Pyramidal Feature Shrinking for Salient Object Detection

Mingcan Ma[1 2], **Changqun Xia**[2*], **Jia Li**[1 2*]
[1]State Key Laboratory of Virtual Reality Technology and Systems, SCSE, Beihang University, Beijing, China
[2]Pengcheng Laboratory, Shenzhen, China
{mingcanma, jiali}@buaa.edu.cn, xiachq@pcl.ac.cn

Abstract

Recently, we have witnessed the great progress of salient object detection (SOD), which benefits from the effectiveness of various feature aggregation strategies. However, existing methods usually aggregate the low-level features containing details and the high-level features containing semantics over a large span, which introduces noise into the aggregated features and generates inaccurate saliency maps. In this paper, we propose a pyramidal feature shrinking network (PFSNet), which aims to aggregate adjacent feature nodes in pairs with layer-by-layer shrinkage, so that the aggregated features fuse effective details and semantics and discard interference information. Specifically, a pyramidal shrinking decoder (PSD) is proposed to aggregate adjacent features hierarchically in an asymptotic manner. Unlike other methods that aggregate features with significantly different information, this method only focuses on adjacent feature nodes in each layer and shrinks them to a final unique feature node. Besides, we propose an adjacent fusion module (AFM) to perform mutual spatial enhancement between the adjacent features to dynamically weight the features and adaptively fuse the appropriate information. Besides, a scale-aware enrichment module (SEM) based on the features extracted from the backbone is utilized to obtain rich scale information and generate diverse initial features with dilated convolutions. Extensive quantitative and qualitative experiments demonstrate that the proposed intuitive framework outperforms 14 state-of-the-art approaches on 5 public datasets.

Introduction

The purpose of salient object detection (SOD) is to estimate visually important objects and regions in an image. This is the basic work of many visual tasks such as object tracking (Liang et al. 2016), object recognition (Rutishauser et al. 2004), semantic segmentation (Yao and Gong 2019) and so on. Similar to many other computer vision tasks, SOD is also dominated by the CNN methods, although the results are excellent but not perfect.

SOD algorithms based on deep-learning usually depend on various strategies to fuse features extracted from the backbone. And there is a big gap between high-level features and low-level features. Therefore, the key to the SOD algorithm lies in how to make full use of semantic and detailed information. FPN (Lin et al. 2017) considers this problem, and subtly combines semantic information and detailed information by gradually merging features of different levels from bottom to top. Besides, many works such as (Zhang et al. 2018a; Zhao et al. 2020) based on the FPN have achieved milestone results. However, there are still problems in feature fusion in FPN based networks. Considering the merging operation of the last two features, the low-level features with rich details and noise will be combined with the high-level features after the previous fusion processing. As shown in Fig. 1, We visualize these two features of MINet (Pang et al. 2020) based on FPN and find that they are completely different, and the result shows the direct combination of features with large differences will produce noise and even cause performance degradation. We define the above-mentioned feature fusion operations as leaping feature fu-

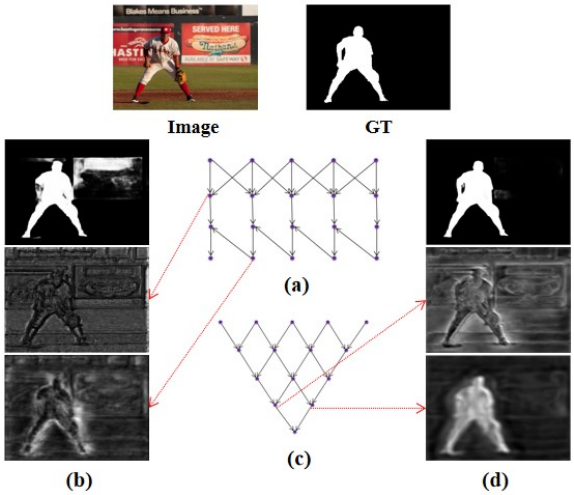

Figure 1: Visual comparison of features before the last feature fusion. (a) MINet. (b) the result of MINet. From top to bottom, (b) shows the predicted map, low-level features and high-level features. (c) PFSNet. (d) denotes our results, and the specific meaning is consistent with (b).

*Correspondence should be addressed to Changqun Xia (E-mail: xiachq@pcl.ac.cn) and Jia Li (E-mail: jiali@buaa.edu.cn). URL: http://cvteam.net.

sion. Of course, not only FPN-based networks suffer from this problem, but many methods that need to integrate high-level and low-level features fall into this dilemma.

We are inspired by the evolution of biological species and propose a method to avoid leaping feature fusion operations. Under the constraints of natural selection, organisms evolve in a direction suitable for the current environment. In this process, there are two characteristics: 1) Only creatures with similar characteristics can produce offspring; 2) Natural selection will enhance genes suitable for the environment while inhibiting genes not suitable for the current environment. Considering the similarity between feature fusion and the biological evolution process, we propose a new SOD network based on the above natural phenomena.

First of all, we propose a pyramid shrinking decoder (PSD) as shown in Fig. 1 (c). We define five features extracted from the backbone as five feature types, adjacent features as similar features, and non-adjacent features as isolated features. PSD only shrinks similar features in each layer. After several layers of shrinking, the features most suitable for the current input are retained. To make the fusion process enhance the features suitable for the current sample and suppress the features not suitable, we design an adjacent fusion module (AFM). It first allows adjacent features to complement each other spatially, and then assigns different weights to different features and compresses the features through convolution. Besides, to make the initial features more diverse, we design a scale-aware enrichment module SEM. It can make a full use of the size information of the initial features and supplement rich multi-scale information. Our main contributions can be summarized as follows:

- We propose a pyramid shrinking decoder PSD, which shrinks adjacent features layer by layer and rejects any isolation feature fusion operation to avoid the problem of leaping feature fusion.
- We introduce the adjacent fusion module AFM to fuse adjacent features in pairs, which can enhance the features suitable for the current input sample and weaken the inappropriate features.
- We designed a simple but effective initial scale-aware enrichment module SEM to supplement the features extracted from the backbone with rich multi-scale information.
- A comprehensive comparison with 14 latest methods on five datasets demonstrates the superiority of our proposed framework.

