ISSN 1330-3651 (Print), ISSN 1848-6339 (Online)
https://doi.org/10.17559/TV-20230210000345
Original scientific paper

A Lightweight Convolutional Neural Network for Salient Object Detection

Fengchang FEI, Wei LIU, Lei SHU*

Abstract: U-shape networks are widely used in salient object detection. Recently, CTDNet with a Comprehensive Triangular Decoder improved detection efficiency, which made some improvement with respect to the complexity and slow training of U-shape networks. However, CTDNet is still not lightweight enough, and the use of Global Average Pooling for top-level semantic features can lead to the loss of global structural information. This paper proposes Trilateral Enhanced Network (TENet), a faster salient detection model based on CTDNet, for industrial application. TENet uses MobileNetV3 as a backbone network so that TENet only needs 3.72M parameters, which lightweight the network consequently. TENet contains a feature fusion module called Channel Attraction Enhanced Feature Fusion Model, which integrates high-level semantics to improve accuracy. Additionally, Convolutional Block Feature Enhancement Module is proposed, which can further enhance accuracy. In comparison with CTDNet, TENet is a lightweight network with faster detection speed and more detection accuracy. TENet robustly detects defects in salient texture images, indicating insensitivity to texture interference. Experiments show TENet maintains strong performance on salient textures detection, demonstrating suitability for industrial optical inspection.

Keywords: CTDNet; industrial optical inspection; MobileNetV3; salient object detection; TENet

1 INTRODUCTION

Salient object detection aims to detect foreground objects that are significantly different from the objects in the background in the image. Hence, a colour image is transformed into a saliency map, and the values in different positions in the map represent the degree of eye-catching attention of the image [1, 2].Early salient object detection models mostly rely on manually extracted features to represent local details and global contexts [3, 4]. Owing to the lack of high-level abstract semantic information, the detection effect in complex scenes is deficient. In recent years, convolutional neural network has been widely used in various fields, including salient object detection, due to its automatic extraction of multi-level semantic information[5-8]. The pyramid-like network structure not only contains rich and detailed underlying information, but also reflects more high-level semantic information, which can better locate the exact location of salient objects. Therefore, many salient object detection algorithms based on pyramid structure has been proposed [9-12].

The reason why U-shape [13] is widely used is that it only needs a small number of training samples to complete the training of the model. However, in order to achieve high accuracy, the model has to be complex with a large number of parameters. In 2021, Zhao proposed CTDNet [14], which uses ResNet-18 as the backbone network so as to overcome the shortcomings brought by U-shape with many parameters and to improve the speed of the model.CTDNet has three branches. The first is Boundary Path where the edge supervision is enhanced to improve the accuracy of boundary segmentation. The second is Spatial Path which adds the supervision of spatial features so that more spatial details can be preserved. The third is Semantic Path which contains the supervision of high-level features to reduce the loss of deep semantic information and context structure information.

Although CTDNetis effective for salient object detection, there are still some points which are worthy of discussion:

(1) The GAP operation at the highest level in the semantic path increases the receptive field and enhances the semantic features, but the image becomes a point. Will the position information of the object be lost?

(2) The high-level semantic features of objects are completely different from their spatial representation in morphology. The low-level features contain many spatial details, and the high-level features are abstract semantic features. Is it appropriate to use spatial information (the ground truth of salient objects) to guide the learning of high-level features in semantic path?

(3) CTDNet can use ResNet-18 as the backbone network to lighten the model and improve the running speed of the model. Can the backbone network be modified to further speed up the model?

(4) The results of salient object detection and object segmentation are to generate a binary image. Can the algorithm be used for object segmentation?

Therefore, based on the CTD framework, this paper proposes a TENet saliency detection model, which has a lightweight structure and superior detection performance. TENet has made the following improvements:

(1) The backbone network is replaced with a more streamlined MobileNetV3 to reduce network parameters and improve detection speed. Compared with the 11.82M parameters of CTDNet-R18, TENet has only 3.72M parameters.

(2) A new semantic feature fusion model is designed.

(3) High level semantic features are not supervised by the ground truth to reduce the loss of semantic information in the model.

(4) An attention module is added after the middle convolution layer to further improve the accuracy of salient object detection.

2 RELATED WORKS

U-Net based on U-shape adopts data augmentation, which realizes biomedical segmentation with only veryfew annotated images. Hence U-shape structure has been widely used in image segmentation and salient object detection algorithms [15-17]. Tsunyi Lin proposed aFPN network framework similar to U-shape structure for target detection [9]. FPN was initially used for object detection.Then many algorithms have adopted FPN structure in salient object detection and image segmentation [7, 18-22]. In addition to U-shape and FPN, other salient object detection algorithms have been proposed [23, 24]. However, in order to improve algorithm performance, the networks above are becoming more complex, with more parameters and slower training speeds. As a consequence, the algorithm testing must rely entirely on GPU to complete.However, these complex saliency detection algorithms are difficult to be adopted by the industry.

2.1 CTDNet

In 2021, Zhao proposed a CTDNet which contained a Comprehensive Triangular Decoder [14] for salient object detection. CTDNet is a lightweight algorithm for salient object detection [14]. The key of CTDNet is that it contains an efficient decoder---Comprehensive Trilateral Decoder (CTD). CTDNet is divided into three paths: Semantic Path, Spatial Path and Boundary Path. These three branches aim to solve the problems of unclear semantic information, loss of spatial information and lack of boundary information respectively. These three branches include the processing of different depth features in convolutional networks, which complement one another. Semantic Path is used to capture rich semantic context information and global context information with large receptive fields. Spatial Path is designed to preserve more spatial details. The combination of Semantic Path and Spatial Path forms a comprehensive and powerful middle and high-level fusion feature. For Boundary Path, CTDNet uses low-level spatial features and high-level fusion features to extract the location boundary features of objects and adds additional edge supervision to improve the accuracy of boundary acquisition.

In order to realize the fusion of different features, CTDNet has three feature fusion modules: Feature Fusion Module (FFM) to fuse the features of different convolution layers of the network; Cross Aggregation Module (CAM) to fuse spatial features and advanced semantic features to realize the combination of the Semantic Path and Semantic Path; Boundary Refining module (BRM) to merge boundary features and high-level features.

