AENet: Attention Enhancement Network for Industrial Defect Detection in Complex and Sensitive Scenarios

Yi Wan
Chongqing University of Posts and Telecommunications
Lingjie Yi
Chongqing University of Posts and Telecommunications
Bo Jiang
Chongqing University of Posts and Telecommunications
Junfan Chen
Chongqing University of Posts and Telecommunications
Yi Jiang
Chongqing University of Posts and Telecommunications
Xianzhong Xie
xiexzh@cqupt.edu.cn

Chongqing University of Posts and Telecommunications

Research Article

Keywords: Encoder-decoder model, Attention mechanism, Attention fusion, Industrial defect detection

Posted Date: August 31st, 2023

DOI: https://doi.org/10.21203/rs.3.rs-3305080/v1

Additional Declarations: No competing interests reported.

Version of Record: A version of this preprint was published at The Journal of Supercomputing on February 3rd, 2024. See the published version at https://doi.org/10.1007/s11227-024-05898-0.

AENet: Attention Enhancement Network for Industrial Defect Detection in Complex and Sensitive Scenarios

Yi Wan[1], Lingjie Yi[1], Bo Jiang[1], Junfan Chen[1], Yi Jiang[1], Xianzhong Xie[1*]

[1*]College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China.

*Corresponding author(s). E-mail(s): xiexzh@cqupt.edu.cn;
Contributing authors: d200201017@stu.cqupt.edu.cn;
ylj52237@foxmail.com; s210131076@stu.cqupt.edu.cn;
chenjf584@outlook.com; jiangyi@cqupt.edu.cn;

Abstract

Conventional image processing and machine learning based on handcrafted features struggle to meet the real-time and high-accuracy requirements for industrial defect detection in complex, sensitive, and dynamic environments. To address this issue, this paper proposes AENet, a novel real-time defect detection network based on an encoder-decoder model, which achieves high detection accuracy and efficiency while demonstrating good convergence and generalization. Firstly, A spatial channel attention (SCA) module in the encoding network is designed to integrate spatial attention and channel attention using a multi-head 3D self-attention mechanism. This improves parallelism and detection efficiency. Secondly, the decoding network of AENet incorporates the Cross-Level Attention Fusion (CLAF) module, which fuses input features from different layers. Combined with multi-level upsampling, this enhances the representation of defect features. Furthermore, we insert a simplified aggregator into the encoder-decoder network of AENet to extract feature information at different scales with low computational cost. This aggregation process aids in training and inference on industrial defect datasets by incorporating contextual information. Extensive experimental results demonstrate that AENet outperforms other segmentation models in accomplishing defect recognition and segmentation in challenging optical environments. It exhibits a faster convergence than other networks and a balance between accuracy and speed. It achieves a recognition precision of over

96% for almost all types of defects in the actual industrial environment on the NVIDIA Tesla V100 GPU.

Keywords: Encoder-decoder model, Attention mechanism, Attention fusion, Industrial defect detection

1 Introduction

Defect detection technology has become an important aspect in the manufacturing industry. Surface defects are inevitable during the product processing. They have a detrimental impact on the appearance quality and service life of industrial products. Therefore, detecting surface defects is an essential part of product quality control. Conventional surface defect detection often require a significant amount of manpower and resources. Besides, it usually suffers from issues such as low detection efficiency and poor accuracy. In recent years, machine vision-based detection methods have gradually replaced manual labor due to their advantages of high precision, real-time capabilities, and strong objectivity. This has become a developmental trend in surface defect detection technology [1–5]. Machine vision-based detection methods are now widely applied in various fields, including industrial production, medicine, and military security.

In the actual industrial environment, compared to the surfaces of smooth materials such as Liquid Crystal Display (LCD), Polypropylene (PP), and precision optical components, photographs of metal surfaces often exhibit phenomena such as uneven illumination, strong reflections, and significant background noise [3, 6]. Additionally, advanced defect assessment standards not only require determining the presence of surface defects but also obtaining precise measurements of their size and type. The random and unpredictable shapes of metal surface defects make it impractical to apply visual-semantic methods used in optical character recognition (OCR) [7] or generate reference images through conditional generative models [8]. This poses significant challenges for real-time detection of metal surface defects.

Currently, machine vision techniques for surface defect detection mainly rely on conventional image processing and machine learning. Conventional image processing methods detect and segment defects based on the principle of local abnormality reflection. They can be further categorized into structural methods, threshold methods, spectral methods, and model-based methods [9–11]. These methods require setting multiple thresholds or boundary conditions in the algorithms to identify various types of defects. However, these thresholds or boundary conditions are highly sensitive to factors such as lighting conditions and background colors, lacking adaptability to real-world detection environments. In different detection environments, they need adjustment, and the detection algorithms may even require redesigning. The commonly used machine learning methods are mostly based on handcrafted features or shallow learning techniques. Learning methods based on handcrafted features necessitate manual design and annotation of features. Shallow learning techniques may lack

sufficient discriminative power for complex environmental features, leading to underfitting and overfitting issues when dealing with a large amount of complex data. Both of these approaches are often tailored to specific scenarios, lacking adaptability, real-time capability, and robustness in real-world metal defect detection environments.

To address the above issues, this paper proposes an industrial complex surface defect detection network based on deep neural networks: AENet (Attention-based Encoder-Decoder Network). AENet adopts an encoder-decoder architecture and integrates spatial attention, channel attention, cross-level attention mechanisms, along with a simplified aggregator. This design achieves higher detection accuracy and efficiency while demonstrating good convergence and generalization capabilities. The main contributions of this paper are as follows:

1. AENet, a real-time network for metal surface defect edge enhancement and detection, is designed based on an encoder-decoder structure. In the encoder, the edge attention module (SCA) is inserted, which incorporates a multi-head spatial attention mechanism that integrates local and global attention in multiple dimensions. It increases the parallelism and detection efficiency.
2. AENet incorporates a simple aggregator, MPP, between the encoder and decoder to aggregate global information at a lower computational cost. It acquires image information from different levels to ensure high inference accuracy.
3. The decoder includes the Cross-Level Attention Fusion Module (CLAF), which utilizes spatial attention to achieve weighted fusion of cross-level features and combines multi-level upsampling to enhance the representation of defect features.
4. Inference results on real industrial datasets demonstrate that AENet achieves the learning process more efficiently. AENet performs well in aluminum surface defect detection and segmentation under industrial environmental conditions, achieving a recognition precision of over 96%. It ensures both inference segmentation accuracy and exhibits excellent real-time capability and robustness. On the public datasets Vehicle component defect and CrackForest, it achieves mIoU (mean Intersection over Union) of 67.56% and 82.00%, respectively.

2 Related work

2.1 Encoder-decoder based work

As a neural network model, the encoder-decoder architecture holds great potential in machine vision for metal defect detection. Its image feature extraction and encoding-decoding characteristics make it capable of achieving high detection accuracy and reliability, making it suitable for metal surface defect detection.

Boukdir et al. [12] proposed a text generation method based on an encoder-decoder model. The encoder and decoder utilize Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) structures, respectively. Experimental results on a sign language dictionary dataset demonstrated that this model can generate highly clear text and exhibits high accuracy and fluency. Lin et al. [13] introduced a wafer pattern counting, detection, and classification method based on an encoder-decoder Convolutional Neural Network (CNN) architecture. The encoder network extracts

features from the wafer patterns, and the decoder network performs counting, detection, and classification. Wang et al. [14] proposed a high-precision unmanned aerial vehicle (UAV) localization algorithm based on an encoder-decoder model, utilizing point cloud super-resolution and image semantic segmentation as auxiliary techniques. Jie et al. [15] presented a novel Convolutional Neural Network (CNN) model for object detection, which combines Atrous Spatial Pyramid Pooling (ASPP) with an encoder-decoder structure.

