizontal Flip	—	0.6
Vertical Flip	—	0.4
Transpose	—	0.4
Brightness-Contrast Adjustment	Brightness variation range: ±50%, Contrast variation range: ±50%	1.0
Hue-Saturation Adjustment	Hue: ±40◦, Saturation: ±50%, Brightness: ±50%	0.9
Affine Transformation	Zoom range: 0.5 to 1.5 times; Translation: ±30%; Rotation: ±90°; Shear: ±15°	0.9
Gaussian Blur	Kernel size: 3–9	0.4
CLAHE	Grid size: 8 × 8	0.4
Gamma Correction	Gamma range:70–130	0.4
ISONoise	Intensity range:0.1–0.3	0.3
CoarseDropout	Max holes: 8, Max hole size: 32 × 32 pixels	0.2

All enhancement operations should be synchronized to adjust the bounding box, and the minimum visibility threshold should be set to 0.3. This ensures that the target area within the enhanced annotation box can be effectively identified.
Based on the second-stage enhanced data, 8 third-stage enhanced versions are generated for each image. The final training set expands to 6,786 images, and the validation set expands to 1,821 images. Some of the effects of data augmentation are shown in the following figure. Compared with the original data, the richness of the enhanced data has been significantly improved (As shown in Figure 6).

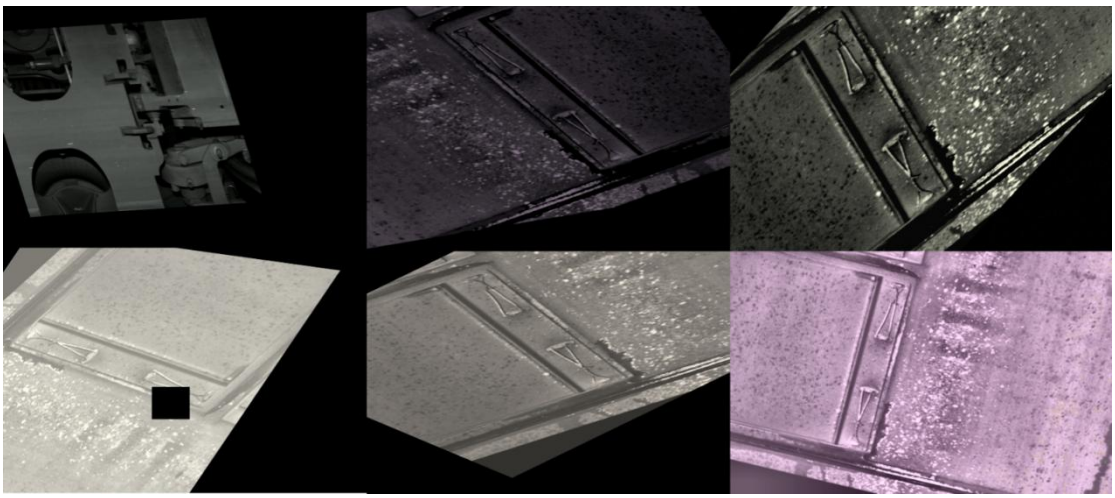


Figure 6 Images from the Enhanced Data Set of High-Speed Trains

3.2 Model Training Configuration

This paper uses YOLOv12 as the basic model for oil spill target detection. The relevant configuration environment and parameter settings are shown in Tables 4 and 5.

