Global Context-Aware Progressive Aggregation Network for Salient Object Detection

Zuyao Chen,[1] **Qianqian Xu,**[2] **Runmin Cong,**[3] **Qingming Huang**[1,2,4,5*]

[1]University of Chinese Academy of Sciences, Beijing, China
[2]Key Lab. of Intelligent Information Processing, ICT, CAS, Beijing, China
[3]Institute of Information Science, Beijing Jiaotong University, Beijing, China
[4]Key Lab. of Big Data Mining and Knowledge Management, CAS, Beijing, China
[5]Peng Cheng Laboratory, Shenzhen, Guangdong, China
chenzuyao17@mails.ucas.ac.cn, xuqianqian@ict.ac.cn, rmcong@bjtu.edu.cn, qmhuang@ucas.ac.cn

Abstract

Deep convolutional neural networks have achieved competitive performance in salient object detection, in which how to learn effective and comprehensive features plays a critical role. Most of the previous works mainly adopted multiple-level feature integration yet ignored the gap between different features. Besides, there also exists a dilution process of high-level features as they passed on the top-down pathway. To remedy these issues, we propose a novel network named *GCPANet* to effectively integrate low-level appearance features, high-level semantic features, and global context features through some progressive context-aware Feature Interweaved Aggregation (FIA) modules and generate the saliency map in a supervised way. Moreover, a Head Attention (HA) module is used to reduce information redundancy and enhance the top layers features by leveraging the spatial and channel-wise attention, and the Self Refinement (SR) module is utilized to further refine and heighten the input features. Furthermore, we design the Global Context Flow (GCF) module to generate the global context information at different stages, which aims to learn the relationship among different salient regions and alleviate the dilution effect of high-level features. Experimental results on six benchmark datasets demonstrate that the proposed approach outperforms the state-of-the-art methods both quantitatively and qualitatively.

Introduction

Salient object detection aims to detect interesting regions that attract human attention in an image (Cong et al. 2018a). As an efficient preprocessing technique, salient object detection benefits a wide range of applications such as image understanding (Zhang, Du, and Zhang 2014), image retrieval (Gao et al. 2015), and object tracking (Hong et al. 2015).

In recent years, the development of deep learning, especially the emergence of Fully Convolutional Network (Long, Shelhamer, and Darrell 2015), has greatly boosted the progress of salient object detection (Zhao et al. 2015; Li and Yu 2016; Wang et al. 2016). Fully Convolutional Network (FCN) stacks multiple convolution layers and pooling
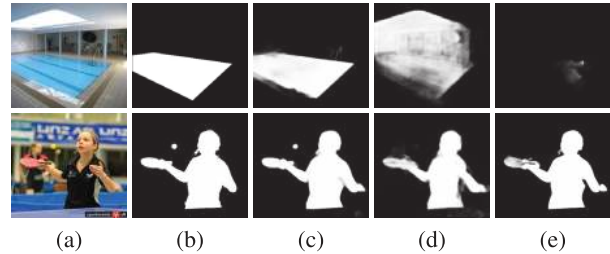
Figure 1: Sample results of our method compared with others. (a) Image; (b) Ground truth; (c) GCPANet (Ours); (d) PiCANet-R (Liu, Han, and Yang 2018); (e) BASNet (Qin et al. 2019).

layers to gradually enlarge the receptive fields of network and extracts high-level semantic information. As pointed out in previous works (Luo et al. 2017; Zhang et al. 2017), due to the pyramid-like CNNs structure, low-level features usually have larger spatial size and more fine-grained details, while high-level features tend to gain more semantic knowledge and discard some meaningless or irrelevant detail information. Generally speaking, the high-level features are beneficial to the coarse localization of salient objects, whereas the low-level features that contain the spatial structural details are suitable to refine boundaries. However, there remains several problems for the FCN-based methods: (1) Due to the gap between different level features, the simple combination of semantic information and appearance information is insufficient and lacks consideration of the different contribution of different features for salient object detection; (2) Most of the previous works ignored the global context information, which benefits for deducing the relationship among multiple salient regions and producing more complete saliency result.

To remedy the above mentioned issues, we propose a novel network named Global Context-Aware Progressive Aggregation Network (*GCPANet*), which consists of four modules: Feature Interweaved Aggregation (FIA) module, Self Refinement (SR) module, Head Attention (HA) module, and Global Context Flow (GCF) module. Considering the characteristics difference between multiple level features,

we design the FIA module to fully integrate the high-level semantic features, low-level detail features, and global context features, which is expected to suppress the noises but recover more structural and detail information. Before the first FIA module, we add a HA module on the top layer of the backbone to strengthen the spatial and channel-wise response on the salient object. After aggregation, features will be fed into a SR module to refine the feature maps via leveraging the inner characteristics within features. Taken into account that the context information can benefit for capturing the relationship among multiple salient objects or different parts of salient object, we design a GCF module to exploit the relationship from global perspective, which is conducive to improving the completeness of salient object detection. Besides, as pointed out in (Liu et al. 2019), the high-level features will be diluted as they passed on the top-down pathway. By introducing GCF, the features containing global semantics are delivered to feature maps at different stages, which alleviates the effect of features dilution. As shown in Fig. 1, the proposed method can handle some challenging scenarios, such as complex scene understanding (the high-luminance ceiling interference), or multiple objects relationship reasoning (the ping-pong bat and ball).

From the above, the contributions of our work can be summarized as follows:

1. A global context-aware progressive aggregation network is proposed to achieve saliency detection, which includes the Feature Interweaved Aggregation (FIA) module, the Self Refinement (SR) module, the Head Attention (HA) module, and the Global Context Flow (GCF) module.