Related Work

In recent years, more and more salient object detection networks based on deep-learning such as (Xia et al. 2017) have been proposed. Compared with many traditional algorithms that rely on low-level features, deep-learning-based methods can use detailed information and semantic information more effectively. Especially after the emergence of a Fully Convolutional Neural Network (FCN), salient object detection based on deep-learning can better reflect its advantages.

The key of the SOD algorithm based on deep-learning is to obtain powerful feature expression. To achieve this goal,

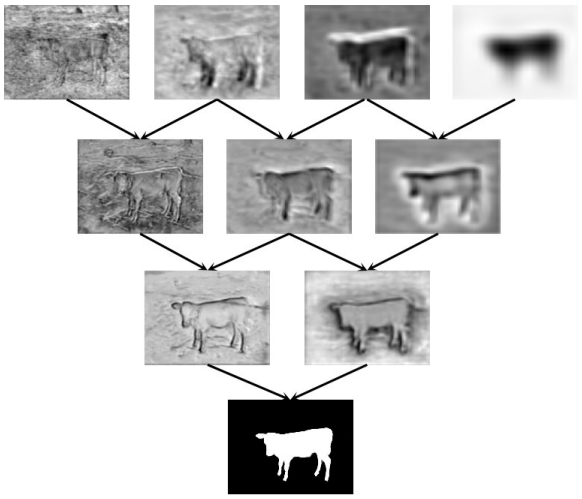

Figure 2: Visual display results of detailed intermediate features of PSD. The feature maps correspond to the output features of each node in Fig. 1 (c). The whole process shows our idea of shrinking features.

U-net (Ronneberger, Fischer, and Brox 2015) is proposed to gradually supplement the feature information in the encoder to the decoder. Zhao et al. (2019) sets spatial and channel attention to help the network pay more attention to features suitable for the current sample. Zhang et al. (2017) achieved better performance by improving the simple fusion module in U-Net. Luo et al. (2017) Predict salient regions through local and global features. Qin et al. (2019) proposed a more comprehensive loss function to optimize the results of saliency detection. Zhang et al. (2018a) proposed a Bi-Directional Message Passing Model to make better use of multi-layer features. Liu et al. (2019) capture more useful features by combining simple pooling operations. Pang et al. (2020) considered extracting scale information from adjacent features and designed an AIM module to extract more information between adjacent features. They also designed SIM to obtain more feature information from the features themselves. Wei et al. (2019) consider the differences between the features and design an FCM module, which uses the feature multiplication method to avoid introducing noise as much as possible. Zhou et al. (2020) propose Interactive Two-Stream Decoder (ITSD) to make full use of the relationship between the salient object boundary map and the salient object map.

With the development of SOD, the accuracy is closer to the upper limit. Therefore, more details need to be considered to achieve further breakthroughs. However, many previous methods only focus on combining rich feature information to obtain better feature expression but ignore the problem of leaping feature fusion. Although some methods such as MINet (Pang et al. 2020) and F3Net (Wei, Wang, and Huang 2019) have realized that leaping feature fusion may bring negative effects, they only propose related modules to reduce the impact of the problem. Fundamentally, they did not avoid leaping feature fusion operations. Differently, we