2.2 Attention Mechanism

In human visual perception, the human eye can focus on the objects projected on the retina quickly. In convolution neural networks, the convolution blocks can be filtered to highlight more effective features. Human eye attention includes the location of interest and the characteristics of interest. Therefore, in convolutional neural networks, researchers have proposed two attention mechanisms: spatial attention module and channel attention module.

In spatial attention model, each pixel learns a weight value in the spatial domain of the image, and each weight value represents the importance of a pixel position. Spatial attention model first appeared in the paper [25] in 2015, in which Max Jaderberg proposed a structure of spatial transformer, which can realize the spatial transformation of feature map without any additional learning process, opening a new way for neural network model.

Channel attention model is to apply a weight to the feature map of each channel in the convolution block. Each weight represents the importance of the feature map. In convolutional neural networks, the higher the number of layers, the lower the resolution of the feature map, but the more the number of channels. It is necessary to select the channel feature map to determine which feature map is more important. Hence the channel attention mechanism is used to achieve this task. Through the study of channel attention, Jie Hu proposed a Sequence and Exception Block (SEB) structure [26], which adaptively recalibrates the feature response of the channel by establishing the interdependence between the channels. This structure can be easily added to convolution network to form SENet, which can improve the accuracy, but it also brings higher model complexity and more computation. Based on SEB, Qilong Wang proposed a more effective channel attention model ECANet [27]. Compared with SEB, ECANet structure reduces network parameters and has faster operation speed with higher accuracy. In order to further improve the efficiency of attention model, Sanghyun Woo maintains that the feature matrix obtained after convolution is the result of mixing the spatial features and channel features of image, and that the feature matrix should be enhanced independently according to channel dimension and spatial dimension, hence Convolutional Block Attention Module (CBAM) is proposed [28].

2.3 MobileNetV3

Compared with heavyweight networks, lightweight networks have less parameters, less computation, and short time consumption. Lightweight networks are more suitable for scenarios with limited computing power, storage, and power consumption, such as mobile devices and industrial pipelines. MobileNet is an outstanding lightweight network, which has developed into the third version after the accumulation of the first and second versions [29]. MobileNetV3 adds the channel attention mechanism, which is excellent in performance and speed. It is favoured by the academic and industrial circles and has become one of the most widely used lightweight networks. The TENet proposed in this paper also uses MobileNetV3 as the backbone network.

3 PROPOSED METHOD

3.1 Algorithm Framework

The overall framework of TENet proposed in this paper is shown in Fig. 1. Since CTD structure is more compact than U-shape structure network, TENet adopts CTD structure. TENet mainly includes encoder and decoder. The network framework will be explained in detail as follows:

3.1.1 Encoder

If the algorithm is to be used in the industry, it is very necessary to improve the speed of the algorithm. Therefore, MobileNetV3 is adopted as the backbone network of TENet. The experimental results show that TENet has fast execution speed and low requirements for computer hardware.

The image is input to MobileNetV3. After multi-stage convolution processing in the network, corresponding convolution blocks can be obtained. These convolution blocks are also called feature matrices. The deeper the layers of the backbone network, the lower the spatial resolution of the feature matrix obtained after convolution of the input image. The spatial resolution of the convolution block generated after convolution of each stage is 1/2 of the spatial resolution of the convolution block generated by convolution of the previous stage. The convolution blocks obtained after convolution in the first stage has the highest spatial resolution. However, if this convolution block is involved in decoder, the number of network weights will be greatly increased, and the calculation amount of the network will become large. Therefore, the paper only focuses on the convolution block obtained by convolution in the last four stages. The spatial resolution of the feature matrix in the last four stages is 1/4, 1/8, 1/16 and 1/32 of the resolution of the input image. We use $\{E^{(2)}, E^{(3)}, E^{(4)}, E^{(5)}\}$ to represent feature matrixes from these four stages.

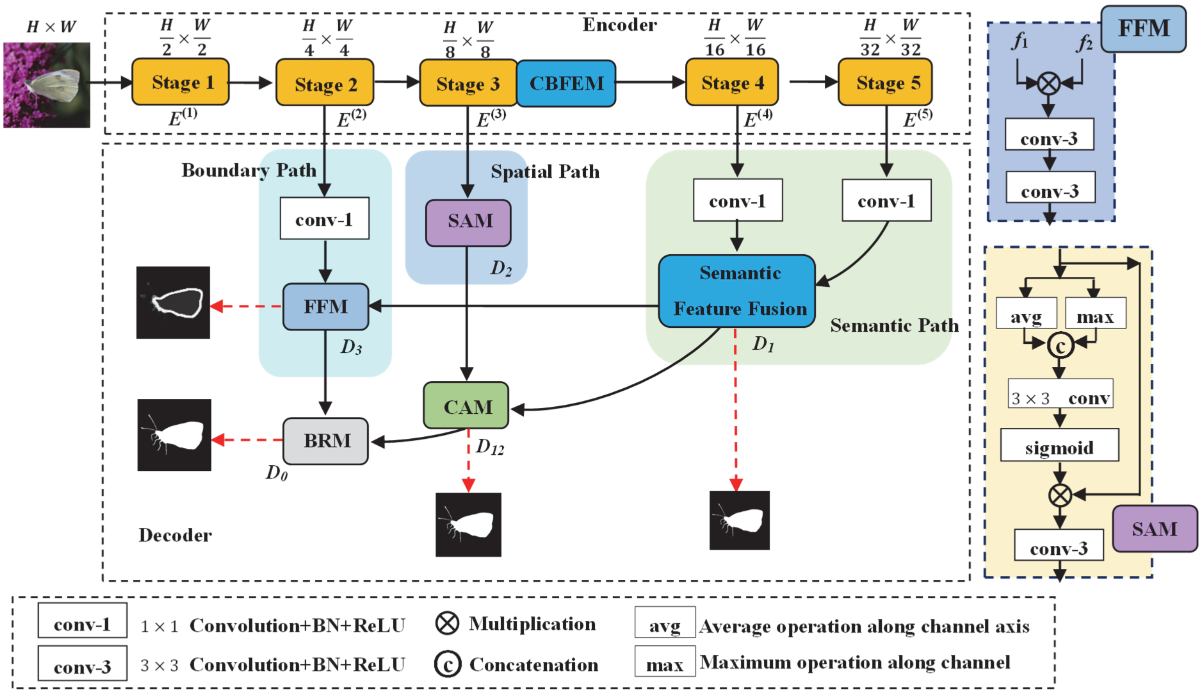

Figure 1 The framework of Trilateral Enhanced Network (TENet)

The features obtained by stage 3 are between spatial features and high-level semantic features, that is, it contains spatial features and integrate high-level semantic features. In order to enhance the significant features in $E^{(3)}$ and reduce the influence of insignificant features on the experimental results, we embed a Convolutional Block Feature Enhancement Module (CBFEM) after $E^{(3)}$ to strengthen the significant features and suppress the insignificant features through the CBFEM. In the following experiments, we will also compare the results of the encoder without CBFEM and the encoder with CBFEM.

