AANet: Adjacency auxiliary network for salient object detection

Xialu Li [1], Ziguan Cui[1,*], Zongliang Gan[1], Guijin Tang[1], and Feng Liu[2]
[1] College of Telecommunications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing, 210003- China
[e-mail: 1260610097@qq.com, cuizg@njupt.edu.cn, ganzongliang@gmail.com, tanggj@njupt.edu.cn]
[2] College of Educational Science and Technology, Nanjing University of Posts and Telecommunications, Nanjing, 210003- China
[e-mail: liuf@njupt.edu.cn]
*Corresponding author: Ziguan Cui

Received May 25, 2021; revised August 15, 2021; accepted Septebmer 15, 2021; published October 31, 2021

Abstract

At present, deep convolution network-based salient object detection (SOD) has achieved impressive performance. However, it is still a challenging problem to make full use of the multi-scale information of the extracted features and which appropriate feature fusion method is adopted to process feature mapping. In this paper, we propose a new adjacency auxiliary network (AANet) based on multi-scale feature fusion for SOD. Firstly, we design the parallel connection feature enhancement module (PFEM) for each layer of feature extraction, which improves the feature density by connecting different dilated convolution branches in parallel, and add channel attention flow to fully extract the context information of features. Then the adjacent layer features with close degree of abstraction but different characteristic properties are fused through the adjacent auxiliary module (AAM) to eliminate the ambiguity and noise of the features. Besides, in order to refine the features effectively to get more accurate object boundaries, we design adjacency decoder (AAM_D) based on adjacency auxiliary module (AAM), which concatenates the features of adjacent layers, extracts their spatial attention, and then combines them with the output of AAM. The outputs of AAM_D features with semantic information and spatial detail obtained from each feature are used as salient prediction maps for multi-level feature joint supervising. Experiment results on six benchmark SOD datasets demonstrate that the proposed method outperforms similar previous methods.

Keywords: Salient Object Detection, Deep Learning, Convolutional Neural Network, Multi-scale Information, Multi-level Feature Fusion

This work was supported by the National Natural Science Foundation of China (61501260), 1311 Talent Program of NJUPT, and Research Fund of Nanjing University of Posts and Telecommunications (NY220215).

http://doi.org/10.3837/tiis.2021.10.014 **ISSN : 1976-7277**

1. Introduction

Visual saliency detection is designed to detect the entire objects in the images that most attract people's attention [1]. In recent years, many researchers have studied this topic, and it is a very important preprocessing operation in computer vision tasks such as image retrieval [2], scene classification [3], target tracking [4], and person re-identification [5].

Recently, with the rapid development of machine learning technology, the use of deep learning and CNN to solve various image processing problems has made good achievements [6]. Some researchers have also applied the full convolution network (FCN) to saliency detection [7-9], FCN is adopted to build models to obtain high-level semantic information for SOD. Some previous SOD methods [10-12] use a series of convolutional with single-scale and max pooling operations to obtain deep features, in which the receptive field of features is limited due to frequent pooling operations in convolution neural networks. With that reason and the scale and position of salient objects are variable, the structural information of images is often lost, resulting in inaccurate detection of target boundary areas. Multi-scale convolution features can be added to the design of saliency detection model to obtain better salient objects areas and structural characteristics. For example, methods such as [13-15] extend the spatial pyramid pooling [16] to extract multi-scale context information of the image. By using convolution kernels with atrous, the receptive field can be enlarged by atrous spatial pyramid pooling (ASPP). However, it only realizes the feature change and integration in space. Besides, the resolution of features on the spatial scale axis is not dense enough either. When the scene is complex, the extracted features may not be able to capture salient objects and their boundary accurately. To solve the above problems, we proposed parallel connection feature enhancement module (PFEM), which captures multi-scale context information by realizing the feature connection between space and channel in the multi-scale region of the feature maps.

In addition, most of the existing SOD methods are based on the encoder-decoder architecture [9, 10, 17, 18, 19]. From the point of view of the fusion method of different features, skip connection [7] is used to introduce features of the early encoding layers into the later decoding layer. It is indeed possible to recover the spatial precision by the earlier feature representation, which may be lost and blurred at the deeper layer. If only a single layer feature is fused into the decoding layer to mark the categories in the image, it may lead to some ambiguous representations in the decoding process and reduce the prediction quality. It is known that, in general, high-level features do not contain fine spatial details. As shown in **Fig. 1**(a), the information conveyed cannot express the target's detail of limbs and "round" head shape, which will cause terrible errors in identifying the target shape as "human". On the other hand, although the low-level features can obtain sharp boundaries, they cannot accurately capture salient objects. For **Fig. 1**(b), in addition to the nearby "car" highlighted, the outline of the distant "island" is also extracted. Therefore, in this paper, the adjacency auxiliary module (AAM) is proposed to process the feature information of the two adjacent layers. The latter layer's information may distinguish between the azimuth and the rough outline of the target, and the features of the former layer can refine the accuracy of the boundary of the target.

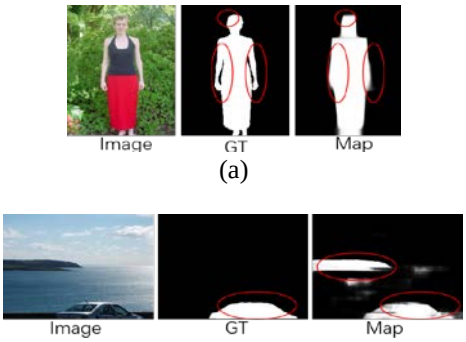


Fig. 1. Visualization of high-and low-level feature prediction (Map obtained only by using a single-layer information. (a) Visualization of failure to get fine detail (b) Visualization of failure to get fine semantic feature)

To validate the performance of our proposed adjacency assistant network (AANet), we show experimental results on six popular SOD datasets. A series of ablation experiments were conducted to assess the contribution of each module. From the quantitative metric and visualizations of the experimental results, it can be seen that our adjacent auxiliary network (AANet) can obtain better saliency maps compared to the previous state-of-the-art methods. In short, our main contribution can be summed up as:

1) We propose an adjacent auxiliary network (AANet) based on multi-scale feature fusion, which includes parallel connection feature enhancement module (PFEM), adjacent auxiliary module (AAM), and adjacent decoder module (AAM_D).

2) The PFEM has better multi-scale extraction ability by combining channel attention and multi-scale features. What differs from the ASPP module is that it not only focuses on the spatial scale transformation, but also connects the channel attention features and makes serial connections between the branches, which can obtain more dense features.

3) We design AAM to enable the complementary fusion of information between the two adjacent layers of features with different characteristics. So that semantic ambiguity and noise in the features can be suppressed and explicit confidence can be provided for the correct semantic target detection. On the basis of AAM, we proposed adjacency decoding module (AAM_D), which fully integrates and refines the obtained features, and achieves the goal of refining the salient objects.

4) Our network adopts multi-level feature joint supervision to improve the optimization ability. The experimental results show that our AANet has good performance on six salient detection datasets, which verifies the superiority and effectiveness of the proposed method.

The following paragraphs of this paper are organized as follows: Section 2 summarizes the related works. Section 3 describes the salient object detection framework and specific details of the AANet. Section 4 discusses the performance of our model compared with the previous state-of-the-art methods and Section 5 is the conclusion of this paper.

2. Related Work

Conventional salient detection methods are mainly based on low-level hand-crafted features, such as color contrast [20], background contrast [21], center prior [22], and so on. Wei et al. [23] use two low-level features of color and luminance to calculate the contrast between the current pixel and its neighborhood of different sizes, to determine the salient value of the pixel. In addition, researchers build graph models to calculate the salient prediction of image pixels

[24, 25, 26]. For example, in [25], seeds for manifold sorting are obtained by using background weight maps, and a third-order smoothness structure is also designed to strengthen the performance of manifold sorting. However, these methods which only focus on low-level features cannot capture the rich semantic information in the image, and the detection in complex scenes may fail.