The aforementioned works all utilize encoder-decoder models for image feature extraction, applying different neural network models to various detection environments, achieving high detection accuracy and efficiency. Encoder-decoder models have the capability to adaptively extract features from a large amount of data and exhibit strong generalization ability on new data. However, they are susceptible to interference from detection environments, particularly in metal industrial production lines where the environmental conditions are generally poor.

2.2 Attention-based work

The attention mechanism refers to a data-based transfer learning approach used in deep learning models, which distinguishes different parts of the learned data during training. Its function is to assign different weights to different parts of the input data, allowing the deep learning model to focus more on important regions while reducing interference from irrelevant information.

In the fields of natural language processing and image processing, the attention mechanism has achieved significant performance improvements in modeling sequence data and classifying data. Wang et al. [16] proposed an object detection algorithm based on cascaded feature fusion and multi-level self-attention mechanism. The cascaded feature fusion method is used to extract features with different abstraction levels, and the multi-level self-attention mechanism enhances the representation ability of feature expressions. Togo et al. [17] also introduced a multi-scale subway tunnel image defect detection algorithm based on spatial attention mechanism. Peng et al. [18] presented a novel textile defect detection network based on attention mechanism and multi-task fusion.

The above works all introduce a single local attention mechanism, enabling the model to focus more on local regions, thus improving detection accuracy. However, single attention mechanisms are often designed based on specific scenarios and may not adapt well to different deep learning tasks and application scenarios.

2.3 Attention fusion

Attention fusion combines multiple attention mechanisms to achieve comprehensive modeling and exploration of data across different dimensions and features. For various deep learning tasks and application scenarios, appropriate attention fusion methods can significantly enhance the performance of the model.

Some researchers have proposed different attention fusion methods, such as CBAM (Channel and Spatial Attention Module) [19], SENet (Squeeze and Excitation Networks) [20], and GLAM (Global Local Attention Model) [21]. In the process of

implementing attention fusion, conventional methods concatenate or parallel different attention mechanisms, such as SAGAN (Self-Attention Generative Adversarial Networks) [22] and CBAM-ResNet [23]. Furthermore, some researchers have developed iterative and recursive-based attention fusion methods, such as BAN (Bi-Directional Attention Network) [24] and SAN (Stacked Attention Networks) [25]. These methods enhance the accuracy and robustness of the fused attention mechanisms but introduce complex network structures, which may affect the real-time capability of inference tasks.

To address real-time industrial defect detection tasks in complex production environments, we propose a deep neural network based on an encoder-decoder structure that integrates multiple attention mechanisms. This approach achieves high detection accuracy and efficiency while demonstrating excellent convergence and generalization capabilities.

3 Algorithm design

This section firstly provides a detailed introduction to the proposed network architecture of AENet. It then introduces the attention-enhanced mechanism for metal surface defect detection in the encoder called the Spatial Channel Attention Enhancement module (SCA). Next, the Simplified Multi-Scale Pooling module (MPP) is presented, which enhances the model's receptive field. Finally, the Cross-Level Attention Fusion module (CLAF) in the decoder is introduced.

3.1 Network architecture

The main model of AENet proposed in this paper is shown in Figure 1. AENet primarily consists of three major modules: encoder, aggregator, and decoder.

Given an input image with a size of $C \times H \times W$, AENet utilizes a lightweight network called STDCNet [26] as the backbone encoder to extract hierarchical features. STDCNet consists of 5 stages. Stage 1 and 2 are simple convolution-batch normalization-pooling implementations. Stage 3/4/5 are more complicated structures. Each of them is separated into several blocks. Among them, downsampling is implemented in the first block, in which the feature size compression can be achieved. The position encoding module is commonly used in semantic recognition networks to identify the meaning of longer words. Here in AENet, we use position encoding to classify defects of different shapes. Additionally, the SCA module is introduced in the encoding structure, which incorporates an edge attention enhancement mechanism. This allows the network to not only learn global features but also achieve precise recognition of local regions in the image, enhancing robustness in defect recognition under poor lighting conditions. The SCA module is added as branches to the downsampling blocks in stage 3/4/5.

AENet employs the aggregator MPP to extract information from different scales in the input feature maps. MPP takes the output features of the encoder as input and generates a globally enriched feature map for the decoder. The decoder of AENet is designed using the FLD structure [27], consisting of three upsampling layers and two CLAF stacks. The upsampling layers are used to reduce the channel dimension and

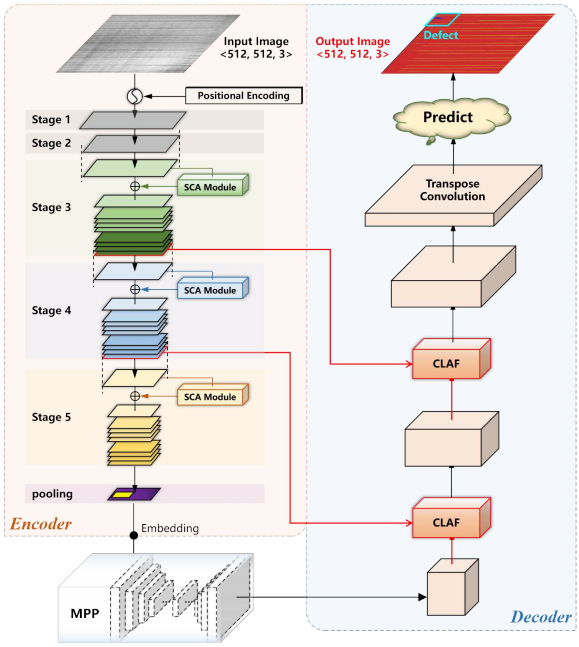

Fig. 1 Architecture of AENet

restore the image size. CLAF is employed to fuse low-level and high-level features for better restoration of defect features. In the segmentation head, the sample features with 1/8th of the original size are dimensionally reduced to reduce the computational cost when preserving core features. Subsequently, the feature size is expanded to match the input image through upsampling for pixel-level classification. The design of channel of features in the decoder achieves a balance in computational cost with the encoder.

3.2 Attention-enhancement mechanism

In this paper, we insert the SCA module as a branch in the backbone network of the AENet encoder to enhance the representation of both global and local features, accomplishing the attention-enhancement mechanism. The attention enhancement is mainly achieved through the fusion of spatial attention and channel attention, which operate in different dimensions. The detailed structure of the SCA module is shown in Figure 2. The input features of the SCA module can be described as Equation (1).

$$X = \{X_i\}_{i=1}^{H \times W} \in R^{C \times H \times W} (C = d_{model}) \tag{1}$$

Here, C, W, and H represent the number of feature channels, the width, and the height of the feature map, respectively. d_{model} represents the dimensionality of the encoder's output features. The SCA module mainly performs four operations on the input feature map x, including (a) feature acquisition across different channels, (b) updating the feature map based on spatial dependencies, (c) constructing multidimensional weights fused with attention, and (d) adding broadcast elements for feature

fusion. The logical algorithm can be represented by Equation (2).

$$y_i = \text{Concat}(x_i, w_{v_2} \cdot \text{RELU}(\text{LN}(w_{v_1} \cdot \sum_{\forall j} \left(\cdot \sum_{\forall m} e^{W_k x_m} \right)^{-1} \cdot e^{W_k x_j} \cdot x_j))) \quad (2)$$

x and y represent the inputs and outputs of the global up-down module, respectively, with the same feature size. i is the index of the query position. j and m enumerate the positions of all pixels. w_{v_1}, w_{v_2}, and w_k represent linear transformations achieved through 1×1 convolutions. LN(·) denotes the normalization layer [28]. $\delta(\cdot) = w_{v_2} \cdot \text{RELU}(\text{LN}(w_{v_1} \cdot \sigma(\cdot)))$ represents the calm transformation, aiming to capture dependencies between channels and pixels. $\sigma(\cdot)$ represents the generated multidimensional weights fused with dual attention. The Concat operation concatenates the downsampled features from the encoding network with the attention-enhanced features in terms of channels. From this, we can obtain a new feature map.