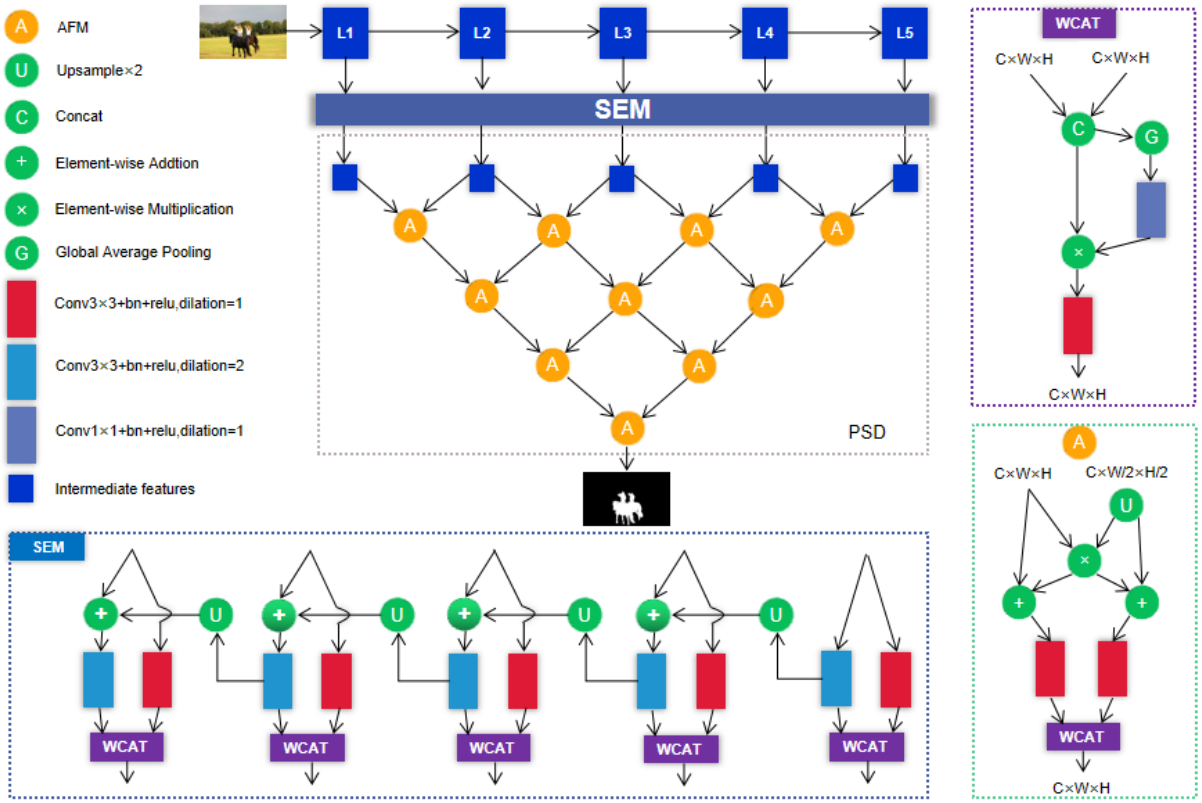

Figure 3: The overall display of PFSNet, which is based on ResNet-50. SEM is used to enhance the initial features, L1, L2,..., L5 represents five features extracted from the backbone. AFM is used to shrink and merge adjacent features. WCAT is part of SEM and AFM. The arrow indicates the direction of data flow. The saliency map indicates the location of the supervision.

propose a decoder that avoids the operation of leaping feature fusion. PFSNet shrinks adjacent features hierarchically in an asymptotic manner, and finally achieves the purpose of multi-feature fusion under the condition of avoiding leaping feature fusion operations as much as possible.

Method

As shown in Fig. 3, based on the concept of smoothly merging multiple features, we construct a pyramid shrinking decoder PSD to shrink adjacent features in pairs layer-by-layer. In the decoder, we design an adjacent fusion module AFM to retain useful information in adjacent feature nodes and reduce noise. In order to make the initial features more diverse, we propose a scale-aware enrichment module SEM to pre-process the features extracted from the backbone and get rich multi-scale features. In this session, we will introduce PSD, AFM, and SEM in turn.

Adjacent Fusion Module (AFM)

First of all, we define features to be merged by AFM as parent features, and merged features are child features. AFM has two main tasks: 1) Child features should inherit features suitable for the current input sample, and discard inappropriate futures; 2) The child features should maintain the same dimensions as those of their parents. Since all feature merging operations in PSD are performed on adjacent features, the features to be merged have great similarities. Therefore, the shared features in parents are features that require special attention from the child. Therefore, we first extract shared features from parent features through element-wise multiplication and then add shared features back to parent features through element-wise addition to enhance them. We merge the two processed features through the concatenation operation, and then let them pass through the global average pooling, 1×1 convolution, and $softmax$ activation function in turn to generate a weight vector. Finally, the weight vector and the features are correspondingly multiplied to obtain the weighted features. After feature weighting, we use 3×3 convolution to compress the channels of the child features consistent with the parent features. Since different features have different weights, after convolution calculation, elements with smaller weights are rarely inherited by child features. In this way, we achieved the goal of letting children inherit important features and discard more noise. For more details about AFM, please refer to Fig. 3. The detailed definition of AFM can be expressed as:

$$\mathbf{f}_c = (\delta(conv(gap(\mathbf{f}_a))) \bullet \mathbf{c}) \bullet \mathbf{f}_a, \quad (1)$$

$$\mathbf{f}_a = concat(\tilde{\mathbf{f}}_1, \tilde{\mathbf{f}}_2), \quad (2)$$

$$\tilde{\mathbf{f}}_1 = conv(\mathbf{f}_1 \oplus \mathbf{f}_1 \otimes up(\mathbf{f}_2)), \quad (3)$$

$$\tilde{\mathbf{f}_2} = conv(up(\mathbf{f}_2) \oplus \mathbf{f}_1 \otimes up(\mathbf{f}_2)), \quad (4)$$

where $\mathbf{f}_1$ and $\mathbf{f}_2$ denote adjacent features, $conv$ represents a convolution with a batch normalization layer and the ReLU activation function, gap denotes global average pooling, $\mathbf{c}$ denotes the number of channels of $\mathbf{f}_a$, $\oplus$ denotes element-wise addition, $\otimes$ means element-wise multiplication, up represents up-sample operation, and δ denotes softmax activation function. In short, the AFM first makes both input features pay more attention to the common elements through element-wise multiplication, then weights all features through the global average pooling operation, and finally adjusts the number of channels through the convolution operation and obtains the final result.

Pyramidal Shrinking Decoder (PSD)

The design focus of many previous methods is to aggregate detailed features and semantic features with different fusion strategies. But most of them directly fuse features over a large span, such as PFANet (Zhao and Wu 2019). Although this operation can supplement rich features to the network. It will also bring great negative effects. Many methods try to challenge this problem, such as MINet (Pang et al. 2020) and F3Net (Wei, Wang, and Huang 2019). But they only use adjacent features to extract more scale information or propose related modules to reduce the impact. There is still a leap of fusion of low-level features and high-level features in these two methods. This paper proposes for the first time to extend adjacent features into hierarchical fusion. In this way, we can use the advantages of adjacent feature fusion to achieve multi-layer feature fusion and avid leaping fusion operations. Besides, from the location of the last feature fusion, it can be seen that the framework based on FPN directly integrates low-level features containing noise, while PFSNet has eliminated a lot of noise.

