Selective focus saliency model driven by object class-awareness

Danpei Zhao\textsuperscript{1,2,3} | Bo Yuan\textsuperscript{1,2,3} | Zhenwei Shi\textsuperscript{1,2,3} | Zhiguo Jiang\textsuperscript{1,2,3}

\textsuperscript{1} Image Processing Center, School of Astronautics, Beihang University, Beijing, China
\textsuperscript{2} Beijing Key Laboratory of Digital Media, Beihang University, Beijing, China
\textsuperscript{3} Key Laboratory of Spacecraft Design Optimization and Dynamic Simulation Technologies, Ministry of Education, Beihang University, Beijing, China

Abstract

Current many salient object detection (SOD) models only focus on highlighting visual conspicuous region but fail to make saliency detection for specific targets. In this paper, a selective focus saliency model driven by object class-awareness (SF-OCA) to run saliency detection is proposed. The framework consists of a visual saliency detection flow, a segmentation-classification flow, and a class-awareness selection module. It combines bottom-up visual perception with a top-down task-driven manner, which is capable of detecting specific category salient targets and eliminating the interference from other saliency areas, providing a new idea for saliency detection. Experimental results show that the method achieves comparable performance with state-of-the-art models on four public saliency datasets. In addition, a new dataset was also built to test the proposed framework for the selective focus saliency detection. Compared with other SOD methods, the method not only highlights visual saliency regions but can choose more important or more noteworthy targets in a class-awareness manner. The method also shows better robustness under a variety of conditions including multi-targets, small targets and complex background.

1 | INTRODUCTION

How to pick out the interested regions in a complex visual scene and ignore uninterested areas? Given multiple foreground objects, we human often choose to focus on the interested targets and overlook uninterested ones based on our knowledge or motivation. Figure 1 shows an example that simultaneously contains two foreground objects zebra and elephant. On account of different motivations like 'We need to find out the interested foreground salient targets (e.g. zebra or elephant in Figure 1) in the image!', the detection model should confine the saliency areas on the interested target (e.g. zebra) and ignore unrelated one (e.g. elephant). In this paper we propose to add subjective category selection capabilities to a saliency detection framework from the perspective of visual saliency perception. We aim not only to perceive a saliency area from a visual point of view, but also to add a subjective willingness to make a choice among multiple salient objects.

Salient object detection aims to imitate human visual system (HVS) to localise and segment the most conspicuous regions, acting as an important pre-processing step in computer vision field such as visual tracking [1, 2], video compression [3, 4] and image segmentation [5, 6]. Generally, SOD methods can be divided into two classes: methods based on bottom-features and those based on deep-features. Due to the rapid development of convolutional neural network (CNN), the methods based on CNNs like [7–13] already made remarkable advancement on visual saliency detection. However, there are still challenges like: (1) unable to randomly to choose interested categories salient targets; (2) most saliency detection methods suffer from heterogeneous salient target interior and boundary detail loss. To address these issues, in this paper we present a framework named SF-OCA that unites bottom-up visual saliency with a top-down task-driven manner to run selective focus saliency detection.
Related Work

The proposed method executes multiple tasks simultaneously. We design a universal saliency detection framework, of which interested salient targets and eliminating unrelated ones. Objects coexisting, the proposed method is able to focus on the saliency detection. Under circumstance of multiple salient segmentation and classification. Furthermore, the designed SCNet shares mutual conv layers with the VSNet but run object feature map with the same size with the input. The proposed saliency map by extracting conv features and upsamping the follows a simplest encoder-decoder manner, generates visual level features to train a valid CNN based saliency detection of deep CNNs like VGG [17] and ResNet [18], extracting high-resolution module (CASM) that utilizes the fusion of multiple features (SCNet). Additionally, we also design a class-awareness selection (VSNet) as well as object segmentation and classification (SCNet). Additionally, we also design a class-awareness selection module (CASM) that utilizes the fusion of multiple features to output class-aware saliency map. Benefiting by advancement of deep CNNs like VGG [17] and ResNet [18], extracting high-level features to train a valid CNN based saliency detection model is popular and practical. The proposed VSNet, which follows a simplest encoder-decoder manner, generates visual saliency map by extracting conv features and upsampling the feature map with the same size with the input. The proposed SCNet shares mutual conv layers with the VSNet but run object segmentation and classification. Furthermore, the designed CASM unites the VSNet and SCNet to the final selective focus saliency detection. Under circumstance of multiple salient objects coexisting, the proposed method is able to focus on the interested salient targets and eliminating unrelated ones.

Our main contributions are summarised as follows:

- We propose a selective focus saliency model, which combines bottom-up visual perception with a top-down task-driven manner to selectively highlight specific saliency areas in accordance with detection task. The proposed method utilizes multiple cues including visual saliency, region segmentation and classification, which jumps out of the limitation that using visual information only.

- We design a universal saliency detection framework, of which the choice of CNN baseline is not crucified. The layout of the proposed framework comprehensively takes account of saliency detection, task-driven and boundary maintenance, which is able to not only highlight class-awareness salient target but also retain homogeneous interior and distinct boundary.

- The proposed method executes multiple task simultaneously. Saliency detection and a segmentation module are already designed together in the proposed framework, so that our method can run selective focus saliency detection with no extra post-processing.

Extensive experiments on four universal saliency datasets, including ECSSD [19], DUTS-TE [20], DUT-OMRON [21], and HKU-IS [22], validate the effectiveness of the proposed framework. To run selective focus detection, we built an image dataset named MCSD to test our SF-OCA. The proposed dataset contains saliency binary masks, object segmentation labels and corresponding category labels simultaneously, and the details are introduced in Section 4.3.