Recently, SOD methods using deep neural networks have developed rapidly. Convolution neural network (CNN) is used to extract multi-level and multi-scale features, so that the model can capture the salient area accurately, and has a good expression in speed and performance. Inspired by image semantic segmentation, Zhang et al. [17] introduced FCN network framework in semantic segmentation into saliency detection and achieved better detection results. Lee et al. [27] proposed a network framework combining low-level features by handcrafted and high-level features by backbone network, and then combines them to detect the salient maps by concatenation and convolution. Zhang et al. [12] use an encoder-decoder framework for better saliency prediction. To learn uncertain features, a 'redefined dropout' is added to the encoder and the decoder is also designed with hybrid up-sampling scheme to avoid checkerboard artifacts. However, the simplest single-stream transmission methods may gradually lose the spatial local information of the image due to the deepening of the network in the process of feature extraction, which will affect the final detection results, especially for the inaccurate detection of the salient target boundary.

To cope with the above problems, many algorithms adopt the way of side fusion. The so-called side fusion network refers to the fusion of multi-layer feature information of the backbone network for saliency prediction. Hou et al. [7] introduced the short connection to the skip-layer structures to detect salient maps, and added the output from higher level to the shallow output using short connection. In this way, lower side output can better locate salient areas with the help of higher-level features as well as lower-level features enrich the details of deeper side outputs. Zhang et al. [28] proposed to aggregate multi-level convolutional features for saliency detection. All levels of feature maps are integrated into multiple resolutions and rough semantic and detail information are combined to predict saliency maps. Wang et al. [29] proposed global recurrent localization network, which uses multi-scale convolution kernels to extract different scales context information, and then connects the information to decoding layers. And local context information for each spatial position can be adaptively learned because of the local boundary refinement network. Zhang et al. [30] designed symmetrical fully convolutional neural networks to learn complementary saliency features with the guidance of lossless feature reflection. This side fusion method is similar to the principle of U-Net [31] in semantic segmentation, which gradually fuses the rich spatial detail features of the bottom layers in the decoding stage to improve the fineness of the salient maps. Inspired by pyramid scene parsing network [32], Liu et al [33] designed a U-shaped architecture based on feature pyramid structure according to top-down and bottom-up design, which uses pyramid pooling module to capture global guidance information and help more accurately locate the salient objects. Zhang et al. [13] proposed a feature extraction module for multi-level feature mapping, and then design a bi-directional structure to deliver messages between features of different layers. The characteristics of representative traditional and CNN-based SOD models are summarized in **Table 1**. In short, how to obtain multi-layer information and multi-scale information and integrate all this information more effectively is a very important point in salient object detection.

Table 1. Characteristics of different methods for SOD

Algorithm	Type	Characteristics
RCRR [24]	traditional models	saliency reversion correction (RC) process, regularized random walk ranking model
DSC [26]	traditional models	using deformed smoothness constraint to find low contrast object in the label propagation model, refined map to refined coarse map
BMP [13]	CNN-based	using convolution layers with various reception fields, inter-level exchange through a gated bi-directional pathway
Amulet [28]	CNN-based	recursively embedding multi-level features maps
ELD [27]	CNN-based	concatenating high-level and encoded hand-crafted feature distance map
LFR [30]	CNN-based	using lossless feature reflection (FR) to guide a symmetrical FCN, weighted structural loss
C2Snet [18]	CNN-based	multi-task learning, adding a SOD branch to a contour detection model, contour and SOD branches progressively supervise and update each other
UCF [12]	CNN-based	FCN-based, R-dropout operation for uncertainty in encoder, hybrid up-sampling for smoothing
Capsal [19]	CNN-based	integrating semantic context from a captioning network
RFCN [10]	CNN-based	recurrent fully convolutional networks, and use segmentation tasks as supervision for training

Inspired by the above work, firstly, in order to obtain better multi-scale information, we propose PFEM, based on ASPP, to parallel add channel extraction features to transform on the spatial scale to get better multi-scale features. Then we designed the AAM, which complements and suppresses each other by making use of the different properties of low and high-level features to remove fuzzy interference information. This can overcome the ambiguity problem caused by the previous algorithm framework such as [29, 30, 33, 34] which directly integrates the single-layer features of the coding layer into the decoder for decoding. Finally, we designed the AAM_D to effectively use the refined features of multi-layer and multi-scale for saliency detection, so that the detection results have better accuracy of salient target area location and boundary segmentation.

3. The proposed method

3.1 AANet Network Structure

In this work, our model is based on the encoder-decoder architecture. ResNet-50 converges quickly and has small parameters relatively, for which we adopt it as the backbone network. To achieve the task of saliency detection, we have made some modifications to it. The last pooling layer and the fully connected layer are removed. The input image with size of H×W is fed into the backbone network to obtain multi-level feature mapping: {Conv-1, Res-2, Res-3, Res-4, Res-5}. The feature mapping f_5 obtained from Res-5 has the smallest spatial dimension $\frac{H}{2^5} \times \frac{W}{2^5}$. Specifically, according to the five feature maps with different resolutions, in order to further extract the scale information contained in different features, we designed the parallel connection feature enhancement module (PFEM). The output channel of different feature maps after reprocessing by the PFEM module is 128. Then, the adjacency auxiliary module (AAM) is designed to further obtain clear boundaries and consistent semantics. The feature map f_5 is transformed into a feature map with 256 channels by a 3×3 convolution operation, and we take it as the next layer input of the first adjacent auxiliary module. Finally, the adjacency auxiliary features and the feature mapping from the previous encoding layer are

fused by the adjacency decoder (AAM_D). The features from the last decoding layer are used to obtain our salient maps by 1×1 convolution layer and up-sampling operation. In addition, in order to better learn the loss, we output all the feature maps obtained from the four decoding modules as an auxiliary loss. The overall structure is illustrated in **Fig. 2**.

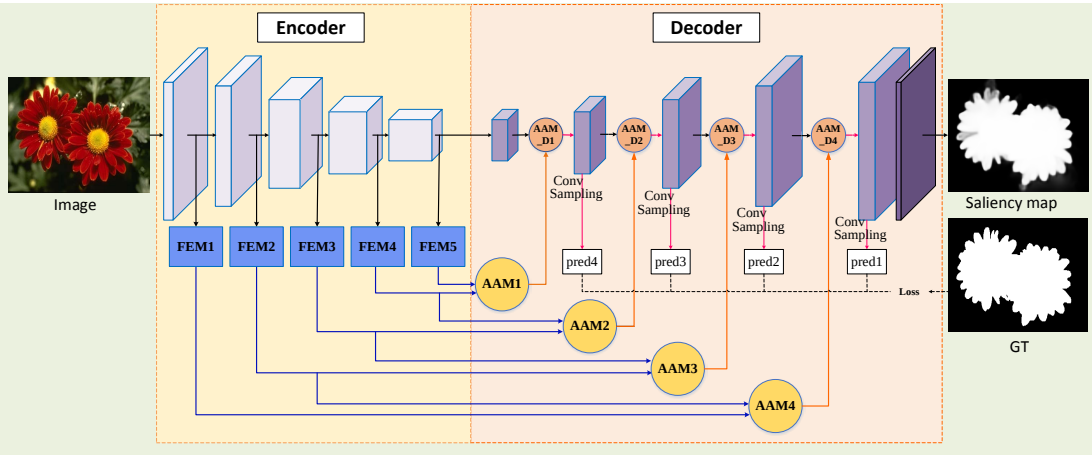


Fig. 2. The proposed AANet network structure

3.2 Parallel Connection Feature Enhancement Module

The capture of semantic information is very important for salient object detection. The simple convolution neural network model learns features of objects by stacking multiple convolution layers and pooling layers. There are large variations in the scales and position of salient objects in different images. If we directly encode and decode the features extracted from the backbone network to generate salient targets. The feature extraction is only a single-scale convolution and merging operation, which may result that some complex scene problems are handled poorly, leading to the problem of information loss. Some works have been done to extract multi-scale features from spatial pyramid pools and apply them to saliency detection, which uses dilated convolution of different sizes of kernels to obtain feature maps with different scale receptive fields. However, the insertion of "holes" will lead to the problem of sparse features. In addition, these transformations only extract spatial information from different channels, and may not be able to accurately convey the channel features. In order to solve these two problems, PFEM is designed along the backbone to learning multi-scale semantic information.