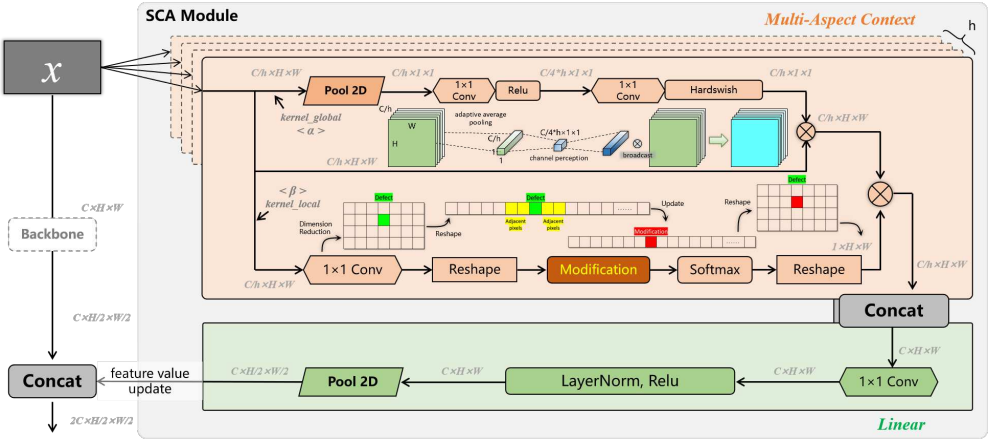

Fig. 2 Design of SCA module

The SCA module adopts a multi-head attention design to enhance attention, where the number of attention heads is h, corresponding to the different Multi-Aspect Contexts in Figure 2. Each individual Multi-Aspect Context is further divided into two branches: channel attention and spatial attention. The size information of each layer in SCA is shown in Table 1.

For the channel attention branch, the feature map of size $C/h \times H \times W$ undergoes size compression through an adaptive average pooling module, resulting in a compressed map of size $C/h \times 1 \times 1$. Next, the features are processed through two consecutive convolutional layers with activation to perceive channel information. Here we maintain the size of the output features as $C/h \times 1 \times 1$. At this stage, feature values distribution that is only related to the channel can be obtained. Then, we broadcast the size $C/h \times 1 \times 1$ feature map onto the input feature map of size $C/h \times H \times W$ and

Table 1 Size information of each node in SCA

Operation (Channel/Spatial)	Features Size		Kernel Size	
Global	$<C/h,H,W>$[1]		$C/h \times H \times W$	
Pool/Conv	$<C/h,1,1>$	$<1,H,W>$	\	$1\times1\times1$
Conv+Relu/Reshape	$<C/(4h),1,1>$	$<1,1,H\times W>$	$C/(4h)\times1\times1$	\
Conv+Hardswish/ Modification+Softmax	$<C/h,1,1>$	$<1,1,H\times W>$	$C/h\times1\times1$	\
Reshape/Reshape	$<C/h,H,W>$	$<1,H,W>$	\	\
A×B	$<C/h,H,W>$		\	\
Concat	$<C,H,W>$		\	
Conv	$<C,H,W>$		$C\times1\times1$	
Layernormal+Relu	$<C,H,W>$		\	
Pool	$<C,H/2,W/2>$		\	

perform element-wise multiplication. Finally, we obtain the feature map with channel attention, which has the same size as the input feature of the SCA module.

For the spatial attention branch, we first use a standard convolution with a kernel size of 1×1×1 to reduce the dimensionality of the image. This can result in a feature map of size $1\times H\times W$ for further processing. Next, we reshape the feature map to lower the feature dimension for ease of computation. The reshaped feature map size becomes $1\times H\,W$. Then, the Modification and Softmax module of SCA utilizes Algorithm 1 to update the feature values. In the process, each feature value is able to incorporate the features of its surrounding pixels. In regions with abrupt grayscale changes, the output features, after the completion of Algorithm 1, can more sensitively reflect the spatial grayscale differences in that area. The spatial attention branch eventually restores the updated features to a size of $1\times H\times W$. By multiplying the results of the two branches, the three-dimensional features are combined with dual attention.

Algorithm 1: Spacial Modification.
temp_avg[ ] and temp_max[ ] are registers for local averaging and maximization.
k is the size of local receptive field.
L = length(HW).

Data: InputFeatures [1*HW]
Result: OutputFeatures [1*HW]

```
for i in range(L) do
    start = max(0, i - k);
    end = min(n, i + k + 1);
    avg = sum(InputFeatures [start: end])/(end - start);
    temp_avg[i] = avg;
    temp_max[i] = max(InputFeatures[start: end]);
    OutputFeatures[i] = temp_max[i] - temp_avg[i];
end
```

Two branches each have their own global weights, defined as α and β. The significance of global weights lies in the fact that the grayscale features of the image vary for different attention heads, leading to different requirements for different attention

mechanisms. Equation (3) represents the mathematical expression of global weights. α is related to the feature values of each channel. By linearly transforming the feature values of each channel after the average pooling module, the weight values for each channel are obtained. β is related to the feature values of different pixels within the same channel. By linearly transforming the feature values of each channel after dimension reduction, the weight values for different rows and columns are obtained. Here, d_h is a scaling factor used to counterbalance the impact of different variances in SCA. It can be computed as $d_h = d_{model}/h$. The role of Softmax is to capture the proportion relationships between different channels and between different pixels within the same channel, and output the weight matrices with dimensions of $C/h \times 1 \times 1$ and $1 \times H \times W$.

$$\begin{aligned} \alpha &= \text{Softmax}(\frac{W_{\alpha 1} x_1}{\sqrt{d_h}}, \frac{W_{\alpha 1} x_1}{\sqrt{d_h}}, \frac{W_{\alpha 1} x_1}{\sqrt{d_h}}) \\ \beta &= \text{Softmax}(\frac{W_{\beta 1} x_1}{\sqrt{H \times W}}, \frac{W_{\beta 1} x_1}{\sqrt{H \times W}}, \frac{W_{\beta 1} x_1}{\sqrt{H \times W}}) \end{aligned} \tag{3}$$

Linear is designed for further extract features and perform downsampling on the attention-enhanced results. The SCA module concatenates the updated feature data with the downsampled feature maps from the encoding module to achieve channel expansion. Through the attention-enhancement mechanism, the feature maps at each stage undergo feature updates, enhancing the feature extraction capability of the encoder.