2 | RELATED WORK

Due to the strong generalisation capacity, CNN based approaches are popular in achieving high performance saliency detection. However, most methods focus on pure visual feature [10, 22], leading to ambiguous boundary and inhomogeneous salient object interior. On the other hand, exploring which target is more significative with respect to the task is a meaningful and unexploited in the saliency detection research. We introduce the related work in the following two topics.

2.1 | Salient object detection

In former years, traditional SOD methods like [23, 24] utilise image low-level features such as color, intensity and shape to detect saliency area, whereas these methods generally unable to adapt to complex scenes. With the rapid development of deep learning techniques, SOD models based on deep neural networks have substantially improved detection performance. For instance, SOD methods built on CNNs have been explored in [10, 15, 16, 25–31]. These methods developed various networks using CNN as baseline and achieved satisfied advancement. According to the architectures, these methods are designed with a single-stream network [8, 32–36] or a multi-stream network [37–40]. The former is a standard architecture consisting of a sequential cascade of convolution layers, pooling layers and non-linear activation operations. While the latter has multiple network streams for explicitly learning multiscale saliency features from inputs of varied resolutions or with different structures. Some multi-stream networks handle different tasks at separate pathways. The outputs from different streams are combined to form final prediction [41]. In this paper, we propose to run visual saliency detection using a multi-stream architecture. However, many CNN based methods like [22, 25, 42, 43] are generally easy to be influenced by background and lose target boundary details. To address these problems, Li, et al [29] proposed to combine saliency detection and semantic segmentation by sharing common convolutional layers, meanwhile the segmentation results are used to enhance the saliency map. For achieving accurate saliency region boundary, Qin, et al [27] added salient edge information to saliency network to refine visual saliency map. While Feng, et al [15] used multi-scale convolutional features including boundary-aware information and [44] fused multiple scale outputs from the network to generate accurate boundary. Wu, et al. [16] designed an intertwined multi-supervision form to merge saliency detection, edge...
FIGURE 3 Overview of the proposed SF-OCA. The framework comprises VSNet, SCNet and CASM. Of which the VSNet follows an encoder-decoder manner, and shares mutual conv layers, that is, Res-1 to Res-5, with SCNet. VSNet and SCNet output visual saliency map, object mask and corresponding category, respectively. And the CASM realise selective saliency detection task in a Concatenation (C) and Refinement (R) manner.

2.2 Task-driven saliency detection

Most SOD methods are object-level methods, that is, designed to detect pixels that belong to the salient objects without being aware of the individual instances [41]. Based on our motivation, we aim to realise saliency detection with being aware of the individual instances. Recently task-driven saliency detection researches reveal human subjective focus area, such as eye fixation models [48–50], which aim to predict the locations of human eye-fixations. Cao, et al [51] presented a feedback CNN to uncover the circulate mechanism of deep neural networks and to capture visual attention on salient objects. From another point of view, distributing finite attention resources to interest regions is also a way to run task-driven saliency detection. Zhang, et al [14] proposed excitation backprop scheme to pass along top-down signals downwards in the network hierarchy. By weighing connection between high-level and low-level neurons, neural attention in CNNs can be focused on specific targets. While Fernando, et al [52] proposed a task-driven visual saliency method by learning contextual semantic information and relationship among different tasks together in a generative adversarial manner. However, the methods above can localise specific salient target but generally with ambiguous target boundary. Some recent works start to rethink the relation among multiple salient objects. Wang, et al [49] infers object saliency using the fixation prior, which imitates human visual attention mechanisms and allows the suggested model to explicitly segment out the most visually important objects in an interpretable manner. Yo Umeki, et al [53] suppress inaccurate saliencies and estimate importance degree for multi-object images.

From above mentioned approaches, we proposed a method that simultaneously localise class-awareness salient target meanwhile preserve homogeneous saliency area as well as clear target boundary, as described in Section 3.

3 PROPOSED FRAMEWORK

3.1 Architecture overview

In this paper, we propose a framework named SF-OCA. As seen in Figure 3, the SF-OCA mainly consists of three modules, that is, VSNet, SCNet and CASM. The proposed framework is capable of detecting class-aware salient targets with homogeneous interior and distinct boundary. The following subsections start from the visual saliency detection network in Section 3.2, then the segmentation and classification network in Section 3.3, and the class-awareness selection module in Section 3.4.
3.2 Visual saliency detection network

The essential concept of the proposed SF-OCA is to overall utilise visual saliency detection, object segmentation and classification to selective focus saliency detection. To underline the significance of the object segmentation and classification module on promoting visual saliency performance, we design the VSNet following a simplest encoder-decoder form. Each encoder in the encoder part performs convolution with a filter bank to produce a set of feature maps, which are then batch normalised and then an ReLU ($max(0,x)$) is applied. Following that, max-pooling with a 2×2 window and stride 2 is performed and the resulting output is sub-sampled by a factor of 2. In the encoder part, we do not pay much attention to boundary information, which we will resolve later in our SCNet. While each decoder in the decoder network upsamples its input feature map using the memorised max-pooling indices from the corresponding encoder feature map. It goes without saying that a more complicated visual saliency detection network could improve the performance, however, it is not a major concern in this paper. The purpose of the VSNet is to locate the visual saliency area, which is donated as $S$, and lay the basis for the following segmentation and optimisation. The encoder part can be various feedforward convolutional networks like ResNet [18] based or VGG [17] based saliency detection networks. Since the state-of-the-art performance, we take ResNet-50 [18] as an example to extract deep saliency features. In detail, the encoder part includes five convolution blocks as being introduced in [18] and [54], which are Res-1 to Res-5. The pooling layers are placed for parameters reduction. Correspondingly, in the decoder part we apply upsampling with stride 2 and deconv layers to output visual saliency map, which has the same size with the input. And we use ReLU in the hidden layers. For training, we employ ImageNet pretrained model and use a saliency dataset to train VSNet The loss for VSNet is the binary cross entropy [55]:

$$L_{VS} = l_{bce}(S, G) = -\sum_{(r,c)} \left[ G(r,c) \log(S(r,c)) + (1 - G(r,c)) \log(1 - S(r,c)) \right],$$

(1)

where $G(r,c) \in \{0,1\}$ represents the ground truth label of the pixel $(r,c)$ and $S(r,c)$ is the predicted probability of being salient object.