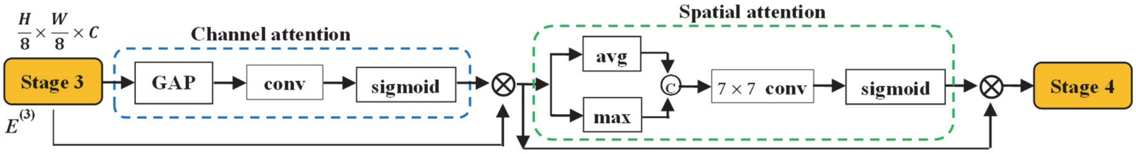

Figure 2 Convolutional Block Feature Enhancement Module (CBFEM)

Convolutional Block Feature Enhancement Module (CBFEM) is proposed in this paper to enhance convolution features through attention mechanism. It is a mixed attention mechanism. The structure is shown in Fig. 2. CBFEM includes two steps of enhancement. First, channel attention enhancement is performed on convolution block. Here we refer to ECANet [27]. Next, spatial attention enhancement is performed. Here we refer to CBAM [28]. CBFEM can improve the significant features in the two dimensions of channel and space in the convolution block and suppress the insignificant features.

In the channel attention module of CBFEM, the GAP (Global Average Pooling) [30] operation is first performed on the feature matrix $E^{(5)}$. The GAP operation is to average each feature map in $E^{(5)}$ to one point, and then the feature matrix $E^{(5)}$ is transformed into a 1D vector $E^{(5)}_{\text{GAP}}$. Next, 1D convolution of size 5 is performed on this 1D vector, and a sigmoid activation function is executed to obtain a channel attention weight map Map_C. The purpose of using 1D convolution of size 5 is to capture cross-channel feature extraction, further reduce the complexity of the model and improve the running speed. $E^{(5)}$ is multiplied by Map_C. to obtain $E_c^{(5)}$. The purpose of spatial attention enhancement is to improve the spatial saliency of $E_c^{(5)}$. Firstly, the maxpool and avgpool feature maps of $E_c^{(5)}$ are obtained along the channel axis. Two 2D matrices $E_{\max}^{(5)}$ and $E_{\text{avg}}^{(5)}$ are obtained. Then $E_{\max}^{(5)}$ and $E_{\text{avg}}^{(5)}$ are concatenated. The spatial attention weight map Map_S is obtained through a 7 × 7 convolution and sigmoid activation function. The feature matrix $E_c^{(5)}$ is multiplied by Map_S to get $E_{\text{CBFE}}^{(5)}$, which will be used as the input of stage 4. The whole process can be described as:

$$Map_{\text{CA}} = \sigma\left(C_{1d}\left(GAP\left(E^{(5)}\right)\right)\right) \tag{1}$$

$$E_c^{(5)} = Map_{\text{CA}} \otimes E^{(5)} \tag{2}$$

$$Map_S = \sigma\left(C_{7\times7}\left(\left[avg_c\left(E_c^{(5)}\right);\max{}_c\left(E_c^{(5)}\right)\right]\right)\right) \tag{3}$$

$$E_{\text{CBFE}}^{(5)} = Map_S \otimes E_c^{(5)} \tag{4}$$

where C_{1d} represents 1D convolution, σ denotes sigmoid activation function, ⊗ denotes element-wise multiplication, avg_c and $\max_c$ represent average and maximum operations along the channel axis, and $C_{7\times7}$ represent 7 × 7 convolution.

3.1.2 Decoder

TENet decoder framework refers to CTDNet and includes three paths: Semantic Path, Spatial Path and Boundary Path. TENet enhances abstract semantics, spatial features and boundary features respectively in these three paths.

1) Semantic Path

In Semantic Path, we need to fuse the feature matrix $E^{(5)}$ of Stage5 with the feature matrix $E^{(4)}$ of Stage4 to complete the decoding process. $E^{(5)}$ is a high-level semantic feature, which contains the overall context texture information. In order to reduce the computation, E5 feature matrix passthrougha 1 × 1 convolution and is compressed to 64 channels. Since the spatial resolution of $E^{(5)}$ is only one-half of $E^{(4)}$, the convoluted $E^{(5)}$ needs to be upsampled once in order to fuse $E^{(4)}$. The whole process can be described as:

$$E_1^{(5)} = Up\left(C_{1\times1}\left(E^{(5)}\right)\right) \tag{5}$$

$$E_1^{(4)} = C_{1\times1}\left(E^{(4)}\right) \tag{6}$$

where $C_{1\times1}$ represents 1 × 1 convolution, and Up represents upsampling.

Next, semantic feature fusion is performed for $E_1^{(4)}$ and $E_1^{(5)}$ with 64 channels. Here, we compare three semantic feature fusion schemes. The first is FFM [24], and the second and third semantic fusion schemes are proposed in this paper.

(1) Feature Fusion Module (FFM)

FFM can fuse two groups of features. The structure is shown in Fig. 1. In FFM, $E_1^{(5)}$ and $E_1^{(4)}$ are fused by element-wise multiplication to amplify the features. The fused features pass through two 3 × 3 convolution layers to obtain the final feature matrix. Each convolution in FFM is followed by a batch normalization and ReLU function activation. The process of FFM can be formulized as:

$$D_1 = \text{FFM}(E_1^{(5)}, E_1^{(4)}) = C_{3\times3}\left(C_{3\times3}\left(E_1^{(5)} \otimes E_1^{(4)}\right)\right) \tag{7}$$

where D_1 represents the final fused features obtained after FFM.

