IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 24 August 2022, accepted 11 September 2022, date of publication 15 September 2022,
date of current version 23 September 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3206945

RESEARCH ARTICLE

MPTC-FPN: A Multilayer Progressive FPN With Transformer-CNN Based Encoder for Salient Object Detection

XIAOQI YANG AND LIANGLIANG DUAN
School of Information and Control Engineering, Qingdao University of Technology, Qingdao 266520, China
Corresponding author: Liangliang Duan (hengxingdll9@163.com)

This work was supported by the National Natural Science Foundation of Shandong Province under Grant ZR2019PF019 and Grant ZR2020QF044.

ABSTRACT Due to the development of Convolutional Neural Networks (CNN), significant progress has been made in Salient Object Detection (SOD). However, methods based on CNN are difficult to achieve good results in learning global context information. Recently, with the rapid development of vision transformer, it provides a new perspective for the performance improvement of salient object detection. Benefiting from the powerful capability of global modeling, transformer can supplement rich global contextual information. For lacking the ability to learn local details, it is suboptimal to only adopt transformer as encoder. Therefore, how to skillfully combine local details and global context information is crucial. We conbine CNN and transformer to propose a Multilayer Progressive FPN with Transformer-CNN Based Encoder For Salient Object Detection (MPTC-FPN). Similar to most of the previous methods, we adopt the FPN network as the basic structure. But the difference from previous methods is that we have six initial features before feature fusion, instead of the traditional four or five. We use a low-level feature generation module (LFGM) to generate a lower-level feature to supplement local details. In addition, we also propose a module to reduce the difference between features (DRM), making the features more conducive to fusion. On the basis of FPN, we add a large number of feature fusion nodes, which makes the process of feature fusion smoother. Moreover, we adjust the supervision strategy, use multiple supervision points, and adopt an appropriate weight distribution strategy among the multiple supervision points. A series of comprehensive experimental results demonstrates that our proposed method outperforms previous state-of-the-art methods on five datasets.

INDEX TERMS Transformer-CNN, hybrid encoding, salient object detection, feature aggregation, feature pyramid network.

I. INTRODUCTION

Salient Object Detection (SOD) aims to locate and segment the most important objects or regions in a given image or video [1], [2], [3]. It has been applied to numerous vision problems, including visual tracking [4], image retrieval [7], content-aware image editing [5], robot navigation [6]. Traditional salient object detection (SOD) methods [8], [9], [10], [11], [12], [36] mostly rely on hand-crafted features, such as color contrast, boundary background. However, during saliency maps generating, the lack of high-level semantic information limits its accuracy. In recent years, the rapid development of convolutional neural networks has injected new vitality into the field of salient object detection, which has greatly improved its performance compared to traditional methods. In the field of salient object detection, Encoder-decoder network architectures dominate. These methods usually include two parts: encoder and decoder. The encoder usually uses a pre-trained convolutional neural network model as the backbone network to extract features at different levels, such as VGG [14], ResNet [15]. Decoders are usually carefully designed by researchers spending plenty

The associate editor coordinating the review of this manuscript and approving it for publication was Yiming Tang.

of time to combine low-level features with rich spatial details and high-level features with semantic information. The features of different levels are fused by the decoder to generate the final predicted saliency map.

But a widespread problem is that semantic information in high-level features is gradually ablated during feature fusion. Meanwhile, low-level features introduce background noise, which has tremendous negative impact on the generated significant accuracy and become the first problem to be addressed when designing the network structure. Therefore, maintaining the clarity of high-level semantic information in the feature integration phase and suppressing the background noise introduced by low-level features are the keys to the excellent performance of methods in the field of salient object detection. However, existing methods [22], [51], [52] with convolutional neural network as the backbone network is limited because of the lack of powerful global modeling ability.

The recently popular transformer networks [16], [17] provides a new perspective for solving above existing problem, which can break the performance bottleneck. Transformer is introduced from the field of natural language processing (NLP). It obtains global context information through self-attention mechanism and establishes a long-distance dependency. Transformers treat image patches in the same way that they process tokens in natural language processing applications [17]. Transformers have been applied to many computer vision tasks due to their powerful capabilities in global modeling. In the meantime it has been applied to the field of salient object detection, and have achieved considerable results. However, Transformer still lacks in learning local detail information, which inhibits further improving the performance of salient object methods. Because not only the global context information, but also the local detail information is still critical to generate the final saliency map. CNN is lacking in global modeling, while Transformer is insufficient in local detail learning. Therefore, how to effectively combine CNN and Transformer is the critical factor to improve the performance of salient object detection.

We improve on the basis of FPN [13], and combine CNN and Transformer by using a hybrid encoding method. Most existing salient object detection methods use four or five levels of features extracted from the backbone network. Our hybrid encoding method adopts Swin Transformer [18] as the backbone network and uses five levels of different features extracted from it. In addition, we process the original image through a CNN module named LFGM, which generates a low-level feature. It makes our proposed method initially have six different levels of features, which we label as $F1$, $F2$, $F3$, $F4$, $F5$, $F6$. On the basis of this, we propose a feature difference reduction module (DRM) to reduce the gap between different features, making the result of feature fusion more accurate and effective. It is precisely because of the increase in the number of initial features that the network depth is been further deepened. In the feature fusion stage, different from FPN [13], we add a large number of feature fusion nodes and adopt a layer-by-layer fusion method. The purpose of this is to reduce the span between different level features during the feature fusion stage, so that the process of feature fusion is smoother. Simultaneously, we use the proposed CAT module to fuse features at two different levels, and reduce the number of channels between layers to spare computational resource consumption. Because of the addition of a large number of feature fusion nodes, we have more supervision points to choose than FPN. Therefore, we adopt a multi-supervised point strategy and use an appropriate weight distribution strategy for supervised training. Which further improves the accuracy of the final generated saliency map.

Our main contributions can be summarized as follows:

- A hybrid encoding method is adopted to combine Transformer and CNN.The low-level feature generation module (LFGM) is used to generate a lower-level feature while the transformer is used to capture long-range dependencies.
- Based on FPN, we propose a novel deep network structure called MPTC-FPN. The structure of MPTC-FPN is more suitable for multi-supervised strategies.
- A feature difference reduction module DRM is proposed to reduce the gap between different levels of features and make it beneficial to feature fusion.
- The CAT module is used for feature fusion, and a layer-by-layer progressive strategy is adopted during fusion. In addition, in the feature fusion stage, we continuously reduce the number of channels to save computing resources.

II. RELATED WORK

In this section, we will introduce some recent salient object detection methods and the application of transformer in computer vision.

A. SALIENT OBJECT DETECTION

The vast majority of traditional salient object detection methods [8], [9], [10], [11], [12], [36] are based on hand-crafted features, such as color contrast, edge priors, background information, etc. Based on multi-level image segmentation, Jiang *et al.* [8] used a supervised learning method to map regional feature vectors to saliency scores, and then fused the saliency scores of different levels to generate a saliency map. Perazzi *et al.* [10] proposed a clear and intuitive algorithm for contrast-based saliency estimation. Starting from the perspective of reconstruction error, Li *et al.* [11] calculated the dense and sparse reconstruction errors of each image region, and then obtained the final result through a series of calculations. Although these methods have achieved good results from different perspectives, the lack of high-level semantic information limits the improvement of these methods in accuracy.