3.3 Segmentation and classification network

The VSNet design is able to output all visual saliency targets but cannot achieve class-aware saliency detection and suffers from boundary challenge. In this case, we propose the SCNet that shares mutual cony layers with the VSNet to help gaining saliency region with distinct boundary and homogeneous interior. The SCNet consists of two branches, which are capable of simultaneously running object segmentation and classification. For statement convenience, we donate target segmentation result and category label as $M$ and $L$, respectively. In addition, a Canny [56] edge detector is introduced to extract the contour from object segmentation result, and the contour output is donated as $C$. In detail, after Res-5, the feature maps are feeded into region proposal network (RPN) [57]. RPN is a way to generate candidate region of interest (RoI). We donate $P$ for original proposal, $\hat{P}$ for predicted ground truth and $G$ for labeled round truth. For each proposal, we use RoIAlign [54] to extract a small feature map and align the extraction features with the input. RoIAlign uses bilinear interpolation to extract feature map from each RoI, and the feature map has floating point form coordinates after pooling. Next we present a two-branch design to obtain the object segmentation results and category labels, respectively. On the one hand, as seen in Figure 4, we operate one upsampling on RPN’s output by 2×2 with stride 2 to enlarge the scale of the feature map, which help promoting segmentation resolution. After upsampling, the map was imported into a fully convolutional network to gain segmentation map, where the edge extractor was employed to extract foreground contour. On the other hand, we employ fully connected layers on RPN’s output and then use softmax function for classification to acquire the object category labels. Notice that ReLU is used in the hidden layers.

In our implementation, for each testing image, SCNet generates $N$ proposals and runs box prediction on each proposal but preserve the highest scoring $N_0$ proposals, which are used to run object segmentation. Particularly, for each RoI, the segmentation flow predicts $K$ masks (where $K$ represents object class number), we follow the instructions in [54] and only preserve the $k$-th mask for class $k$ that predicted by classification branch.

For training the SCNet, we follow the suggestions in [54]. The loss for SCNet includes three parts, which are $L_B$ for bounding box, $L_C$ for classification and $L_M$ for mask, respectively. Specifically, due to the need of generating mask for each object, $L_M$ adopts an average binary cross-entropy loss. For an input image, assuming it contains $K$ categories, the mask branch generate $K \times h \times h$ binary masks. For an RoI, assuming its ground truth is $k$-th, only the $k$-th mask contributes to its $L_M$. $L_C$ is a cross entropy loss and $L_B$ is a regression loss following the suggestion in [58].

$$L_C(p, p_i^*) = -\log[p_i^*] + (1 - p_i^*) (1 - p_i),$$

(2)

$$L_B(t, t_i^*) = \text{smooth}_1 \left( t - t_i^* \right),$$

(3)

where ground truth label $p_i^*$ is 1 if the anchor is positive otherwise 0. $p_i$ is the predicted value. $t_i$ is a coordinate vector of the predicted bounding box, and $t_i^*$ is that of the ground truth box.
associated with a positive anchor. The full objective loss on each sampled RoI is:

\[ \mathcal{L}_{SC} = \mathcal{L}_C + \mathcal{L}_B + \mathcal{L}_M. \]

### 3.4 Class-awareness selection module

As introduced in Sections 3.2 and 3.3, we gain visual saliency map \( S \), object masks \( M_i \), and contours \( C_i \), as well as object category labels \( L_i, i \in \{1, 2, ..., K\} \). As shown in Figure 5, we design a Class-Awareness Selection Module to generate class-aware saliency map. Firstly, on the basis of detection task, we use \( L \) as instruction to localise the corresponding targets. Particularly, we take target category label \( L_T \) as the criteria to pick out the target \( M_T \). Assuming an object instance belongs to an RoI, \( L_T \) is defined as:

\[
L_T = \begin{cases} 
L_i & \text{if } T = i, i \in \{1, 2, ..., K\}, \\
None & \text{otherwise}
\end{cases}
\]

Thus we obtain target category label for the class-aware saliency map. We use \( L_T \) to sort out the corresponding \( M_T \), which ought to be a binary mask map. And \( C_T \) is gained by using Canny [56] detector on \( M_T \). Moreover, we apply \( C_T \) on \( S \) to pick out the task-relevant saliency region, which is donated as \( S_T \).

\[
M_T = \begin{cases} 
M_i & \text{if } L_T = L_i, i \in \{1, 2, ..., K\}, \\
None & \text{otherwise}
\end{cases}
\]

\[
S_T = S \cap A_{C_T},
\]

where \( A_{C_T} \) represents the non-zero area in \( M_T \) limited by \( C_T \). And \( \cap \) indicates region intersection. In other words, we only preserve saliency area in \( S \) that also belongs to mask region in \( M_T \) limited by \( C_T \). Pixel values in \( S_T \) are the same to the values those defined in \( S \). Ultimately, \( M_T \) and \( S_T \) are merged by weight factors.

\[
S_{\text{final}} = \lambda_1 \cdot S_T + \lambda_2 \cdot M_T,
\]

where \( S_T \) and mask \( M_T \) have the same size \([M,N]\). \( \lambda_1 \) and \( \lambda_2 \) are weight factors of \( S_T \) and \( M_T \), respectively. And \( S_{\text{final}} \) is the class-aware saliency map in accordance with the detection task. Comparing with \( S \), \( S_{\text{final}} \) achieves more homogeneous interior and distinct boundary.

