DAANet: Dual Attention Aggregating Network for Salient Object Detection

Yijie Li[a,*], Hewei Wang[a,*], Shaofan Wang[b], Soumyabrata Dev[c,d,**]

[a]*Beijing-Dublin International College, Beijing University of Technology, Beijing 100124, China*
[b]*Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China*
[c]*The ADAPT SFI Research Centre, Dublin D04V1W8, Ireland*
[d]*School of Computer Science, University College Dublin, Dublin D04V1W8, Ireland*

Abstract

Convolutional neural networks have been introduced for salient object detection (SOD) for several years which have been proven to have the ability to achieve better performance than traditional methods. In the early stage of the development of CNN in the SOD task, most of the convolutional neural networks use simple structured feature extraction methods with fully-connected layers to generate salient masks which failed to capture and aggregate sensitive information at the different down-sample stages. The feature pyramid network (FPN) based structure with encoder and decoder is more popular for semantic segmentation and salient object detection tasks, the FPN structure with attention modules can efficiently capture the important area of input feature and achieve better performance. In this paper, to improve the overall performance of salient object detection tasks, we propose a dual attention aggregating network (DAANet), which is an FPN-based deep convolutional neural network with a dual attention aggregation module (DAAM) and boundary-joint training. The DAAM considers the salient map prediction from the low-level output as pseudo-attention which can efficiently aggregate multi-scale information. The Convolution block attention module (CBAM) in DAAM can refine the aggregation of pseudo-attention which enables better performance. We evaluate our proposed DAANet on six benchmark datasets and analyze the effectiveness of each module of DAANet which shows that DAANet outweighs other previous approaches in many aspects. In addition, the lightweight configuration of the model can achieve an MAE of 0.051 on the DUTS-TE dataset with only 15.8 MB of parameters. In the spirit of reproducible research, the model code, dataset, and results of the experiments in this paper are available at: `https://github.com/Att100/DAANet`.

Keywords: Convolutional neural network, salient object detection, attention, boundary-joint training

1. Introduction

Salient object detection refers to the identification of vital visual information analog to the human attention mechanism. Most of the proposed methods of this computer vision task have been widely used in film and television production, and image matting. The early approaches for salient object detection are usually based on traditional visual extraction methods. Some of those methods use handcraft features or filters to extract important regions, and others may adopt the graph-based method to transfer the computer vision task to a graph optimization problem. When the convolutional neural networks

*Authors contributed equally to this research.
**Corresponding author. Email: soumyabrata.dev@ucd.ie, Tel.: + 353 1896 1797.
Email addresses: `yijie.li@ucdconnect.ie` (Yijie Li), `hewei.wang@ucdconnect.ie` (Hewei Wang), `wangshaofan@bjut.edu.cn` (Shaofan Wang), `soumyabrata.dev@ucd.ie` (Soumyabrata Dev)

have been widely used in all kinds of computer vision tasks, many research works on CNN-based salient object detection have been released. Before the introduction of classical image classification networks, such as VGG [1] and ResNet [2], many CNN-based approaches use simple design with the combination of CNN and fully connected layers to generate salient masks. Those early methods usually yield limited performance. Since 2016, many classical pre-trained image classification models have been used as feature extraction backbone in salient object detection which significantly improves the overall performance of prediction. In recent years, FPN [3] and UNet [4] have become the most popular structure in SOD. The shortcut connection between the encoders and decoders enhances the ability of feature aggregation. However, the encoder-decoder-based approach with straightforward design in decoders is hard to capture the details of a specific salient region. In addition, the prediction of ordinary multi-class semantic segmentation is more discreet than salient object detection because every pixel should be labeled with a class and the pixel with the same class can be everywhere in the image. As for salient object detection, the pixels with the positive label (salient object) are usually located at a small scope which means the model should pay more attention to the overall structure. Those drawbacks motivate us to use dual attention to remit these issues.

In this paper, we propose a dual attention aggregation network (DAANet) with boundary-joint training to enhance the feature aggregation in decoding stages and improve the overall performance for salient object detection. The DAANet includes a backbone network as an encoder and several dual attention aggregating modules (DAAM) as decoders. If boundary-joint training is enabled, the network will contains additional boundary-joint decoders (BJD) to predict boundaries. The DAAM has two attention while one of which is a pseudo-attention while it considers the salient map from the previous decoding stage as attention weights. Another attention is a modified version of the convolution block attention module (CBAM) [5] which refines the previous aggregation results to enable more accurate prediction. Inspired by BASNet introduced by Qin *et al.* [6], we adopt boundary joint-training with a U-shaped encoder-decoder which shares the same backbone with our original network, while the boundary is generated by the Prewitt operator from salient ground-truth. We also use the combined binary cross entropy loss and IOU loss to improve the supervision on the overall structure.