Table 4 YOLOv12 Training Environment Configuration

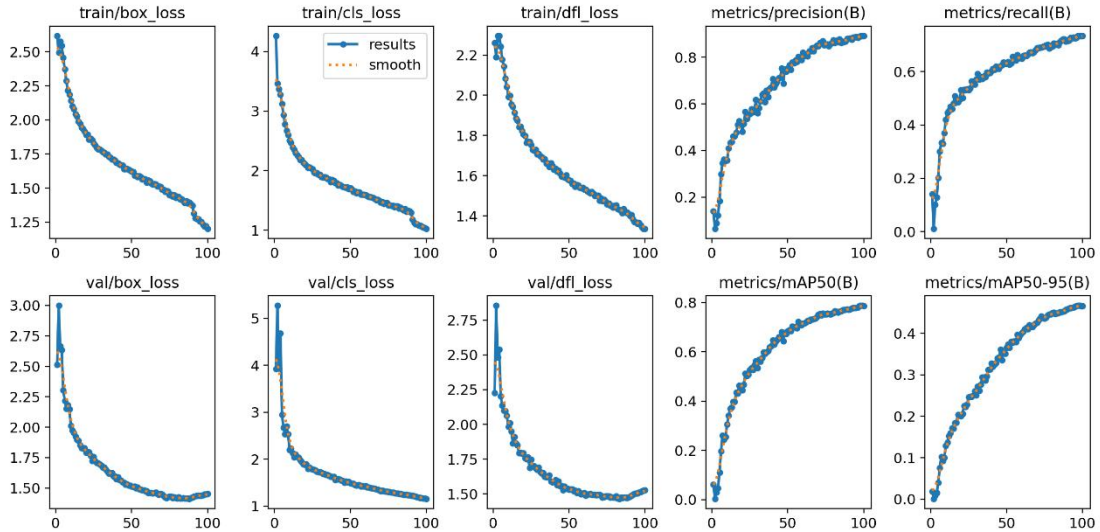
Category	Configuration Item	Parameter
Hardware Environment	Memory	128GB DDR4
Hardware Environment	GPU	NVIDIA RTX 4090D
Software Environment	Operating System	Ubuntu 20.04
Software Environment	CUDA Version	12.1
Software Environment	Python Version	3.10
Software Environment	Deep Learning Framework	PyTorch 2.1.0

Table 5 YOLOv12 Training Parameter Configuration

Parameter Category	Parameter Name	Parameter Value
Data Preprocessing	Input Image Resolution	640 × 640 pixels
Data Preprocessing	Batch Size	16
Optimization Strategy	Optimizer	SGD
Optimization Strategy	Initial Learning Rate	0.01
Optimization Strategy	Momentum	0.937
Optimization Strategy	Weight Decay	0.0005
Training Control	Number of Epochs	100

3.3 Training Results

3.3.1 Convergence of loss function



Fgure 7 The Trend Chart of Various Indicators during the Training Process of YOLOv12

According to Figure 7, in order to accurately study the changes in different indicators, the initial values during training and the approximate end values during training are summarized in the following table (see Table 6).

Table 6 Evaluation Results of Each Metric During Training

Training Metric	Approx. Initial Value	Approx. Final Value
train/box_loss	2.65	1.23
train/cls_loss	4.50	1.00
train/dfl_loss	2.25	1.35
val/box_loss	2.50	1.50
val/cls_loss	5.40	1.00
val/dfl_loss	2.25	1.56
metrics/precision(B)	0.07	0.88
metrics/recall(B)	0.06	0.70
metrics/mAP50(B)	0.00	0.80
metrics/mAP50-95(B)	0.00	0.50

The table shows that the initial values of train/box_loss and train/cls_loss are 2.65 and 2.50 respectively. They decrease rapidly with each training iteration and stabilize at around1.23 and 1.50 respectively at the end of training. There is no significant overfitting phenomenon. This indicates that the model has a good ability to locate the oil-soiled areas. The initial values of the classification loss (cls_loss) for the training set and validation set are 4.50 and 5.40 respectively. They decrease to nearly 1.00 by the end of training. The convergence speed of the classification loss is fast, indicating that the model can effectively distinguish the oil-soiled areas from the background. The initial values of the distribution focusing loss (dfl_loss) for the training set and validation set are both close to 2.25. The dfl_loss of the training set converges to around 1.35 at the end of training. The validation set value drops to around 1.56. The stable decrease of this loss indicates that the prediction accuracy of the model for the distribution of bounding boxes gradually improves. The three types of loss functions converge well during the training process. The downward trend of validation set loss is consistent with that of the training set loss. The model does not show severe overfitting and the training is sufficient and effective.
val/dfl_loss shows a slight increase at the 100th training round, while train/dfl_loss continues to decrease. This indicates that the model is in a critical state between underfitting and overfitting. Continued training may lead to overfitting. Therefore, this paper chooses to stop training at the 100th round, and the generalization effect of the model can be improved at this time.