(2) Channel Attention Enhanced Feature Fusion Model (CAEFFM)

$E^{(5)}$ and $E^{(4)}$ are important semantic features. In attention analysis, features that attract people's attention need to be focused, hence $E^{(5)}$ and $E^{(4)}$ should be enhanced.CAEFFM enhances the features after the fusion of $E^{(5)}$ and $E^{(4)}$ with channel attention module. The processof CAEFFM can be described as:

$$E_{1\text{mul}} = C_{3\times3}\left(E_1^{(5)} \otimes E_1^{(4)}\right) \tag{8}$$

$$\begin{aligned} CAmap_1 &= \sigma\left(\text{MLP}\left(\text{avg}_c\left(E_{1\text{mul}}\right)\right) + \text{MLP}\left(\max{}_c\left(E_{1\text{mul}}\right)\right)\right) \\ &= \sigma\left(W_1\left(W_0\left(\text{avg}_c\left(E_{1\text{mul}}\right)\right)\right) + W_1\left(W_0\left(\max{}_c\left(E_{1\text{mul}}\right)\right)\right)\right) \end{aligned} \tag{9}$$

$$D_1 = \text{CAEFFM}\left(E_1^{(5)}, E_1^{(4)}\right) = C_{3\times3}\left(E_{1\text{mul}} \otimes CAmap_1\right) \tag{10}$$

where D_1 represents the features obtained after CAEFFM. avg_c and $\max_c$ represent average and maximum operations along the channel axis. MLP represents the multi-layer perceptron, and W_0 and W_1 represent the weight of the multi-layer perceptron [28]. In CAEFFM, $E_1^{(5)}$ and $E_1^{(4)}$ are fused by element-wise multiplication. The fused features pass through a 3 × 3 convolution layers to get the feature matrix $E_{1\text{mul}}$. Then the channel attention module [28] is performed on $E_{1\text{mul}}$ followed by a 3 × 3 convolution to obtain the feature matrix D_1. Each convolution is followed by a block normalization and ReLU function activation.

(3) Spatial Attention Enhanced Feature Fusion Model (SAEFFM)

In addition to the features that easily attract human eye attention and need to be enhanced, the spatial locations that attract human eye attention also need to be enhanced.

SAEFFM applies spatial attention module to multi feature fusion. The process of SAEFFM can be described as:

$$E_{1\text{mul}} = C_{3\times3}\left(E_1^{(5)} \otimes E_1^{(4)}\right) \quad (11)$$

$$SAmap_1 = \sigma\left(C_{7\times7}\left(\left[\text{avg}_s\left(E_{1\text{mul}}\right); \max_s\left(E_{1\text{mul}}\right)\right]\right)\right) \quad (12)$$

$$D_1 = \text{SAEFFM}\left(E_1^{(5)}, E_1^{(4)}\right) = C_{3\times3}\left(E_{1\text{mul}} \otimes SAmap_1\right) \quad (13)$$

where avg_s and $\max_s$ represent average and maximum operations along the spatial axis. $E_{1\text{mul}}$ generate two different spatial context feature matrices by using average-pooling and max-pooling operations along the spatial axis. We concatenate these two feature matrices and compute the spatial attention map $SAmap_1$ by a 7 × 7 convolution and a sigmoid function. $SAmap_1$ and $E_{1\text{mul}}$ are fused by element-wise multiplication followed by a 3 × 3 convolution to obtain the final fusion matrix D_1. Each 3 × 3 convolution here is followed by a block normalization and ReLU function activation.

TENet uses ground truth to supervise the Semantic Path to guide the learning of the network. Compared with CTDNet, we do not embed a Global Average Pooling (GAP) layer on $E^{(5)}$. This is because although the receptive field is increased by GAP operation, the feature map is compressed to one pixel point. The local position information will be lost when an image is reconstructed by the pixel point. Hence, GAP operation will reduce the accuracy of object segmentation. We also do not use ground truth to supervise $E^{(5)}$. This is because $E^{(5)}$ are semantic features, while ground truth are spatial features. These are two completely different features. Such supervision may mislead the learning direction of the network. Therefore, the Semantic Path part of TENet is simpler than CTDNet.

2) Spatial Path

Spatial Path is used to enhance the features of the middle layer. The spatial resolution of $E^{(3)}$ is only one-eighth of the input image. $E^{(3)}$ passes through the Spatial Attention Module (SAM) [14] to obtain the features D_2. In fact, a spatial attention mechanism is added to the SAM, which enhances the spatial features of $E^{(3)}$. In SAM, two single-channel feature maps F_{avg} and F_{max} are generated by $E^{(3)}$ using average-pooling and max-pooling operationsalong the channel axis. Then we concatenate F_{avg} and F_{max} and compute the attention map Map_{SAM} by a 3 × 3 convolution and a sigmoid function. Each feature map in $E^{(3)}$ is multiplied by the corresponding weight in Map_{SAM}, and then the feature matrix obtained pass through a 3 × 3 convolution to obtain the final output D_2 with 64 channels. The process of SAM can be described as:

$$Map_{\text{SAM}} = \sigma\left(C_{3\times3}\left(\left[\text{avg}_c\left(E^{(3)}\right); \max_c\left(E^{(3)}\right)\right]\right)\right) \quad (14)$$

$$D_2 = C_{3\times3}\left(Map_{\text{SAM}} \otimes\right)E^{(3)} \quad (15)$$

D_2 contains spatial information, while D_1 contains semantic information. Spatial information and semantic information are two completely different features. The spatial resolutions of D_2 and D_1 are 1/8 and 1/16 of the input image. We use the Cross Aggregation Module (CAM) proposed in the paper [14] to fuse D_2 and D_1. In CAM, D_1subsampled to the same resolution as D_2 by bilinear interpolation. We apply a 3 × 3 convolution with stride 2 followed by a batch normalization and a ReLU activation function to D_2. The resolution of $D_2^{'}$ is the same as D_1. $D_2^{'}$ and D_1 are fused by element-wise multiplication to obtain C_1. D_2 and $D_1^{'}$ are fused by element-wise multiplication to obtain C_2. The resolution of C_2 is the same as D_2, and the resolution of C_1 is the same as D_1. Then C_1 and C_2 pass through a 3 × 3 revolution respectively followed by a batch normalization and a ReLU activation function to obtain $C_1^{'}$ and $C_2^{'}$. The upsampled $C_1^{'}$ and $C_2^{'}$ are fed into FFM to obtain D_{12}. The above process can be described as:

$$D_1^{'} = Up\left(D_1\right), D_2^{'} = F_{3\times3,2}\left(D_2\right) \quad (16)$$

$$C_1 = D_1 \otimes D_2^{'}, C_2 = D_2 \otimes D_1^{'} \quad (17)$$

$$C_1^{'} = F_{3\times3}\left(C_1\right), C_2^{'} = F_{3\times3}\left(C_2\right) \quad (18)$$

$$D_{12} = \text{FFM}\left(Up\left(C_1^{'}\right), C_2^{'}\right) \quad (19)$$

where Up represents upsampling and $F_{3\times3,2}$ represents a 3 × 3 convolution with stride 2.