In recent years, convolutional neural networks have developed rapidly, and most salient object methods based on convolutional neural networks have achieved excellent results. Profit from the powerful feature extraction

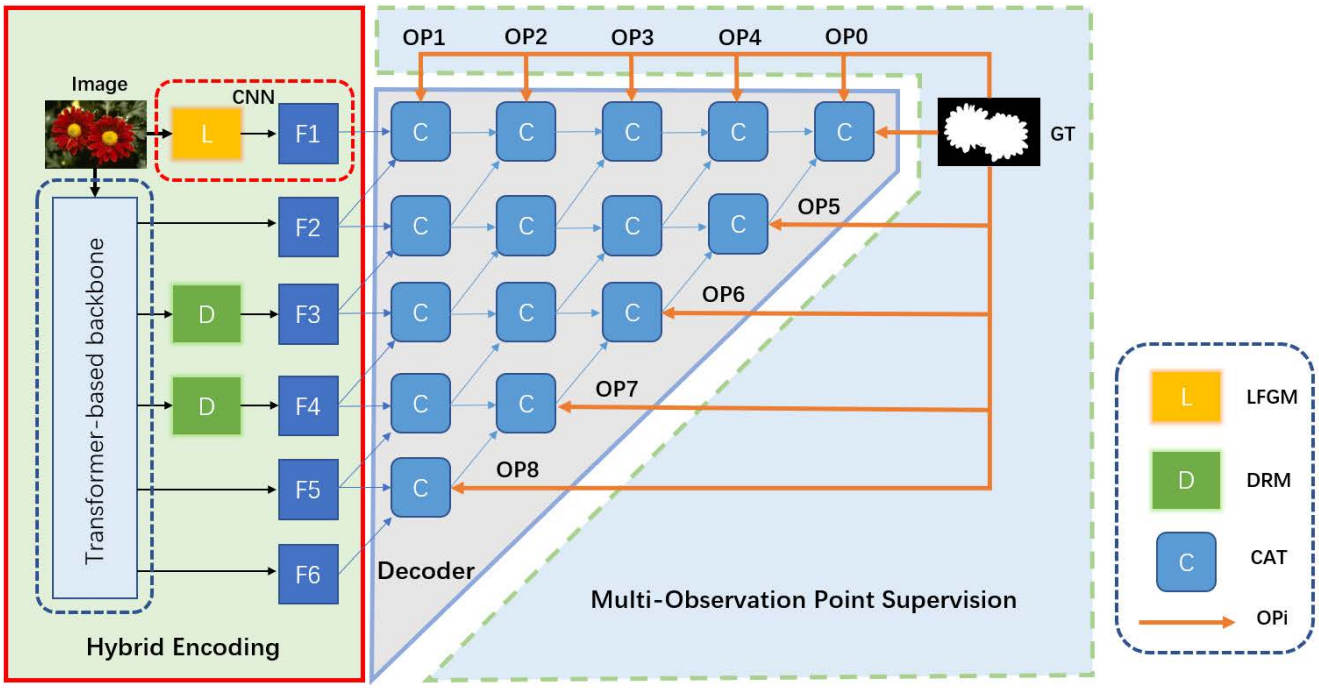

FIGURE 1. Overall architecture diagram of our proposed MPTC-FPN. LFGM is the low-level feature generation module. DRM and CAT are difference reduction module and feature aggregation module. OPi is defined as the number of observation points. The value range of i is 0 to 8.

capabilities of convolutional neural networks, convolutional neural networks can extract multi-level information in original images. With effective high-level semantic information, salient objects or regions can be located more accurately. Therefore, the traditional salient object detection methods based on hand-crafted features have been gradually abandoned. Li and Yu [35] used CNN to extract multi-scale features and compute saliency values for each superpixel. Zhao *et al.* [19] proposed a multi-context deep learning framework that uses both global and local context modeling. Wang *et al.* [20] used two CNNs with different functions to combine local estimation and global search and generate the final saliency map. Compared to traditional methods, these methods have achieved remarkable progress. However, these methods ignore important overall spatial information because they process the image in a patch-level manner, which limits the continued improvement of performance.

To address the limitations of CNNs on image pixel-level segmentation, fully convolutional neural networks (FCN) [21] are proposed. It inspired the field of salient object detection, and researchers have put more effort into pixel-level saliency map generation. As we know, the low-level features generated by shallow networks have rich local details and can be used to refine the boundaries of salient objects or regions, while the high-level features generated by deep networks have rich semantic information and are mainly used to locate salient objects or regions. Therefore, the ability to effectively fuse low-level features and high-level features is the key to generating high-quality saliency maps. Liu *et al.* [22] extended the role of pooling layers in convolutional neural networks and proposed an efficient network model. Wei *et al.* [23] designed a cascaded feedback decoder to refine previous features with high-resolution and high-level semantic features. In addition, a pixel position-aware loss function is designed to assign different weights to pixels at different positions. Zhao *et al.* [24] uses a multi-level gating unit to control the information of the encoder to flow to the decoder reasonably. Ma *et al.* [25] designed a unique multi-scale information extraction module. The network structure they proposed only fuses adjacent feature nodes during the feature fusion process, effectively suppressing the introduction of background noise. However, the lack of global modeling ability makes CNN-based salient object detection methods encounter a bottleneck in performance improvement again. This paved the way for the introduction of Transformer in the field of salient object detection.

B. APPLICATION OF TRANSFORMER IN COMPUTER VISION

Transformer [16] was first proposed in the field of natural language processing and applied to machine translation. In many natural language processing tasks, it has achieved remarkable results. Dosovitskiy *et al.* [17] first introduced Transformer to the field of computer vision and achieved state-of-the-art methods on multiple standard datasets for image classification. Compared with CNN-based methods, their proposed Vision Transformer (ViT) requires less computational resources [17]. Wang *et al.* [26] proposed the

Pyramid Vision Transformer (PVT), a pure Transformer backbone network that can be used for a variety of pixel-level dense prediction tasks. Liu *et al.* [18] proposed a hierarchical Vision Transformer (Swin Transformer), which aims to become a general computer vision backbone network.Transformer has a very wide range of applications in the field of computer vision [57], such as object detection [58], segmentation [59], pose estimation [60]. Besides, Transformer was introduced into SOD field. Liu *et al.* [27] proposed a unified model (VST) for salient object detection from a new perspective of sequence-to-sequence modeling based on a pure Transformer structure. Mao *et al.* [28] used the Swin Transformer [18] as the backbone network to conduct research on salient object detection and camouflaged object detection. Benefiting from the powerful global modeling capabilities of transformer, Transformer-based salient object detection methods have achieved remarkable results. However, most of these methods ignore local detail information, which plays a key role in refining the boundaries of salient objects. Abundant local details can make the generated saliency map more refined. Therefore, how to supplement local detail information in the process of feature fusion is significant for Transformer-based salient object detection methods.

In order to solve the above problems, we adopt the form of hybrid encoding to effectively combine CNN and Transformer. While utilizing the powerful global modeling ability of Transformer, the low-level features generated by CNN are used to supplement local details.

III. METHOD

In this section, we will describe our proposed module. In the first part, we give an overall overview of the proposed network structure. In the second part, we will describe the encoder part in more detail, especially the hybrid coding strategy adopted. In the third part, we will introduce the decoder part and the modules used in detail. In the fourth part, we will elaborate on the proposed DRM module at length. In the final fifth part, we give a brief introduction to the loss function we use. A more intuitive representation of the entire network is shown in Figure 1.

A. OVERVIEW OF NETWORK

Our network structure is an MPTC-FPN structure formed by improving the FPN [13] structure. As a result of the use of hybrid coding, the network structure is further deepened. DRM is not applied to all initial features, only added to $F3$ and $F4$. In feature fusion, we add a large number of feature fusion nodes. Finally, we supervise a total of nine feature points at the top and both sides of the network, and use a reasonable strategy to adjust the weight ratio between each supervision point.