3.3.2 Detection accuracy analysis

Figure 7 shows that the initial value of metrics/precision(B) is less than 0.20, and it increases rapidly, reaching 0.85 at the 90th round, and stabilizes at around 0.88 at the end of training. This indicates that the model's confidence in identifying the oil-soiled areas is relatively high. The initial value of metrics/recall(B) is approximately 0.06, and it converges to around 0.70. This shows that most of the real oil-soiled areas can be detected by the model. The initial value of metrics/mAP50(B) in the training is close to 0.00, and it reaches around 0.80 at the end of training. This indicates that at an IoU threshold of 0.5, the model has a high detection accuracy. The initial value of metrics/mAP50-95(B) is close to 0.00, and it reaches around 0.50 at the end of training. This change indicates that the model can maintain good detection performance under strict IoU requirements.

3.3.3 Model performance analysis

The analysis of the training process of YOLOv12 shows that the model exhibits good performance in the detection of oil-soiled categories. In the training results of the validation set, the recall rate of the oil-soiled category reaches 0.79, indicating that the model can effectively identify 79% of the oil-soiled samples, demonstrating a good ability to detect oil- soiled targets. This result verifies the effectiveness of the data augmentation strategy adopted in this paper. After multiple- stage data augmentation to expand the training samples, the model can learn the key visual features of the oil-soiled areas. Figure 8 shows the annotation results of some validation sets. Figure 9 shows the annotation results of some test sets.