3) Boundary Path

Boundary Path is used to decode $E^{(2)}$. The spatial resolution of $E^{(2)}$ is one fourth of the input map. Low-level features $E^{(2)}$ contain a lot of spatial information. In order to fuse spatial features and semantic features, FFM is still used to fuse $E^{(2)}$ and D_1 which is the output of the Semantic Path. We apply a 1 × 1 convolution followed by a batch normalization and a ReLU function to D_1 to adjust the number of channels. The result after convolution is fused with the upsampled result of D_1 by FFM to obtain D_3. The whole process can be formulated as:

$$D_3 = \text{FFM}\left(Up\left(D_1\right), F_{1\times1}\left(E^{(2)}\right)\right) \quad (20)$$

In order to enhance the boundary segmentation accuracy, TENet adds the learning of object boundaries like CTDNet [14].

4) Through the above three steps, there are two outputs D_3 and D_{12}, which are the fused features of low-level and high-level features and those of mid-level and high-level features respectively. Since BRM [14] can better fuse the two types of features, TENet also uses BRM to fuse D_3 and D_{12}. In BRM, D_3 is fused with upsampled D_{12} by addition operation to obtain B_1. B_1 pass through global average pooling (GAP), that is, we apply average-pooling operation along the special axis to B_1, and then obtain a 1D vector. The 1D vector pass through a 1 × 1 convolution followed

by a batch normalization and a sigmoid function to adjust channel, by which 1D vector Map_{B1} is obtained. B_1 is multiplied by Map_{B1} to obtain an enhanced feature matrix B_2. We combine B_2 and B_1 by addition operation to further enhance the significant features. Finally, the combined features pass through two 3 × 3 convolution layers, which generates the final output D_0. Each 3 × 3 convolution is followed by a batch normalization and a ReLU activation function. The whole process is described as:

$$B_1 = Up\left(D_{12}\right) + D_3 \tag{21}$$

$$Map_{B1} = \sigma\left(F_{1\times1}\left(\mathrm{GAP}\left(B_1\right)\right)\right) \tag{22}$$

$$B_2 = Map_{B1} \otimes B_1 \tag{23}$$

$$D_0 = F_{3\times3}\left(F_{3\times3}\left(B_2 + B_1\right)\right) \tag{24}$$

3.2 Loss Function

During the training of TENet, there are four output feature matrices D_0, D_1, D_{12} and D_3. These four feature matrices pass through a 3 × 3 convolution and sigmoid function and then are converted to the single channel prediction masks. We adopt the combination of three loss functions, one is regression focal loss [31], and the other two are IoU loss and BCE loss [14].

The characteristic of Regression focal loss (RF loss)is that it can deal with class imbalance problems. Because the dataset we use has class imbalance problem, RF loss is adopted in TENet. The definition of RF loss formula is as follows:

$$L_{\mathrm{RF}}\left(P,G\right) = \frac{-1}{N}\sum_{i=1}^{H}\sum_{j=1}^{W}\begin{cases}\left(1-P\left(i,j\right)\right)^{\alpha} log\left(P\left(i,j\right)\right), \\ \quad \text{if } P\left(i,j\right)=1 \\ \left(1-G\left(i,j\right)\right)^{\beta}\left(P\left(i,j\right)\right)^{\alpha} log\left(1-P\left(i,j\right)\right), \\ \quad \text{if } P\left(i,j\right)=0\end{cases} \tag{25}$$

where P denotes the prediction mask, G represents the ground truth, H and W represent the row number and column number of the ground truth, $P(i, j)$ represents the probability that the pixel of the ith row and the j-th column is predicted as objects, and $G(i, j)$ represents the value of the ith row and the j-th column in G. N represents the total number of pixels covered by the object in G. α and β are the hyperparameters of focal loss [32], which is used to reduce the weight of easily classified samples. Their values are 2 and 4 [31].

IoU loss indicates the overall similarity between the prediction mask and the ground truth. The IoU loss formula is as follows:

$$L_{\mathrm{IoU}}\left(P,G\right) = 1-\frac{\sum_{i=1}^{H}\sum_{j=1}^{W}P\left(i,j\right)G\left(i,j\right)}{\sum_{i=1}^{H}\sum_{j=1}^{W}\left(P\left(i,j\right)+G\left(i,j\right)-P\left(i,j\right)G\left(i,j\right)\right)} \tag{26}$$

RF loss and IoU loss have their own advantages, thus for D_0, D_1, and D_{12}, we use the combination of RF loss and IoU loss to supervise their prediction masks. The combined loss function is as follows:

$$L^{(k)} = \gamma L_{\mathrm{RF}}^{(k)} + L_{\mathrm{IoU}}^{(k)} k \in \left\{D_0, D_1, D_{12}\right\} \tag{27}$$

where γ is the weight used to balance the two loss functions. When γ is 0.6, the experimental result is the best.

In order to enhance the accuracy of segmented object contour, we use the boundary map of ground truth to supervise the prediction of D_3. Here, BCE loss [14] is used as the loss function of this supervision, and the formula is as follows:

$$L_{\mathrm{BCE}}\left(P,G\right) = -\sum_{i=1}^{H}\sum_{j=1}^{W}\left[G\left(i,j\right)\log\left(P\left(i,j\right)\right)+\left(1-G\left(i,j\right)\right)\log\left(1-P\left(i,j\right)\right)\right] \tag{28}$$

According to Eqs. (26), (27) and (28), the total loss function is:

$$L = L_{\mathrm{BCE}} + \sum_{k\in\left\{D_0, D_1, D_{12}\right\}} \mu_k L^{(k)} \tag{29}$$

where μ_k represents the weight of the corresponding three loss functions, and we set them to 1.0, 0.5 and 0.25 respectively.