The module contains multiple dilated convolutions (we have adopted parallel connections for several dilated convolutions) and channel attention mechanisms. We extract five levels of feature mapping from the ResNet-50 network. For each level of feature maps, different scale information can be obtained by dilated convolution with different dilated rates. The output of the dilated convolution layer at the prior scale variation is fed to the next unexecuted dilated convolution branch, and then calculate the scale of another dilated convolution branch after element-wise addition. The previous multi-scale feature extraction methods, such as ASPP [16], use four dilated convolution layers to expand the receptive field without losing resolution and increasing the amount of computation. However, when the dilated rate is relatively large, the 3×3 filter no longer captures the global context effectively. So that our convolution kernels are not all the size of 3×3. We set the convolution kernels of the first dilated convolution to 1×1 and the other three to 3×3. In addition, our dilated rates are set to 1, 3, 5, 7, while those of ASPP are often 6, 12, 18, 24. Furthermore, as shown in **Fig. 3**, we do not simply adopt four ways to obtain different scale information $\{f_1^c, f_2^c, f_3^c, f_4^c\}$. We add the scale features

obtained from the previous branches to the input feature and then carry out the next scale transform. Through such feature connection, the neurons of each feature map can extract and purify semantic information from multiple scales. In this way, the feature information will be denser in terms of feature resolution, and the receiving field will be larger.

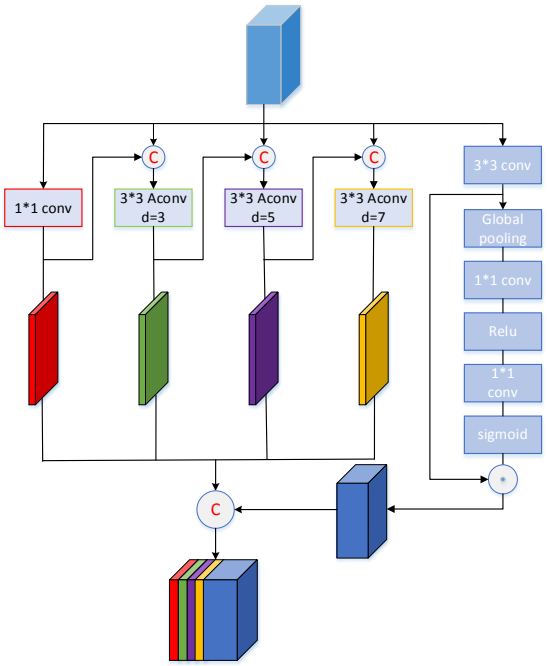


Fig. 3. Structure of PFEM

The above operation improves the effectiveness of multi-scale feature extraction in the spatial domain. Besides, we add channel-wise attention to make the feature pay more attention to the salient target from the channel. As shown in **Fig. 3**, the weight information on the channel is obtained by pooling, channel convolution, and sigmoid of the input feature. At the same time, it can be seen that the number of the input feature's channel is changed to 128 through a convolution operation before entering into the calculation of channel feature, the reason of which is to avoid the layer with fewer channels being submerged by the layer with more channels. What's more, it was found in our test that the target location was usually very accurate when the salient detection based on the ResNet-50 backbone network was carried out, while the target boundary detection was often not accurate enough. Therefore, when designing PFEM, we should appropriately reduce the number of high-level features channels and appropriately increase the number of low and middle-level features channels, so that the semantic location and boundary detection of significant targets can achieve better performance at the same time, and the computation and network parameters can also be reduced. And finally, we combine several features through cross-channel cascading to get enhanced features. The calculation formula of the whole process of PFEM can be written as:

$$
\begin{aligned}
f_i^{d_j} &= \begin{cases} \mathrm{Re}\,LU(BN(Aconv\_1(f_i))), & j=1 \\ \mathrm{Re}\,LU(BN(Aconv\_j(Concat(f_i, f_i^{d_{j-1}})))), & j=2,3,4 \end{cases} \\
f_i^{ca} &= \sigma(conv(\mathrm{Re}\,LU(conv(avg(f_i))))) \times conv(f_i) \\
f_i^{c} &= BN(Conv(Concat(f_i^{d_1}, ..., f_i^{d_4}, f_i^{ca})))
\end{aligned} \tag{1}
$$

where f_i is the feature mapping of each layer obtained from the backbone network, f_i^d is the output of each dilated convolution branch, f_i^{ca} indicates the output of the channel attention branch, and f_i^c is the output of PFEM, Aconv_1() is dilated convolution with different convolution kernels and dilated rates, ReLU() and $BN()$ are ReLU nonlinear and batch normalization operations respectively, Concat() is a cascading operation at the channel level. Details of the size and number of channels are shown in **Table 2**.

Table 2. Details of PFEM

Block	Input	Layer	Channel size	Output
FEM1	Conv-1 (72*72*64)	Aconv*4+CA	64→32,96→32, 96→32,96→32, 64→128	72*72*256
FEM2	Res-1 (72*72*256)	Aconv*4+CA	256→32,288→32, 288→32,288→32, 256→128	72*72*256
FEM3	Res-2 (36*36*512)	Aconv*4+CA	512→32,544→32, 544→32,544→32, 512→128	36*36*256
FEM4	Res-3 (18×18*1024)	Aconv*4+CA	1024→32,1056→32, 1056→32,1056→32, 1024→128	18*18*256
FEM5	Res-4 (9*9*2048)	Aconv*4+CA	2048→32,2080→32, 2080→32,2080→32, 2048→128	9*9*256

3.3 Adjacency Auxiliary Module

By adopting PFEM, we can capture effective context information of multi-scale and multi-receptive fields. If the resulting feature mapping is passed directly to the decoder for decoding, the spatial accuracy may be lost in the deep layer. However, if f_i is only connected with the corresponding decoder, as mentioned in Section 1, when the local edge information is blurred or the location information is not accurate, some information ambiguity in this layer is also transmitted to the decoding parts, which will result that ambiguity appears and lead to poor detection results, especially inaccurate detection of the target edge part.

We know that the low-level features have rich details but background noise, while the high-level features have better semantic information but the boundaries are fuzzy and rough. For this reason, we design the adjacency auxiliary module AAM to make adjacent layer features f_{i+1} and f_i supplement each other to strengthen the common effective information. So that the invalid or interfering information can be suppressed and the ambiguity can be resolved. Through the combination of the feature information of the latter layer with the feature of the

current layer, the ambiguity and noise can be filtered. The feature expression of saliency targets can be more accurate, and better saliency prediction can be obtained.