2. The FIA module integrates the low-level detail information, high-level semantic information, and global context information in an interweaved way, where the global context information is produced by the GCF module to capture the relationship among different salient regions and improve the completeness of the generated saliency map.

3. Compared with 12 state-of-the-art methods on six public benchmark datasets, the proposed network *GCPANet* achieves best performance in quantitative and qualitative evaluations.

Related Work

In this section, we will review the related works on deep learning based salient object detection methods, which have achieved remarkable progress on saliency detection thanks to its powerful representation capability.

Inspired by image semantic segmentation, Zhao et al. (Zhao et al. 2015) proposed a fully connected CNN to integrate local and global features to predict the saliency map. Wang et al. (Wang et al. 2016) adopted a recurrent CNN to refine the predicted saliency map step by step. For further enhance the saliency map, several recent works (Hou et al. 2017; Zhang et al. 2017; Deng et al. 2018; Hu et al. 2018; Li et al. 2018; Zhang et al. 2018a; 2018b) integrate features in multiple layers of CNN to exploit the context information at different semantic levels. Among them, Hou et al. (Hou et al. 2017) introduced short connections to the skip-layer structure for capturing fine details. Zhang et al. (Zhang

et al. 2017) concatenated multi-level feature maps based on multiple resolution and introduced a boundary refinement strategy. Deng et al. (Deng et al. 2018) proposed an iterative method to optimize the saliency map, leveraging features generated by deep and shallow layers. Hu et al. (Hu et al. 2018) recurrently concatenated multi-layer features for saliency detection. Li et al. (Li et al. 2018) proposed a contour-to-saliency transferring method that simultaneously predict the contours and saliency maps. Zhang et al. (Zhang et al. 2018a) built a bi-directional message passing model for better integrating multi-level features. Zhang et al. (Zhang et al. 2018b) designed an attention guided network that selectively integrates multi-level contextual information in a progressive manner. Lately, Wu et al. (Wu, Su, and Huang 2019) proposed a cascade partial decoder that utilizes attention mechanism to refine high-level features. Qin et al. (Qin et al. 2019) proposed a boundary-aware model to segment salient object regions and predict the boundaries simultaneously. Liu et al. (Liu et al. 2019) extended the FPN structure equipped with pyramid pooling module to fuse the coarse-level semantic features and fine-level features.

Methodology

In this section, we first outline the proposed network. Then, we elucidate how each component made up and illustrate its effect for saliency detection.

Overview of the Proposed Network

As Fig. 2 shows, the proposed network is a symmetrical encoder-decoder architecture, where the encoder component is based on ResNet-50 to extract the multi-level features, and the decoder component progressively integrates the multi-level comprehensive features to generate the saliency map in a supervised way. Specifically, we first use a HA module to strengthen the spatial regions and feature channels with high response on salient objects, and a SR module to generate the first-stage high-level features through the feature refinement and enhancement. Then, we progressively cascade a FIA module and a SR module in three times to learn more discriminative features and generate more accurate saliency map. In the FIA module, the low-level detail information, high-level semantic information, and global context information are fused in an interweaved way. The SR module successive to each FIA module is to refine the coarse aggregation features. Note that, the global context information is produced by the proposed GCF module, which captures the relationship among different salient regions and constrains more complete saliency prediction. To facilitate the optimization, we combine auxiliary loss branches of each sub-stage with dominant loss.

Feature Interweaved Aggregation Module

As we all know, low-level features include more detail information, such as texture, boundary, and spatial structure, but they also contain more background noises. By contrast, high-level features can provide abstract semantic information, which is beneficial to locate the salient object and suppress the noises. Thus, these two level features are always
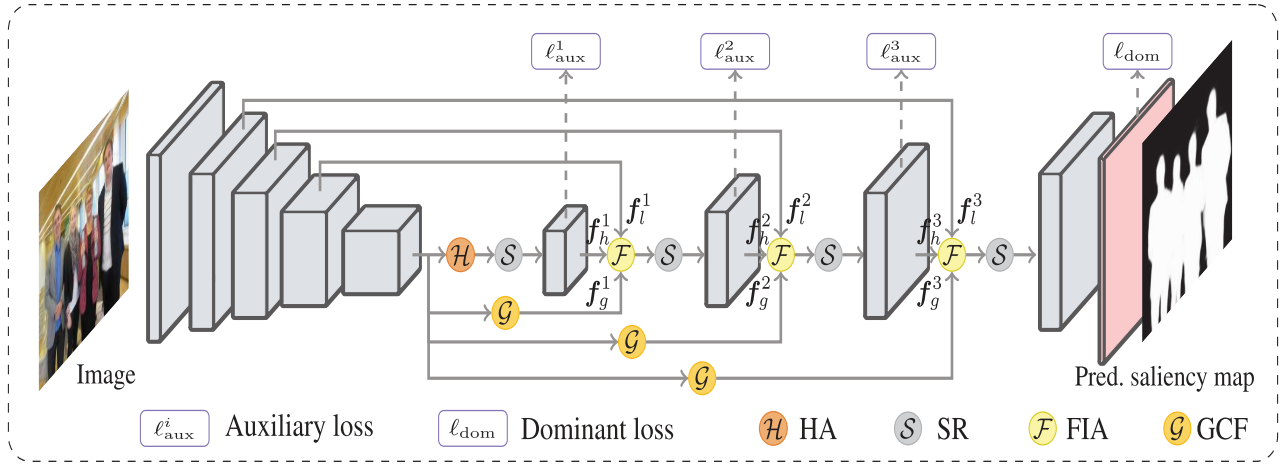
Figure 2: The overall pipeline of the proposed network *GCPANet*. $\boldsymbol{f}_l^t, \boldsymbol{f}_h^t, \boldsymbol{f}_g^t$ ($t = 1, 2, 3$) denote the low-level detail, high-level semantic, and global context features, respectively.