4 EXPERIMENTS

The lightweight TENet proposed in this paper is compared with the current excellent CTDNet. We mainly compare their effectsin salient object detection and industrial optical inspection.

4.1 Datasets

The results of salient object detection and image segmentation are both a binary image, thus we test the TENet from two aspects: salient object detection and industrial optical inspection. In the experiment, we use two test datasets, one is the salient detection dataset DUTS [33] and the other is the industrial optical inspection dataset DAGM [34]. There are 10553 training images and 5019 test images in the DUTS. DAGM is mainly used to detect miscellaneous defects on various statistically textured backgrounds. This dataset has 10 types of textures, and each type of texture contains non-defective images and defective images.

4.2 Evaluation Metrics

Mean Absolute Error (MAE) is generally used as the evaluation metrics for salient object detection algorithm, but TENet will also be used in industrial optical inspection. Salient object detection and industrial optical inspection can be considered as object segmentation task. The evaluation metrics of object segmentation algorithm generally uses Mean Intersection over Union (MIoU) similar to MAE. Therefore, this paper uses MIoU as the evaluation metrics of the algorithm.MIoU represents the coincidence ratio of the intersection and union of two sets. When used to evaluate object segmentation algorithms, it represents the coincidence ratio of the intersection and union of the real segmentation results and the predicted segmentation results.

$$MIoU = \frac{1}{k}\sum_{i=1}^{k}\frac{P\cap G}{P\cup G} \tag{30}$$

where, k represents the number of classes, P represents the prediction result, and G represents the ground truth. For the object segmentation model, the larger the MIoU value, the better the segmentation effect of the model. When the MioU value is equal to 1, it means that the predicted value is completely coincident with the real value.

4.3 Implementation Details

We use MobileNetV3 as the backbone network. All training images are resized to 400 × 400 with random cropping to feed into model. We use adaptive moment estimation optimizer with the cosine annealing decay learning rate strategy to train our model and training epoch is 400.

4.4 Results Analysis

In order to test the feasibility of the algorithm used in industry, only low-performance CPU is used to test the model effect in the experiment. The computer configuration for the test is Intel i7-7700 (3.6 GHz) and 8 GB memory. In order to take the accuracy and speed of the model into consideration, the Trilateral Enhanced Network in Fig. 1 is partially modified several times in this paper to obtain multiple variant networks. All the networks are compared, and all experimental data are shown in Tab. 1 and Tab. 2.

Table 1 Experimental results of salient object detection (DUTS dataset)

	MIoU	FPS
CTDNet_r18 [14]	0.587	3.13
CTDNet-MobileNetV3	0.601	**3.33**
TENet without CBFEM	0.601	**3.33**
TENet-CAEFFM without CBFEM	0.612	**3.33**
TENet-SAEFFM without CBFEM	0.607	**3.33**
TENet with CBFEM	**0.615**	3.23

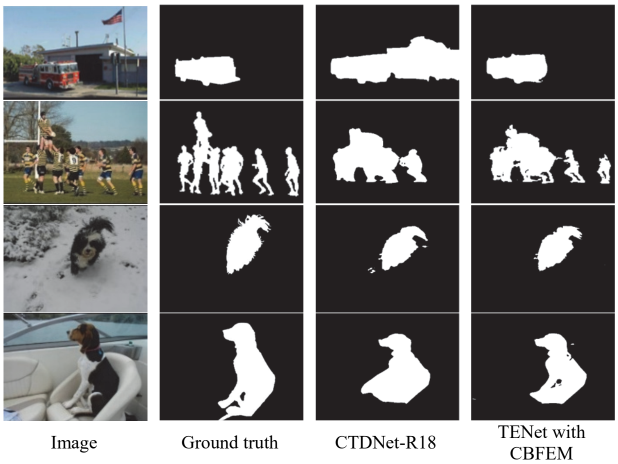

Figure 3 Comparison of algorithm results on DUTS dataset

The first column in Tab. 1 represents the algorithms for comparison. The second column is the MIoU value of DUTS dataset using different algorithms. The third column is the frame rate (Frames per Second) when only a single CPU (Intel i7-7700 3.6GHz) is used for detection. The first line is the evaluation metrics. The second line shows the experimental results of DUTS dataset using CTDNet-R18 [14]. The result in line 3 is produced by CTDNet framework where MobileNetV3 is used as the backbone network and thus this network is called CTDNet-MobileNetV3. The fourth line is the result of Trilateral Enhanced Network (TENet) using MobileNetV3 as the backbone network proposed by us in Fig. 1, but the CBFEM module is not embedded after the stage 3 of the network. The fifth line is also TENet without embedded CBFEM, but the FFM of the Semantic Path in the network is replaced by CAEFFM. Line 6 is TENet without CBFEM, but FFM of Semantic Path in the network is replaced by SAEFFM. Line 7 shows the experimental results of the complete TENet with CBFEM embedded behind stage 3.

Table 2 Experimental results of industrial optical inspection (DAGM dataset)

	MIoU					FPS
	Class1	Class2	Class3	Class4	Class5	
TENet with CBFEM	**0.9641**	0.9438	0.9511	0.9643	0.9528	**3.23**
CTDNet-R18 [14]	0.9531	**0.9505**	**0.9594**	**0.9698**	**0.9564**	3.13
	Class6	Class7	Class8	Class9	Class10	
TENet with CBFEM	**0.9835**	0.9550	0.9460	0.9748	**0.9677**	**3.23**
CTDNet-R18 [14]	0.9648	**0.9718**	**0.9480**	**0.9813**	0.9665	3.13