As shown in Fig. 3, the core goal of PSD is to achieve multi-feature integration while avoiding leaping feature fusion operations as much as possible. PSD is a structure composed of AFMs, and the process of merging features is carried out in adjacent node pairs. We construct PSD on the backbone of ResNet-50. Assuming that the size of the input picture is $C * W * H$, we can get five features $\{\mathbf{f}_i | i = 1, 2, ..., 5\}$ with sizes $[\frac{M}{2^i}, \frac{N}{2^i}]$ from the backbone. In the first step, we merge $\mathbf{f}_i$ and $\tilde{\mathbf{f}}_{i+1}$ through AFM to obtain $\{\mathbf{f}_i | i = 1, 2, 3, 4\}$, and then perform similar operations to obtain $\{\mathbf{f}_i | i = 1, 2, 3\}$, and continue to perform the above process twice to get the final result $\mathbf{f}_1$. During this process, feature shrinking is only performed between adjacent features. In this way, our decoder can avoid leaping fusion operations as much as possible.

Specifically, PSD contains four layers of feature shrinking processes, of which there are 10 adjacent feature fusion modules. If we consider 10 fusion results separately, we can get the results of gradual fusion of any adjacent features of the backbone. As shown in Fig. 2, node in the third row and first column can get the fusion result of the first three features of the backbone. Taking into account the weighted selection effect of AFM, PSD can selectively enlarge or reduce the influence of features extracted from the backbone, thereby obtaining a diversified feature expression.

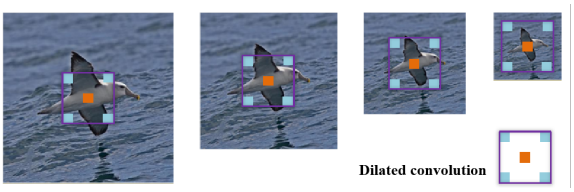

Figure 4: The effect of dilated convolution with dilated rate $D = 2$ under different size images.

Scale-aware Enrichment Module (SEM)

After obtaining the effective feature combination decoder PSD, we design SEM to enable the decoder to obtain richer initial feature information. The features extracted from the backbone have sufficient size information and the hierarchical information between details and semantics. SEM makes full use of them to provide more useful information for the initial features. As shown in Fig. 4, the same dilated convolution can get different results on images of different sizes, and the superposition of different convolution results can produce more complete feature expression. Based on this, we propose SEM in this paper. The specific structure is shown in Fig. 3. The specific definition of the model can be expressed as:

$$\mathbf{l}_i = (\delta(conv(gap(\mathbf{t}_i)) \bullet \mathbf{c}) \bullet \mathbf{t}_i, \quad (5)$$

$$\mathbf{t}_i = concat(\tilde{\mathbf{f}_i}, conv(\mathbf{f}_i))), \quad (6)$$

$$\tilde{\mathbf{f}_i} = \begin{cases} dconv(\mathbf{f}_i), & i = 5 \\ dconv(\mathbf{f}_i \oplus up(\tilde{\mathbf{f}}_{i+1})), & i = 1, ..., 4 \end{cases}, \quad (7)$$

where $\mathbf{f}_i$ means the i-th feature extracted from the backbone, $dconv$ denotes the combination of dilated convolution, batch normalization and Relu activation function, $\mathbf{c}$ means the number of channels of $\mathbf{t}_i$.

Loss Function

Unlike many other methods, our network only needs to be supervised in one location. PFSNet uses Binary Cross Entropy and Intersection Over Union as loss functions which are commonly used in the SOD field such as (Wei, Wang, and Huang 2019) to train our network. It can be defined as:

$$\mathcal{L}_{tot} = \mathcal{L}_{bce} + \mathcal{L}_{iou}, \quad (8)$$

where intersection over union loss can be expressed as:

$$\mathcal{L}_{iou} = 1 - \frac{\sum_{i=0}^{H}\sum_{j=0}^{W} mul(i,j)}{\sum_{i=0}^{H}\sum_{j=0}^{W}(sum(i,j) - mul(i,j)) + \epsilon}, \quad (9)$$

where (i, j) represents the pixel position of the image. H and W represent the height and width of the image, respectively. $sum(i, j)$ represents the sum of the predicted saliency map and the ground-truth map at (i, j) pixel, and $mul(i, j)$ represents the product of the predicted saliency map and the

ground-truth at (i, j) pixel. ϵ is set to 1e-6 to prevent division by zero. $\mathcal{L}_{bce}$ can be expressed as:

$$\begin{aligned}\mathcal{L}_{bce} = -\sum_{i=0}^{H}\sum_{j=0}^{W}(g(i,j)log(p(i,j)) \\ + (1-g(i,j))log(1-p(i,j))),\end{aligned} \tag{10}$$

where g represents ground-truth map, p represents the predicted saliency map. Our training goal is to minimize $\mathcal{L}_{tot}$.

Experiment

Datasets

In order to prove the superiority of the method proposed in this paper, we choose the five most convincing benchmark datasets in the SOD field as test sets. DUT-OMRON (Yang et al. 2013) contains 5,168 challenging images with pixel-level annotations, and ECSSD (Yan et al. 2013) contains 1,000 semantically complex images and annotation maps. HKU-IS (Li and Yu 2015) contains 4,447 images with multiple discontinuous salient objects that will intersect the image boundary. PASCAL-S (Li et al. 2014) includes 850 natural images. These images are selected from PASCAL VOC 2010 segmentation challenge and are pixel-wise annotated. DUTS-TE (Wang et al. 2017), the test set in DUTS, contains 5,019 challenging images and their annotations. Like many previous methods, we choose DUTS-TR, the training set of DUTS (Wang et al. 2017), to train our network, which contains 10,553 images and corresponding annotated maps.