Here, it is significant to address the discrepancy between object segmentation and saliency detection. The former aims to classify every single pixel to compose multiple sets, and each set represents an object. Nevertheless, the latter outputs saliency map, whose pixel value varies from 0 to 255 (e.g. 8-bit map) and higher value indicates higher saliency. In the proposed method, we use object segmentation work to enhance visual saliency detection so that the final detection is more approximate to the ground truth.

### 4 EXPERIMENTS

#### 4.1 Training strategy

As introduced in Section 3, the proposed SF-OCA runs class-aware saliency detection. We train our framework as follows.

**Step 1:** For visual saliency detection, we import a saliency detection dataset to train the VSNet. The dataset should contain images and corresponding binary saliency masks.

**Step 2:** For class-aware SOD implementation, we proceed to train the SCNet with the supervision of \( \mathcal{L}_B \), \( \mathcal{L}_M \) and \( \mathcal{L}_C \), as suggested in Section 3.3. We import MS-COCO [59] to train the SCNet.

Implementation details are introduced in Section 4.2.

#### 4.2 Implementation details

##### 4.2.1 Parameter settings

The VSNet and SCNet share common CNN baseline and we use ResNet-50 [18] as the backbone. \( \lambda_1 \) and \( \lambda_2 \) are set as 0.3 and 0.7, respectively. And in our implementation, \( N = 300, N_0 = 100 \). On the one hand, we train the VSNet on DUTS-TR [20] dataset for visual saliency detection. In this stage, we set batch size as 16, learning rate as 0.01 and epoch as 30. On the other hand, we train the SCNet by setting batch size as 1, epoch as 160 and learning rate as 0.01 which is decreased by 10 after the 120 epoch. Due to the training strategy, the proposed framework can be trained within 11G GPU memory in 8 GPU days.

##### 4.2.2 Running environment

We implement our method using Python with the PyTorch tool-box. The train and test environment is a 3.2GHz CPU, 32GB RAM and GTX 1080Ti GPU.

#### 4.3 Datasets and evaluation metrics

We carry out our experiments on four public saliency detection datasets, which are ECSSD [19], DUTS-TE [20], DUT-OMRON [21] and HKU-IS [22]. Of which, ECSSD contains
1000 image pairs, DUTS-TE includes 5019 test images and ground truths. DUT-OMRON comprises 5168 test samples and HKU-IS consists of 4447 image pairs. The four datasets are widely used in visual saliency detection. We train our VSNet on DUTS-TR [20] set, which contains 10,553 image pairs. Meanwhile, we train the SCNet on MS-COCO [59] as introduced in Section 4.2. In addition, we built a new dataset named MCSD for evaluating class-aware saliency detection. The proposed MCSD contains 200 images, which are extracted from MS-COCO [59]. Considering that each image contains one or multiple saliency instances, we re-label each salient object, and label the object segmentation as well as category ground truth by Wada’s toolbox [60]. Thus the MCSD contains three kinds of annotations, that is, 293 saliency binary masks, object segmentation masks and corresponding category labels. The details are introduced in Table 1.

For the purpose of visual saliency evaluation, we use four effective metrics for testing, including the precision-recall (PR) curve, F-measure, mean absolute error (MAE) and area under curve (AUC).

### 4.3.1 PR curve

PR curve is a universally used metric to evaluate saliency performance. It binarises saliency map with threshold from 0 to 255 (e.g. 8-bit map) and compares the binary maps with ground truth.

### 4.3.2 F-measure

It is an overall measurement computed from precision and recall by weighted combination, as follows:

\[
F_\beta = \frac{(1 + \beta^2) \cdot \text{precision} \cdot \text{recall}}{\beta^2 \cdot \text{precision} + \text{recall}},
\]

where we set $\beta^2 = 0.3$ to weigh the precision more over the recall.

### 4.3.3 MAE

It calculates mean absolute error between prediction and ground truth pixel by pixel, as follows:

\[
MAE = \frac{1}{M \times N} \sum_{x=1}^{M} \sum_{y=1}^{N} |(S(x,y) - G(x,y)|,
\]

where $M$ and $N$ indicate the size of the visual saliency map $S$. $S(x,y)$ and $G(x,y)$ represent prediction values and ground truth values of coordinate $(x,y)$, respectively.

### 4.3.4 AUC

It represents the area of ROC curve, sliding from 0.5 to 1. And higher value means better accuracy.

Considering the proposed framework simultaneously proceeds visual saliency detection and class-awareness segmentation, we particularity introduce a new metric, that is, FAP, to make evaluation on both saliency accuracy and segmentation precision. The FAP is calculated as follows:

\[
FAP = \rho_1 \cdot F_\beta + \rho_2 \cdot AP,
\]

where $F_\beta$ indicates F-measure for visual saliency detection and mask $AP$ means averaged over IoU thresholds for segmentation. $\rho_1$ and $\rho_2$ are constant coefficients, which we set 0.5 both in our experiments. The range of FAP is [0, 1] and higher value means better performance.