combined together to generate the complementary features. In addition to these two level features, the global context information is very useful to infer the relationship among different salient objects or parts from the global perspective, which is conducive to generate more complete and accurate saliency map. Moreover, using the context features can alleviate the effect of feature dilution. Hence, we develop the FIA module to fully integrate these three level features, which in turn produces a discriminative and comprehensive feature with global perception. Specifically, as shown in Fig. 3, the FIA module receives three parts input, i.e., the high-level features from the output of the previous layer, the low-level features from the corresponding bottom layer, and the global context feature generated by the GCF module. Note that, the production of global context feature will be introduced in the latter subsection.

We first introduce the aggregation strategy for high-level features and low-level features. Different from previous works (Qin et al. 2019; Liu et al. 2019) that often simply fuse the high-level features after up-sampling with the low-level features by concatenation or addition operation, we adopt a more aggressive yet efficient operation, i.e., multiplication. The multiplication operation can strengthen the response of salient objects, meanwhile suppress the background noises. Specifically, for the consistency of multiplication operation, the low-level feature maps $\boldsymbol{f}_l^t$ ($t = 1, 2, 3$) are firstly fed into a 1×1 convolution layer $conv_1$, which compress the features to have the same number of channels as of the high-level features $\boldsymbol{f}_h^t$. Then, a 3×3 convolution layer is applied to high-level features $\boldsymbol{f}_h^t$ to obtain a semantic mask $\boldsymbol{W}_h^t$ after up-sampling. Further, we multiply the mask $\boldsymbol{W}_h^t$ to the compressed low-level features $\tilde{\boldsymbol{f}}_l^t$. Besides, considering high-level features will discard some detail information relevant to salient objects, we apply the above fusion strategy in a mirror way. The mirror path different from the above mentioned is that a detail mask $\boldsymbol{W}_l^t$ is generated by low-level features through a 3×3 convolution layer and then, the mask $\boldsymbol{W}_l^t$ is multiplied to the high-level features $\tilde{\boldsymbol{f}}_h^t$ af-
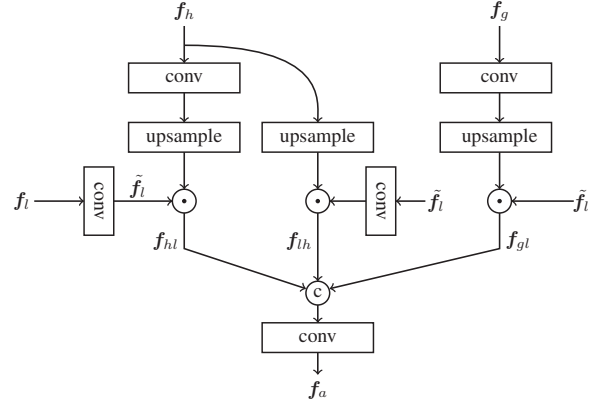

Figure 3: Illustration of the FIA module, where symbol "c" denotes concatenation.

ter up-sampling. The mirror path is supposed to add fine-grained detail information to the predicted saliency maps. The above process can be described as

$$\boldsymbol{W}_h^t = upsample(conv_2(\boldsymbol{f}_h^t)) \qquad (1)$$

$$\boldsymbol{f}_{hl}^t = \delta(\boldsymbol{W}_h^t \odot \tilde{\boldsymbol{f}}_l^t) \qquad (2)$$

$$\boldsymbol{W}_l^t = conv_3(\tilde{\boldsymbol{f}}_l^t) \qquad (3)$$

$$\boldsymbol{f}_{lh}^t = \delta(\boldsymbol{W}_l^t \odot upsample(\boldsymbol{f}_h^t)) \qquad (4)$$

where $\tilde{\boldsymbol{f}}_l^t = conv_1(\boldsymbol{f}_l^t)$ denotes the compressed low-level features, $\odot$ denotes element-wise multiplication, δ denotes the ReLU activation function, $upsample$ is the up-sampling operation via bilinear interpolation, and t is the stage index.

Further, to model the relationship between different parts of salient objects and alleviate the dilution process of high-level features, we introduce the global context features $\boldsymbol{f}_g^t$ at each stage. We employ the global context features $\boldsymbol{f}_g^t$ to generate a context mask $\boldsymbol{W}_g^t$. Then, the mask $\boldsymbol{W}_g^t$ is multiplied
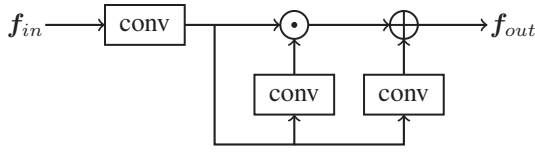
Figure 4: Structure of the SR module.

to the compressed low-level features $\tilde{\boldsymbol{f}}_l^t$.

$$\boldsymbol{W}_g^t = upsample(conv_4(\boldsymbol{f}_g^t)) \tag{5}$$

$$\boldsymbol{f}_{gl}^t = \delta(\boldsymbol{W}_g^t \odot \tilde{\boldsymbol{f}}_l^t) \tag{6}$$

Finally, these three level features are concatenated and then passed through a 3×3 convolution layer to obtain the final fusion features:

$$\boldsymbol{f}_a^t = conv_5(concat(\boldsymbol{f}_{hl}^t, \boldsymbol{f}_{lh}^t, \boldsymbol{f}_{gl}^t)) \tag{7}$$

Each of the above mentioned convolution layers except $conv_2$, $conv_3$, and $conv_4$ is equipped with a batch normalization layer and the ReLU activation function. The output of FIA module is then passed to the SR module.