Evaluation Metric

In order to conduct experimental evaluation more comprehensively, we have selected four widely used evaluation metrics to evaluate the performance of the algorithm. They are Precision-Recall, Mean Absolute Error (MAE), F-measure (F_β^*) and E-measure(E_ξ). The saliency map predicted by the network is a non-binary map, so when a different threshold is selected to binarize the prediction map, different precision and recall values can be obtained. So we use the precision-recall curve to comprehensively evaluate the prediction results. The second evaluation metric MAE is defined as the average absolute error between all elements between the predicted image and the ground-truth map.

$$MAE = \frac{1}{H \times W}\sum_{i=0}^{H}\sum_{j=0}^{W}\|p(i,j) - g(i,j)\|, \tag{11}$$

where p represents the predicted saliency map, and g represents the corresponding ground-truth map. (H, W) represents the size of the image. Another indicator F-measure score is a metric that comprehensively considers recall and precision, and is defined as follows:

$$F_\beta = \frac{(1+\beta^2)Pricision + Recall}{\beta^2 Precision + Recall}, \tag{12}$$

where β^2 is usually set to 0.3 to emphasize the proportional relationship between recall and precision. This coefficient is the recommended value by (Yang et al. 2013). A higher F-measure value indicates a more accurate prediction result. Here we choose the largest value calculated with different thresholds as the evaluation result. The E-measure proposed in (Fan et al. 2018) is also a general evaluation metric in the SOD field and the value of E-measure is directly proportional to the quality of the predicted result. This paper will also use it to evaluate experimental results.

Implementation Details

DUTS-TR is used to train PFSNet. We use PyTorch to construct our network and train it on a PC with a GTX1080TI GPU. Set batch size to 20, epochs to 50, use Stochastic Gradient Descent (SGD), set the momentum to 0.9, and weight decay to 0.0005. Horizontal flipping, random cropping, and multi-scale input images are used to pre-process the images. ResNet-50 pre-trained on ImageNet was used as the backbone. We set the maximum learning rate of the backbone to 0.005 and the other parts to 0.05. And the learning rate first increases and then decreases with the training process. The size of each image is adjusted to 352 x 352 to predict the saliency map without any post-processing.

Comparison with State-of-the-arts

In order to fully demonstrate the superiority of the proposed method, we evaluated and tested 14 recent methods with the aforementioned evaluation metrics, including C2S (Li et al. 2018), BMPM (Zhang et al. 2018a), GAPR (Zhang et al. 2018b), CiPA-R (Liu, Han, and Yang 2018), BASNet (Qin et al. 2019), CPD-R (Wu, Su, and Huang 2019), PoolNet (Liu et al. 2019), BANet (Su et al. 2019), MINet (Pang et al. 2020), U2Net (Qin et al. 2020), ITSDNet (Zhou et al. 2020), DFINet (Liu, Hou, and Cheng 2020), GateNet (Zhao et al. 2020), GCPANet (Chen et al. 2020). In order to ensure fairness, we use a unified evaluation code to evaluate the salient prediction map published by each method.

Quantitative Comparison. Tab. 1 shows the comparison results of 15 methods on the three evaluation metrics. Our method performs well on multiple datasets, especially ECSSD, HKU-IS, and DUTS-TE, which demonstrates the superiority of the proposed model. For the relatively simple dataset ECSSD, the performance improvement is valuable. The best result of the MAE under ECSSD in 2019 was 0.035, and it reached 0.033 in 2020, and we reached 0.031. For the complex dataset DUTS-TE, F-measure reached 0.880 in 2019 and 0.888 in 2020, while PFSNet reached 0.898, which has exceeded the best improvement rate of the previous year. Besides, Fig. 5 shows the precision-recall curves and F-measure curves of various methods. As can be seen, our method still performs well under multiple thresholds, and two types of curves exceed most comparison methods. In short, considering the comprehensive experimental results, our method has absolute advantages in the case of fair comparison of multiple datasets and multiple evaluation metrics.

Qualitative Comparison. In order to more intuitively illustrate the advantages of the proposed algorithm, we visualize the prediction results of 8 state-of-the-art methods in different scenarios. As shown in Fig. 6, our method can

-	ECSSD			HKU-IS			DUTS-TE			DUT-OMRON			PASCAL-S		
Method	$F_\beta^* \uparrow$	$MAE \downarrow$	$E_\xi \uparrow$	$F_\beta^* \uparrow$	$MAE \downarrow$	$E_\xi \uparrow$	$F_\beta^* \uparrow$	$MAE \downarrow$	$E_\xi \uparrow$	$F_\beta^* \uparrow$	$MAE \downarrow$	$E_\xi \uparrow$	$F_\beta^* \uparrow$	$MAE \downarrow$	$E_\xi \uparrow$
Ours	0.952	0.031	0.928	0.943	0.026	0.956	0.898	0.036	0.902	0.823	0.055	0.875	0.881	0.063	0.857
$GateNet_{20}$	0.945	0.040	0.924	0.933	0.033	0.949	0.888	0.040	0.889	0.818	0.055	0.862	0.875	0.068	0.852
$U2Net_{20}$	0.951	0.033	0.924	0.935	0.031	0.948	0.873	0.045	0.886	0.823	0.054	0.871	0.862	0.076	0.841
DFI_{20}	0.949	0.035	0.924	0.934	0.031	0.951	0.886	0.039	0.892	0.818	0.055	0.865	0.885	0.065	0.857
$MINet_{20}$	0.947	0.033	0.927	0.935	0.029	0.953	0.884	0.037	0.898	0.810	0.055	0.865	0.873	0.064	0.852
$GCPANet_{20}$	0.948	0.035	0.920	0.938	0.031	0.949	0.888	0.038	0.891	0.812	0.056	0.860	0.876	0.061	0.850
$ITSDNet_{20}$	0.947	0.034	0.927	0.934	0.031	0.952	0.883	0.041	0.895	0.821	0.061	0.863	0.876	0.064	0.853
$BANet_{19}$	0.945	0.035	0.928	0.931	0.032	0.950	0.872	0.040	0.892	0.803	0.059	0.860	0.870	0.070	0.855
$BASNet_{19}$	0.942	0.037	0.921	0.928	0.032	0.946	0.860	0.048	0.884	0.805	0.056	0.869	0.857	0.076	0.847
$PoolNet_{19}$	0.944	0.039	0.924	0.933	0.032	0.949	0.880	0.040	0.889	0.808	0.056	0.863	0.869	0.074	0.850
CPD_{19}	0.939	0.037	0.925	0.925	0.034	0.944	0.865	0.043	0.887	0.797	0.056	0.866	0.864	0.072	0.849
$C2SNet_{18}$	0.910	0.055	0.914	0.896	0.048	0.927	0.807	0.063	0.846	0.758	0.072	0.829	0.845	0.082	0.840
$BMPM_{18}$	0.928	0.044	0.914	0.920	0.039	0.937	0.852	0.049	0.859	0.774	0.063	0.839	0.857	0.073	0.838
$PAGR_{18}$	0.927	0.061	0.914	0.918	0.048	0.939	0.854	0.055	0.880	0.771	0.071	0.842	0.851	0.092	0.846
$PICA_{18}$	0.935	0.046	0.913	0.918	0.043	0.936	0.860	0.051	0.862	0.803	0.065	0.841	0.863	0.075	0.833