### 4.4 Performance comparison

The proposed SF-OCA can run visual saliency detection with homogeneous interior and accurate target boundaries. To demonstrate the effectiveness of the proposed method, we compare our method with nine state-of-the-art visual saliency models, including two unsupervised approaches, which are RC [23] and wCtr [24], and seven deep learning based methods, which are UCF [8], SRM [9], Amulet [7], PiCANet [10], MLM-SNet [16], BASNet [27] and PFA [30]. On the ECSSD [19], DUTS-TE [20], DUT-OMRON [21] and HKU-IS [22], we evaluate [7–9, 16, 27, 30] by making use of the source test results provided by the official releases.

In the proposed framework, we design the VSNet for visual saliency detection by a simplest encoder–decoder architecture without additional tricks, which is marked as Ours-baseline. We also replace the VSNet with two up-to-date visual saliency detection approaches [16, 30], which are donated as Ours-MLM and Ours-PFA. Figure 6 illustrates PR curves and F-measure scores, our method performs favourably against other methods on all four datasets. Table 2 shows various metrics across four datasets. Our method is very comparable with the state-of-the-arts over four metrics. For instance, Ours-baseline achieves 0.8801 F-measure and 0.0488 MAE on ECSSD [19] dataset, higher than [7, 10]. While Ours-MLM and Ours-PFA achieve state-of-the-art performance on DUT-OMRON [21] and the proposed MCSD.

However, the most significant advantages of the proposed SF-OCA can be introduced as follows. On the one hand, our method can visual saliency detection. On the other hand, with respect to multiple classes coexisting, our method could preserve task-relevant salient targets meanwhile eliminate insignificant background and irrelevant targets in accordance with detection task. In Figure 7, we present several visual
FIGURE 6  PR curves, precision, recall and F-measure scores on ECSSD, DUTS-TE, DUT-OMRON, HKU-IS
### TABLE 2  
Performance of the proposed SF-OCA and the state-of-the-art methods on four public datasets and the proposed MCSD. We take F-measure scores, MAE and AUC as evaluation metrics. The best two scores are marked in bold and italic, respectively. Ours-baseline: Using the VSNet for visual saliency detection. Ours-MLM: Replacing VSNet with MLMSNet [16]. Ours-PFA: Replacing VSNet with PFA [30].

| Methods         | ECSSD         | DUTS-TE       | DUT-OMRON     | HKU-IS        | MCSD          |
|-----------------|---------------|---------------|---------------|---------------|---------------|
|                 | $F_p$ ↑       | MAE ↓         | AUC ↑         | $F_p$ ↑       | MAE ↓         | AUC ↑         | $F_p$ ↑       | MAE ↓         | AUC ↑         | $F_p$ ↑       | MAE ↓         | AUC ↑         | $F_p$ ↑       | MAE ↓         | AUC ↑         |
| RC [23]         | 0.3412        | 0.3115        | 0.6283        | 0.2725        | 0.2558        | 0.6530        | 0.2776        | 0.2643        | 0.6579        | 0.3455        | 0.2756        | 0.6542        | 0.3409        | 0.2563        | 0.6935        |
| wCtr [24]       | 0.6762        | 0.1714        | 0.8812        | 0.5041        | 0.1533        | 0.8552        | 0.5272        | 0.1440        | 0.8850        | 0.6763        | 0.1424        | 0.8960        | 0.5695        | 0.1607        | 0.8738        |
| UCF [8]         | 0.8439        | 0.0691        | 0.9759        | 0.6307        | 0.1122        | 0.9559        | 0.6206        | 0.1240        | 0.9376        | 0.6783        | 0.1177        | 0.8763        | -             | -             | -             |
| SRM [9]         | 0.8923        | 0.0544        | 0.9708        | 0.7527        | 0.0588        | 0.9463        | 0.7068        | 0.0694        | 0.9166        | 0.8738        | 0.0459        | 0.9692        | -             | -             | -             |
| Amulet [7]      | 0.8683        | 0.0589        | 0.9731        | 0.6775        | 0.0846        | 0.9498        | 0.6471        | 0.0976        | 0.9398        | 0.8408        | 0.0507        | 0.9753        | -             | -             | -             |
| PiCANet [10]    | 0.8489        | 0.0588        | 0.9680        | 0.7357        | 0.0533        | 0.9486        | 0.6909        | 0.0666        | 0.9630        | 0.8479        | 0.0518        | 0.9620        | 0.7768        | 0.0664        | 0.9105        |
| BASNet [27]     | 0.8796        | 0.0370        | 0.9607        | 0.7911        | 0.0476        | 0.9348        | 0.7556        | 0.0565        | 0.9136        | 0.8955        | 0.0322        | 0.9582        | -             | -             | -             |
| MLMSNet [16]    | 0.8685        | 0.0446        | 0.9737        | 0.7455        | 0.0490        | 0.9582        | 0.6917        | 0.0636        | 0.9109        | 0.8710        | 0.0387        | 0.9735        | 0.9012        | 0.0401        | 0.9423        |
| PFA [30]        | 0.8874        | 0.0448        | 0.9756        | 0.7837        | 0.0409        | 0.9680        | 0.7751        | 0.0415        | 0.9660        | 0.8837        | 0.0324        | 0.9814        | 0.9106        | 0.0385        | 0.9437        |
| Ours-baseline   | 0.8801        | 0.0488        | 0.9682        | 0.7603        | 0.0583        | 0.9437        | 0.7012        | 0.0655        | 0.9029        | 0.8690        | 0.0430        | 0.9674        | 0.8748        | 0.0459        | 0.9038        |
| Ours-MLM        | 0.8740        | 0.0449        | 0.9714        | 0.7604        | 0.0482        | 0.9517        | 0.7034        | 0.0625        | 0.9072        | 0.8761        | 0.0395        | 0.9686        | 0.9036        | 0.0403        | 0.9355        |
| Ours-PFA        | 0.8884        | 0.0437        | 0.9756        | 0.7869        | 0.0409        | 0.9653        | 0.7786        | 0.0413        | 0.9650        | 0.8854        | 0.0322        | 0.9798        | 0.9115        | 0.0384        | 0.9433        |