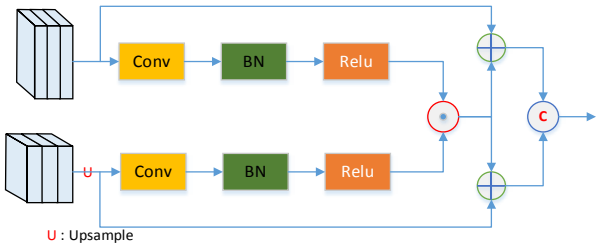


Fig. 4. Structure of AAM

Now explain how to obtain refinement features from the AAM, the structure figure is shown in **Fig. 4**. Obviously, the features of the outputs from PFEM are the same channel dimensions, but different in spatial dimensions. We conduct operation of up-sampling with a factor of 2 to the feature f_{i+1}^{c}, so a new feature map $f_{(i+1)^*}^{c}$ can be got, whose spatial dimensions is the same as f_i^c. Then we obtain the feature f_m^i by element-wise multiplication between $f_{(i+1)^*}^{c}$ and f_i^c. Lastly, the combined features are obtained by adding f_m^i with the two input features and cascaded them along channels, so that the features have better spatial and semantic information. The whole process could be shown as follows.

$$\begin{aligned} f_m^i &= BN(ReLU(conv(f_i^c))) \otimes BN(ReLU(conv(Up(f_{i+1}^c)))) \\ f_i^A &= Concat(Up(f_{i+1}^c) + f_m^i, f_i^c + f_m^i) \end{aligned} \tag{2}$$

where $conv$ refers to the convolution of kernel of 3×3, $\otimes$ is the operation of element-wise multiplication, $Up()$ is bilinear interpolation sampling. Details such as the size of input and output and the number of channels in AAM can be seen in **Table 3**.

Table 3. Details of AAM (k: kernel; s: stride; p: padding)

Block	Input	Layer	Channel size	Output
AAM1	FAM4, FAM5	Conv(k=3×3, s=1,p=1) Bilinear interpolation	256→128→256	18×18×256
AAM2	FAM3, FAM4	Conv(k=3×3, s=1,p=1) Bilinear interpolation	256→128→256	36×36×256
AAM3	FAM2, FAM3	Conv(k=3×3, s=1,p=1) Bilinear interpolation	256→128→256	72×72×256
AAM4	FAM1, FAM2	Conv(k=3×3, s=1,p=1) Bilinear interpolation	256→128→256	72×72×256

3.4 Adjacent Decoding Module

In addition, to effectively merge the multi-scale features obtained from AAM modules, we design adjacency auxiliary decoding (AAM_D) network based on AAM (shown in **Fig. 5**).

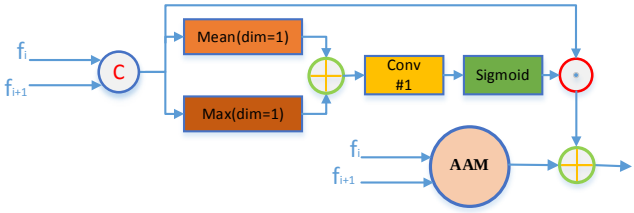


Fig. 5. Structure of AAM_D

As mentioned above, AAM modules can be used to solve problems in which feature information is ambiguous. We can also convolute the features obtained from the top layer of the encoder to obtain the appropriate number of channels, and then regard the output of adjacent auxiliary features as a group of adjacent features with different properties. Taking the first decoding module as an example, we get new features f_{5^*} from feature f_5 obtained from Res-5 by convolution operation with the kernel of 3×3 and channel number of 256. We treat it as one of the AAM's inputs: f_5^c, and treat the features f_4^A obtained from the adjacency auxiliary module as another AAM's input: f_4^c. And then they are fed into the AAM. The input features of the latter decoder are analogous in turn, with the difference that the output of the previous decoder is regarded as a higher-level feature. Furthermore, in order to obtain more detailed spatial information, we obtained the spatial attention weights after concatenating the higher features and lower features, and weighted them into the connection features. The process of obtaining spatial attention is to take the maximum value and average value of each spatial pixel of feature mapping on all channels respectively to get two features with the size of $h \times w \times 1$, cascade them through a convolution operation with the number of output channels being 1, and finally apply sigmoid to the features to get the spatial weight value. This module can further enhance the difference between salient objects and background, so as to obtain better salient prediction maps.

Table 4. Details of AAM_D

Block	Input	Layer	Channel size	Output
AAM_D1	Res-4*, AAM1	AAM+SA	256→512→256	18*18*256
AAM_D2	AAM2, AAM_D1	AAM+SA	256→512→256	36*36*256
AAM_D3	AAM3, AAM_D2	AAM+SA	256→512→256	72*72*256
AAM_D4	AAM4, AAM_D3	AAM+SA	256→512→256	72*72*256

Table 4 shows the details of the AAM_D. The salient maps outputted by AAM_D can be seen in **Fig. 6**. Among them, pred1-4 is the saliency maps predicted by different decoding layers, which can be seen that the AANet formed by adjacent decoders can gradually obtain clearer saliency maps. For example, pred4, which is predicted by high-level features, has fine semantic target identification and location ability, but the target edge detection is not accurate enough. While pred3, pred2, and pred1 gradually integrate more low-level information on the basis of high-level features, the detection of target edge becomes increasingly accurate. The pred1 incorporates the most layers of information and produces the most accurate and clear saliency map, which is the reason that we ended up using the output of the last decoder as our final saliency maps.

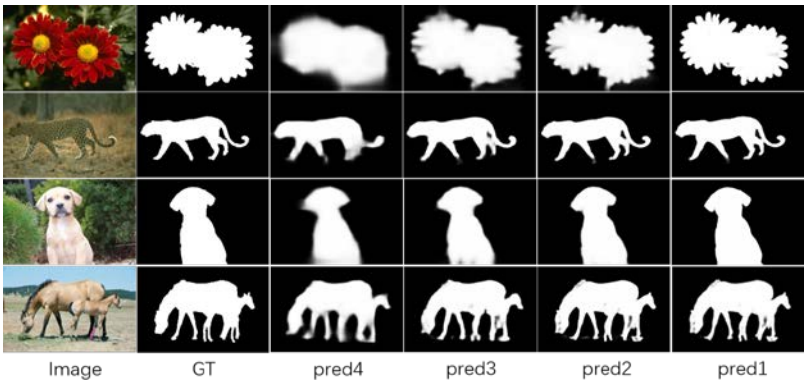


Fig. 6. Visualization of output layers of different decoders

3.5 Loss Function

In the above sections, the PFEM is to obtain the feature-enhanced multi-scale information from the output of each layer of the backbone network, and salient prediction maps are obtained through the AAM and the AAM_D. In order to optimize the features of each layer, the network adopts the way of multi-level joint supervision, which outputs all the AAM_D of the four layers to predict the saliency maps and calculates the loss with the ground truth after up-sampling. The predicted maps were recorded as {pred4, pred3, pred2, pred1} respectively, and pred1 was taken as the final salient result. At present, binary cross-entropy loss (BCE) loss is the target function of many saliency detection methods during training, which is defined as:

$$L_{bce} = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} -[G_{ij} \log(S_{ij}) + (1 - G_{ij}) \log(1 - S_{ij})] \tag{3}$$

where H×W represent the scale of the input image, G_{ij} is the ground truth label of the pixel (i, j), and S_{ij} represents the corresponding saliency value in position (i, j). Because the multi-level feature joint supervision is adopted in this paper, according to the description of the characteristics of saliency maps output by different decoding layers in section 3.4, we set the loss weights of different levels as 1, 0.5, 0.25, and 0.125 in turn. So, the final loss function is:

$$\begin{aligned} L_{total} = {} & L_{bce}(pred1, g) + 0.5 L_{bce}(pred2, g) + 0.25 L_{bce}(pred3, g) + \\ & 0.125 L_{bce}(pred4, g) \end{aligned} \tag{4}$$

where pred1, pred2, pred3, and pred4 are the prediction map of each layer, and g is the corresponding ground truth.