Table 1: The comparison of quantitative results includes the maximum F-measure (F_β^*, the larger the better), MAE (the smaller the better) and E-measure (E_ξ, the larger the better). The best and runner-up results are marked with red and blue, respectively.

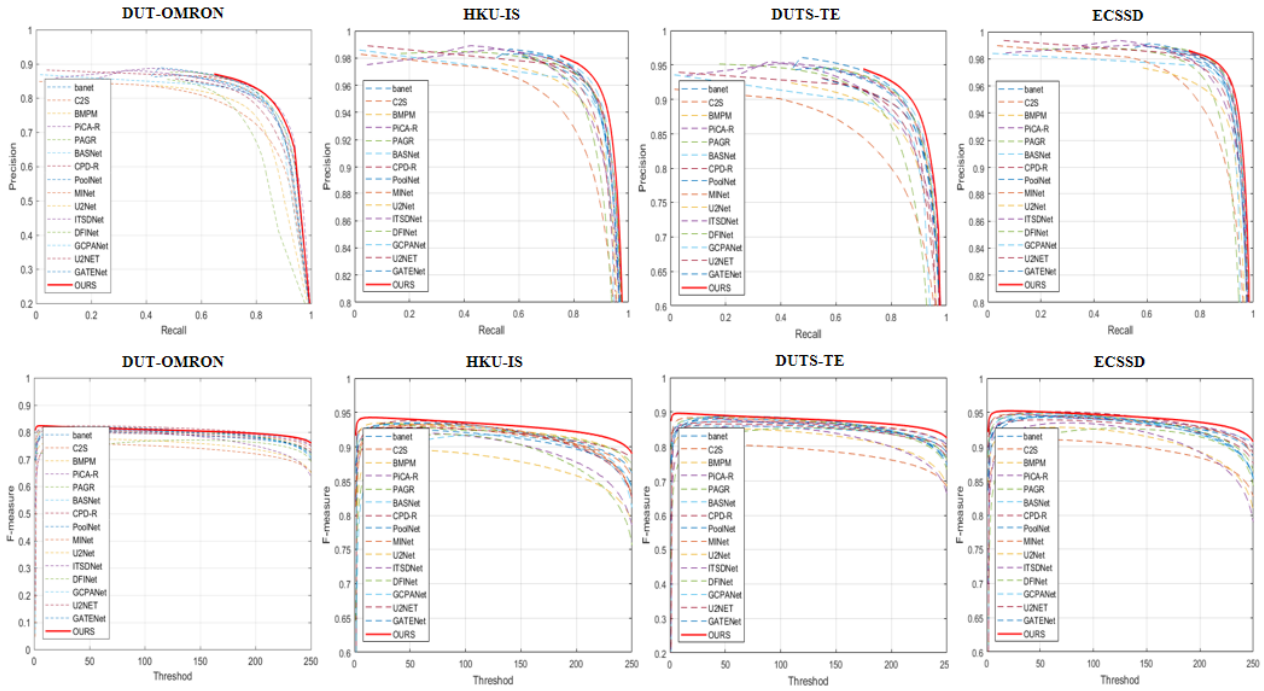

Figure 5: The precision-recall curve and F-measure curve of 15 methods under four benchmark datasets. The results show that PFSNet has surpassed the previous method with a variety of threshold options.

obtain outstanding results for pictures containing objects of different scales. In addition, our method can detect all targets more comprehensively in a scene containing multiple salient objects. From the comparison in the seventh row, we can also see that in complex scenes, our results can better shield background noise and accurately capture salient objects. From the comparison result in the sixth row, we can see that our model can accurately distinguish confusing objects. In short, our method is prominent in multi-scene images, multi-object images, complex background images, and images containing confusing objects. This can also illustrate the effectiveness of the proposed algorithm.