B. ENCODER

As we mentioned earlier, high-level features contain semantic information, which can precisely locate salient objects or regions. The low-level features are rich in local detail information, which can well complement the local details in the generated saliency map. In previous works, most of the methods used five-level features extracted from the backbone network, and then used a well-designed decoder for feature fusion, and finally generated a better saliency map. For some reasons, some methods abandon the use of the first-level features, and use the fourth-level features extracted by the backbone network, and then perform feature fusion to generate the final result.

While our encoder structure is different from the previous methods, in this part, we adopt a hybrid encoding method. These include the five-level features encoded with Transformer and our newly added one-level features. Transformers use self-attention to capture long-term dependencies in the data, which has important implications for capturing global contextual information. Swin Transformer constructs hierarchical feature maps, and this hierarchical architecture reduces the computational complexity related to image size to linear. This greatly improves computational efficiency and can serve as a general computer vision backbone. We choose Swin-B pre-trained on the ImageNet-1K dataset [29] as the backbone network. The image input size is $384 * 384$, and the five-level feature maps extracted by the backbone network are $96 * 96$, $48*48$, $24*24$, $12*12$, and $12*12$ respectively. The number of channels is 128, 256, 512, 1024, 1024, respectively. We label these five-level feature maps as $F2$, $F3$, $F4$, $F5$, and $F6$, respectively. To refine the generated saliency map, supplementing local spatial details, we introduce a lower-level feature. We adopt low-level feature generation module (LFGM) to generate a feature map of size $192 * 192$ and the number of channels is 64. This lower-level feature contains a large amount of local detail information, which is complementary to the powerful global modeling ability of transformer. This good complementary form can not only accurately locate salient objects and regions but also supplement local spatial details during feature fusion. Therefore, higher-quality saliency maps can be generated, which greatly improves the accuracy. We label this lower-level feature as $F1$. LFGM can be expressed as follow:

$$X^* = ReLU(BN(Conv(I)))$$
$$O = ReLU(BN(Conv(MaxPool(X^*))))$$

I represents the input original image, while X* represents the intermediate state generated during processing. O represents the final generated result. Conv, BN, and ReLU are represented as convolutional layers, normalization layers, and RelU activation functions, respectively.

C. DECODER

In the decoder part, we added a large number of feature fusion nodes, and formed a layer-by-layer progressive structure. Because of the introduction of feature $F1$ in hybrid encoding, the depth of our network is further deepened. From six feature nodes in leftmost side of network structure, it is

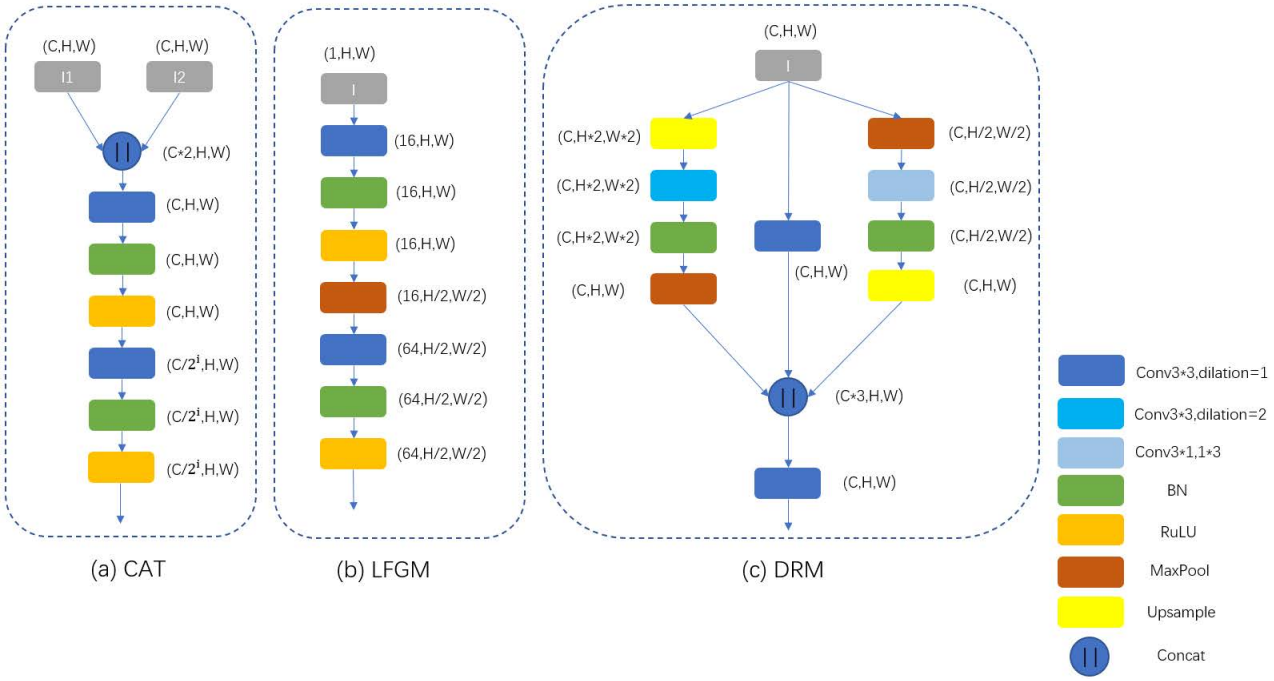

FIGURE 2. Visual presentation of the details about our proposed three modules.(a) The CAT module is used to fuse the two features and reduce the number of channels. (b) LFGM is used to generate lower-level feature representation maps in hybrid coding. (c) DRM is used to reduce the difference between different levels of features and improve the feature fusion effect.

reduced layer by layer to one feature node in the far right. When performing feature fusion, we adopt a simple feature fusion module namely CAT. The higher-level feature maps are first upsampled to the same size as the adjacent lower-level features at first. Then two feature maps of the same size are connected, and the feature maps generated after the connection are passed through a series of convolutional layers to complete feature fusion. The added large number of feature fusion nodes can alleviate the problem of too large span between different features, and can better fuse features at all levels. In the process of progressive layer by layer, we are continuously reducing the number of channels to save the consumption of computing resources. After the intermediate process graph generated by the connection operation passes through a series of convolutional layers, it will be reduced to $1/2$, $1/4$, $1/8$, and $1/16$ of the original number of channels in different layers. The CAT module we use can be formulated as follows:

$$\begin{aligned} X^* &= Concat(X, Y) \\ Z &= ReLU(BN(Conv(X^*))) \\ O &= ReLU(BN(Conv(Z))) \end{aligned}$$

X and Y represent the two feature points to be fused, O represents the process map generated during the convolution process, and $Z*$ represents the final result generated after the fusion of the two feature points.

D. DIFFERENCE REDUCTION MODULE

Excessive differences between different-level features will also affect the effect of feature fusion. Therefore, we design a Difference Reduction Module (DRM) to reduce the difference between features, so that the effect of feature fusion is further improved. Probably different from many previous works, we only add DRM module to the middle two features, not to all the features. $F3$ and $F4$ are two intermediate-level features, which play a critical role in communicating high-level features and low-level features, and are a bridge between high-level features and low-level features. For this reason, we only add our proposed module to the middle layers. It makes the processed $F3$ and $F4$ more communicative and adaptable between high-level features and low-level features, which is more conducive to feature fusion.