3.3 Simplified aggregator

The aggregator proposed in this paper is a simplified Multiscale Pyramid Pooling module (MPP), as shown in Figure 3. The design idea is inspired by the human visual perception mechanism. When observing a scene, the human eye typically simultaneously attends to different regions of the scene, such as foreground, background, and peripheral vision. These regions often exhibit significant scale differences. Therefore, to comprehensively understand the scene, it is necessary to consider information from multiple scales simultaneously. However, in contrast to semantic recognition, defect detection often focuses on extremely small features, requiring feature extraction at smaller scales.

The main function of the pyramid pooling is to extract information at different scales from the input feature map. MPP first utilizes the pyramid pooling module to integrate the input features. MPP is designed with three global average pooling operations, with pooling window sizes of 2×2, 4×4, and 8×8. The 2×2 pooling divides the input feature map into four rectangular regions and performs pooling operations on each region, preserving local detailed information. Similarly, the 4×4 and 8×8 poolings divide the input feature map into 16 and 64 rectangular regions, respectively, to capture finer-grained local information. The pooled results are then subjected to convolution and upsampling operations. In the convolution operation, the same convolutional kernel is applied to extract feature information, with a kernel size of 1×1. Finally, the feature vectors at different scales are concatenated and further processed with convolution to generate refined features.

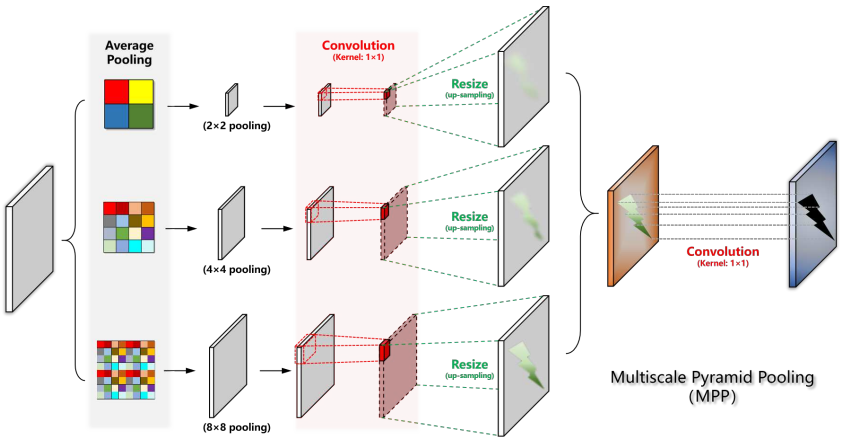

Fig. 3 Design of multiscale pyramid pooling module

Compared to the conventional pyramid pooling module, MPP in this paper removes the 1×1 pooling and replaces it with an 8×8 pooling to extract more pixel features within smaller rectangular regions, thus improving accuracy. MPP reduces the number of intermediate and output channels, eliminates shortcuts, and replaces the concatenation operation with addition, thereby improving computational efficiency. It is more suitable for real-time industrial inspection.

3.4 Cross-level attention fusion module

In the decoder, there is a stack of basic blocks along with the Cross-Level Attention Fusion module (CLAF). As mentioned above, the fusion of multi-level features is crucial for achieving high-precision segmentation. CLAF employs spatial attention to enrich the fused feature representation.

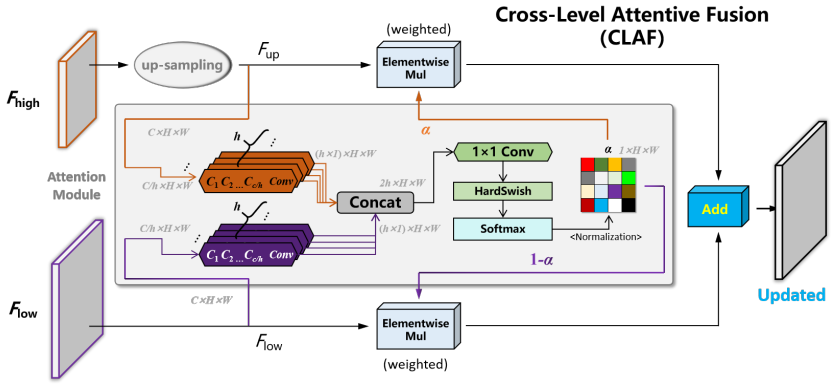

Fig. 4 Design of cross level attention fusion module

The CLAF network architecture is shown in Figure 4. The motivation behind the spatial attention module is to generate a 2-dimensional weight matrix that represents the importance of each pixel in the input features by leveraging the spatial relationships between pixels. CLAF utilizes a spatial attention module to generate a pair of normalized weights, which are then fused with the cross-level input features through Elementwise_Mul and Elementwise_Add operations. Specifically, we denote the two-level input features as F_{high} and F_{low}. F_{high} represents the output from a deeper-level module. F_{low} corresponds to the corresponding part from the encoder. They have the same channels but different sizes. CLAF first employs bilinear interpolation to upsample F_{high} to the same size as F_{low}. Then, the attention module performs dimension reduction on F_{up} and F_{low} using 1×1 convolutions. Here, we design a multi-head mechanism for attention, where the number of attention heads is denoted as h and can be customized. Each head processes C/h channels with a 1×1 convolution and outputs intermediate features of size $1 \times H \times W$. Concat is used to concatenate all the obtained intermediate features along the channel dimension, followed by a 1×1 convolution to obtain a feature of size $1 \times H \times W$. Finally, Softmax is applied to produce a normalized weight matrix σ. The aforementioned process can be represented by the Equation (4).

$$\begin{aligned} \sigma &= \text{attention}(F_{up}, F_{low}) \\ F_{out} &= F_{up} \cdot \sigma + F_{low} \cdot (1 - \sigma) \end{aligned} \tag{4}$$

To obtain the weighted features, we apply Elementwise_Mul operations to F_{up} and F_{low} separately. Finally, CLAF performs Elementwise_Add on the weighted features across levels and outputs the fused feature. The formulaic representation of the spatial attention module is shown in Equation (5).

$$\begin{aligned} F_{cat} &= \text{Concat}(Conv_1(F_{up})...\text{Conv}_\text{h}(F_{up}), \text{Conv}_{\text{h}+1}(F_{low})...\text{Conv}_{2\text{h}}(F_{low})) \\ \alpha &= \text{Softmax}(\text{HardSwish}(\text{Conv}(F_{cat}))) \end{aligned} \tag{5}$$

The feature map after fusion through CLAF, when upsampled, achieves better restoration of the original image and emphasizes the features, thereby further improving the segmentation accuracy.

4 Experiments

4.1 Data and implementation details

Vehicle component defect dataset. The vehicle component defect dataset is a small-scale industrial defect detection segmentation dataset. It consists of 864 high-quality images with pixel-level annotations. In this paper, the dataset is divided into training, validation, and test sets in an 8:1:1 ratio, containing 691, 86, and 87 images, respectively. The dataset contains three different defect categories: peeling, crack and scratch. It can be used for experimental research in the field of industrial defect detection.

BSData dataset. BSData is dataset for instance segmentation and industrial wear forecasting. It consists of 1104 channel 3 images with 394 image-annotations for the

surface damage type “pitting”. We divide these 394 images into 2 categories in a 4:1 ratio for training and image segmentation.

RSDDs dataset. RSDDs is a dataset of surface defects on railway tracks. It consists of two types. Type I RSDDs dataset is captured from the fast lane, which contains 67 images. Type II RSDDs dataset is captured from regular/heavy transportation tracks, which contains 128 images. Each image of these two types contains at least one type of defect. We choose the type II RSDDs dataset, and divide it a 4:1 ratio.