Ablation Study

In the SEM model, in order to obtain the multi-scale information of the image, we use the convolution with the dilated rate $D = 2$. In order to verify the influence of this hyper-parameter, we designed the following experiment. We set D to 1, 2, 3, 4 in turn, and the other structures remain unchanged. The evaluation results are shown in Tab. 2. We can see that the best results can be obtained when $D = 2$, and as D increases, the performance decreases. Therefore, $D = 2$ is used when referring to this parameter in the subsequent experiments. In addition, an ablation experiment is designed

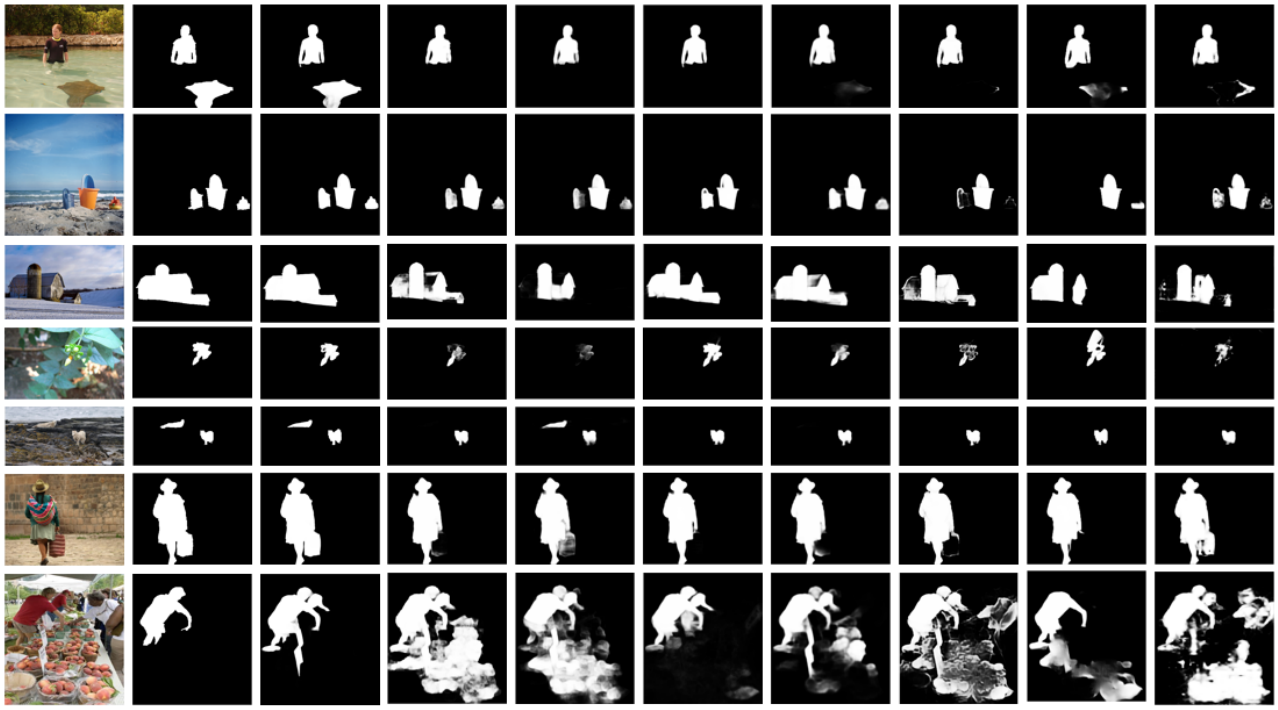

Figure 6: Visual comparison with seven state-of-the-art methods. Obviously, our method can get a clearer and more accurate result in various complex scenarios.

-	ECSSD			HKU-IS		
D	$F_\beta^* \uparrow$	$MAE \downarrow$	$E_\xi \uparrow$	$F_\beta^* \uparrow$	$MAE \downarrow$	$E_\xi \uparrow$
1	0.950	0.034	0.926	0.941	0.029	0.954
2	0.952	0.031	0.928	0.943	0.026	0.956
3	0.950	0.032	0.927	0.940	0.027	0.954
4	0.946	0.035	0.927	0.937	0.028	0.951

Table 2: The effect of dilated convolution in SEM is compared when dilated rate D takes different values. The best result is obtained when $D = 2$.

-	DUTS-TE		
method	$F_\beta^* \uparrow$	$MAE \downarrow$	$E_\xi \uparrow$
basenet	0.833	0.055	0.856
basenet+PSD	0.885	0.039	0.890
basenet+SEM	0.880	0.038	0.887
basenet+PSD+AFM	0.891	0.038	0.898
basenet+PSD+AFM+SEM	0.898	0.036	0.902

Table 3: Ablation study results of the gradual combination of the modules mentioned on the DUTS-TE dataset.

to better verify the effectiveness of each module in PFSNet. We use MAE, F_β^* and E_ξ to compare different methods under the DUTS-TE dataset. The experiment selects ResNet-50 as the backbone and selects direct concatenation operation to merge the features of each layer of ResNet-50 as the basic network. Then PFSD, AFM, and SEM are embedded into PFSNet in turn. As shown in Tab. 3, PSD plays a major role in PFSNet, which also proves the effectiveness of the No Leaping Fusion Operations. Besides, with the addition of modules, the test performance gradually improves, which demonstrates the effectiveness of the proposed modules.

Conclusion

In this paper, we propose a pyramidal feature shrinking network PFSNet for salient object detection. We emphatically consider the differences of features at different levels and propose a decoder PSD that gradually shrinks adjacent feature nodes in an asymptotic manner. In the decoder, we design an adjacent fusion module AFM to retain useful information in adjacent feature nodes and reduce noise. In addition, In order to make the initial features more diverse, we design a scale-aware enrichment module SEM to preprocess the features extracted from the backbone. Finally, extensive quantitative and qualitative experiments demonstrate that the proposed intuitive framework outperforms 14 state-of-the-art approaches on five public datasets.

Acknowledgements

This work is partially supported by the National Natural Science Foundation of China under the Grant 61922006, and Baidu academic collaboration program.

References

Chen, Z.; Xu, Q.; Cong, R.; and Huang, Q. 2020. Global context-aware progressive aggregation network for salient object detection. *arXiv preprint arXiv:2003.00651* .

Fan, D.-P.; Gong, C.; Cao, Y.; Ren, B.; Cheng, M.-M.; and Borji, A. 2018. Enhanced-alignment measure for binary foreground map evaluation. *arXiv preprint arXiv:1805.10421* .