Self Refinement Module

In FIA module, we combine the complementary characteristics between different level features and obtain the comprehensive feature expression. As a simple and intuitionistic way, one can directly apply a softmax layer after FIA module to obtain the saliency maps, while it still exists some defects. For instance, there are some holes in the predicted salient objects, which are caused by the contradictory response of different layers. Hence, we develop a SR module to further refine and enhance the feature maps after passing the HA module and FIA modules by utilizing the multiplication and addition operation (see Fig. 4). In detail, we firstly apply a 3×3 convolution layer to squeeze the input features $\boldsymbol{f}_{in}$ into feature vector $\tilde{\boldsymbol{f}}$ with the channel dimension of 256, meanwhile remaining useful information. Then, the feature $\tilde{\boldsymbol{f}}$ is fed into two convolution layers to obtain the mask $\boldsymbol{W}$ and bias $\boldsymbol{b}$ for multiplication and addition operation. The main process can be described as

$$\tilde{\boldsymbol{f}} = conv_6(\boldsymbol{f}_{in}) \tag{8}$$

$$\boldsymbol{f}_{out} = \delta(\boldsymbol{W} \odot \tilde{\boldsymbol{f}} + \boldsymbol{b}) \tag{9}$$

where $\boldsymbol{f}_{out}$ is the refined feature maps.

Head Attention Module

Since the top layers features of the encoder component usually are redundant for salient object detection, we design a HA module following the top layer to learn more selective and representative features by leveraging the spatial and channel-wise attention mechanisms.

Specifically, we first apply a convolution layer to the input feature maps $\boldsymbol{F}$ to obtain a compressed feature representation $\tilde{\boldsymbol{F}}$ with 256 channels. Then, we generate a mask $\boldsymbol{W}$

and bias $\boldsymbol{b}$ as similar as the way used in the SR module. The output of the first stage is obtained by

$$\boldsymbol{F}_1 = \delta(\boldsymbol{W} \odot \tilde{\boldsymbol{F}} + \boldsymbol{b}) \tag{10}$$

Further, the input feature $\boldsymbol{F}$ is down-sampled into a channel-wise feature vector $\boldsymbol{f}$ through average pooling, which has strong consistency and invariance. Then, two successive fully connected layers $fc_1(\cdot), fc_2(\cdot)$ are applied to project the feature vector $\boldsymbol{f}$ into an output vector $\boldsymbol{y}$. The final output feature maps $\boldsymbol{F}_{out}$ will be obtained via weighting with vector $\boldsymbol{y}$. The second stage can be described as the following equations,

$$\boldsymbol{y} = \sigma \circ fc_2 \circ \delta \circ fc_1 \circ (\boldsymbol{f}) \tag{11}$$

$$\boldsymbol{F}_{out} = \boldsymbol{F}_1 \odot \boldsymbol{y} \tag{12}$$

where $fc_i(\cdot)$ denotes i-th FC layers, δ denotes the ReLU activation function, σ is the sigmoid operation, and $\circ$ denotes function composition.

Global Context Flow Module

For the challenging scenarios in salient object detection, such as cluttered background, foreground disturbance, and multiple salient objects, simple integration of high-level and low-level features may fail to completely detect the salient regions due to lacking the global semantic relationship among different parts of salient object or multiple salient objects. Besides, since the top-down pathway is built upon the bottom-up backbone, the high-level features will be gradually diluted as they are transmitted to lower layers.

To remedy these issues, we design the GCF module to capture the global context information embedded into the FIA module at each stage. Different from (Liu et al. 2019), we take into account the different contributions at different stages. We firstly employ global average pooling (Lin, Chen, and Yan 2013) to obtain the global contextual information and then reassign different weights to different channels of the global contextual feature maps for each stage. More specifically, for each stage, the process can be described as

$$\boldsymbol{y}^t = \sigma \circ fc_4 \circ \delta \circ fc_3(\boldsymbol{f}_{gap}) \tag{13}$$

$$\tilde{\boldsymbol{f}}^t = conv_{10}(\boldsymbol{f}_{top}) \tag{14}$$

$$\boldsymbol{f}_g^t = \tilde{\boldsymbol{f}}^t \odot \boldsymbol{y}^t \tag{15}$$

where $\boldsymbol{f}_{top}$ refers to the top layer features, and $\boldsymbol{f}_{gap}$ refers to the features generated by the top layer features through global average pooling, which includes global contextual information. Then, the output $\boldsymbol{f}_g^t$ is fed into the FIA module, which has been elaborated in the previous section.

Loss Function

In saliency detection, binary cross-entropy loss is often used as the loss function to measure the relation between the generated saliency map and the ground truth, whcih can be formulated as

$$\ell = -\frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} [G_{ij} \log(S_{ij}) \tag{16}$$
$$+ (1 - G_{ij}) \log(1 - S_{ij})]$$

where H, W denote the height and width of the image, respectively, G_{ij} is the ground truth label of the pixel (i, j), and S_{ij} represents the corresponding probability of being salient objects in position (i, j). To facilitate the optimization of the proposed network, we add auxiliary loss at three decoder stages. Specifically, a 3×3 convolution operation is applied for each stage to squeeze the channel of the output feature maps to 1. Then these maps are up-sampled to the same size as the ground truth via bilinear interpolation and sigmoid function is used to normalize the predicted values into $[0, 1]$. The total loss consists of two parts, i.e., the dominant loss corresponding to the output and the auxiliary loss of each sub-stage.

$$\ell_{\text{total}} = \ell_{\text{dom}} + \sum_{i=1}^{3} \lambda_i \ell_{\text{aux}}^i \tag{17}$$

where λ_i denotes the weight of different loss, and $\ell_{\text{dom}}, \ell_{\text{aux}}^i$ denote the dominant and auxiliary loss, respectively. The auxiliary loss branches only exist during the training stage, whereas they are abandoned when inference.