CrackForest dataset. CrackForest is a fast road crack detector that achieves excellent accuracy. It is an annotated database of road crack images that can roughly reflect the condition of urban road surfaces. In this paper, the dataset is divided into training and test sets, containing 94 and 24 images, respectively. The CrackForest dataset contains one defect category.

A high-end aerospace aluminum defect dataset. This dataset is an individual industrial defect detection segmentation dataset and consists of 5,000 surface images of aluminum materials. It contains 8 different defect categories. We used all the images for training, and subsequently achieved real-time on-site defect detection using the hardware system.

Hardware system. For vehicle component defect dataset, BSData dataset, RSDDs dataset and CrackForest dataset, we use CUDA 10.2 and CUDNN 8.2 on an NVIDIA Tesla V100 GPU for experiments. For real-time high-end aluminum defect detection, we use CUDA 10.2 and CUDNN 8.2 on an NVIDIA Tesla V100 GPU for training. Then we deploy AENet in the system shown in the following Figure 5(a) to complete on-site defect detection and segmentation. Figure 5(b) shows the actual scene of defect detection in aluminum materials.

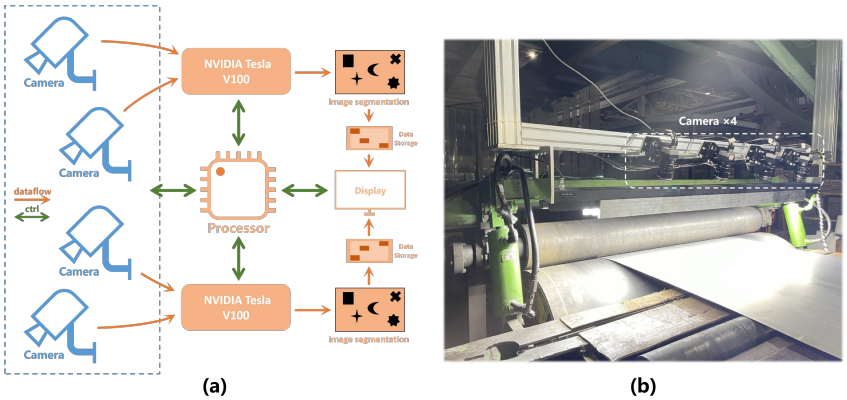

Fig. 5 (a) System architecture for on site defect detection, (b) high end aluminum material defect detection

Training settings. We choose the stochastic gradient descent (SGD) algorithm with a momentum of 0.9 as the optimizer and employ a warm-up strategy and "poly" learning rate scheduler. To conduct model training, we set the batch size to 16, the maximum number of iterations to 20,000, the initial learning rate to 0.005, the random scaling range is [0.5, 2.0] and weight decay to 5e-4. We also apply data augmentation

techniques to the dataset, including random scaling, random cropping, random horizontal flipping, random color jittering, and normalization. For the Vehicle Component Defect Dataset, the cropping resolution is set to 512×512. For the BSData Dataset, the cropping resolution is set to 1130×460. For the CrackForest Dataset, the cropping resolution is set to 480×320. For the RSDDs Dataset, the cropping resolution is set to 55×1250. For the Aluminum Surface Defect Dataset, the cropping resolution is set to 2048×1000. All experiments are conducted using PaddlePaddle [29] on an NVIDIA Tesla V100 GPU.

Inference settings. To fairly compare the performance of different models on the datasets introduced above. We perform the model conversion and inference in the same environment. The predicted images are resized back to the original input image size. The overall inference time includes image preprocessing, inference and post-processing time. We evaluate the segmentation accuracy using the standard mean Intersection over Union (mIoU) as the comparison metric and the time consumption per image (s/img) as the time comparison metric.

4.2 Detection results on public datasets

4.2.1 Comparison with state-of-the-art models

Under the same training and inference settings, we compared our model with the state-of-the-art segmentation networks on the public datasets mentioned before. We present the performance metrics of mIoU and inference latency in Table 2 and Table 3. As shown in Table 2, The performance of mIoU after convergence varies for different datasets. For datasets with environmentally sensitive surface minor defects like vehicle component defect and RSDDs, AENet can achieve the highest mIoU among these compared networks. In Table 3, all experiments were conducted on NVIDIA Tesla V10 for fair comparison.

Table 2 mIoU of advanced instance segmentation methods on industrial defect datasets

	Encoder	mIoU(%)			
		Vehicle component defect	BSData	RSDDs	CrackForest
Enet	-	35.4	40.64	59.01	83.35
EspnetV2	ESPNetV2	56.28	37.16	55.28	80.28
BiSeNetV1	ResNet18	58.38	32.96	53.38	81.38
BiSeNetV2	-	62.05	32.48	57.50	78.79
Sfnet	DF1	63.89	34.96	57.89	81.02
STDCSegV1	STDC1	57.48	29.34	59.05	81.35
STDCSegV2	STDC2	58.72	36.55	59.03	81.10
Our model	**STDC2+SCA**	**67.56**	**37.92**	**59.12**	**82.00**

Figure 6 shows the mIoU comparison of different neural networks at different training epochs. We consider 20000 iters as the training endpoint. As the training progresses, the mIoU of AENet steadily increases. Besides, compared to other networks, AENet has faster convergence. Figure 7 displays the results of inferring industrial defect images under different contrast, and brightness conditions. The green band in

Table 3 Inference latency of advanced instance segmentation methods on industrial defect datasets

	Hardware	Latency (s/img)			
		Vehicle component defect	BSData	RSDDs	CrackForest
Enet	NVIDIA Tesla V100 GPU	0.0282	0.0523	0.0074	0.0187
EspnetV2		0.0255	0.0559	0.0067	0.0173
BiSeNetV1		0.0267	0.0513	0.0071	0.0179
BiSeNetV2		0.0238	0.0477	0.0069	0.0162
Sfnet		0.0344	0.0726	0.0100	0.0231
STDCSegV1		0.0369	0.0714	0.0092	0.0242
STDCSegV2		0.0454	0.0927	0.0128	0.0319
Our model		**0.0392**	**0.0762**	**0.0106**	**0.0237**

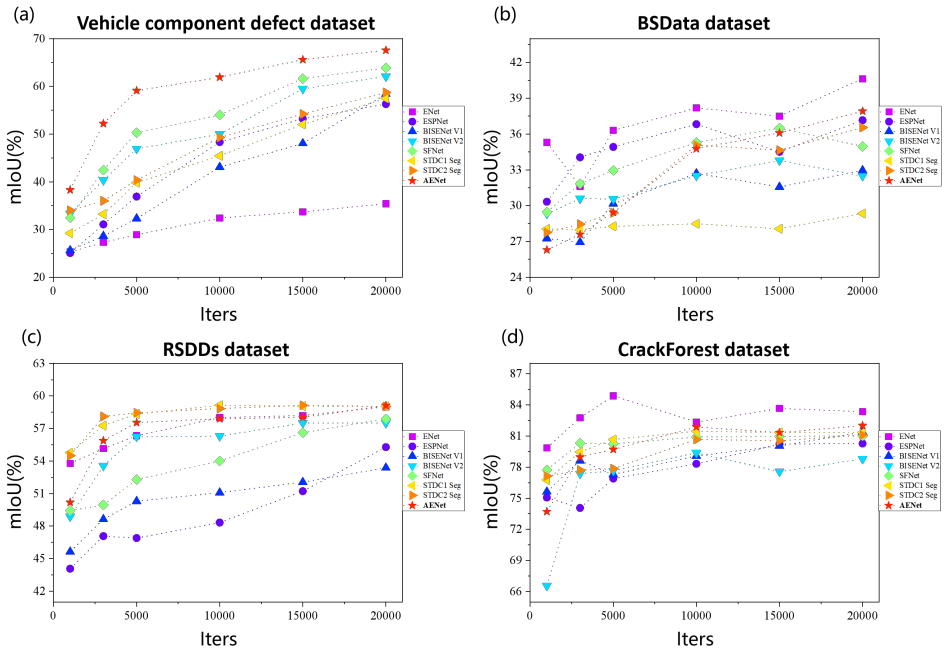

Fig. 6 Comparison of mIoU of different networks with increasing training times on (a)Vehicle component defect, (b)BSData, (c)RSDDs and (d)CrackForest