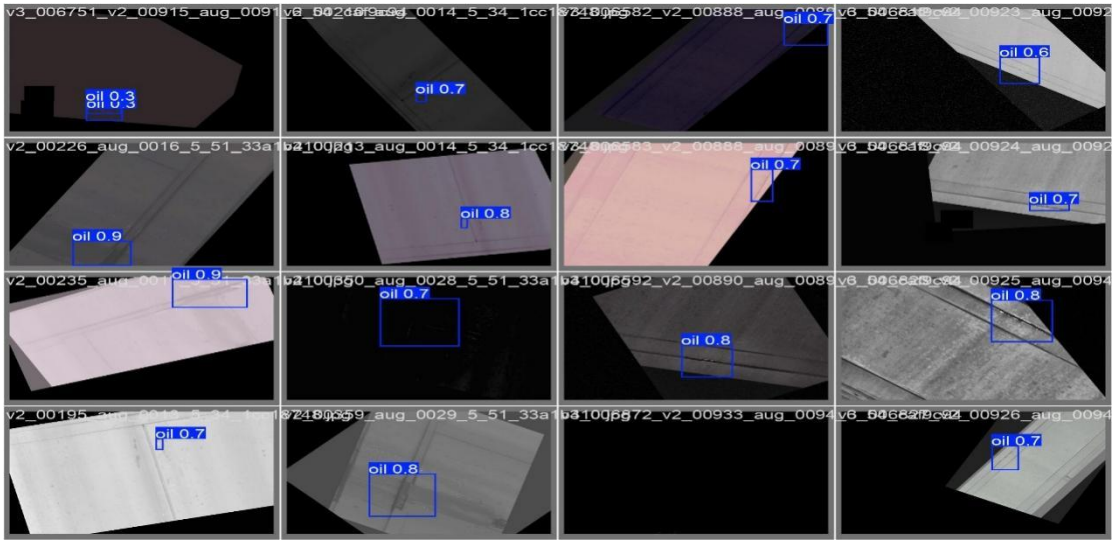


Figure 8 Validation Set Annotation Results of YOLOv12

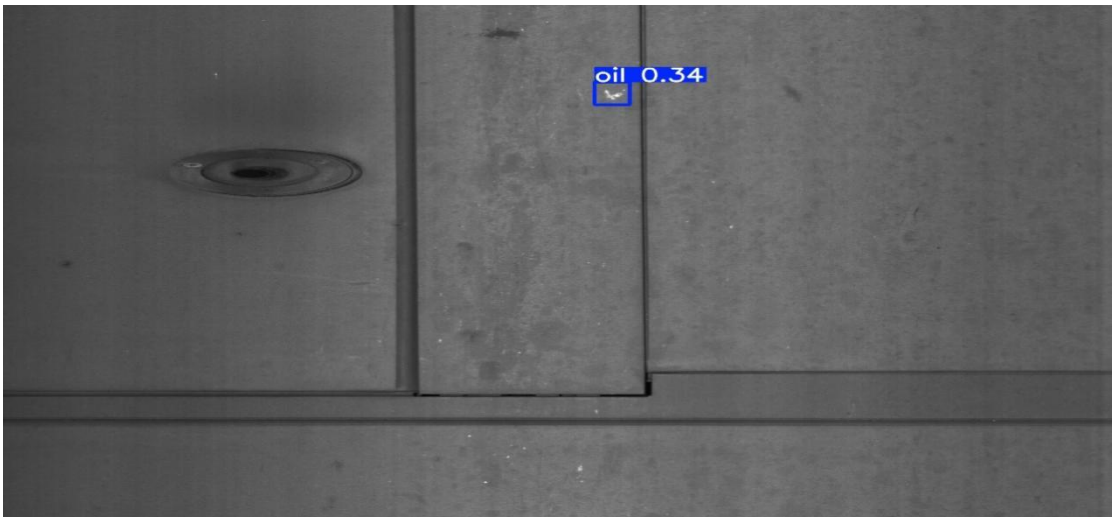


Figure 9 Validation Results of YOLOv12 Test Set Images

The evaluation results of the validation set show that this model can effectively identify and label the oil stain areas in the images that are interfered by reflections. The results of the test set also prove that the model has good generalization ability. The model can also accurately detect and locate the reflective oil stain targets that are hidden in visual features and difficult to be distinguished by the naked eye.

4 DEEPLABV3+ SEMANTIC SEGMENTATION MODULE

Although the recall rate of the YOLOv12 oil contamination detection model on the validation set is relatively high, the visual features of the oil contamination area are highly similar to some backgrounds, such as rust marks, shadows, and residual oil stains. It is difficult for YOLOv12 to distinguish these features through end-to-end detection alone. Therefore, this paper introduces the DeepLabV3+ semantic segmentation module and constructs a joint detection framework of "target detection + semantic segmentation". This framework uses YOLOv12 to quickly locate and preliminarily detect the key component areas in the image, and then inputs the detection results into the DeepLabV3+ network for pixel-level semantic segmentation, to refine the oil contamination boundaries.

4.1 Dataset Preparation and Format Conversion

4.1.1Data source

The training data for the DeepLabV3+ stage comes from the training set and test set that have been labeled after the training of YOLOv12 is completed. This dataset contains 6,786 training images and 1,821 validation images, and all images have the boundary box information of the oil contamination area labeled. To meet the training requirements of the semantic segmentation task, the boundary box labels need to be converted into pixel-level mask labels.

4.1.2Conversion from YOLO format to mask format

This paper designs a format conversion script to convert the YOLO format boundary box labels into the single-channel grayscale masks required by DeepLabV3+. The script creates a blank mask matrix of the same size as the original image,

then parses the normalized bounding box coordinates in the YOLO annotation file, and inversely normalizes them to pixel-level coordinates. Then, draw filled rectangles in the mask matrix to set the pixel values of the oil contamination area to 255. The converted mask file is saved in PNG format to ensure that its name is consistent with the original image and the size is the same.

4.2 Model Training Configuration

This paper uses DeepLabV3+ as the base model for semantic segmentation, and its parameter settings are as follows.

Table 7 Training Parameter Configuration for DeepLabV3+ Model

Parameter Name	Parameter Value
Input Image Size	512 × 512 pixels
Batch Size	8
Number of Epochs	100
Learning Rate	0.0001
Learning Rate Scheduling	Polynomial Decay Optimizer
Momentum	0.9
Weight Decay	0.0005
Backbone Network	MobileNetV2 (ImageNet pretrained)
Input Image Size	512 × 512 pixels

In the training of the DeepLabV3+ model, this paper adopts a learning rate of 0.0001. This value can retain the general features of the pre-trained model during fine-tuning and avoid excessive parameter updates that would damage the already acquired effective representations. The oil stain segmentation task itself has a serious class imbalance problem. The proportion of oil stain pixels in the entire image is low. This learning rate can slow down the model's overfitting to the background category, stabilize the training process, and improve the final convergence effect.

4.3 Training Results

4.3.1 Distribution analysis of segmentation indicators on the validation set

The average IoU of the DeepLabV3+ model on the vali- dation set reached 0.4535, and the average Dice coefficient was 0.4872. These quantitative indicators indicate that the overlap between the predicted oil stain area and the real mask area is approximately 45%. This verifies the feasibility and effectiveness of DeepLabV3+ in the oil stain segmentation task of EMU TEDS images.

During the overall training process, by statistically analyzing the distribution of evaluation indicators for different samples, the training situation of the model can be intuitively obtained. The results are shown in Figure 10.