Experiments

In this section, we first describe the implementation details, introduce the benchmark datasets, evaluation metrics. Then, we conduct experiments on these datasets to evaluate the effectiveness of the proposed method.

Implementation Details   We adopt ResNet-50 (He et al. 2016) pretrained on ImageNet (Deng et al. 2009) as our network backbone. In the training stage, we resize each image to 320×320 with random horizontal flipping, then randomly crop a patch with the size of 288×288 for training. During the inference stage, images are simply resized to 320×320 then fed into the network to obtain prediction without any other post-processing (e.g., CRF). We use Pytorch (Paszke et al. 2017) to implement our model. Mini-batch Stochastic gradient descent (SGD) is used to optimize the whole network with the batch size of 32, the momentum of 0.9, and the weight decay of $5e$-4. We use the warm-up and linear decay strategies with the maximum learning rate $5e$-3 for the backbone and 0.05 for other parts to train our model and stop training after 30 epochs. The inference of a 320×320 image takes about 0.02s (over 50 fps) with the acceleration of one NVIDIA Titan-Xp GPU card. The code is now available. [1]

Datasets   We conduct experiments on six public saliency detection benchmark datasets, and the detailed introduction is provided as follows:

- *ECSSD* (Yan et al. 2013) consists of 1,000 natural images which are manually collected from the Internet;

- *PASCAL-S* (Li et al. 2014) has 850 natural images that are carefully selected from the PASCAL VOC dataset (Everingham et al. 2010);

- *HKU-IS* (Li and Yu 2015) includes 4,447 images and most of them have low contrast or more than one salient object;

---

[1] https://github.com/JosephChenHub/GCPANet.git

- *DUT-OMRON* (Yang et al. 2013) contains 5,168 high quality images. Images of this dataset have one or more salient objects and relatively cluttered background, and thus salient object detection on this dataset is very challenging;

- *SOD* (Movahedi and Elder 2010) is composed of 300 images, many of which contain multiple objects either with low contrast or touching the image boundary;

- *DUTS* (Wang et al. 2017) is currently the largest saliency detection benchmark dataset, which consists of 10,553 training images (*DUTS-TR*) and 5,019 testing images (*DUTS-TE*).

As with other works in salient object detection (Qin et al. 2019; Liu et al. 2019), we employ *DUTS-TR* as our training dataset and evaluate our model on other datasets.

Evaluation Metrics   To quantitatively evaluate the effectiveness of our proposed model, we adopt precision-recall (PR) curves , F-measure (F_β) score and curves, Mean Absolute Error (MAE), and structural similarity measure (S_m) as our performance measures. With different thresholds, pairs of precision and recall value can be computed by comparing the binarized map with the ground truth. Then, we can plot the precision-recall curve (Cong et al. 2019). The second metric F-measure score takes both precision and recall into account, which is defined as $F_\beta = \frac{(1+\beta^2) \cdot Precision \cdot Recall}{\beta^2 \cdot Precision + Recall}$ where β^2 is set to 0.3 to emphasize the precision over recall, as suggested in the previous work (Cong et al. 2018b). Larger F-measure score indicates better performance. For precision-recall pairs, we calculated each corresponding F-measure score and choose the maximum as the evaluation score on the whole dataset. Another metric MAE is defined as the average pixel-wise absolute difference between the prediction map and the ground truth (Cong et al. 2017), i.e., $MAE = \frac{1}{H \times W} \sum_{y=1}^{H} \sum_{x=1}^{W} |S(x, y) - G(x, y)|$ where S denotes the predicted saliency map, G indicates the corresponding ground truth, and H, W are the height and width of the saliency map respectively. The smaller MAE indicates better performance. Since the F_β and MAE are based on pixel-wise errors and ignore the structural similarities, we adopt the structural similarity measure proposed by (Fan et al. 2017) as one of our metrics. The structural similarity measure is defined as $S_m = \alpha * S_o + (1 - \alpha) * S_r$, where α is set to 0.5 to balance the object-aware structural similarity (S_o) and region-aware structural similarity (S_r).

Compared with the State-of-the-arts

We compare the proposed model with 12 state-of-the-art methods, including Amulet (Zhang et al. 2017), C2S (Li et al. 2018), RANet (Chen et al. 2018), PAGR (Zhang et al. 2018b), PiCANet-R (Liu, Han, and Yang 2018), DGRL (Wang et al. 2018), R³Net (Deng et al. 2018), BMPM (Zhang et al. 2018a), RADF (Hu et al. 2018), CPD-R (Wu, Su, and Huang 2019), BASNet (Qin et al. 2019) and PoolNet (Liu et al. 2019). For fair comparison, the saliency maps of different methods are provided by authors or obtained by running their released codes under the default parameters.