**FIGURE 7**  
The visualisation results of the proposed SF-OCA and state-of-the-art visual saliency detection methods. Illustrations by comparing our baseline framework with other methods. For the images with simple background and legible texture, our method has little gap with the state-of-the-arts. Nevertheless, under circumstances like eye-catching background and low contrast, possibly many the state-of-the-art visual saliency detection methods failed to locate and detect foreground salient objects. However, our method can effectively detect foreground salient areas. We take the last three rows in Figure 7 as examples. The state-of-the-art models are easily effected by the striking background and give improper predictions (e.g. the background tower on the last row), but in contrast our method successfully outputs the foreground salient
targets, which demonstrates the robustness of our method. Except naive visual saliency detection, our method could run class-aware saliency detection. Additionally, for the purpose of detecting saliency map for specific targets, we test our model on the proposed MCSD. Comparing with popular saliency datasets, the MCSD consists of 200 samples containing multiple labels, including saliency binary masks, object segmentation annotations and category labels, and the details are introduced in Section 4.3 and Table 1. The test results on the proposed dataset are shown in Figure 8. As can be seen the output saliency maps are not only visual saliency but also class-awareness, that is, for different objects belonging to variance categories, the proposed method is able to generate saliency map for certain categories with homogeneous interior and distinct boundary. For example, for the first image on the first row in Figure 8, there are two salient objects including person and sports ball. Our model can detect and output class-aware saliency map for the corresponding target, respectively.

4.5 Ablation study

4.5.1 Baselines

The VSNet in the proposed SF-OCA framework plays the role of providing visual saliency areas. And different baselines of the VSNet could achieve different performance on visual saliency detection and the final class-awareness results. The test dataset is the proposed MCSD, which support visual saliency detection and class-aware saliency detection, simultaneously. From Table 3, it can be seen that for both VSNet only and VSNet+SCNet+VASM, the ResNet-50 version achieves the best $F_\beta$. As for VSNet+SCNet+VASM, scores of $F_\beta$ (0.9087 vs. 0.6768) and $MAE$ (0.0446 vs. 0.0864) are significantly improved comparing with VSNet only, which proves the effectiveness of our multi-cue guidance in improving saliency detection performance. Figure 9 illustrates PR-curves and correlative metrics including Precision and Recall. However, the curves in Figure 9(b) drops fast against high recall. To our best knowledge, this is because the CASM eliminates the irrelevant targets and areas, which allows the proposed framework focuses on task-relevant saliency target with homogeneous interior.

As seen in Figure 9(b), employing various baselines in VSNet has little effect on promoting performance. To our best guess, we evaluate the impact of $\lambda_1$ and $\lambda_2$ on ECSSD [19] and the

---

**FIGURE 8** The class-aware saliency detection results of the proposed SF-OCA. The output saliency maps indicate that our method is able to generate class-aware saliency map with homogeneous interior and distinct boundary. Zoom in for better review.

**TABLE 3** The performance of the proposed method by using different baselines in the VSNet. The test dataset is MCSD. We use $F_\beta$, $MAE$ and $AUC$ as evaluation metrics. The best scores are marked in bold.
TABLE 4  Impact of $\lambda_2$ on the final saliency map. The test dataset are ECSSD and the proposed MCSD, and notice that $\lambda_1+\lambda_2=1$. The best results are marked in bold.

| $\lambda_2$ | ECSSD | MCSD |
|-------------|-------|------|
|             | $F_\beta \uparrow$ | $\text{MAE}_1$ | $F_\beta \uparrow$ | $\text{MAE}_1$ |
| 0           | 0.7759 | 0.1077 | 0.6383 | 0.0925 |
| 0.1         | 0.8491 | 0.0739 | 0.8964 | 0.0440 |
| 0.2         | 0.8568 | 0.0616 | 0.8884 | 0.0430 |
| 0.3         | 0.8655 | 0.0503 | 0.8847 | 0.0421 |
| 0.4         | 0.8741 | 0.0489 | 0.8817 | 0.0412 |
| 0.5         | **0.8823** | **0.0485** | 0.8763 | 0.0405 |
| 0.6         | 0.8812 | 0.0490 | 0.8759 | 0.0397 |
| 0.7         | 0.8801 | 0.0488 | **0.8961** | **0.0419** |
| 0.8         | 0.8402 | 0.0526 | 0.8771 | 0.0308 |
| 0.9         | 0.8313 | 0.0665 | 0.8791 | 0.0369 |
| 1.0         | 0.8145 | 0.0684 | 0.8856 | **0.0360** |

TABLE 5  Runtime analysis (fps) of the proposed method. The images those being fed into the network are rescaled to 224×224.

|        | VSNet | SCNet | CASM |
|--------|-------|-------|------|
| Train  | 45.7  | 2.6   | -    |
| Test   | 87.5  | 4.6   | 24.4 |

proposed MCSD. As seen in Table 4, diverse weights of $\lambda_2$ make different attributions to the final performance. Comparing with employing the VSNet only, that is, $\lambda_2 = 0$, the testing indexes indicate that the proposed SCNet and CASM could effectively improve the visual saliency performance. Nevertheless, simply increasing the weight of $\lambda_2$ is not always benefiting the final result, which shows that the segmentation work could enhance the saliency detection performance but there still is limitation. For example, on ECSSD dataset, when $\lambda_2 = 0.5$, the proposed method achieves the best $F_\beta$ and $\text{MAE}_1$. While on the proposed MCSD dataset, $F_\beta$ score is the highest when $\lambda_2 = 0.7$. However, the best $\text{MAE}_1$ score occurs when $\lambda_2 = 1.0$. So for different datasets, the best results may be achieved by different parameter settings. In addition, the runtime analysis of each part of the proposed framework is exhibited in Table 5. Please note that the testing batch size is set as 1.