the figure represents the segmentation results. The adjustment of image contrast and brightness is done to simulate real industrial inspection environments. The conditions of low contrast and low brightness increase the difficulty of defect recognition. AENet can outline defects under poor optical conditions. This demonstrate that our proposed AENet has superior segmentation performance, higher learning efficiency, and good real-time capability and robustness.

Figures 8-10 show the original images of three public defect datasets as well as the results of inference using AENet. Defects are marked in green. AENet can accurately distinguish shadows and defects, even if they have similar grayscale features.

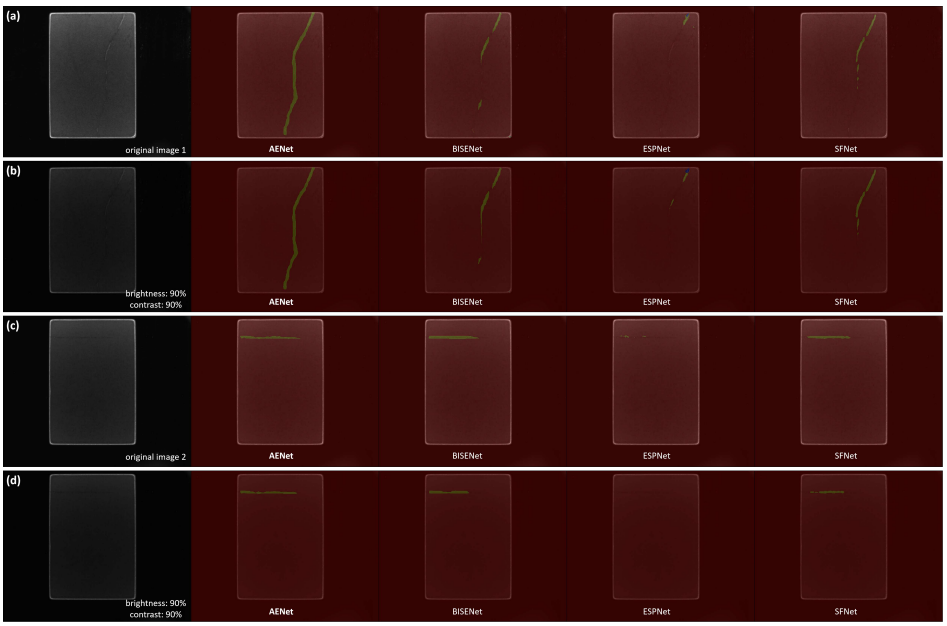

Fig. 7 The segmentation results of vehicle component defect dataset. Groups (a) and (c) represent the inference results of two different original images in different neural networks, while groups (b) and (d) represent the inference results in different neural networks after 90% contrast and brightness changes are made to the original images

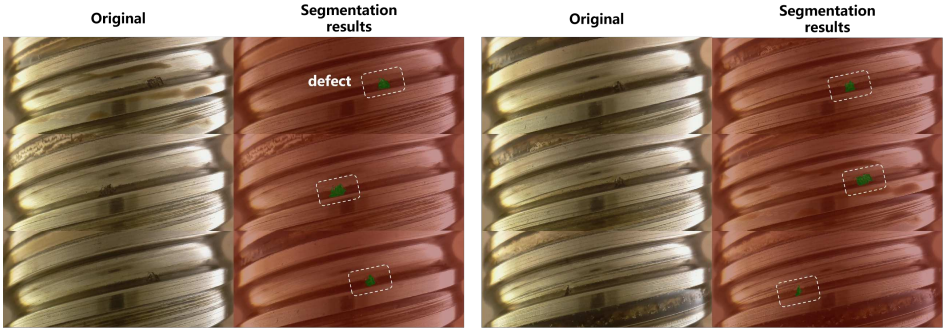

Fig. 8 The segmentation results of BSData dataset using proposed AENet

These random images demonstrate AENet's ability of detection and segmentation of industrial defects with different sizes and shapes.

4.2.2 Ablation study

This ablation experiment validates the effectiveness of three design modules in AENet, namely the Edge Attention Module (SCA), the Simplified Pyramid Pooling Module (MPP), and the Cross-Level Attention Fusion Module (CLAF). The experiment used the same training and inference settings as before, with STDC2-Seg as the baseline

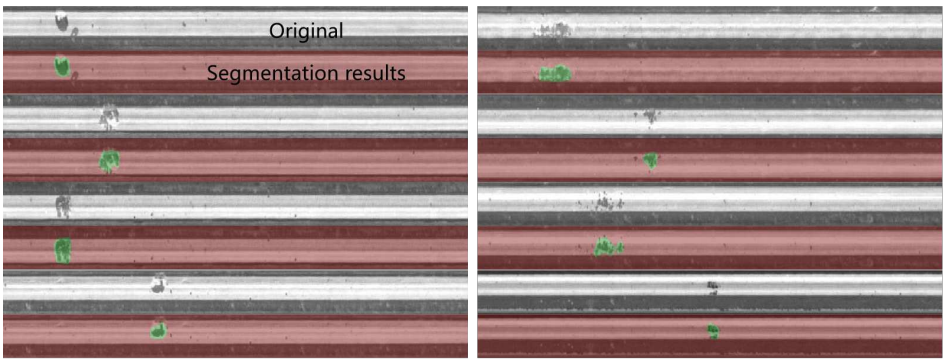

Fig. 9 The segmentation results of RSDDs dataset using proposed AENet

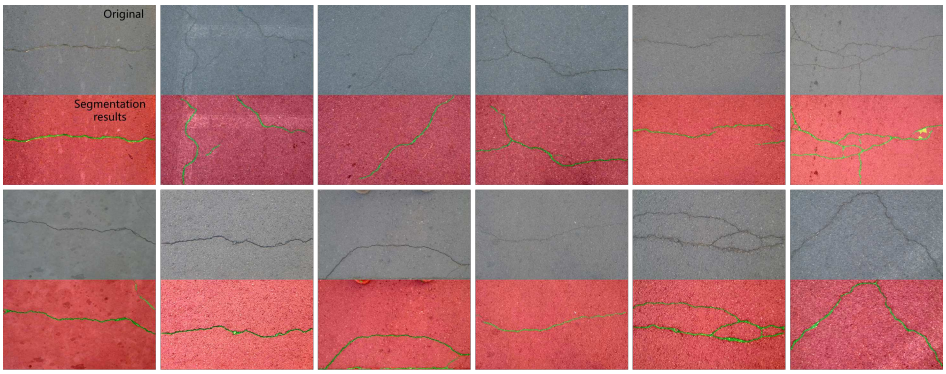

Fig. 10 The segmentation results of CrackForest dataset using proposed AENet

model. The results in Table 4 demonstrate that the mIoU consistently improves when SCA, MPP, and CLAF are sequentially incorporated. For the ablation experiment task, we train AENet on Vehicle component defect dataset.