4. Experiment results and analysis

4.1 Datasets

In this paper, experiments are evaluated on six public salient object detection datasets and some introduction is under the following. ECSSD [35] contains of 1000 complex images. PASCAL-S [36] has 850 images containing multiple objects and cluttered backgrounds. DUTS [37] is the largest SOD dataset currently, which contains 10553 training images and 5019 test images. DUT-OMRON [38] is a very challenging dataset for salient target detection at present, including 5168 complex images with cluttered backgrounds and one or more salient objects. SOD dataset constructed by V. Movahedi et al. [39] contains 300 images, many of

which reflect common challenges in real-world scenarios, such as low contrast and salient targets to the side. The HKU-IS [40] includes 4447 images with low contrast or multiple salient objects.

4.2 Implementation Details

ResNet-50, which is pre-trained on ImageNet, is adopted as the backbone network of this paper. In the training stage, the size of the picture after a random horizontal flip is adjusted to 320*320 and randomly crop to 288×288 patches for training. To optimize the whole network, we adopt mini-batch Stochastic gradient descent. The batch and momentum are set 8 and 0.9 respectively. And the weight attenuation and epoch are set to 5e-4 and 30. Warm-up strategy and linear decline strategy are adopted to adjust the learning rate. The model is implemented on Pytorch1.1. We use the training set part of DUTS as our training data set. Because the framework is an end-to-end network, in the test stage, the images are adjusted to 320×320 and sent directly to the network to predict the maps.

4.3 Evaluation Metrics

Three metrics are used to evaluate and compare the performance of our proposed approach with some previous representative saliency detection algorithms, including F-measure score, precision-recall (PR) curves and mean absolute error (MAE) score. With different thresholds, by comparing the binarized image with the ground truth (GT), the precision and recall pairs can be calculated. The F measure score is a weighted harmonic average of precision and recall, which can be calculated as:

$$F_\beta = \frac{(1+\beta^2)\times Precision \times Recall}{\beta^2 \times Precision + Recall} \tag{5}$$

To put more emphasis on precision, the value is set to 0.3. The MAE score is the average pixel-wise absolute difference between the prediction map and GT, and the formula is as follows:

$$MAE = \frac{1}{H \times W}\sum_{y=1}^{H}\sum_{x=1}^{W}|\, S(x,y) - G(x,y)\,| \tag{6}$$

where $S(x,y)$ and $G(x,y)$ represent the pixels of the prediction maps and GT, respectively. The smaller MAE indicates better performance.

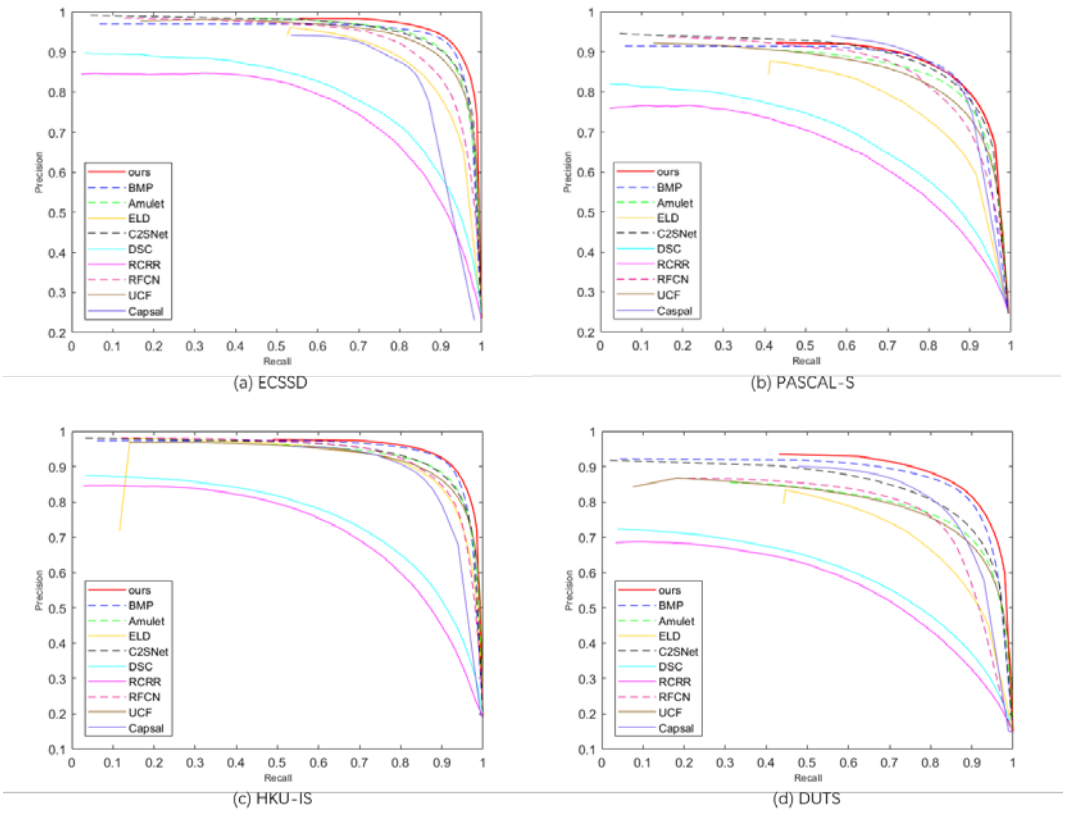
4.4 Performance Comparison

The algorithm proposed in this paper was compared with 10 classical salient target detection models, including RCRR [24], DSC [26], BMP [13], LFR [30], Amulet [28], ELD [27], C2Snet [18], UCF [12], Capsal [19] and RFCN [10]. Core ideas and types about these algorithms are listed in **Table 1**. We run the code provided by the author to get the saliency map or directly use the saliency map provided by it, and then calculate the evaluation results on different test sets according to the saliency maps.

4.4.1 Quantitative Comparison

Table 5. Compares with different saliency detect methods on 6 datasets. The best two results are highlighted in bold.

Method	ECSSD		PASCAL-S		HKU-IS		DUTS-TE		DUT-OMRON		SOD	
	F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE
	conventional methods											
RCRR	0.6956	0.1837	0.6633	0.2267	0.6638	0.1711	0.5170	0.1901	0.5260	0.1823	0.5749	0.2585
DSC	0.7185	0.1907	0.7185	0.2394	0.6796	0.1895	0.5247	0.2182	0.5355	0.2150	0.5916	0.2630
	deep SOD methods											
BMP	0.8666	0.0545	0.7674	**0.0736**	**0.8707**	**0.0387**	0.7451	**0.0490**	**0.6917**	**0.0635**	0.7637	**0.1079**
LFR	**0.8794**	**0.0525**	0.7641	0.1066	0.8706	0.0396	0.7211	0.0834	0.6776	0.1030	**0.7892**	0.1233
Amulet	0.8684	0.0589	0.7632	0.0977	0.8428	0.0521	0.6775	0.0846	0.6472	0.0976	0.7445	0.1443
ELD	0.8169	0.0783	0.7181	0.1216	0.7826	0.0719	0.6509	0.0924	0.6139	0.0909	0.7124	0.1552
C2Snet	0.8653	0.0593	0.7672	0.0802	0.8395	0.0516	0.710	0.066	0.6647	0.0734	0.7638	0.1239
UCF	0.8439	0.0691	0.7675	0.1155	0.8235	0.0612	0.6351	0.1119	0.6206	0.1203	0.7388	0.1476
Capsal	0.8205	0.0728	**0.8196**	**0.0729**	0.8407	0.0613	**0.7550**	0.0692	0.5474	0.0873	0.6684	0.1307
RFCN	0.8334	0.1070	0.7468	0.1316	0.8349	0.0889	0.7109	0.0900	0.6265	0.1105	0.7430	0.1697
Ours	**0.9060**	**0.0411**	**0.8017**	0.0774	**0.8915**	**0.0351**	**0.7957**	**0.0482**	**0.7374**	**0.0687**	**0.8086**	**0.1020**