4.5.2  Segmentation performance

The proposed SCNet runs object segmentation and classification. The segmentation result contributes to the final
TABLE 6  Performance of average precision of the proposed method, we employ segmentation evaluation metrics including mask AP (IoU from 0.5 to 0.95 with increments of 0.05), AP50, AP75. The test datasets are COCO minival and the proposed MCSD.

| Metrics | COCO Minival | MCSD |
|---------|--------------|------|
| AP(%)  | 28.7         | 39.0 |
| AP50 (%) | 49.6         | 62.8 |
| AP75 (%) | 30.4         | 41.7 |

TABLE 7  The proposed SF-OCA performance on MCSD using FAP for evaluation. FAP, FAP50, FAP75 indicate that using mask AP (IoU from 0.5 to 0.95 with increments of 0.05), AP50, AP75, respectively.

| Metric   | Ours-Baseline | Ours-MLM | Ours-PFA |
|----------|---------------|----------|----------|
| FAP      | 0.6324        | 0.6468   | 0.6508   |
| FAP50    | 0.7514        | 0.7658   | 0.7698   |
| FAP75    | 0.6459        | 0.6603   | 0.6643   |

redclass-aware saliency segmentation performance. For purpose of evaluating the SCNet, we take average precision (AP) as index on COCO minival set [59] and the proposed MCSD dataset. The results are shown in Table 6. The SCNet achieves 28.7 mask AP and 39.0 mask AP on two testing datasets, respectively.

As introduced in Section 4.3, we use FAP to evaluate the selective focus saliency detection result, which simultaneously takes into account of visual saliency and class-awareness segmentation. The results are shown in Table 7. Comparing with our baseline model, that is, using VSNet for coarse visual saliency detection, more robust visual saliency detection models like MLMSNet [16] and PFA [30] lead to better performance due to the high quality visual saliency output.

4.5.3  Tough circumstances

The proposed SF-OCA utilises multi-cue to the final robust prediction. Figure 10 displays some test samples in several difficult circumstances. Comparing with state-of-the-art saliency detection models [10, 16, 30], the row (a) demonstrates that our method can effectively eliminate interference of complex background and generate preferable saliency map. The rows (b) and (c) illustrate that our model can be applied to small objects and low contrast scenes, respectively. While the row (d) indicates testing results on multiple objects situation, and for different target categories (e.g. giraffe and elephant on the last row), our method successfully detect both kinds of salient targets. Moreover, under the circumstance of multiple objects adhesion, our method can segment and classify each salient target individually, offering the choice of outputting class-aware saliency map with distinct boundary. Figure 11 gives some failure examples of the proposed method, we see that under circumstances like small, multiple targets coexist in large visual field, or small target with strong interference and suchlike challenging situations, the proposed method may not be very qualified. For example, row (a) contains two kinds of salient objects. Due to the dense crowd and the small sports ball, the proposed method does not segment all person target separately and fails to segment the sports ball. While row (b) shows a situation that with strong background inference and the train set of SCNet do not contain the signboard label which leads to a bad segmentation result. These issues may will be further explored in the future work.

5  CONCLUSION

In this paper, we propose a novel framework named SF-OCA, aiming to achieve selective focus saliency detection. Our framework balances salient object detection, segmentation and classification to output saliency target with homogeneous interior and distinct boundary. Experimental results show that our
framework achieves comparable performance with the state-of-the-arts. To sum up, this paper is proposed to overall consider human visual attention and subjective task-driven manner to run class-aware saliency detection driven by object class-awareness, going further than visual saliency detection only. Additionally, the backbone of the proposed framework is not unchangeable, there is still potential to improve the performance by using more effective network architectures.

ACKNOWLEDGEMENTS
This work was supported by the National Key R&D Program of China under Grant 2017YFC1405600, the National Natural Science Foundation of China under Grant 6167103, and the Aeronautical Science Foundation of China under Grant 2017ZC51046.