Li, G.; and Yu, Y. 2015. Visual saliency based on multiscale deep features. In *Proceedings of the IEEE conference on CVPR*, 5455–5463.

Li, X.; Yang, F.; Cheng, H.; Liu, W.; and Shen, D. 2018. Contour knowledge transfer for salient object detection. In *Proceedings of the ECCV*, 355–370.

Li, Y.; Hou, X.; Koch, C.; Rehg, J. M.; and Yuille, A. L. 2014. The secrets of salient object segmentation. In *Proceedings of the IEEE Conference on CVPR*, 280–287.

Liang, P.; Pang, Y.; Liao, C.; Mei, X.; and Ling, H. 2016. Adaptive objectness for object tracking. *IEEE Signal Processing Letters* 23(7): 949–953.

Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; and Belongie, S. 2017. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on CVPR*, 2117–2125.

Liu, J.-J.; Hou, Q.; and Cheng, M.-M. 2020. Dynamic Feature Integration for Simultaneous Detection of Salient Object, Edge and Skeleton. *arXiv preprint arXiv:2004.08595* .

Liu, J.-J.; Hou, Q.; Cheng, M.-M.; Feng, J.; and Jiang, J. 2019. A Simple Pooling-Based Design for Real-Time Salient Object Detection. In *IEEE CVPR*.

Liu, N.; Han, J.; and Yang, M.-H. 2018. Picanet: Learning pixel-wise contextual attention for saliency detection. In *Proceedings of the IEEE Conference on CVPR*, 3089–3098.

Luo, Z.; Mishra, A.; Achkar, A.; Eichel, J.; Li, S.; and Jodoin, P.-M. 2017. Non-local deep features for salient object detection. In *Proceedings of the IEEE Conference on CVPR*, 6609–6617.

Pang, Y.; Zhao, X.; Zhang, L.; and Lu, H. 2020. Multi-Scale Interactive Network for Salient Object Detection. In *Proceedings of the IEEE/CVF Conference on CVPR*, 9413–9422.

Qin, X.; Zhang, Z.; Huang, C.; Dehghan, M.; Zaiane, O. R.; and Jagersand, M. 2020. U2-Net: Going deeper with nested U-structure for salient object detection. *Pattern Recognition* 106: 107404.

Qin, X.; Zhang, Z.; Huang, C.; Gao, C.; Dehghan, M.; and Jagersand, M. 2019. BASNet: Boundary-Aware Salient Object Detection. In *Proceedings of the IEEE/CVF Conference on CVPR*.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, 234–241. Springer.

Rutishauser, U.; Walther, D.; Koch, C.; and Perona, P. 2004. Is bottom-up attention useful for object recognition? In *Proceedings of the 2004 IEEE Computer Society Conference on CVPR, 2004. CVPR 2004.*, volume 2, II–II. IEEE.

Su, J.; Li, J.; Zhang, Y.; Xia, C.; and Tian, Y. 2019. Selectivity or Invariance: Boundary-aware Salient Object Detection. In *ICCV*.

Wang, L.; Lu, H.; Wang, Y.; Feng, M.; Wang, D.; Yin, B.; and Ruan, X. 2017. Learning to detect salient objects with image-level supervision. In *Proceedings of the IEEE Conference on CVPR*, 136–145.

Wei, J.; Wang, S.; and Huang, Q. 2019. F3Net: Fusion, feedback and focus for salient object detection. *arXiv preprint arXiv:1911.11445* .

Wu, Z.; Su, L.; and Huang, Q. 2019. Cascaded Partial Decoder for Fast and Accurate Salient Object Detection. In *Proceedings of the IEEE/CVF Conference on CVPR*.

Xia, C.; Li, J.; Chen, X.; Zheng, A.; and Zhang, Y. 2017. What is and what is not a salient object? learning salient object detector by ensembling linear exemplar regressors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4142–4150.

Yan, Q.; Xu, L.; Shi, J.; and Jia, J. 2013. Hierarchical saliency detection. In *Proceedings of the IEEE conference on CVPR*, 1155–1162.

Yang, C.; Zhang, L.; Lu, H.; Ruan, X.; and Yang, M.-H. 2013. Saliency detection via graph-based manifold ranking. In *Proceedings of the IEEE conference on CVPR*, 3166–3173.

Yao, Q.; and Gong, X. 2019. Saliency Guided Self-attention Network for Weakly-supervised Semantic Segmentation. *arXiv preprint arXiv:1910.05475* .

Zhang, L.; Dai, J.; Lu, H.; He, Y.; and Wang, G. 2018a. A bi-directional message passing model for salient object detection. In *Proceedings of the IEEE Conference on CVPR*, 1741–1750.

Zhang, P.; Wang, D.; Lu, H.; Wang, H.; and Ruan, X. 2017. Amulet: Aggregating multi-level convolutional features for salient object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, 202–211.

Zhang, X.; Wang, T.; Qi, J.; Lu, H.; and Wang, G. 2018b. Progressive attention guided recurrent network for salient object detection. In *Proceedings of the IEEE Conference on CVPR*, 714–722.

Zhao, T.; and Wu, X. 2019. Pyramid Feature Attention Network for Saliency Detection. In *Proceedings of the IEEE/CVF Conference on CVPR*.

Zhao, X.; Pang, Y.; Zhang, L.; Lu, H.; and Zhang, L. 2020. Suppress and balance: A simple gated network for salient object detection. *arXiv preprint arXiv:2007.08074* .

Zhou, H.; Xie, X.; Lai, J.-H.; Chen, Z.; and Yang, L. 2020. Interactive Two-Stream Decoder for Accurate and Fast Saliency Detection. In *Proceedings of the IEEE/CVF Conference on CVPR*, 9141–9150.