Some previous work [53], [54], [55] has demonstrated that enriching the receptive fields of convolution kernels enhances neural network learning for objects of different scales and sizes during feature processing. Therefore we use three different types of convolutions in the DRM module, including ordinary convolutions, asymmetric convolution [30] and atrous convolutions [31]. We deal with the input features in three ways. First of all, we duplicate the input features twice, and upsample and downsample the copied features respectively. After upsampling and downsampling, there will be three different sizes of features. Then we use the atrous convolutional layer to process the features after upsampling, and use the asymmetric convolutional layer to process the features after downsampling. For the original size feature, we use original convolution for processing. Then, we restore the previously up-sampling and down-sampling features to their original size using down-sampling and up-sampling, respectively. The three convolutional features of different types are connected

together through the connection operation, and then passed through a common $3 * 3$ convolutional layer. DRM can be formalized as follows:

$$\begin{aligned} X_1 &= MaxPool(Conv_{atr}(Up(I))) \\ X_2 &= Conv_{3*3}(I) \\ X_3 &= Up(Conv_{asy}(MaxPool(I))) \\ O &= Conv_{3*3}(Concat(X_1, X_2, X_3)) \end{aligned}$$

We employ atrous convolution with a dilation rate of 2, which is defined as $Conv_{atr}$. For the convolution kernel of asymmetric convolution, we adopt the dimensions of $1*3$ and $3*1$, which are defined as $Conv_{asy}$. The size of the convolution kernel of ordinary convolution, we use the ordinary size of $3 * 3$. For the downsampling process, we use a max pooling layer for processing. And Upsampling is denoted as Up.

E. LOSS FUNCTION

Our proposed MPTC-FPN structure facilitates the realization of a strategy of multiple supervision points for supervision. Therefore, we used nine supervision points in the training process and adjusted the weight ratio between each supervision point. We used binary cross-entropy loss function, IOU loss function and progressive self-guided (PSG) loss [56].

The binary cross-entropy loss function ignores the similarity in image structure, and only calculates the difference between individual pixels. The IOU loss function can be used to calculate the similarity of the overall structure between two images, so combining it with the binary cross-entropy loss function can achieve better training results.In addition, we use PSG loss as an anxiliary loss function for training. The binary cross-entropy loss function, IOU loss function and the PSG loss are denoted as $\mathcal{L}_{\text{bce}}$, $\mathcal{L}_{\text{iou}}$ and $\mathcal{L}_{\text{psg}}$, respectively. The final loss function is defined as $\mathcal{L}_{\text{total}}$ The overall loss function we use is as follows:

$$\mathcal{L} = \mathcal{L}_{\text{bce}} + \mathcal{L}_{\text{iou}} + \mathcal{L}_{\text{psg}} \tag{1}$$

$$\mathcal{L}_{\text{total}} = \alpha \times \mathcal{L}^{(0)} + (1-\alpha) \times \sum_{k=1}^{8} \mathcal{L}^{(k)} \tag{2}$$

We adjust the weight ratio of different supervision points. In our paper, we set α to 0.6. The binary cross-entropy loss can be formulated as:

$$\begin{aligned} \mathcal{L}_{\text{bce}} = -\sum_{i,j}^{H,W} &[G(i,j)\log(P(i,j)) \\ &+ (1-G(i,j))\log(1-P(i,j))] \end{aligned} \tag{3}$$

H and W represent the height and width of the image, respectively. G represents the ground-truth map, and P represents the saliency map generated by prediction. The IoU loss function can be expressed as:

$$\mathcal{L}_{\text{iou}} = 1 - \frac{\sum_{x=1}^{H}\sum_{y=1}^{W} P(x,y)G(x,y)}{\sum_{x=1}^{H}\sum_{y=1}^{W}[P(x,y)+G(x,y)-P(x,y)G(x,y)]} \tag{4}$$

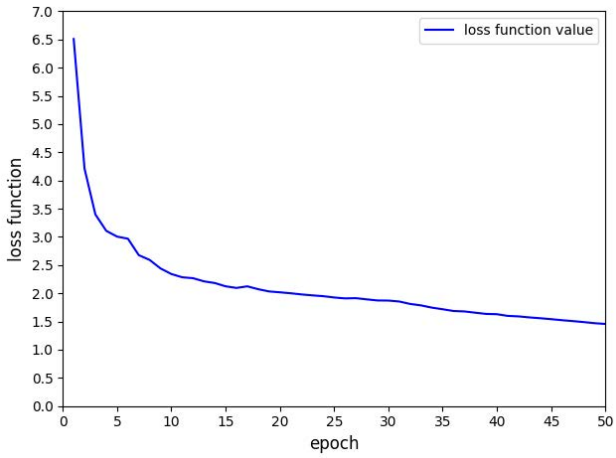

FIGURE 3. Convergence curve of the loss function.

The symbols in the formula have the same meaning as those in the binary cross-entropy loss function. The PSG loss can be formulated as follows:

$$\mathcal{L}_{\text{psg}} = L(SM_{pred}, f(SM_{pred})) \tag{5}$$

$L(*,*)$ represents the main loss function used. So $L(*,*)$ can be expressed as follows:

$$L(*,*) = \mathcal{L}_{\text{bce}}(*,*) + \mathcal{L}_{\text{iou}}(*,*) \tag{6}$$

$f(*)$ is defined as a simulated morphological closing operatoin. It can be described as follows:

$$f(SM_{pred}) \approx maxpool(SM_{pred}) \cap SM_{gt} \tag{7}$$

SM_{pred} and SM_{gt} represent the generated saliency map and ground truth, respectively. Our goal is to reduce the loss function as the number of training epochs increases.

IV. EXPERIMENTS

In this section, we will describe the content of the five subsections. These include experimental details, selection of datasets, evaluation metrics, performance comparisons, and experimental studies of ablation. We will demonstrate the superiority of our proposed method by comparing our method with some previous state-of-the-art methods. In addition, we conduct a series of ablation experiments to explore the impact of each module or strategy used by our proposed MPTC-FPN on the experimental result. In addition, for the training process of the model, we draw the loss function convergence curve, as shown in Figure 3.

A. IMPLEMENTATION DETAILS

The proposed approach is implemented by the Pytorch. The SGD optimizer [32] with weight decay of 5e-4 and momentum of 0.9 is adopted to optimize the network. We use a warm up strategy, and warming up epochs is 6. Meantime, poly is adopted to adjust the learning rate. The learning rate change

TABLE 1. Quantitative comparison results between our method and sixteen previous state-of-the-art methods on five datasets.Higher values for F_β, F_β^ω and mean better, and lower values for S_m mean better.The best performing values are marked in **bold**.

Methods	ECSSD		DUTS-TE		HKU-IS		DUT-OMRON		PASCAL-S	
	F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE
Amulet	0.915	0.059	0.778	0.085	0.897	0.051	0.743	0.098	0.839	0.099
UCF	0.903	0.069	0.773	0.112	0.888	0.062	0.730	0.120	0.824	0.116
BRN	0.922	0.041	0.828	0.050	0.910	0.036	0.774	0.062	0.856	0.073
C2SNet	0.896	0.059	0.790	0.066	0.883	0.051	0.733	0.079	0.840	0.088
AFNet	0.935	0.042	0.863	0.046	0.923	0.036	0.797	0.057	0.871	0.071
BASNet	0.942	0.037	0.860	0.048	0.928	0.032	0.805	0.056	0.860	0.079
F3Net	0.945	0.033	0.891	0.035	0.937	0.028	0.813	0.053	0.882	0.064
CAGNet-R	0.937	0.037	0.866	0.040	0.926	0.030	0.791	0.054	0.873	0.069
GCPANet	0.948	0.035	0.888	0.038	0.938	0.031	0.812	0.056	0.882	0.063
ITSD	0.947	0.034	0.883	0.041	0.934	0.031	0.821	0.061	0.882	0.066
LDF	0.950	0.034	0.898	0.034	0.939	0.027	0.820	0.051	0.887	0.062
MINet	0.947	0.033	0.884	0.037	0.935	0.029	0.810	0.055	0.880	0.066
GateNet	0.952	0.035	0.898	0.035	0.942	0.029	0.829	0.051	0.888	0.065
VST	0.951	0.033	0.890	0.037	0.942	0.029	0.825	0.058	0.890	0.062
PAKRN	0.953	0.032	0.907	0.033	0.943	0.027	0.834	0.050	0.888	0.068
PFSNet	0.952	0.031	0.896	0.036	0.943	0.026	0.823	0.055	0.887	0.065
Ours	**0.961**	**0.023**	**0.919**	**0.024**	**0.953**	**0.021**	**0.847**	**0.042**	**0.905**	**0.052**