Table 1: Performance comparison with 12 state-of-the-art methods on 6 benchmark datasets. The best results on each dataset are highlighted in **boldface**.

| Methods	ECSSD			HKU-IS			PASCAL-S			DUT-OMRON			DUTS-TE			SOD		
	$F_\beta\uparrow$	$S_m\uparrow$	MAE↓	$F_\beta\uparrow$	$S_m\uparrow$	MAE↓	$F_\beta\uparrow$	$S_m\uparrow$	MAE↓	$F_\beta\uparrow$	$S_m\uparrow$	MAE↓	$F_\beta\uparrow$	$S_m\uparrow$	MAE↓	$F_\beta\uparrow$	$S_m\uparrow$	MAE↓
Amulet (Zhang et al. 2017)	0.915	0.894	0.059	0.894	0.882	0.053	0.832	0.815	0.097	0.744	0.781	0.097	0.779	0.803	0.085	0.803	0.754	0.139
C2SNet (Li et al. 2018)	0.911	0.896	0.053	0.898	0.888	0.046	0.848	0.834	0.080	0.759	0.799	0.072	0.811	0.831	0.062	0.819	0.757	0.121
RADF (Hu et al. 2018)	0.912	0.888	0.064	0.903	0.882	0.052	0.832	0.804	0.104	0.785	0.811	0.071	0.811	0.820	0.073	0.822	0.756	0.136
RANet (Chen et al. 2018)	0.920	0.894	0.055	0.912	0.888	0.045	0.830	0.792	0.102	0.785	0.812	0.063	0.831	0.839	0.060	0.847	0.761	0.122
DGRL (Wang et al. 2018)	0.925	0.906	0.043	0.913	0.896	0.037	0.853	0.834	0.074	0.779	0.810	0.063	0.834	0.846	0.051	0.844	0.770	0.104
PAGR (Zhang et al. 2018b)	0.926	0.889	0.061	0.918	0.887	0.048	0.851	0.813	0.092	0.771	0.775	0.071	0.854	0.838	0.056	0.836	0.714	0.145
R^{3}Net (Deng et al. 2018)	0.929	0.910	0.051	0.910	0.894	0.047	0.837	0.809	0.101	0.793	0.819	0.073	0.829	0.837	0.067	0.837	0.765	0.129
BMPM (Zhang et al. 2018a)	0.929	0.911	0.044	0.920	0.906	0.039	0.857	0.840	0.073	0.775	0.809	0.063	0.852	0.862	0.049	0.852	0.784	0.105
PiCANet-R (Liu, Han, and Yang 2018)	0.935	0.917	0.046	0.919	0.904	0.043	0.863	0.849	0.075	0.803	0.832	0.065	0.860	0.869	0.051	0.853	0.787	0.102
CPD-R (Wu, Su, and Huang 2019)	0.939	0.918	0.037	0.925	0.906	0.034	0.864	0.842	0.072	0.797	0.825	0.056	0.865	0.869	0.043	0.857	0.765	0.110
BASNet (Qin et al. 2019)	0.943	0.916	0.037	0.928	0.909	0.032	0.857	0.832	0.076	0.805	0.836	0.057	0.859	0.866	0.048	0.849	0.766	0.112
PoolNet (Liu et al. 2019)	0.944	0.921	0.039	0.933	0.917	0.032	0.869	0.845	0.074	0.808	0.836	0.056	0.880	0.883	0.040	0.867	0.795	0.100
Ours	**0.949**	**0.927**	**0.035**	**0.938**	**0.920**	**0.031**	**0.876**	**0.861**	**0.061**	**0.812**	**0.839**	**0.056**	**0.888**	**0.891**	**0.038**	**0.872**	**0.802**	**0.087**
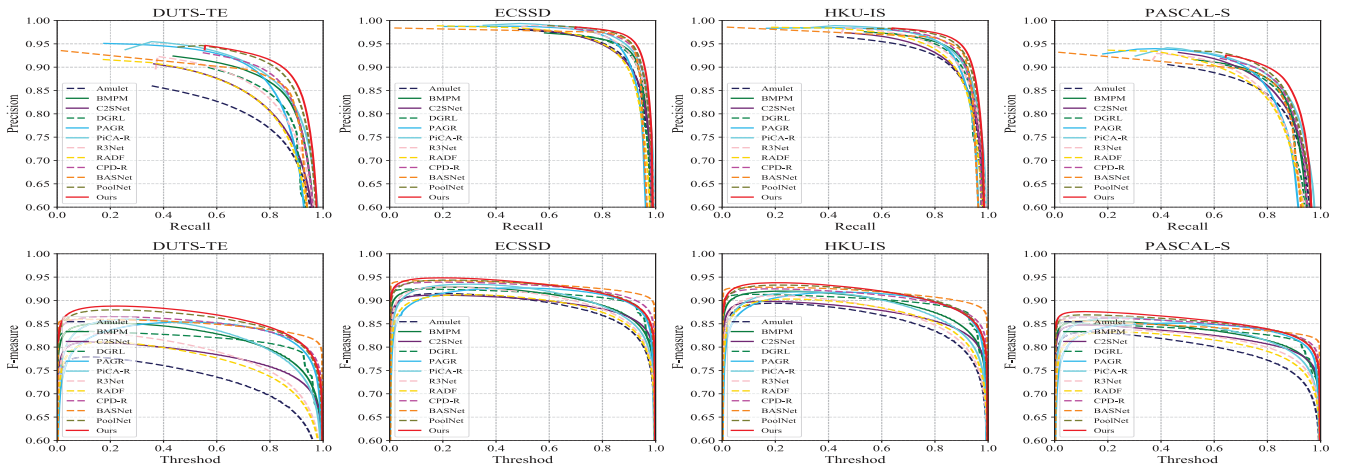
Figure 5: Illustration of PR curves (the first row), F-measure curves (the second row) on four datasets.