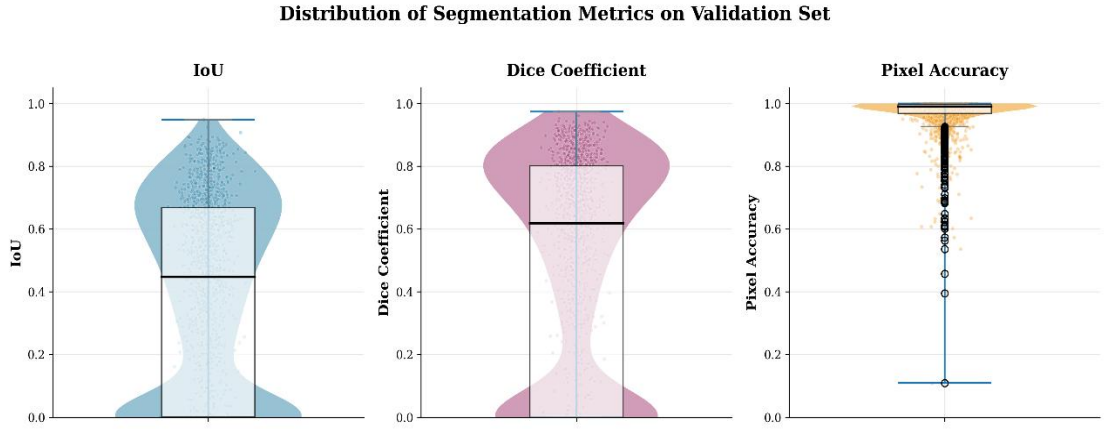


Figure 10 Distribution of Segmentation Metrics on Validation Set

According to Figure 10, the IoU values of most of the validation samples are concentrated in the range of 0.5 to 0.8, the Dice values are concentrated in the range of 0.6 to 0.9, and the pixel accuracy rates are concentrated in the range of 0.9 to 1.0. Approximately 75% of the samples have IoU values higher than 0.35, and the model's segmentation performance on most images is relatively stable. There is no extreme long-tail phenomenon in the statistical distribution. The model performs relatively stably when dealing with image scale changes, changes in lighting conditions, and different oil stain forms. There are peaks at the 0 value for IoU and Dice values. The authors believe that this phenomenon is caused by samples with extremely small oil stain areas, highly blurred boundaries, or strong reflective interference. However, for large areas of oil stains, the overall segmentation effect of the model is within an acceptable range. The pixel accuracy rate remains at a relatively high level overall. The model performs well in the pixel classification task and can accurately identify the vast majority of background pixels.