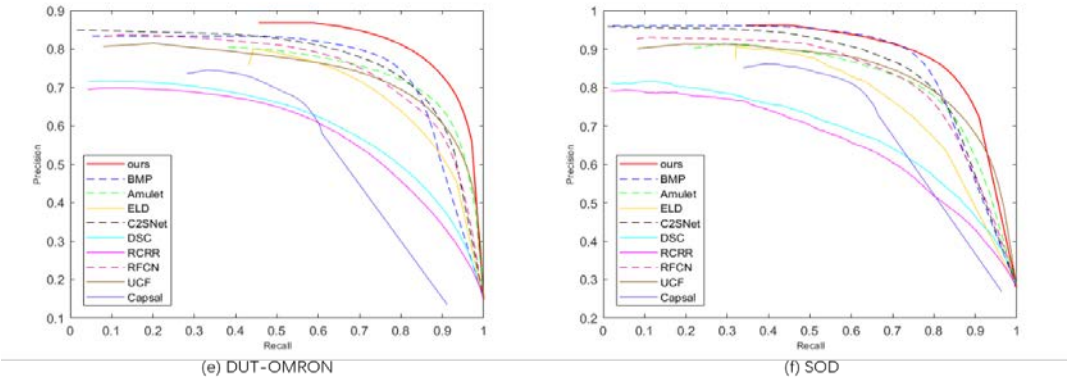


Fig. 7. The comparison of PR curves with previous methods on six datasets

Table 5 shows the results of quantitative comparison of the algorithms in terms of F_β and MAE scores on six datasets. It is obvious that our algorithm achieves the best results on almost all six datasets. Compared with these representative algorithms, our method is closest to the ideal value of 1 in F-measure score. For example, on the DUT-TE and DUT-OMRON datasets, our algorithm improved by about 0.05 over the best value in the comparison algorithm, which is a significant improvement, and the other datasets also improved by at least 0.02. In regard to MAE scores, our method reduces 0.01 on ECSSD dataset and 0.006 on DUT- OMRON dataset, and the MAE scores of other data sets also decrease. The above phenomenon shows that the number of prediction errors of our method is far less than that of other methods. Besides, it can be observed that our method performs better on some more complex and difficult datasets, such as HKU-IS, DUT-TE, DUT-OMRON, and SOD, in which many images have multiple salient objects and complex backgrounds. This indicates that, compared with the comparison methods, our algorithm is a more competitive algorithm, which shows the effectiveness of the model.

In addition, **Fig. 7** is the PR curve of our algorithm and other typical salient detection algorithms. Obviously, our algorithm has higher precision and recall. As can be seen from the figure, compared with other methods at different thresholds, the PR curve (red curve) obtained by our method is prominent in most cases, which is consistent with the measurements reported in **Table 5**. As you can see, our method is relatively weak in MAE score on PASCAL-S datasets, but we have better measurements on F-measure. Moreover, our model has much higher precision especially when the recall score is closer to 1, which indicates that the false positive rate of our method is lower than other methods. This proves the robustness of our algorithm on challenging datasets. In addition, our proposed method can detect salient areas of the image very well without any post-processing.

4.4.2 Qualitative Evaluation

Fig. 8 shows some visualization examples of different methods. The detection results shown in **Fig. 8** by different methods are given in different challenging scenarios, such as small targets, multi-targets, irregular shapes, low contrast, foreground interference, and so on. Specifically, for those objects with small salient targets and low contrast, the comparison methods are difficult to detect complete salient objects. On the contrary, our method can detect the complete salient object with consistent salient values. As shown in the fourth row, other methods such as Amulet, UCF, RCRR, and other algorithms cannot accurately detect the target

at the center point, but our method can accurately grasp the target, which indicates that our PFEM can well extract effective information and extract the salient object. It is worth noting that our method has stronger robustness to background/foreground interference (like second/third row), and can capture the relationship between different objects (like fifth/ninth row). It shows that our proposed PFEM can fully extract the semantic information of the image, and the AAM can fully eliminate some ambiguous information. For example, in the third row, our method can accurately detect 'ducks' in the water, while other methods can not eliminate its reflection. It is worth mentioning that our method has better ability to detect edge details in the prominent central area. Because we also add spatial attention mechanism in the AAM, such as the images in the eighth and ninth rows, which do not consider the spatial position of the pictures equally but pay more attention to the foreground area, we can detect 'the people on the sofa' in the eighth row of pictures, and our method can also get the results closer to the ground truth for the details in the images. For example, our method can clearly detect 'the deer's horns' and 'limbs' in the ninth row of **Fig. 8**.

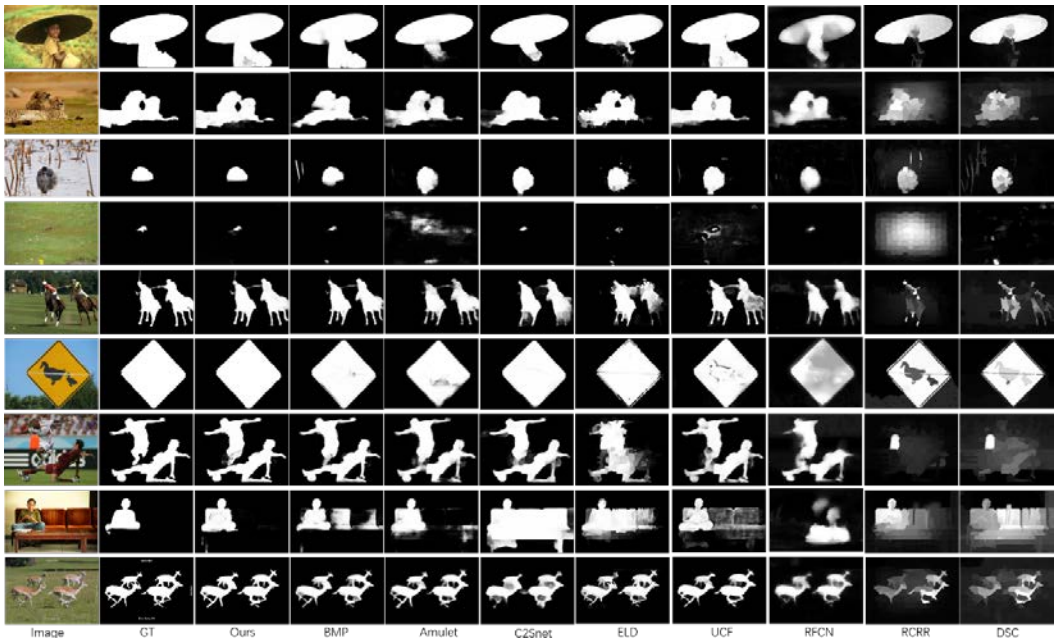


Fig. 8. Visualization comparison of the proposed model with other methods.

4.4.3 Speed Performance

We compare the running times of our method with other SOD methods. The evaluation is conducted with an NVIDIA GTX 1080Ti GPU. The results are shown in **Table 6**. As it can be seen, our method is much faster than most of the compared methods. The testing process only costs 0.04s for each image.

Table 6. Comparison of running times

Method	RCRR	DSC	BMP	LFR	Amulet	ELD	C2Snet	UCF	RFCN	**Ours**
Times(s)	1.21	1.89	0.06	0.08	0.06	0.59	**0.03**	0.11	0.65	**0.04**

4.5 Ablation Analysis

We carry out ablation experiments to investigate the effectiveness of the modules designed in the proposed network. Our comparative experiments were conducted on the test set of DUTS dataset and the ECSSD dataset. We can see from **Table 7** that the best performance can be achieved when the model contains all the components (PFEM, AAM, and AAM_D).