Methods	ECSSD		DUTS-TE		HKU-IS		DUT-OMRON		PASCAL-S	
	F_β^ω	S_m	F_β^ω	S_m	F_β^ω	S_m	F_β^ω	S_m	F_β^ω	S_m
Amulet	0.840	0.894	0.658	0.804	0.817	0.886	0.626	0.781	0.739	0.819
UCF	0.806	0.883	0.596	0.782	0.779	0.875	0.573	0.760	0.698	0.806
BRN	0.891	0.903	0.774	0.842	0.875	0.894	0.709	0.806	0.802	0.837
C2SNet	0.839	0.882	0.701	0.818	0.818	0.873	0.640	0.780	0.762	0.826
AFNet	0.886	0.913	0.785	0.867	0.869	0.905	0.717	0.826	0.804	0.850
BASNet	0.904	0.916	0.803	0.866	0.889	0.909	0.751	0.836	0.797	0.834
F3Net	0.912	0.924	0.835	0.888	0.900	0.917	0.747	0.838	0.823	0.857
CAGNet-R	0.902	0.908	0.817	0.864	0.893	0.904	0.728	0.815	0.816	0.839
GCPANet	0.903	0.927	0.821	0.891	0.889	0.920	0.734	0.839	0.819	0.864
ITSD	0.910	0.925	0.824	0.885	0.894	0.917	0.750	0.840	0.823	0.859
LDF	0.915	0.924	0.845	0.892	0.904	0.919	0.752	0.838	0.829	0.859
MINet	0.911	0.925	0.825	0.884	0.897	0.919	0.738	0.833	0.818	0.854
GateNet	0.906	0.929	0.828	0.897	0.893	0.925	0.749	0.849	0.821	0.865
VST	0.910	0.932	0.828	0.896	0.897	0.928	0.755	0.850	0.827	0.871
PAKRN	0.918	0.928	0.861	0.900	0.909	0.923	0.779	0.853	0.828	0.858
PFSNet	0.920	0.930	0.842	0.892	0.910	0.924	0.756	0.842	0.829	0.859
Ours	**0.939**	**0.941**	**0.888**	**0.914**	**0.930**	**0.935**	**0.800**	**0.864**	**0.859**	**0.876**

formula can be expressed as follows:

$$lr = lr_{init} \times (1 - \frac{k}{epochs})^{\gamma} \tag{8}$$

lr_{init} represents the initial learning rate. The value of γ is 0.9. We initialize the learning rate to 8e-3. The pre-trained Swin Transformer-B [18] on ImageNet-1K is used as backbone, and we set its initial learning rate as one tenth of that of other parts, which is 8e-4. We follow the way of [25] to initialize the parameters of other parts. All of the image is adjusted to 384 * 384 input the network. For data augmentation, we adopt horizontal flipping and random cropping. The proposed network is trained for 50 epochs on a PC with a RTX 2080Ti. And we set batch size to 6.

B. DATASETS

We use DUTS-TR [33] dataset to train our network. 10553 images and corresponding annotated maps are included in DUTS-TR [33] dataset. To evaluate the superior performance of our proposed method, we select five popular benchmark datasets: DUTS-TE [33], ECSSD [34], HKU-IS [35], DUT-OMRON [36], and PASCAL-S [37], respectively. DUTS [33] is the largest SOD dataset at this stage.

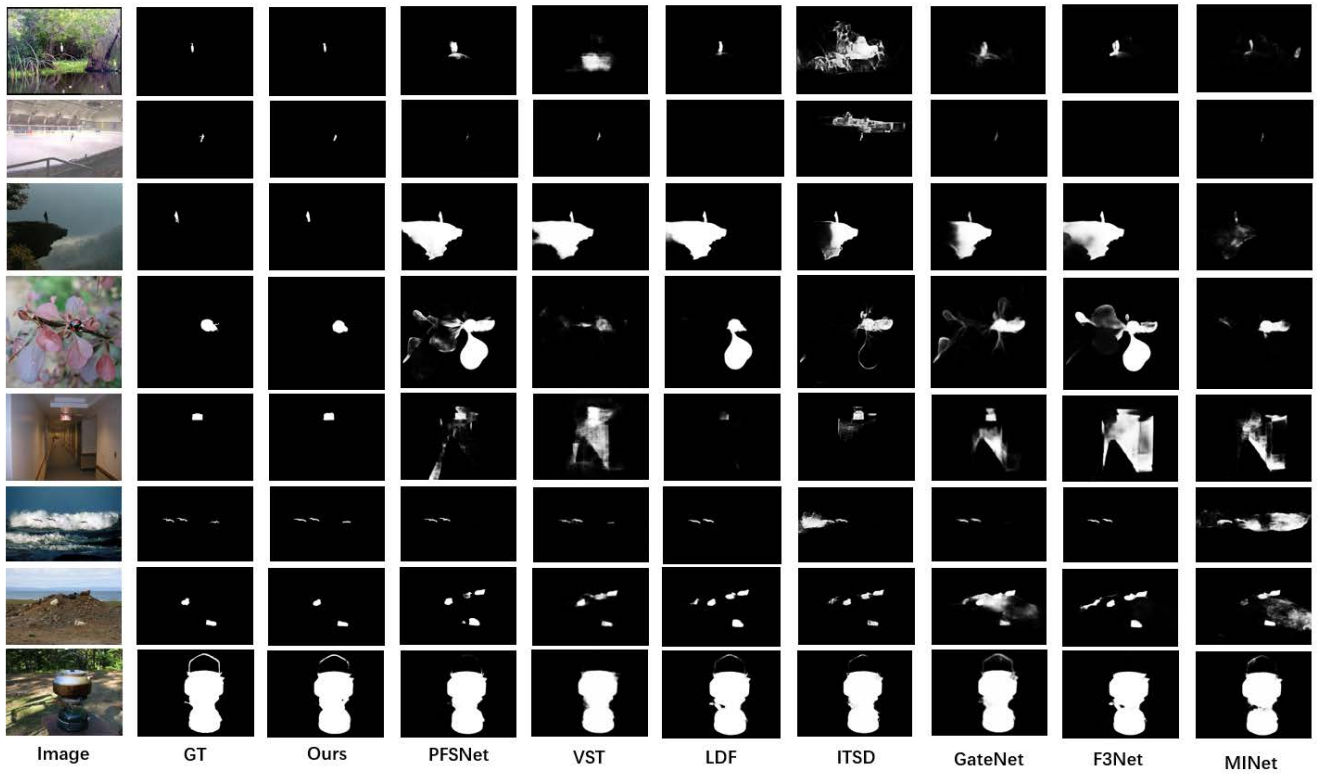

FIGURE 4. Visual comparison of saliency maps generated by our proposed method and seven other state-of-the-art methods. The proposed MPTC-FPN can not only accurately locate salient objects or regions, but also suppress background noise well.