4.3.2 The validation set matrix results

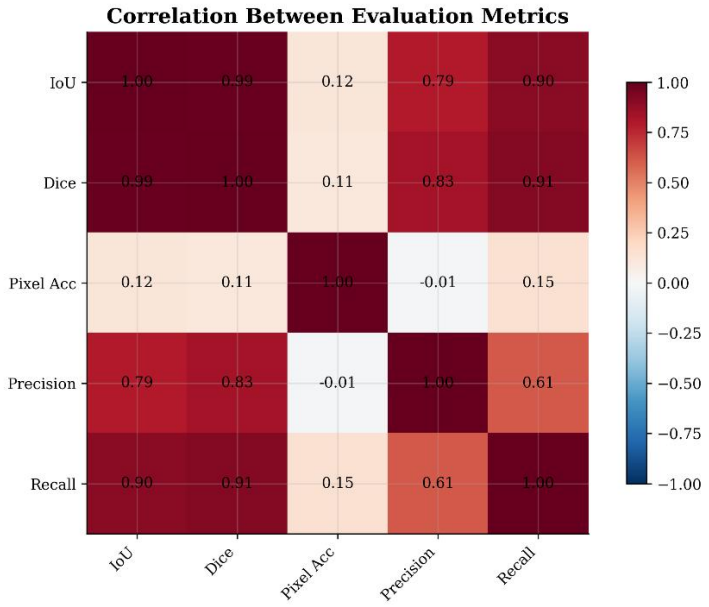


Figure 11 Correlation between Evaluation Metrics

Figure 11's correlation matrix of evaluation indicators shows that the IoU and Dice coefficient have a very strong correlation, with a correlation coefficient of 0.99. Both are highly consistent in evaluating the overlap degree of segmentation, verifying the reliability of the evaluation system. Additionally, the correlation coefficient between recall rate and IoU is 0.90, indicating that the segmentation overlap degree of the model is mainly influenced by the recall rate. The absolute values of the correlation coefficients of pixel accuracy rate and other indicators are lower than 0.15. The authors believe this is due to the fact that the background pixel accounts for more than 96%. Pixel accuracy rate is difficult to sensitively reflect the segmentation quality of the oil stain area. Therefore, in tasks with extremely imbalanced categories, IoU is more suitable as the core evaluation indicator than pixel accuracy rate. In conclusion, the model can provide effective pixel-level mask constraints for the YOLOv12 detection results, filtering out background false-positive areas.

4.4 Test Set Detection Results

This paper evaluated the optimal model that was trained on the test set. Figure 12 shows the segmentation result of a typical sample in the test set, with an IoU of 0.6250.

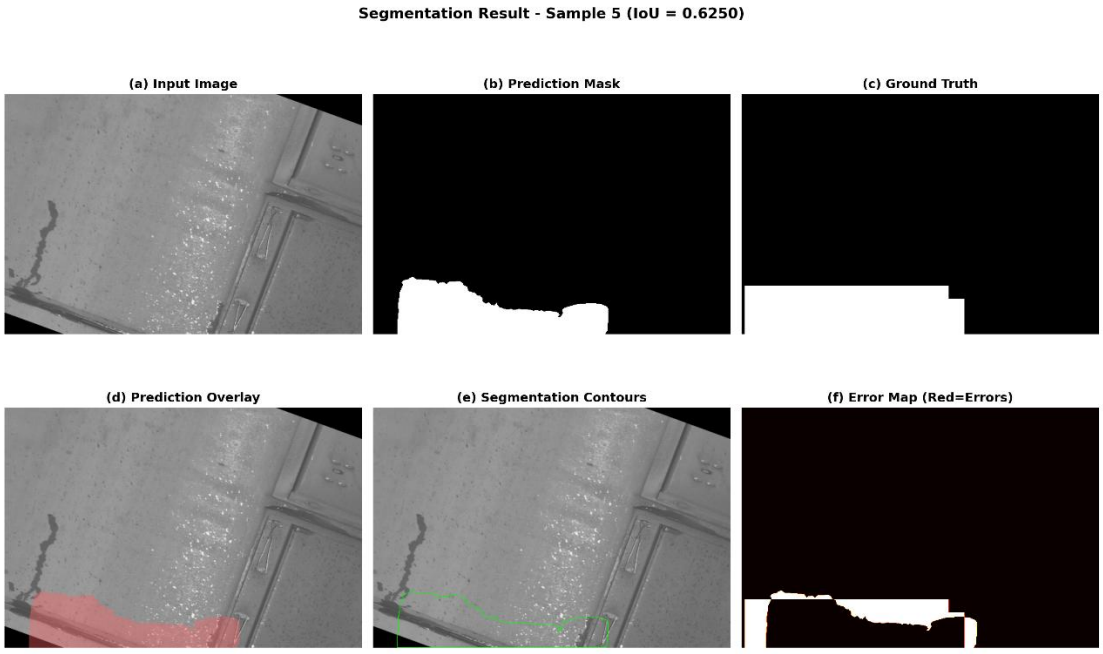


Figure 12 Semantic Segmentation Results of the DeepLabV3+ Test Set

The results in Figure 12 indicate that there is a significant surface reflection phenomenon in the oil stain area of the original input image. The predicted mask by the model basically completely outlines the main shape of the reflective oil

stain area, and there are no obvious breaks or missed detections even in the high-reflective areas. From sub-figure (d), the predicted oil stain area and the actual oil stain position are highly overlapping, and the reflective area does not cause significant interference to the positioning. From the error graph sub-figure (f), it can be seen that the differences between the prediction and the annotation mainly concentrate on the edge parts of the oil stain area. There are oil stain dry-up traces in this image, and the model did not classify them as oil stains. In the actual operation of the EMU, dry-up oil stain traces are usually historical dirt stains. Reflective oil stains are generally recent leakage-generated active oil stains and need to be promptly inspected and handled. The recognition characteristics of the model are in line with reality and can effectively distinguish between the two traces, avoiding invalid alarms for dry-up traces.