REFERENCES
1. Borji, A., et al.: Adaptive object tracking by learning background context. In: 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, pp. 23–30. IEEE, Piscataway (2012)
2. Lin, M., Dai, H.: Object tracking based on visual attention. In: 2016 IEEE International Conference on Information and Automation (ICIA), pp. 1846–1849. IEEE, Piscataway (2016)
3. Guo, C., Zhang, L.: A novel multisresolution spatiotemporal saliency detection model and its applications in image and video compression. IEEE Trans. Image Process. 19(1), 185–198 (2010)
4. Hadizadeh, H., Bajic, I.V.: Saliency-aware video compression. IEEE Trans. Image Process. 23(1), 19–33 (2014)
5. Donoser, M., et al.: Saliency driven total variation segmentation. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 817–824. IEEE, Piscataway (2009)
6. Song, H., et al.: Depth-aware salient object detection and segmentation via multiscale discriminative saliency fusion and bootstrap learning. IEEE Trans. Image Process. 26(9), 4204–4216 (2017)
7. Zhang, P., et al.: Amulet: Aggregating multi-level convolutional features for salient object detection. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 202–211. IEEE, Piscataway (2017)
8. Zhang, P., et al.: Learning uncertain convolutional features for accurate saliency detection. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 212–221. IEEE, Piscataway (2017)
9. Wang, T., et al.: A stagewise refinement model for detecting salient objects in images. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 4039–4048. IEEE, Piscataway (2017)
10. Liu, N., et al.: PicNet: Learning pixel-wise contextual attention for saliency detection. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3089–3098. IEEE, Piscataway (2018)
11. Zeng, Y., et al.: Learning to promote saliency detectors. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1644–1653. IEEE, Piscataway (2018)
12. Xi, X., et al.: Saliency object detection based on an efficient end-to-end saliency regression network. Neurocomputing 323, 46–57 (2019)
13. Liu, Z., et al.: Saliency object detection for RGB-D image by single stream recurrent convolution neural network. Neurocomputing 363, 46–57 (2019)
14. Zhang, J., et al.: Top-down neural attention by excitation backprop. Int. J. Comput. Vision 126, 1084–1102 (2017)
15. Feng, M., et al.: Attributive feedback network for boundary-aware salient object detection. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1623–1632. IEEE, Piscataway (2019)
16. Wu, R.T., et al.: A mutual learning method for salient object detection with intertwined multi-supervision. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8142–8151. IEEE, Piscataway (2019)
17. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv, arXiv:1409.1556, 2015
18. He, K., et al.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778. IEEE, Piscataway (2016)
19. Yan, Q., et al.: Hierarchical saliency detection. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1155–1162. IEEE, Piscataway (2013)
20. Wang, L., et al.: Learning to detect salient objects with image-level supervision. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3796–3805. IEEE, Piscataway (2017)
21. Yang, C., et al.: Saliency detection via graph-based manifold ranking. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3166–3173. IEEE, Piscataway (2013)
22. Li, G., Yu, Y.: Visual saliency detection based on multiscale deep CNN features. IEEE Trans. Image Process. 25(11), 5012–5024 (2016)
23. Cheng, M.M., et al.: Global contrast based salient region detection. In: IEEE Computer Vision and Pattern Recognition (CVPR), vol. 37, pp. 569–582. IEEE, Piscataway (2011)
24. Zhu, W., et al.: Saliency optimization from robust background detection. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2814–2821. IEEE, Piscataway (2014)
25. Wang, L., et al.: Saliency object detection with recurrent fully convolutional networks. IEEE Trans. Pattern Anal. Mach. Intell. 41, 1734–1746 (2019)
26. Hou, Q., et al.: Deeply supervised salient object detection with short connections. IEEE Trans. Pattern Anal. Mach. Intell. 41(4), 815–828 (2019)
27. Qin, X., et al.: Basnet: Boundary-aware salient object detection. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7471–7481. IEEE, Piscataway (2019)
28. Zhang, X., et al.: Progressive attention guided recurrent network for salient object detection. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 714–722. IEEE, Piscataway (2018)
29. Li, X., et al.: Deepsaliency: Multi-task deep neural network model for salient object detection. IEEE Trans. Image Process. 25(8), 3919–3930 (2016)
30. Zhao, T., Wu, X.: Pyramid feature attention network for saliency detection. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3080–3089. IEEE, Piscataway (2019)
31. Wang, W., et al.: Salient object detection in the deep learning era: An in-depth survey. arXiv, arXiv:1904.09146, 2019
42. Li, H., et al.: CNN for saliency detection with low-level feature integration. Neurocomputing 226, 212–220 (2017)
43. Imamoglu, N., et al.: Saliency detection by forward and backward cues in deep-cnn. In: 2017 IEEE International Conference on Image Processing (ICIP), pp. 430–434. IEEE, Piscataway (2017)
44. Fu, K., et al.: Deeplab: A general deep framework for salient object detection. Neurocomputing 356, 69–82 (2018)
45. Chen, L.C., et al.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE Trans. Pattern Anal. Mach. Intell. 40, 834–848 (2018)
46. Zhao, H., et al.: Pyramid scene parsing network. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6230–6239. IEEE, Piscataway (2017)
47. Pang, Y., et al.: Multi-scale interactive network for salient object detection. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9410–9419. IEEE, Piscataway (2020)
48. Kruthiventi, S.S.S., et al.: Saliency unified: A deep architecture for simultaneous eye fixation prediction and salient object segmentation. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5781–5790. IEEE, Piscataway (2016)
49. Wang, W., et al.: Saliency object detection driven by fixation prediction. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1711–1720. IEEE, Piscataway (2018)
50. Cao, C., et al.: Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks. In: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 2956–2964. IEEE, Piscataway (2015)
51. Fernando, T., et al.: Task specific visual saliency prediction with memory augmented conditional generative adversarial networks. In: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1539–1548. IEEE, Piscataway (2018)
52. Umeki, Y., et al.: Salient object detection with importance degree. IEEE Access 8, 147059–147069 (2020)
53. He, K., et al.: Mask R-CNN. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 2980–2988. IEEE, Piscataway (2017)
54. de Boer, P.T., et al.: A tutorial on the cross-entropy method. Ann. Oper. Res. 134(1), 19–67 (2005)
55. Sappa, A.D., Dornaika, F.: An edge-based approach to motion detection. In: International Conference on Computational Science 2006, pp. 563–570. Springer, Berlin Heidelberg (2006)
56. Wang, L., et al.: Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. 39(6), 1137–1149 (2017)
57. Girshick, R.B.: Fast R-CNN. In: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 1440–1448. IEEE, Piscataway (2015)
58. Lin, T.Y., et al.: Microsoft coco: Common objects in context. arXiv, arXiv:1405.0312, 2014
59. Wang, K.: labelme: Image polygonal annotation with Python. https://github.com/wkentaro/labelme (November 25, 2019)

How to cite this article: Zhao D, Yuan B, Shi Z, Jiang Z. Selective focus saliency model driven by object class-awareness. IET Image Process. 2021;15:1332–1344. https://doi.org/10.1049/ipt2.12108