DUTS-TE [33] is a part of DUTS, which contains 5019 test images and corresponding labels. There are 1000 images in ECSSD [34]. These images included in the dataset have meaningful semantic information. HKU-IS [35] contains 4447 images with multiple salient objects. At the same time, the picture background is also very complex. DUT-OMRON [36] contains 5168 images and corresponding labels. The objects in these pictures are often complex in structure and the background of the picture is also complicated. PASCAL-S [37] contains 850 challenging images selected from a dataset originally used for semantic segmentation, namely PASCAL VOC 2010.

C. EVALUATION CRITERIA

To evaluate the accuracy of our proposed network structure, we used four very popular evaluation metrics for a fair comparison.

(1) *MAE* is defined as the absolute error between the binary ground truth and the predicted image. Similarity of the predicted image compared to the binary ground truth is indicated by it. MAE is calculated as

$$MAE = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} |\mathbf{P}(i,j) - \mathbf{G}(i,j)| \tag{9}$$

$\mathbf{P}$ denotes the predicted saliency map and $\mathbf{G}$ denotes the ground-truth. H and W are height and weight of the image.

(2) F-measure is denoted as F_β. It is computed by the weight harmonic mean of the precision and recall. F_β can be formulated as

$$F_\beta = \frac{(1+\beta^2) \times Precision \times Recall}{\beta^2 \times Precision + Recall} \tag{10}$$

As with the previous work, we set β^2 to 0.3 to emphasize the importance of precision. And we report the max values of F_β.

(3) Weight F-measure [38] is denoted as F_β^ω. It can be defined as

$$F_\beta^\omega = \frac{(1+\beta^2) \times Precision^\omega \times Recall^\omega}{\beta^2 \times Precision^\omega + Recall^\omega} \tag{11}$$

F_β^ω uses weighted precision and weighted recall to measure the accuracy of different models, where F_β^ω is also set to 0.3.

(4) S-measure combines the region-aware(S_r) and object-aware(S_o), S_m [39] and focuses on measuring the overall structure similarity. It has the following formula

$$S_m = \alpha S_o + (1-\alpha) S_r \tag{12}$$

Same as some of the previous work, where the α is set to 0.5 as default.

D. PERFORMANCE COMPARISON

We compare our proposed method with 16 previous state-of-the-art methods, including Amulet [40], UCF [41], BRN [42], C2SNet [43], AFNet [61], BASNet [44], F3Net [23],

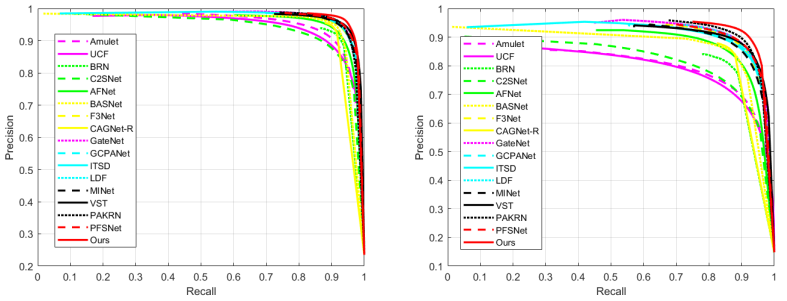

FIGURE 5. Precision recall curves on two saliency datasets,including ECSSD, DUTS-TE.

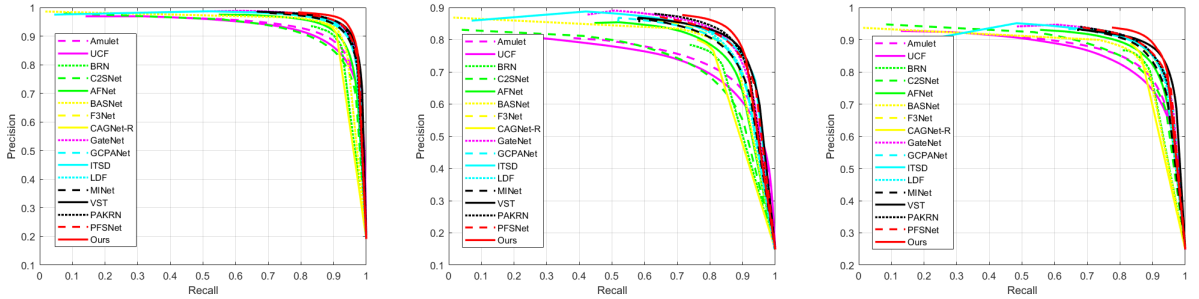

FIGURE 6. Precision recall curves on three common saliency datasets, including HKU-IS, DUT-OMRON and PASCAL-S.

CAGNet-R [45], GCPANet [46], ITSD [47], LDF [48], MINet [49], GateNet [24], VST [27], PAKRN [50], PFSNet [25]. To ensure the fairness of the comparison, all predicted saliency maps are downloaded from the public official website and evaluated under the same evaluation code and environment.

1) QUANTITATIVE COMPARSION

We present the comparison results with the previous 16 state-of-the-art methods on five datasets in Table 1. We adopt four widely-used evaluation metrics, including MAE, F_β, F_β^ω, S_m, where F_β and MAE are in one subtable and F_β^ω and S_m are in another subtable. By comparison, we find that our proposed method significantly outperforms some of the previous methods. The F_β values of MPTC-FPN on five widely adopted datasets, ECSSD, DUTS-TE, HKU-IS, DUT-OMRON, and PASCAL-S, reached 0.961, 0.919, 0.953, 0.847, and 0.905, respectively. Especially on the PASCAL-S dataset, our method outperforms the second best method by 0.015. This is a very significant improvement, while the F_β values improvement on the other four datasets is around 0.01. MAE values We improved by 0.008, 0.009, 0.005, 0.008, 0.010 on the five datasets over the second best data in the table. Among them, the improvement is still the most obvious on the PASCAL-S dataset. The F_β^ω values reached 0.939, 0.888, 0.930, 0.800, and 0.859 on the five datasets, respectively.Among them, the five data sets are improved by 0.019, 0.027, 0.020, 0.021, 0.030 than the second best value. Our method still achieves the most obvious improvement on the PASCAL-S dataset. The S_m values reached 0.941, 0.914, 0.935, 0.864, and 0.876 on the corresponding five datasets, respectively. The improved values are all around 0.010. Through a series of comparisons, the results can demonstrate that our proposed method outperforms previous state-of-the-art methods. More importantly, there has been a great improvement in most cases.

TABLE 2. + represents a simple addition operation. 5 and 6 represent not using mixed encoding and using mixed encoding, respectively. CAT indicates that the CAT module is used for feature fusion. 1ds, 5ds, and 9ds represent the use of one supervision point, five supervision points, and nine supervision points, respectively.

Configuration	DUT-OMRON		PASCAL-S	
	mF_β	MAE	mF_β	MAE
FPN(+)	0.730	0.063	0.805	0.076
5-MPTC-FPN(+)-1ds	0.726	0.059	0.815	0.072
5-MPTC-FPN(CAT)-1ds	0.764	0.049	0.852	0.058
5-MPTC-FPN(CAT)-5ds	0.769	0.047	0.853	0.058
6-MPTC-FPN(CAT)-5ds	0.774	0.049	0.857	0.058
6-MPTC-FPN(CAT)-5ds-DRM	0.765	0.049	0.857	0.057
6-MPTC-FPN(CAT)-9ds	0.773	0.052	0.859	0.056
6-MPTC-FPN(CAT)-9ds-DRM	0.774	0.046	0.858	0.057

2) VISUAL COMPARSION

Some predicted saliency map of the proposed saliency method (MPTC-FPN) and other seven state-of-the-art methods have been shown in Figure 4. In the first and second rows, in the detection scene of small objects, our method can find salient objects more accurately. In the third row, MPTC-FPN can effectively distinguish salient object regions, even when the contrast between salient objects and background is low. In some scenes with complex backgrounds, such as the fourth

TABLE 3. This table shows the effects of different loss functions on the experimental results.