The impact of SCA module. The SCA module significantly improves the mIoU of the network. However, it introduces more parameters compared to MPP and CLAF, and increases the computational complexity of the inference process. If the number of SCA modules is blindly increased, the inference speed of the network will be reduced.

The impact of MPP and CLAF module. Due to the simplified design adopted by MPP itself, there are not many additional parameters added, and so does the FLOPs. CLAF slightly increases network complexity. An attention mechanism is designed in it like SCA. CLAF enables AENet to better display segmentation, trading speed for accuracy.

Although the addition of SCA, MPP and CLAF slightly slows down the inference speed compared to PPLiteSeg, it achieves higher segmentation accuracy. With the support of these three modules, AENet achieves an segementation accuracy of 67.56%, which is a 3.48% improvement over the PPLiteSeg specifically used for semantic recognition. Therefore, these three design modules are effective in defect detection.

Table 4 The results of the ablation experiment

	mIoU(%)	Params(M)	FLOPs(G)	Latency(s)
Baseline	58.72	12.37	1.412	0.0354
PPLiteSeg	64.08	\	\	0.0307
Baseline+S[1]	64.53	13.33	1.533	0.0371
Baseline+S+M[2]	65.96	13.78	1.548	0.0379
Baseline+S+M+C[3]	67.56	14.55	1.590	0.0392

[1]SCA module

[2]MPP module

[3]CLAF module

4.3 Real-time inspection

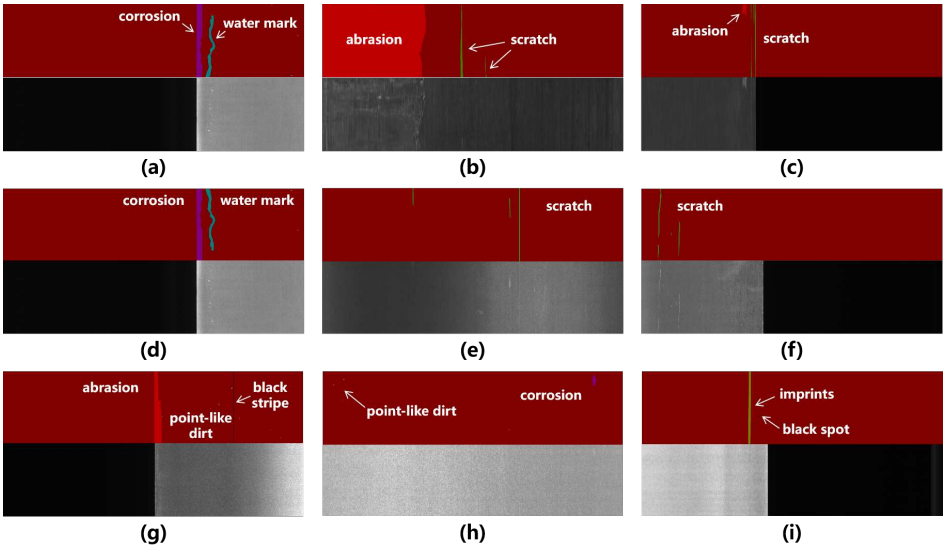

Fig. 11 The results of Real-time industrial aluminum inspection using AENet

We conducted experiments on a high-end aerospace aluminum defect dataset using the training and inference settings mentioned above. We conduct comparisons between the original images of 8 defect categories on the aluminum surface. The segmentation results are shown in Figure 11. The red photographs show the segmentation map, while the black and white photographs are actual aluminum material. It can be observed that whether it is a large area defect, such as water marks, scratches, and corrosion, or a small area defect that is not easily visible to the naked eyes like dot-like-dirt. AENet can effectively recognize and segment them.

We grouped 500 images used for inference verification. And completed the quantity statistics of the five most common defect types, as shown in the total item in Table 5. Then we provided our monitoring results for these defects, as shown in the detected

Table 5 Precision of real-time defect detection for different types of defects in industrial aluminum materials

		corrosion	water mark	abrasion	point-like-dirt	scratch	average
Group 1	detected	11	4	27	79	61	95.79%
	total	12	4	28	84	62	
Group 2	detected	7	3	16	91	54	95.00%
	total	7	3	16	99	55	
Group 3	detected	13	9	10	55	33	97.56%
	total	13	9	11	57	33	
Group 4	detected	8	8	14	56	46	94.29%
	total	9	8	15	59	49	
Group 5	detected	6	3	30	62	66	98.24%
	total	6	3	30	64	67	
Precision		95.74%	100.00%	97.00%	94.49%	97.74%	96.14%

item in Table 5. The experimental results indicate that AENet exhibits good performance against different types of defects. In the real industrial environment, the error of different groups is less than 4%. AENet can accurately capture subtle defect details and segment them. This is of great significance for the quality control of aluminum materials.

5 Conclusion

In this paper, we focus on designing a new real-time industrial defect detection network called AENet, based on an encoder-decoder structure. AENet uses spatial and channel dual attention mechanisms to update feature maps. It fuses features with different levels of abstraction to enhance the representation of features. It can also complete multi-level feature extraction at a lower computational cost. Extensive comparative experiments demonstrate that AENet achieves a good balance between defect segmentation accuracy and inference speed, with higher learning efficiency compared to conventional image segmentation networks, making it robust for real-world industrial inspection environments. The ablation experiments further validate the effectiveness of each module, showcasing the benefits of our design. These findings highlight the promising prospects of encoder-decoder models and attention mechanisms in defect detection and recognition. Overall, AENet can achieve outstanding performance in comparative experiments with advanced networks on multiple datasets. And an error of less than 4% is obtained in on-site industrial aluminum material detection. In future work, we plan to enhance the segmentation accuracy stability of AENet in more challenging visual environments and apply our method to a wider range of tasks.

Acknowledgments. This work is partially supported by the Special Key Project of Technological Innovation and Application Development of Chongqing (CSTB2022TIAD-KPX0057), the Natural Science Foundation of Chongqing, China (2022NSCQLZX0128).

Declarations

- Conflict of interest/Competing interests

Not applicable

- Consent to participate
  Not applicable
- Consent for publication
  Not applicable
- Availability of data and materials
  The datasets used or analysed during the current study are available from the corresponding author on reasonable request.
- Code availability
  The code is available from the corresponding author on reasonable request.
- Authors' contributions
  Yi Wan: Writing-original draft, Writing-review editing, Investigation and Software.
  Lingjie Yi: Software.
  Bo Jiang: Software.
  Junfan Chen: Visualization and Data curation.
  Yi Jiang: Supervision.
  Xianzhong Xie: Methodology and Conceptualization.