Quantitative Evaluation   Table 1 shows the quantitative comparison results in terms of F-measure, S-measure, and MAE score. It's obvious that the proposed method achieves the best performance in terms of different measures, which demonstrates the effectiveness of the proposed model. In addition, as shown in Fig. 5, the PR curves and F-measure curves by our approach (the red curves) are outstanding in most cases compared with other previous methods under different thresholds, which is consistent with the measures reported in Table 1.

Qualitative Evaluation   To further illustrate the advantages of the proposed method, we provide some visual examples of different methods. As Fig. 6 shows, our proposed method can handle various challenging scenarios, including fine-grained structures, cluttered background, foreground disturbance, objects concurrency, and multiple salient objects, etc. Compared with other previous methods, the saliency maps generated by our approach are more complete and accurate. Note that our approach is more robust to background/foreground disturbance (the second/third row) and can capture the relationship among multiple objects (the fifth

row), which illustrates the power of the feature interweaved aggregation strategy and the introducing of global context information.

Ablation Study

In this part, we conduct the ablation study to verify the effectiveness of each key components designed in the proposed model. The ablation experiments are conducted on the ECSSD dataset and ResNet-50 is adopted as the backbone. As shown in Table 3, the proposed model containing all components (i.e., FIA, SR, HA and GCF) achieves the best performance, which demonstrates the necessity of each component for the proposed model to obtain the best saliency detection results.

We adopt the model like U-Net (Ronneberger, Fischer, and Brox 2015) that only concatenates high-level features after up-sampling and low-level features as the baseline model, then add each module progressively. From Table 3, the FIA module largely improves the baseline from 0.0456 to 0.0390 in terms of MAE. Furthermore, the MAE score is improved by 14% compared with the basic model after

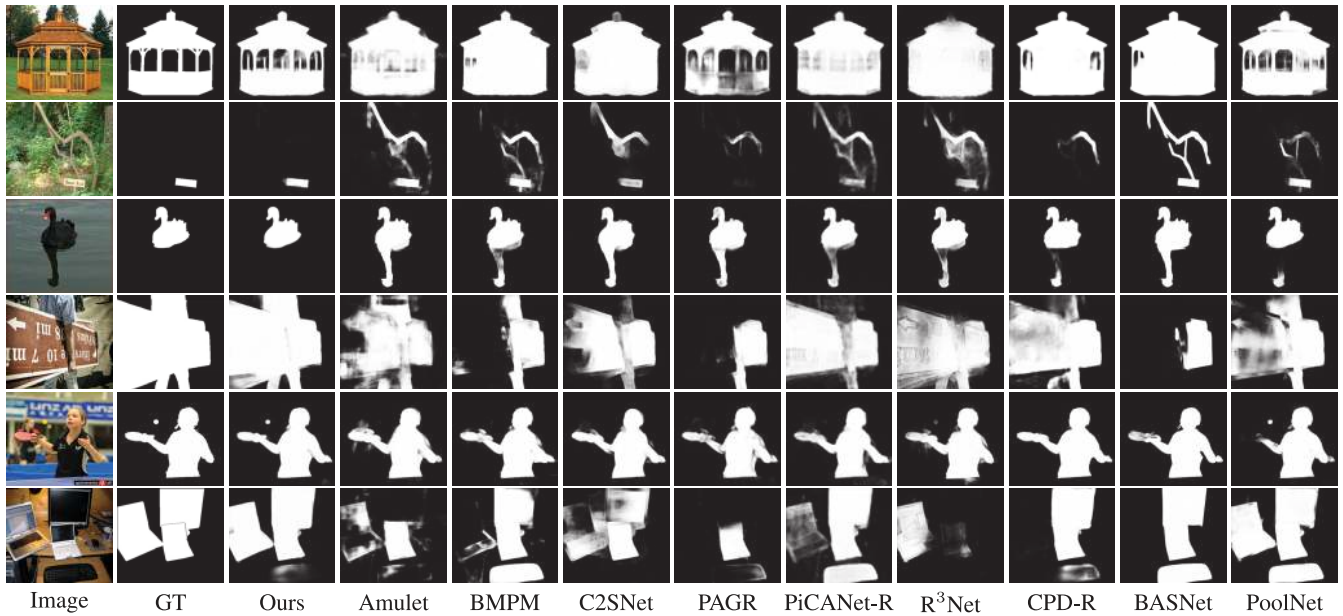| Image | GT | Ours | Amulet | BMPM | C2SNet | PAGR | PiCANet-R | R^3Net | CPD-R | BASNet | PoolNet |

Figure 6: Qualitative comparison of the proposed model with other state-of-the-art methods. Obviously, saliency maps generated by our approach are more accurate and much close to the ground truth in various challenging scenarios.

Table 2: MAE Comparison of the GCF with the shared one.

	ECSSD	HKU-IS	PASCAL-S	DUT-OMRON	DUTS-TE	SOD
with the Shared	0.0361	0.0313	0.0628	0.0590	0.0388	0.0915
with **GCF**	**0.0348**	**0.0309**	**0.0614**	**0.0563**	**0.0380**	**0.0874**

Table 3: Ablation study with different components combinations on ECSSD dataset.

Baseline	FIA	SR	HA	GCF	$MAE \downarrow$
✓					0.0456
✓	✓				0.0390
✓	✓	✓			0.0365
✓	✓	✓	✓		0.0364
✓	✓	✓	✓	✓	**0.0348**

adding the SR module. The combination of FIA and SR has already achieved well performance, while the addition of HA has a slight enhancement. Finally, we add the GCF to the model and obtain the best result.

Moreover, we evaluate the effectiveness of the GCF module compared to another setting, in which the global context features are shared at all stages. From Table 2, the proposed GCF module outperforms the shared one. The potential reason behind this phenomenon is that the parallel scheme of the GCF modules can provide distinct features for different stages, which benefits to learn the comprehensive and discriminative features for salient objects.