Loss function	ECSSD		DUTS-TE		HKU-IS		DUT-OMRON		PASCAL-S	
	F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE
BCE	0.956	0.029	0.909	0.029	0.948	0.025	0.829	0.046	0.901	0.057
IOU	0.957	0.024	0.910	0.025	0.947	0.022	0.827	0.042	0.898	0.053
BCE + IOU	0.960	0.024	0.917	0.026	0.952	0.021	0.841	0.044	0.910	0.051
BCE + PSG	0.959	0.026	0.911	0.028	0.950	0.024	0.832	0.045	0.901	0.055
IOU + PSG	0.960	0.023	0.913	0.025	0.952	0.020	0.836	0.042	0.901	0.052
BCE + IOU + PSG	0.961	0.023	0.919	0.024	0.953	0.021	0.847	0.042	0.905	0.052
Loss function	ECSSD		DUTS-TE		HKU-IS		DUT-OMRON		PASCAL-S	
	F_β^ω	S_m	F_β^ω	S_m	F_β^ω	S_m	F_β^ω	S_m	F_β^ω	S_m
BCE	0.920	0.937	0.851	0.908	0.908	0.932	0.762	0.854	0.838	0.874
IOU	0.936	0.935	0.881	0.906	0.926	0.928	0.787	0.851	0.855	0.868
BCE + IOU	0.937	0.940	0.880	0.912	0.927	0.934	0.789	0.859	0.860	0.879
BCE + PSG	0.929	0.941	0.861	0.909	0.915	0.933	0.770	0.855	0.846	0.877
IOU + PSG	0.942	0.937	0.887	0.908	0.933	0.932	0.801	0.860	0.862	0.872
BCE + IOU + PSG	0.939	0.941	0.888	0.914	0.930	0.935	0.800	0.864	0.859	0.876

and fifth rows, our method can not only locate the salient regions accurately but also avoid introducing background noise. As shown in the seventh row, MPTC-FPN still performs very well with multiple salient objects. In the eighth row, our method is able to accurately locate the correct salient objects in the face of multiple object interference scenes. For small parts of objects, our method can also detect them completely, which is shown in the ninth row. Through visual comparison, it can be seen that the saliency map generated by our method is more accurate and can better suppress background noise.

In addition, in Figure 5 and 6, we also show the PR curve of MPTC-FPN. We plot the PR curves of MPTC-FPN and the previous sixteen state-of-the-art methods on the corresponding five datasets. We can clearly see that our method curve is higher than other state-of-the-art methods. This means that our method performs better than other state-of-the-art methods.

E. ABLATION STUDY

To demonstrate the effectiveness of the proposed module components and the parameter configurations of MPTC-FPN, we conduct a series of ablation experiments on two challenge datasets (DUT-OMRON and PASCAL-S).

1) EFFECTIVENESS OF MPTC-FPN(CAT)

We give the results of the most basic FPN network in the first row, only replacing the backbone network with Transformer. Then on the second and third line, we changed the structure of the decoder to be MPTC-FPN structure. At the same time, we added the CAT module to the feature fusion. By comparison, we can clearly see that the evaluation values on the two datasets have excellent results in Table 2.

2) EFFECTIVENESS OF MULTIPLE SUPERVISION POINTS

Then we changed the supervision strategy to include multiple supervision points. We increase the supervision points to five based on MPTC-FPN(CAT). We put the results after adding supervision points in the fourth row. It can be seen that the results have improved slightly. This shows that the multi-supervision point strategy still has a positive effect on our proposed method.

3) EFFECTIVENESS OF HYBRID ENCODING

On the basis of the previous one, we changed the structure of the encoder and used a hybrid encoding method for encoding. Because of the addition of a lower-level feature, our network depth is increased from five to six layers. Therefore, we further increased the number of supervision points, raising the number of supervision points to nine. The corresponding experimental results, we can see in the fourth and seventh row. Through comparison, we find that both PASCAL-S and DUT-OMRON are improved.

4) EFFECTIVENESS OF DRM

Finally we added DRM modules on $F3$ and $F4$. We can see the final experimental results in the eighth row. By adding the DRM module, the gap between different levels of features is reduced, which is more conducive to feature fusion.We can clearly see the significant improvement in DUT-OMRON dataset. Experimental result strongly proves that DRM can further improve the performance of the saliency network. We added DRM modules for five supervision points and nine supervision points under the hybrid encoding structure. Based on the experimental results, better results can be achieved in the state of nine supervision points.

5) EFFECTIVENESS OF HYPARAMETER α

We further explored the weight distribution parameters between multiple observation points, and the results are shown in Table 4. We found that the total loss with different α have different effects on final saliency results. From Table 4, when the parameter α is 0.6 gets the best result.

TABLE 4. This table shows the effects of five different parameter values on the experimental results.

Values	DUT-OMRON		PASCAL-S	
	mF_β	MAE	mF_β	MAE
0.3	0.761	0.047	0.786	0.056
0.4	0.773	0.047	0.857	0.057
0.5	0.774	0.047	0.857	0.057
0.6	0.774	0.046	0.858	0.057
0.7	0.773	0.047	0.856	0.057

6) COMPARISON OF DIFFERENT LOSS FUNCTIONS

On the basis of the best results obtained previously, we conduct ablation studies on the loss function used. We adopted a strategy of hybrid loss functions, using three different loss functions. They are BCE loss, IOU loss and PSG loss [56] respectively. PSG is used as an auxiliary loss function. The results of the ablation study for the loss function are shown in Table 3. Through the comparison between the first line and the second line, we find that IOU plays a key role in the decline of *MAE* and the increase of F_β^ω. By comparing the first and fourth lines, the second and fifth lines, we can be sure that the addition of PSG can further improve the performance. Because PSG loss function can only be used as auxiliary function [56]. Therefore, we have not set up an experiment using PSG loss function alone. Finally, based on the performance of various numerical values, we decide to use the sum of three loss functions as the hybrid loss function.

V. CONCLUSION

In this paper, we propose a multi-layer progressive architecture and use the CAT module simultaneously to fuse the features at all levels more smoothly. In the encoder stage, we effectively combine Vision Transformer and CNN to further improve the experimental results. We also designed a difference reduction module (DRM) to reduce the difference between features to further improve the performance. In addition, we adopt a multi-supervised point training strategy and incorporate a more advanced loss function. The experimental results show that our proposed method outperforms previous state-of-the-art methods on five widely used datasets. However, our approach has some limitations, as the addition of a large number of feature fusion nodes increases the complexity of the overall model, which makes the overall model very computationally intensive.