References

[1] Xia, L.-m., Wei, C.C.: Abnormal event detection in surveillance videos based on multi-scale feature and channel-wise attention mechanism. The Journal of Supercomputing **78**, 13470–13490 (2022)

[2] Luo, S., Hou, J., Zheng, B., Zhong, X., Liu, P.: Research on edge detection algorithm of work piece defect in machine vision detection system. In: 2022 IEEE 6th Information Technology and Mechatronics Engineering Conference (ITOEC), vol. 6, pp. 1231–1235 (2022). https://doi.org/10.1109/ITOEC53115.2022.9734631

[3] Yu, S., Zhou, W., Liu, J.: A novel defect detection method of liquid crystal display based on machine vision. In: 2022 4th International Conference on Industrial Artificial Intelligence (IAI), pp. 1–6 (2022). https://doi.org/10.1109/IAI55780.2022.9976633

[4] Wu, H., Luo, H., Zhu, W., Wang, Y., Zhang, Q., Ma, B., Yang, Y., Fan, H., Xu, H.: Surface defect detection of plaster coating based on machine vision. In: 2017 IEEE International Conference on Unmanned Systems (ICUS), pp. 277–281 (2017). https://doi.org/10.1109/ICUS.2017.8278354

[5] Chang, K.-C., Chang, F.-H., Wang, H.-C., Amesimenu, G.D.K.: Machine vision welding defect detection based on fpga. In: 2021 16th International Microsystems, Packaging, Assembly and Circuits Technology Conference (IMPACT), pp. 193–196 (2021). https://doi.org/10.1109/IMPACT53160.2021.9696609

[6] Cen, Y., Zhao, R.-Z., Cen, L., Cui, L.-H., Miao, Z.-J., Wei, Z.: Defect inspection for tft-lcd images based on the low-rank matrix reconstruction. Neurocomputing

149, 1206–1215 (2015) https://doi.org/10.1016/j.neucom.2014.09.007

[7] Wu, X., Chen, Q., Xiao, Y., Li, W., Liu, X., Hu, B.: Lcsegnet: An efficient semantic segmentation network for large-scale complex chinese character recognition. IEEE Transactions on Multimedia **23**, 3427–3440 (2021) https://doi.org/10.1109/TMM.2020.3025696

[8] Gu, Z., Chen, H., Xu, Z., Lan, J., Meng, C., Wang, W.: DiffusionInst: Diffusion Model for Instance Segmentation (2022)

[9] Mak, K., Peng, P., Yiu, K.: Fabric defect detection using morphological filters. Image and Vision Computing **27**(10), 1585–1592 (2009) https://doi.org/10.1016/j.imavis.2009.03.007

[10] Yuan, X.-c., Wu, L.-s., Peng, Q.: An improved otsu method using the weighted object variance for defect detection. Applied Surface Science **349**, 472–484 (2015) https://doi.org/10.1016/j.apsusc.2015.05.033

[11] Bai, X., Fang, Y., Lin, W., Wang, L., Ju, B.-F.: Saliency-based defect detection in industrial images by using phase spectrum. Industrial Informatics, IEEE Transactions on **10**, 2135–2145 (2014) https://doi.org/10.1109/TII.2014.2359416

[12] Boukdir, A., Benaddy, M., Meslouhi, O.E., Kardouchi, M., Akhloufi, M.: Character-level arabic text generation from sign language video using encoder-decoder model. Displays **76**, 102340 (2022) https://doi.org/10.1016/j.displa.2022.102340

[13] Lin, Y.: Wafer pattern counting, detection and classification based on encoder-decoder cnn structure. In: 2022 Intermountain Engineering, Technology and Computing (IETC), pp. 1–5 (2022). https://doi.org/10.1109/IETC54973.2022.9796856

[14] Wang, S., Wang, H., She, S., Zhang, Y., Qiu, Q., Xiao, Z.: Swin-t-nfc crfs: An encoder–decoder neural model for high-precision uav positioning via point cloud super resolution and image semantic segmentation. Comput. Commun. **197**(C), 52–60 (2023) https://doi.org/10.1016/j.comcom.2022.10.011

[15] Jie, F., Nie, Q., Li, M., Yin, M., Jin, T.: Atrous spatial pyramid convolution for object detection with encoder-decoder. Neurocomputing **464**, 107–118 (2021) https://doi.org/10.1016/j.neucom.2021.07.064

[16] Wang, C., Wang, H.: Cascaded feature fusion with multi-level self-attention mechanism for object detection. Pattern Recognition **138**, 109377 (2023) https://doi.org/10.1016/j.patcog.2023.109377

[17] Wang, A., Togo, R., Ogawa, T., Haseyama, M.: Multi-scale defect detection from subway tunnel images with spatial attention mechanism. In: 2022 IEEE

International Conference on Consumer Electronics - Taiwan, pp. 305–306 (2022). https://doi.org/10.1109/ICCE-Taiwan55306.2022.9869056

[18] Peng, Z., Gong, X., Lu, Z., Xu, X., Wei, B., Prasad, M.: A novel fabric defect detection network based on attention mechanism and multi-task fusion. In: 2021 7th IEEE International Conference on Network Intelligence and Digital Content (IC-NIDC), pp. 484–488 (2021). https://doi.org/10.1109/IC-NIDC54101.2021.9660399

[19] Zhao, Z., Chen, K., Yamane, S.: Cbam-unet++:easier to find the target with the attention module "cbam". In: 2021 IEEE 10th Global Conference on Consumer Electronics (GCCE), pp. 655–657 (2021). https://doi.org/10.1109/GCCE53005.2021.9622008

[20] Zhong, Z., Lin, Z.Q., Bidart, R., Hu, X., Daya, I.B., Li, Z., Zheng, W.-S., Li, J., Wong, A.: Squeeze-and-Attention Networks for Semantic Segmentation (2020)

[21] Le, N., Nguyen, K., Nguyen, A., Le, B.: Global-Local Attention for Emotion Recognition (2021)

[22] Zhang, H., Goodfellow, I., Metaxas, D., Odena, A.: Self-Attention Generative Adversarial Networks (2019)

[23] Yang, N., He, C.: Malaria detection based on resnet + cbam attention mechanism. In: 2022 3rd International Conference on Information Science, Parallel and Distributed Systems (ISPDS), pp. 271–275 (2022). https://doi.org/10.1109/ISPDS56360.2022.9874134

[24] Cui, Q., Sun, H., Li, Y., Kong, Y.: A deep bi-directional attention network for human motion recovery. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence. IJCAI'19, pp. 701–707. AAAI Press, ??? (2019)

[25] Sun, Q., Fu, Y.: Stacked self-attention networks for visual question answering. In: Proceedings of the 2019 on International Conference on Multimedia Retrieval. ICMR '19, pp. 207–211. Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3323873.3325044 . https://doi.org/10.1145/3323873.3325044

[26] Fan, M., Lai, S., Huang, J., Wei, X., Chai, Z., Luo, J., Wei, X.: Rethinking bisenet for real-time semantic segmentation. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9711–9720 (2021). https://doi.org/10.1109/CVPR46437.2021.00959

[27] Peng, J., Liu, Y., Tang, S., Hao, Y., Chu, L., Chen, G., Wu, Z., Chen, Z., Yu, Z., Du, Y., Dang, Q., Lai, B., Liu, Q., Hu, X., Yu, D., Ma, Y.: PP-LiteSeg: A Superior Real-Time Semantic Segmentation Model (2022)

[28] Lu, N., Yu, W., Qi, X., Chen, Y., Gong, P., Xiao, R., Bai, X.: MASTER: Multi-aspect non-local network for scene text recognition. Pattern Recognition **117**, 107980 (2021) https://doi.org/10.1016/j.patcog.2021.107980

[29] Ma, Y., Yu, D., Wu, T., Wang, H.: Paddlepaddle: An open-source deep learning platform from industrial practice. (2019)