The experimental settings are that the coding layer uses the most primitive backbone network. The decoding layers are the simplest convolutional filters and the results of each layer filter are output for training and testing. The measured results are shown in **Table 7**. Then we replace the convolutional filters with the adjacency decoders we designed, and we can see that there is a great enhancement in both the F_β score and the MAE score. The F_β of the DUTS-TE dataset has increased from 0.6819 to 0.7813(about improved by 15%), indicating that our adjacency decoder has achieved better results. After adding the adjacent auxiliary module, the MAE value of the DUT-TE dataset is reduced by 0.08 points, which indicates that this module plays an effective role in removing the interference part in features and reducing the false positive. When we add the PFEM to the framework, the F_β and MEA scores of DUT-TE and ECSSD datasets are closer to the ideal value to some extent, which shows that our PFEM is very important in the whole network.

Table 7. Ablation study for different components of the proposed AANet (B is the abbreviation of Baseline)

Method	DUTS-TE		ECSSD	
	F_β	MAE	F_β	MAE
Baseline	0.6819	0.0694	0.8178	0.0629
B+AAM_D	0.7813	0.0529	0.9030	0.0439
B+AAM+AAM_D	0.7855	0.0487	0.9048	0.0451
B+PFEM+AAM+AAM_D	**0.7957**	**0.0482**	**0.9060**	**0.0410**

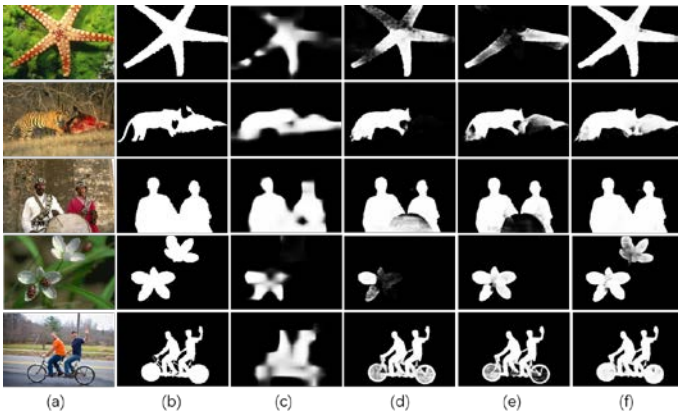


Fig. 9. Visual examples of the ablation experiment. (a) Image. (b) GT. (c) Baseline. (d) B+AAM_D. (e) B+AAM+AAM_D. (f) B+PFEM+AAM+AAM_D

In the visualization example in **Fig. 9**, we can also see that when we gradually add AAM and PFEM to the network, the performance is better and the best result is obtained. In **Fig. 9**, images in **Fig. 9**(c) are saliency maps from the simplest network, which is obvious that the saliency target is very blurred and the boundary is not clear. When AAM_D and AAM are added, it is obvious that the boundary of saliency target becomes sharper (as shown in **Fig. 9**(d) and **Fig. 9**(e)). In the pictures of the second row and fourth lines, **Fig. 9**(e) have added

more target regions with respect to the maps of **Fig. 9**(d), which further shows that the fusion of the features of two adjacent layers through the AAM can really play a supplementary auxiliary role to the information. Although sometimes the discrimination ability of some regions is weakened because of the structural similarity between AAM and AAM_D, our PFEM can make up for this deficiency well and extract the strong semantic information of feature map to help determine the salient regions. For example, in **Fig. 9**(f), the "pentagons of starfish" in the first row, "animal's carcasses" in the second row, and "flowers on the right side" in the fourth row can be completely detected.

4.6 Failure Cases

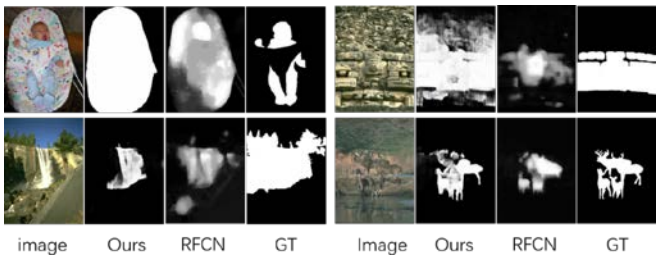


Fig. 10. Some failure cases

Fig. 10 shows some cases of failure. In these scenarios, the foreground and background have very low contrast and the background texture is similar (as shown in the second column of **Fig. 10**), which makes it difficult for our method to identify salient objects and details. In addition, due to the strong "salient-like" objects around the salient objects, our method has errors in the recognition of some scenes (as shown in the first column of **Fig. 10**). For these cases, we may want to use the way of scene understanding to help determine the salient target detection in such complex scenes.

5. Conclusion

In the paper, we propose an adjacency auxiliary network based on multi-scale feature fusion for feature extraction in salient target detection. Firstly, we design the parallel connection feature enhancement module, so that the feature mapping can get complete multi-scale context information, and then we propose the adjacency auxiliary module to fuse the high-and low-level information to remove the possible ambiguity and background noise. Finally, we use the adjacency decoding module to further refine and fuse the features and fully integrate different levels of features. The experimental results on six datasets show that the proposed method is superior to the 10 previous algorithms proposed above in both quantitative and qualitative aspects. In the future work, we will continue to study the new network framework like two-stream network and study how to make more use of contour information to improve the performance of saliency detection.

References

[1] R. Fergus, P. Perona, and A. Zisserman, “Object class recognition by unsupervised scale-invariant learning,” in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. II-264–II-271, Jun.18-20, 2003. Article (CrossRef Link).

[2] Y. Gao, M. Wang, Z. Zha, J. Shen, X. Li and X. Wu, “Visual-textual joint relevance learning for tag-based social image search,” *IEEE Transactions on Image Processing*, vol. 22, no. 1, pp. 363-376, Jan., 2013. Article (CrossRef Link).

[3] C. Siagian and L. Itti, “Rapid biologically- Inspired scene classification using features shared with visual attention,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 2, pp. 300-312, Feb., 2007. Article (CrossRef Link).

[4] S. Hong, T. You, S. Kwak, B. Han, “Online tracking by learning discriminative saliency map with convolutional neural network,” in *Proc. of the 32nd International Conference on Machine Learning*, pp. 597–606, Jul., 2015. Article (Web Link).

[5] R. Zhao, W. Ouyang, X. Wang, “Unsupervised salience learning for person re-identification,” in *Proc. of 2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3586–3593, Jun.23-28, 2013. Article (CrossRef Link).

[6] J. Qiu, Q. Wu, G. Ding, Y. Xu, and S. Feng, “A survey of machine learning for big data processing," *EURASIP Journal on Advances in Signal Processing*, May.28, 2016. Article (CrossRef Link)

[7] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, P.H. Torr, “Deeply supervised salient object detection with short connections,” in *Proc. of 2017 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5300-5309, Jul.21-26, 2017. Article (CrossRef Link).

[8] M. Feng, H. Lu, and E. Ding, “Attentive feedback network for boundary-aware salient object detection,” in *Proc. of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1623-1632, Jun.15-20, 2019. Article (CrossRef Link).

[9] W. Wang, S. Zhao, J. Shen, S.C. Hoi, A. Borji, “Salient object detection with pyramid attention and salient edges,” in *Proc. of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1448–1457, Jun.15-20, 2019. Article (CrossRef Link).

[10] L. Wang, L. Wang, H. Lu, P. Zhang, and X. Ruan, “Salient object detection with recurrent fully convolutional networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 7, pp. 1734-1746, Jul., 2019. Article (CrossRef Link).

[11] L. Wang, H. Lu, X. Ruan, M.-H. Yang, “Deep networks for saliency detection via local estimation and global search,” in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3183–3192, Jun., 2015. Article (CrossRef Link).

[12] P. Zhang, D. Wang, H. Lu, H. Wang, B. Yin, “Learning uncertain convolutional features for accurate saliency detection,” in *Proc. of 2017 IEEE International Conference on Computer Vision*, pp. 212–221, Oct.22-29, 2017. Article (CrossRef Link).