REFERENCES

[1] A. Borji, M.-M. Cheng, H. Jiang, and J. Li, "Salient object detection: A benchmark," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5706–5722, Jan. 2015.
[2] D.-P. Fan, Z. Lin, Z. Zhang, M. Zhu, and M.-M. Cheng, "Rethinking RGB-D salient object detection: Models, data sets, and large-scale benchmarks," *IEEE Trans. neural Netw. Learn. Syst.*, vol. 32, no. 5, pp. 2075–2089, May 2021.
[3] W. Wang, Q. Lai, H. Fu, J. Shen, H. Ling, and R. Yang, "Salient object detection in the deep learning era: An in-depth survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 6, pp. 3239–3259, Jun. 2022.
[4] S. Hong, T. You, S. Kwak, and B. Han, "Online tracking by learning discriminative saliency map with convolutional neural network," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 597–606.
[5] M.-M. Cheng, F.-L. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu, "Repfinder: Finding approximately repeated scene elements for image editing," *ACM Trans. Graph.*, vol. 29, no. 4, pp. 1–8, 2010.
[6] C. Craye, D. Filliat, and J.-F. Goudou, "Environment exploration for object-based visual saliency learning," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2016, pp. 2303–2309.
[7] Y. Gao, M. Wang, D. Tao, R. Ji, and Q. Dai, "3-D object retrieval and recognition with hypergraph analysis," *IEEE Trans. Image Process.*, vol. 21, no. 9, pp. 4290–4303, Sep. 2012.
[8] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li, "Salient object detection: A discriminative regional feature integration approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2083–2090.
[9] W. Zhu, S. Liang, Y. Wei, and J. Sun, "Saliency optimization from robust background detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2814–2821.
[10] F. Perazzi, P. Krähenbühl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 733–740.
[11] X. Li, H. Lu, L. Zhang, X. Ruan, and M.-H. Yang, "Saliency detection via dense and sparse reconstruction," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2976–2983.
[12] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. S. Torr, and S.-M. Hu, "Global contrast based salient region detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 569–582, Mar. 2015.
[13] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 2117–2125.
[14] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
[15] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.
[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.
[17] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.
[18] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted Windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2021, pp. 10012–10022.
[19] R. Zhao, W. Ouyang, H. Li, and X. Wang, "Saliency detection by multi-context deep learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1265–1274.
[20] L. Wang, H. Lu, X. Ruan, and M.-H. Yang, "Deep networks for saliency detection via local estimation and global search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3183–3192.
[21] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3431–3440.
[22] J.-J. Liu, Q. Hou, M.-M. Cheng, J. Feng, and J. Jiang, "A simple pooling-based design for real-time salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 3917–3926.
[23] J. Wei, S. Wang, and Q. Huang, "F^3Net: Fusion, feedback and focus for salient object detection," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 7, pp. 12321–12328.
[24] X. Zhao, Y. Pang, L. Zhang, H. Lu, and L. Zhang, "Suppress and balance: A simple gated network for salient object detection," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 35–51.
[25] M. Ma, C. Xia, and J. Li, "Pyramidal feature shrinking for salient object detection," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 2311–2318.
[26] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2021, pp. 568–578.
[27] N. Liu, N. Zhang, K. Wan, L. Shao, and J. Han, "Visual saliency transformer," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2021, pp. 4722–4732.

[28] Y. Mao, J. Zhang, Z. Wan, Y. Dai, A. Li, Y. Lv, X. Tian, D.-P. Fan, and N. Barnes, "Generative transformer for accurate and reliable salient object detection," 2021, *arXiv:2104.10127*.
[29] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, and A. C. Berg, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
[30] X. Ding, Y. Guo, G. Ding, and J. Han, "AcNet: Strengthening the kernel skeletons for powerful CNN via asymmetric convolution blocks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2019, pp. 1911–1920.
[31] L.-C. 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 Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2016.
[32] L. Bottou, "Stochastic gradient descent tricks," in *Neural Networks: Tricks of the Trade* (Lecture Notes in Computer Science). Berlin, Germany: Springer, 2012, pp. 421–436.
[33] 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. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 136–145.
[34] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 1155–1162.
[35] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Mar. 2015, pp. 5455–5463.
[36] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3166–3173.
[37] Y. Li, X. Hou, C. Koch, J. M. Rehg, and A. L. Yuille, "The secrets of salient object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 280–287.
[38] R. Margolin, L. Zelnik-Manor, and A. Tal, "How to evaluate foreground maps?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 248–255.
[39] D.-P. Fan, M.-M. Cheng, Y. Liu, T. Li, and A. Borji, "Structure-measure: A new way to evaluate foreground maps," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 4548–4557.
[40] P. Zhang, D. Wang, H. Lu, H. Wang, and X. Ruan, "Amulet: Aggregating multi-level convolutional features for salient object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 202–211.
[41] P. Zhang, D. Wang, H. Lu, H. Wang, and B. Yin, "Learning uncertain convolutional features for accurate saliency detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 212–221.
[42] T. Wang, L. Zhang, S. Wang, H. Lu, G. Yang, X. Ruan, and A. Borji, "Detect globally, refine locally: A novel approach to saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3127–3135.
[43] X. Li, F. Yang, H. Cheng, W. Liu, and D. Shen, "Contour knowledge transfer for salient object detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 355–370.
[44] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jagersand, "BASNet: Boundary-aware salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 7479–7489.
[45] S. Mohammadi, M. Noori, A. Bahri, S. G. Majelan, and M. Havaei, "CAGNet: Content-aware guidance for salient object detection," *Pattern Recognit.*, vol. 103, Jul. 2020, Art. no. 107303.
[46] Z. Chen, Q. Xu, and R. Cong, "Global context-aware progressive aggregation network for salient object detection," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 7, pp. 10599–10606.
[47] H. Zhou, X. Xie, J.-H. Lai, Z. Chen, and L. Yang, "Interactive two-stream decoder for accurate and fast saliency detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 9141–9150.
[48] J. Wei, S. Wang, Z. Wu, C. Su, Q. Huang, and Q. Tian, "Label decoupling framework for salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 13025–13034.
[49] Y. Pang, X. Zhao, L. Zhang, and H. Lu, "Multi-scale interactive network for salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 9413–9422.
[50] B. Xu, H. Liang, R. Liang, and P. Chen, "Locate globally, segment locally: A progressive architecture with knowledge review network for salient object detection," in *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35, no. 4, pp. 3004–3012.
[51] Z. Wu, L. Su, and Q. Huang, "Cascaded partial decoder for fast and accurate salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3907–3916.
[52] W. Bo, C. Quan, Z. Min, Z. Zhiqiang, J. Xiaogang, and G. Kun, "Progressive feature polishing network for salient object detection," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 12128–12135.
[53] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*.
[54] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jan. 2016, pp. 2818–2826.
[55] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 2881–2890.
[56] S. Yang, W. Lin, G. Lin, Q. Jiang, and Z. Liu, "Progressive self-guided loss for salient object detection," *IEEE Trans. Image Process.*, vol. 30, pp. 8426–8438, 2021.
[57] K. Han, Y. Wang, H. Chen, X. Chen, J. Guo, Z. Liu, Y. Tang, A. Xiao, C. Xu, Y. Xu, and Z. Yang, "A survey on vision transformer," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Feb. 18, 2022, doi: 10.1109/TPAMI.2022.3152247.
[58] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 213–229.
[59] H. Wang, Y. Zhu, H. Adam, A. Yuille, and L.-C. Chen, "MaX-DeepLab: End-to-end panoptic segmentation with mask transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2021, pp. 5463–5474.
[60] L. Huang, J. Tan, J. Liu, and J. Yuan, "Hand-Transformer: Non-autoregressive structured modeling for 3D hand pose estimation," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 17–33.
[61] M. Feng, H. Lu, and E. Ding, "Attentive feedback network for boundary-aware salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 1623–1632.

XIAOQI YANG is currently pursuing the B.S. degree in software engineering with the School of Information and Control Engineering, Qingdao University of Technology, Qingdao, China. His current research interests include image processing, deep learning, and computer vision.

LIANGLIANG DUAN received the Ph.D. degree from the School of Information Science and Engineering, Yanshan University, Qinhuangdao, China, in 2016. He is currently a Lecturer with the School of Information and Control Engineering, Qingdao University of Technology, Qingdao, China. His research interests include computer vision and machine learning.