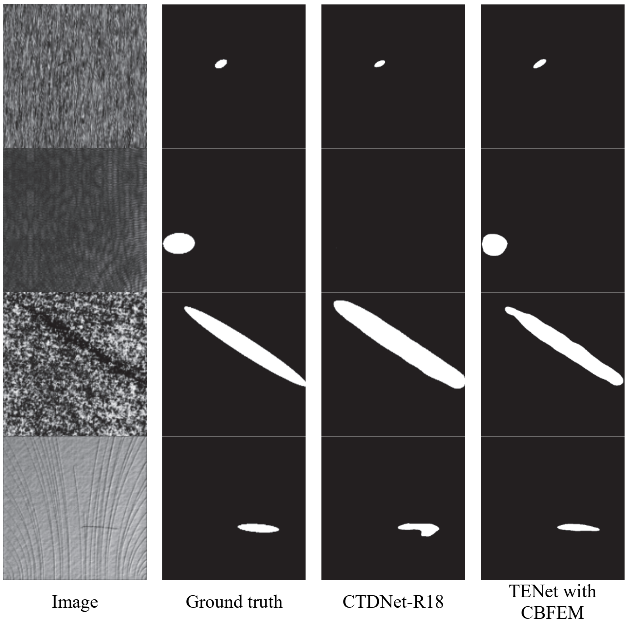

Figure 4 Comparison of algorithm results on DAGM dataset

It can be seen from columns 3 of Tab. 1 that CTDNet-R18 has the slowest speed, 6% slower than the fastest speed, when the CPU is used to detect salient objects. The second slowest is TENet, because the network embeds CBFEM behind stage 3, which increases the complexity of the network, resulting in its speed being 3% slower than the fastest algorithm. The backbone networks of the other four algorithms all use MobileNetv3. Because the number of network parameters does not change much, the algorithm speed is the same and the fastest. The algorithm based on TENet proposed in this paper performs better on DUTS dataset than that based on CTDNet, and the result of CTDNet-R18 is the worst. The detection result based on TENet-CAEFFM is better than that of TENet-SAEFFM, which indicates that the channel attention module is better than the spatial attention module in the high-level semantic feature fusion. It also proves that the feature matrix generated by stage 4 and stage 5 contains relatively less spatial information. The complete TENet embedded with CBFEM performs best on the DUTS dataset, and the MIoU value reaches the highest value of 0.615, which is 4.77% higher than CTDNet-R18. Fig. 3 shows the experimental results of partial images of TENet and CTDNet-R18 on DUTS dataset.

Tab. 2 compares the results of industrial optical inspection generated by TENet and CTDNet-R18, which is partly illustrated by Fig. 4. The results of TENet embedded with CBFEM in Class 1, Class 6 and Class 10 are better than those of CTDNet-R18, while the results of other seven classes are worse than those of CTDNet-R18. The average frame rate (Frames Per Second) of TENet still leads by 3.19%. In the DAGM dataset, Class 1 contains more than ten kinds of cross-texture images. Class 10 presents images which have obvious thin lines from top to bottom. Class 6 is a common texture image similar to Gaussian noise. The remaining diagrams in Class 2-Class 5 and Class 7-Class 9 are ordinary texture images without fixed shapes. This shows that TENet performs worse than CTDNet for ordinary texture images when segmenting abnormal parts in texture images. Therefore, TENet has good performance for obvious texture images, which shows that TENet's performance is not vulnerable to the interference of obvious textures.

5 CONCLUSION

Inspired by CTDNet, this paper proposes Trilateral Enhanced Network, a lightweight model for salient object detection. CTDNet used ResNet18 to improve speed while maintaining performance. This paper utilizes MobileNetV3 as the backbone for even faster detection speeds than CTDNet. Additionally, embedded attention mechanisms in TENet further improve detection accuracy. Experiments demonstrate TENet's high speed and accuracy for detecting defects in textured industrial images. TENet maintains strong performance on textured salient images, indicating robustness to texture interference. With only 3.72M parameters, TENet has low hardware requirements, making it suitable for industrial applications. In summary, TENet achieves fast, accurate, and robust salient object detection in an efficient model. But there is one limitation to be considered: when detecting defects in ordinary texture images, TENet has slightly lower accuracy. In the future research, we will increase various texture image samples to further improve the algorithm's compatibility with different textures.

Acknowledgments

This work is supported by the Science and Technology Research Project of Jiangxi Province Education Department (No. GJJ161552) and Natural Science Foundation of Jiangxi Province (No. 20171BAB202032).

6 REFERENCES

[1] Jung, C. & Kim, C. (2012). A unified spectral-domain approach for saliency detection and its application to automatic object segmentation. *IEEE Transactions on Image Processing a Publication of the IEEE Signal Processing Society*, *21*(3), 1272-83. https://doi.org/10.1109/TIP.2011.2164420

[2] Chen, C., Li, Y., Li, S., Qin, H., & Hao, A. (2017). A novel bottom-up saliency detection method for video with dynamic background. *IEEE Signal Processing Letters*, *25*(2), 154-158. https://doi.org/10.1109/LSP.2017.2775212

[3] Jiang, H., Wang, J., Yuan, Z., Wu, Y., Zheng, N., & Li, S. (2013). Salient object detection: a discriminative regional feature integration approach. *2013 IEEE Conference on Computer Vision & Pattern Recognition.* https://doi.org/10.1109/CVPR.2013.271

[4] Cheng, M., Mitra, J., Huang, X., Torr, H. P., & Hu, S. (2015). Global contrast based salient region detection. *IEEE transactions on pattern analysis and machine intelligence.* https://doi.org/10.1109/TPAMI.2014.2345401

[5] Borji, A., Cheng, M., Hou, Q., Jiang, H., & Li, J. (2019). Salient object detection: a survey. *Computational Visual Media.* https://doi.org/10.1007/s41095-019-0149-9

[6] Wang, W., Lai, Q., Fu, H., Shen, J., Ling, H., & Yang, R. (2021). Salient object detection in the deep learning era: an in-depth survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence.* https://doi.org/10.1109/TPAMI.2021.3051099

[7] Tian, Z., Shen, C., Chen, H., & He, T. (2019). FCOS: Fully convolutional one-stage object detection. *2019 IEEE/CVF International Conference on Computer Vision (ICCV).* https://doi.org/10.48550/arXiv.1904.01355

[8] Chen, H., Sun, K., Tian, Z., Shen, C., & Yan, Y. (2020). Blendmask: top-down meets bottom-up for instance segmentation. *2020 IEEE Conference on Computer Vision and Pattern Recognition.* https://doi.org/10.48550/arXiv.2001.00309

[9] Lin, T. Y., Dollar, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. *IEEE Computer Society.* https://doi.org/10.48550/arXiv.1612.03144