[13] L. Zhang, J. Dai, H. Lu, Y. He, G. Wang, “A bi-directional message passing model for salient object detection,” in *Proc. of 2018 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1741–1750, Jun.18-23, 2018. Article (CrossRef Link).

[14] M. Kampffmeyer, N. Dong, X. Liang, Y. Zhang, and E. P. Xing, “ConnNet: A long-range relation-aware pixel-connectivity network for salient segmentation,” *IEEE Transactions on Image Processing*, vol. 28, no. 5, pp. 2518-2529, May 2019. Article (CrossRef Link).

[15] T. Zhao and X. Wu, “Pyramid feature attention network for saliency detection,” in *Proc. of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3080-3089, Jun.15-20, 2019. Article (CrossRef Link).

[16] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “DeepLab: Semantic image segmentation with deep convolutional nets, Atrous Convolution, and Fully Connected CRFs,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834-848, Apr., 2018. Article (CrossRef Link).

[17] R. Zhang, W. Ouyang, H. Li, and X. Wang, “Saliency detection by multi-context deep learning,” in *Proc. of 2015 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1265– 1274, Jun.7-12, 2015. Article (CrossRef Link).

[18] X. Li, F. Yang, H. Cheng, W. Liu, and D. Shen, "Contour knowledge transfer for salient object detection," in *Proc. of European Conference on Computer Vision*, pp. 355–370, Sept., 2018. Article (CrossRef Link).
[19] L. Zhang, J. Zhang, Z. Lin, H. Lu, and Y. He, "Capsal: Leveraging captioning to boost semantics for salient object detection," in *Proc. of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6024–6033, Jun.15-20, 2019. Article (CrossRef Link).
[20] N. Ayoub, Z. Gao, D. Chen, R. Tobji, and N. Yao, "Visual saliency detection based on color frequency features under bayesian framework," *KSII Transactions on Internet and Information Systems*, vol. 12, no. 2, pp. 676-692, Feb., 2018. Article (CrossRef Link).
[21] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *Proc. of 2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3166–3173, Jun.23-28, 2013. Article (CrossRef Link).
[22] A. Aksac, T. Ozyer, R. Alhajj, "Complex networks driven salient region detection based on superpixel segmentation," *Pattern Recognit*, Vol. 66, pp.268–279, Jun., 2017. Article (CrossRef Link).
[23] Wei Y, Wen F, Zhu W, and Jian S, "Geodesic saliency using background priors," in *Proc. of European Conference on Computer Vision*, pp. 29-42, Oct.7, 2012. Article (CrossRef Link).
[24] Y. Yuan, C. Li, J. Kim, W. Cai and D. D. Feng, "Reversion correction and regularized random walk ranking for saliency detection," *IEEE Transactions on Image Processing*, vol. 27, no. 3, pp. 1311-1322, Mar., 2018. Article (CrossRef Link).
[25] D. Shan, X. Zhang, C. Zhang, "Visual saliency based on extended manifold ranking and third-order optimization refinement," *Pattern. Recognition. Letters*, Vol. 116, pp. 1–7, Dec., 2018. Article (CrossRef Link).
[26] X. Wu, X. Ma, J. Zhang, A. Wang, and Z. Jin, "Salient object detection via deformed smoothness constraint," in *Proc. of the 25th IEEE International Conference on Image Processing (ICIP)*, pp. 2815-2819, Oct.7-10, 2018. Article (CrossRef Link).
[27] G. Lee, Y. Tai and J. Kim, "ELD-Net: An efficient deep learning architecture for accurate saliency detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 40, no. 7, pp. 1599-1610, Jul., 2018. Article (CrossRef Link).
[28] P. Zhang, D. Wang, H. Lu, H. Wang, and X. Ruan, "Amulet: Aggregating multi-level convolutional features for salient object detection," in *Proc. of 2017 IEEE International Conference on Computer Vision*, pp. 202–211, Oct.22-29, 2017. Article (CrossRef Link).
[29] T. Wang, Zhang L, Wang S, "Detect globally, refine locally: A novel approach to saliency detection," in *Proc. of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3127-3135, Jun.18-23, 2018. Article (CrossRef Link).
[30] P. Zhang, W. Liu, H. Lu, C. Shen, "Salient object detection with lossless feature reflection and weighted structural loss," *IEEE Transactions on Image Processing*, Vol. 28, no. 6, pp. 3048–3060, Jan., 2019. Article (CrossRef Link).
[31] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. of International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241, 2015. Article (CrossRef Link).
[32] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, "Pyramid scene parsing network," in *Proc. of 2017 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6230-6239, Jul.21-26, 2017. Article (CrossRef Link).
[33] J. Liu, Q. Hou, M. Cheng, J. Feng, and J. Jiang, "A simple pooling-based design for real-time salient object detection," in *Proc. of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3912-3921, Jun.15-20, 2019. Article (CrossRef Link).
[34] N. Liu, J. Han, and M. Yang, "PiCANet: Learning pixel-wise contextual attention for saliency detection," in *Proc. of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3089-3098, Jun.18-23, 2018. Article (CrossRef Link).
[35] J. Shi, Q. Yan, L. Xu, and J. Jia, "Hierarchical image saliency detection on extended CSSD," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 38, no. 4, pp.717–729, Apr., 2016. Article (CrossRef Link).

[36] Y. Li, X. Hou, C. Koch, J.M. Rehg, and A.L. Yuille, "The secrets of salient object segmentation," in *Proc. of 2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 280–287, Jun.23-28, 2014. Article (CrossRef Link).

[37] L. Wang, H. Lu, Y. Wang, M. Feng, D. Wang, B. Yin, and X. Ruan, "Learning to detect salient objects with image-level supervision," in *Proc. of 2017 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 136–145, Jul., 2017. Article (CrossRef Link).

[38] C. Yang, L. Zhang, H. Lu, X. Ruan, M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *Proc. of 2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3166–3173, Jun.23-28, 2013. Article (CrossRef Link).

[39] V. Movahedi, J. H. Elder, "Design and perceptual validation of performance measures for salient object segmentation," in *Proc. of 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 49-56, Jun.13-18, 2010. Article (CrossRef Link).

[40] G. Li, Y. Yu. "Visual saliency based on multiscale deep features," in *Proc. of 2015 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5455–5463, Jun.7-12, 2015. Article (CrossRef Link).

Xialu Li received the B.S. degree in School of Information science and Engineering of Wuhan University of Science and Technology, China, in 2019. She is currently pursuing the M.S. degree in Electronic and Communication Engineering with Nanjing University of Posts and Telecommunications, China. Her research interests include computer vision, image processing.

Ziguan Cui received the B.S. degree from Shandong University in 2004, the M.S. degree in computer science from Nanjing University of Aeronautics and Astronautics in 2008 and Ph.D. degree in signal and information processing from Nanjing University of Posts and Telecommunications in 2012. He is currently an associate professor at Nanjing University of Posts and Telecommunications. His current research interests include hyperspectral image processing, perceptual image and video quality assessment, machine learning and deep learning.

Zongliang Gan received the B.S., M.S., and Ph.D. degrees from Nanjing University of Posts and Telecommunications in 2001, 2004, and 2007, China. He is currently an associate professor in the College of Communication and Information Engineering at Nanjing University of Posts and Telecommunications. His current research interests include image processing and deep learning.

Guijin Tang received the Ph.D. degree in signal and information processing from Nanjing University of Posts and Telecommunications. He is now an associate professor of Nanjing University of Posts and Telecommunications. His research interests include error concealment and video analysis.

Feng Liu received the M.S., and Ph.D. degrees in optical engineering from Nanjing University of Posts and Telecommunications in 1994, and 1997. He is currently a full professor of Nanjing University of Posts and Telecommunications. His current research interests include image processing, intelligent video analysis and deep learning.