5 CONCLUSION

This paper focuses on the oil stain defect detection task in the TEDS images of EMU (Electric Multiple Unit). Two deep learning models, YOLOv12 and DeepLabV3+, are selected to achieve rapid positioning and fine segmentation of the oil stain areas. To address the issue of insufficient oil stain samples, a multi-stage data augmentation strategy is adopted, expanding the training set from 294 images to 6,786 images. The image enhancement strategy in this paper enables the target recognition accuracy of YOLOv12 to reach 0.88 and the recall rate to 0.70. The DeepLabV3+ model introduced in this paper completes the fine extraction of the oil stain area through pixel-level segmentation. The model achieves an average IoU of 0.4535 on the validation set. The model can effectively identify reflective oil stains that are difficult for humans to distinguish and reduce false alarms for dried traces, demonstrating engineering practicality. The joint training model can well identify and outline the actual oil stains in the EMU images in the test set. The detection framework constructed in this paper combines two deep learning technologies of object detection and semantic segmentation, which can meet the requirements of oil stain defect detection in the complex operating environment of EMUs. This framework provides a reference path for the research on intelligent image detection related technologies in the railway field. Further optimization can be carried out for the model's insufficient detection accuracy for small-sized oil stains and high false detection rate in complex backgrounds.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

REFERENCES

[1] Fan L, Li D Y, Liu B, et al. Research on core technology of TEDS image aided recognition for multiple units based on improved SOLOv2 network. Railway Locomotive & Car, 2025, 45(2): 1-11.
[2] Luo H. Research on intelligent recognition technology of fault image of high-speed EMU components. Beijing: Beijing Jiaotong University, 2022.
[3] Yang Z H. Research on catenary equipment defect identification based on deep learning. Beijing: China Academy of Railway Sciences, 2022.
[4] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016: 770-778.
[5] Yin T Q, Zhou X H, Song L S. Extracting water body information from high-resolution remote sensing images by DeepLab V3+. Zhejiang Hydrotechnics, 2024, 52(6): 88-93.
[6] Zhu Y J, Cai G Q, Han J, et al. Small object detection in complex open-pit mine backgrounds based on improved YOLOv11. Industry and Mine Automation, 2025, 51(4): 93-99.
[7] Transportation Technology Center, Inc. Machine vision for railway inspection. Pueblo: TTCI, 2021.
[8] Falamarzi A, Moridpour S, Nazem M. A review on rail defect detection techniques. Australian Journal of Structural Engineering, 2019, 20(3): 201-213.
[9] Ren S, He K, Girshick R, et al. Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137-1149.
[10] Bochkovskiy A, Wang C Y, Liao H Y M. YOLOv4: Optimal speed and accuracy of object detection. arXiv preprint, arXiv:2004.10934, 2020.
[11] Wang A, Chen H, Liu L, et al. YOLOv10: Real-time end-to-end object detection. arXiv preprint, arXiv:2405.14458, 2024.
[12] Jocher G. YOLOv11. https://github.com/ultralytics/ultralytics, 2024.
[13] Tian Y, Ye Q, Doermann D. YOLOv12: Attention-centric real-time object detectors. arXiv preprint, arXiv:2502.12524, 2025.
[14] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Boston: IEEE, 2015: 3431-3440.
[15] Ronneberger O, Fischer P, Brox T. U-Net: Convolutional networks for biomedical image segmentation//International Conference on Medical Image Computing and Computer-Assisted Intervention. Munich: Springer, 2015: 234-241.

[16] Chen L C, Papandreou G, Kokkinos I, et al. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 40(4): 834-848.
[17] Chen L C, Zhu Y, Papandreou G, et al. Encoder-decoder with atrous separable convolution for semantic image segmentation//Proceedings of the European Conference on Computer Vision. Munich: Springer, 2018: 801-818
[18] Heckhel W, Helali A. Early detection and classification of Alzheimer's disease through data fusion of MRI and DTI images using the YOLOv11 neural network. Frontiers in Neuroscience, 2025, 19: 1554015.
[19] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature, 2015, 521(7553): 436-444.
[20] Pan S J, Yang Q. A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering, 2010, 22(10): 1345-1359.
[21] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016: 779- 788.
[22] Sandler M, Howard A, Zhu M, et al. MobileNetV2: Inverted residuals and linear bottlenecks//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 4510-4520.