[10] Wang, T., Zhang, L., Wang, S., Lu, H., & Borji, A. (2018). Detect globally, refine locally: A novel approach to saliency detection. *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3127-3135. https://doi.org/10.1109/CVPR.2018.00330

[11] Hou, Q., Cheng, M., Hu, X., Borji, A., Tu, Z. & Torr, P. (2019). Deeply supervised salient object detection with short connections. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *41*(4), 815-828. https://doi.org/10.1109/TPAMI.2018.2815688

[12] Hou, Q., Liu, J. J., Cheng, M., Borji, A., & Torr, P. (2018). Three birds one stone: a unified framework for salient object segmentation, edge detection and skeleton extraction. *arXiv:1803.09860.* https://doi.org/10.48550/arXiv.1803.09860

[13] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. *International Conference on Medical Image Computing and Computer-Assisted Intervention.* https://doi.org/10.48550/arXiv.1505.04597

[14] Zhao, Z., Xia, C., Xie, C., & Li, J. (2021). Complementary trilateral decoder for fast and accurate salient object detection.*MM '21: Proceedings of the 29th ACM International Conference on Multimedia.*

[15] Liu, J. J., Hou, Q., Cheng, M., Feng, J., & Jiang, J. (2019). A simple pooling-based design for real-time salient object detection. *2019 IEEE Conference on Computer Vision and Pattern Recognition.* https://doi.org/10.1109/CVPR.2019.00404

[16] Chen, S., Tan, X., Wang, B., Lu, H., & Fu, Y. (2020). Reverse attention based residual network for salient object detection. *IEEE Transactions on Image Processing*, *29*, 3763-3776. https://doi.org/10.1109/TIP.2020.2965989

[17] Zunair, H. & Hamza, A. B. (2021). Sharp u-net: depth wise convolutional network for biomedical image segmentation. *Computers in Biology and Medicine*. https://doi.org/10.48550/arXiv.2107.12461

[18] Bolya, D., Zhou, C., Xiao, F., & Lee, Y. (2019). YOLACT: Real-time instance segmentation. *International Conference on Computer Vision*. https://doi.org/10.48550/arXiv.1904.02689

[19] Tu, Z., Ma, Y., Li, C., Tang, J., & Luo, B. (2020). Edge-guided non-local fully convolutional network for salient object detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 99, 1-1. https://doi.org/10.48550/arXiv.1908.02460

[20] Chen, Z., Zhou, H., Lai, J., & Yang, L., & Xie, X. (2021). Contour-aware loss: boundary-aware learning for salient object segmentation. *IEEE Transactions on Image Processing*, *30*, 431-443. https://doi.org/10.1109/TIP.2020.3037536

[21] Hu, X., Fu, C., Zhu, L., Wang, T., & Heng, P. (2021) SAC-Net: spatial attenuation context for salient object detection. (2020). *IEEE Transactions on Circuits and Systems for Video Technology*, *31*(3), 1079-1090. https://doi.org/10.1109/TCSVT.2020.2995220

[22] Li, Z., Lang, L., Liang, L., Zhao, J., Feng, S., Hou, Q., & Feng, J. (2022). Dense attentive feature enhancement for salient object detection. *IEEE Transactions on Circuits and Systems for Video Technology*, *32*(12), 8128-8141. https://doi.org/10.1109/TCSVT.2021.3102944

[23] Wang, S. (2022). Learning nonlinear feature mapping via constrained non-convex optimization for unsupervised salient object detection. *IEEE Access*, *10*, 40743-40752. https://doi.org/10.1109/ACCESS.2022.3166986

[24] Liu, Y., Wang, P., Gao, Y., Liang, Z., & Lau, R. (2021). Weakly-supervised salient object detection with saliency bounding boxes. *IEEE Transactions on Image Processing*, *30*, 4423-4435. https://doi.org/10.1109/TIP.2021.3071691

[25] Jaderberg, M., Simonyan, K., Zisserman, A., & Kavukcuoglu, K. (2015). Spatial transformer networks. *Neural Information Processing Systems (NIPS 2015).* MIT Press. https://doi.org/10.48550/arXiv.1506.02025

[26] Jie, H., Li, S., & Gang, S. (2018). Squeeze-and-Excitation Networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. https://doi.org/10.1109/CVPR.2018.00745

[27] Wang, Q., Wu, B., Zhu, P., Li, P., & Hu, Q. (2020). ECA-Net: Efficient channel attention for deep convolutional neural networks. *2020 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. https://doi.org/10.1109/CVPR42600.2020.01155

[28] Woo, S., Park, J., Lee, J., & Kweon, I. S. (2018). CBAM: Convolutional block attention module. *2018 European Conference on Computer Vision (ECCV)*. https://doi.org/10.48550/arXiv.1807.06521

[29] Howard, A., Sandler, M., Chu, G., Chen, L. C., Chen, B., Wang, W., Zhu, Y., Pang, R., & Vasudevan, V. (2020). Searching for MobileNetV3. *2020 IEEE/CVF International Conference on Computer Vision (ICCV)*. https://doi.org/10.1109/ICCV.2019.00140

[30] Lin, M., Chen, Q., & Yan, S. (2014). Network in network. *International Conference on Learning Representations*. https://doi.org/10.48550/arXiv.1312.4400

[31] Zhou, X., Wang, D., & Krhenbühl, P. (2019). Objects as Points. https://doi.org/10.48550/arXiv.1904.07850

[32] Lin, T. Y., Goyal, P., Girshick, R., He, K., & P Dollár. (2017). Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, *99*, 2999-3007. https://doi.org/10.48550/arXiv.1708.02002

[33] Xiang, R. (2018, January 1). The DUTS image dataset. *Saliency detection*. http://saliencydetection.net/duts/

[34] Wieler, M., Hahn, T., & Hamprecht, F. (2007). Weakly supervised learning for industrial optical inspection.

Contact information:

Fengchang FEI
College of Modern Economics and Management,
Jiangxi University of Finance and Economics,
Nanchang, China, 330032
E-mail: feifengzhang@jxufe.edu.cn

Wei LIU
Lenovo Research,
Shenzhen, China, 518057
E-mail: liutr01@hotmail.com

Lei SHU
(Corresponding author)
School of Information Technology,
Jiangxi University of Finance and Economics,
Nanchang, China, 330032
E-mail: shulei@jxufe.edu.cn