Conclusion

In this paper, we propose a Global Context-Aware Progressive Aggregation Network (GCPANet) to achieve salient object detection. Considering different characteristics of different level features, we design a simple yet effective aggregation module to fully integrate different level features. We introduce global context information at different stages to capture the relationship among multiple salient objects or multiple regions of salient object and alleviate the dilution effect of features. Experimental results on six benchmark datasets demonstrate that the proposed network outperforms other 12 state-of-the-art methods under different evaluation metrics.

Acknowledgement

References

Chen, S.; Tan, X.; Wang, B.; and Hu, X. 2018. Reverse attention for salient object detection. In *ECCV*, 234–250.

Cong, R.; Lei, J.; Fu, H.; Lin, W.; Huang, Q.; Cao, X.; and Hou, C. 2017. An iterative co-saliency framework for rgbd images. *IEEE TC* 49(1):233–246.

Cong, R.; Lei, J.; Fu, H.; Cheng, M.-M.; Lin, W.; and Huang, Q. 2018a. Review of visual saliency detection with comprehensive information. *IEEE TCSVT*.

Cong, R.; Lei, J.; Fu, H.; Huang, Q.; Cao, X.; and Ling, N. 2018b. Hscs: Hierarchical sparsity based co-saliency detection for rgbd images. *IEEE TMM*.

Cong, R.; Lei, J.; Fu, H.; Porikli, F.; Huang, Q.; and Hou, C. 2019. Video saliency detection via sparsity-based reconstruction and propagation. *IEEE TIP*.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *CVPR*, 248–255.

Deng, Z.; Hu, X.; Zhu, L.; Xu, X.; Qin, J.; Han, G.; and Heng, P.-A. 2018. R^3Net: Recurrent residual refinement network for saliency detection. In *IJCAI*, 684–690.

Everingham, M.; Van Gool, L.; Williams, C. K.; Winn, J.; and Zisserman, A. 2010. The pascal visual object classes (voc) challenge. *IJCV* 88(2):303–338.

Fan, D.-P.; Cheng, M.-M.; Liu, Y.; Li, T.; and Borji, A. 2017. Structure-measure: A new way to evaluate foreground maps. In *ICCV*, 4548–4557.

Gao, Y.; Shi, M.; Tao, D.; and Xu, C. 2015. Database saliency for fast image retrieval. *IEEE TMM* 17(3):359–369.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*, 770–778.

Hong, S.; You, T.; Kwak, S.; and Han, B. 2015. Online tracking by learning discriminative saliency map with convolutional neural network. In *ICML*, 597–606.

Hou, Q.; Cheng, M.-M.; Hu, X.; Borji, A.; Tu, Z.; and Torr, P. H. 2017. Deeply supervised salient object detection with short connections. In *CVPR*, 3203–3212.

Hu, X.; Zhu, L.; Qin, J.; Fu, C.-W.; and Heng, P.-A. 2018. Recurrently aggregating deep features for salient object detection. In *AAAI*, 6943–6950.

Li, G., and Yu, Y. 2015. Visual saliency based on multiscale deep features. In *CVPR*, 5455–5463.

Li, G., and Yu, Y. 2016. Deep contrast learning for salient object detection. In *CVPR*, 478–487.

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

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

Lin, M.; Chen, Q.; and Yan, S. 2013. Network in network. *arXiv preprint arXiv:1312.4400*.

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 *CVPR*, 3917–3926.

Liu, N.; Han, J.; and Yang, M.-H. 2018. PiCANet: Learning pixel-wise contextual attention for saliency detection. In *CVPR*, 3089–3098.

Long, J.; Shelhamer, E.; and Darrell, T. 2015. Fully convolutional networks for semantic segmentation. In *CVPR*, 3431–3440.

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

Movahedi, V., and Elder, J. H. 2010. Design and perceptual validation of performance measures for salient object segmentation. In *CVPRW*, 49–56.

Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; and Lerer, A. 2017. Automatic differentiation in pytorch.

Qin, X.; Zhang, Z.; Huang, C.; Gao, C.; Dehghan, M.; and Jagersand, M. 2019. BASNet: Boundary-aware salient object detection. In *CVPR*, 7479–7489.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 234–241.

Wang, L.; Wang, L.; Lu, H.; Zhang, P.; and Ruan, X. 2016. Saliency detection with recurrent fully convolutional networks. In *ECCV*, 825–841.

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 *CVPR*, 136–145.

Wang, T.; Zhang, L.; Wang, S.; Lu, H.; Yang, G.; Ruan, X.; and Borji, A. 2018. Detect globally, refine locally: A novel approach to saliency detection. In *CVPR*, 3127–3135.

Wu, Z.; Su, L.; and Huang, Q. 2019. Cascaded partial decoder for fast and accurate salient object detection. In *CVPR*, 3907–3916.

Yan, Q.; Xu, L.; Shi, J.; and Jia, J. 2013. Hierarchical saliency detection. In *CVPR*, 1155–1162.

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

Zhang, P.; Wang, D.; Lu, H.; Wang, H.; and Ruan, X. 2017. Amulet: Aggregating multi-level convolutional features for salient object detection. In *ICCV*, 202–211.

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

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

Zhang, F.; Du, B.; and Zhang, L. 2014. Saliency-guided unsupervised feature learning for scene classification. *IEEE TGRS* 53(4):2175–2184.

Zhao, R.; Ouyang, W.; Li, H.; and Wang, X. 2015. Saliency detection by multi-context deep learning. In *CVPR*, 1265–1274.