The main contributions of this work are:

- We propose a novel decoder module for salient object detection: dual attention aggregation module (DAAM), which can achieve better performance than baseline models.
- We combine U-Net and FPN structure to joint training salient object detection and salient boundary detection.
- We do a complete evaluation and comparison with 12 methods on six benchmark datasets. Our method can achieve better performance.
- The lightweight configuration of DAANet can achieve an MAE of 0.051 on the DUTS-TE dataset with only 15.8 MB of parameters.

2. Related Works

Before the widely using of deep learning techniques, there already exist a great number of salient object detection methods such as [7, 8, 9, 10], which usually use gradient variation and features to judge whether a pixel belongs to

foreground or background, and some of these works use simple artificially designed features and algorithms to generate the estimation. Sai *et al.* [7] propose a foreground connectivity measurement to enhance the salient map retrieval. Their approach first builds an objectness map by utilizing the objectness proposals and capturing super-pixels containing the salient object, and then uses their proposed foreground connectivity measure to assign weight to super-pixels. Finally, they apply a saliency optimization to combine the foreground weight and background to get the saliency map. Wei *et al.* [8] illustrate that most of the background area can easily connect to the boundary of the image. On the contrary, it is difficult for the area belonging to the salient object to link the image boundary. This motivates them to redefine the saliency of an image patch as the length of its shortest path to image boundaries. Yang *et al.* [9] propose a graph-based approach that considers the image as a graph with super-pixel as nodes. They rank these nodes based on their similarity to the foreground and background queries to extract the background area and salient objects. In [10], Cheng *et al.* use global contrast as the core methodology to retrieve a salient map that defines the pixel saliency as the contrast ratio. They also introduce histogram-based contrast (HC) which uses color statistics of the input image to define saliency.

When the deep convolutional neural network (CNN) has been introduced to computer vision tasks, including image classification and semantic segmentation, most experiments show that the CNN-based methods can achieve better performance than traditional methods, and many researchers in salient object detection began to use CNN to generate prediction salient map. Li *et al.* [11] introduce a CNN-based approach to extract the salient map, their proposed model is first pre-trained on ImageNet and then uses the pre-trained model to perform the extraction under the multi-scale configuration. Zhao *et al.* [12] propose a multi-context CNN approach that can be modelling both global and local contextual information simultaneously. To enable better performance, they also introduce a task-specific pre-training strategy for their proposed model. Wang *et al.* [13] propose a hybrid approach with deep learning and a traditional method. They first uses a deep learning network and object proposal to generate two rough salient maps and adopt the global search with the fully-connected network to refine the previous results.

After the widely using of deep CNN in the image classification area, VGG [1] and ResNet [2], for example, many saliency detection methods including [14, 15, 16, 17, 18] use more efficient pre-trained backbone networks to extract features. When FCN [19] has been introduced, the encoder-decoder structure has been widely used in salient object detection models. Zhang *et al.* [14] propose a bidirectional model with a contextual feature extraction module that allows the integration of multi-level features and the bi-directional structure to enhance the message passing between features at different levels. They also adopt a gate function to control the passing of information. Hu *et al.* [15] introduce a novel model to recurrently aggregate deep features which can effectively exploit the complementary information extracted by each layer. They compress the combination of outputs from each layer and merge the combination with the feature of each layer to enhance the discrimination of features and salient regions.

Wang *et al.* [20] propose an encoder-decoder liked global recurrent localization networks (RLN) to address the information loss in the concatenation of high and low-level features, which take advantage of the contextual information to improve the accuracy of salient object localization. Liu *et al.* [21] propose a pixel-wise contextual attention network that can selectively learn the attention map for each pixel, while the attention weight represents the contextual relevance at the context location. Zhang *et al.* [22] introduced an FPN-based attention-guided network to remit the sub-optimal-result problem resulting from the interference of redundant details, which uses both channels and spatial attention in different

decoder stages. Feng *et al.* [23] illustrate that most of the FCN-based methods can accurately locate the overall salient object location, but failed to focus on the boundary, who designed attentive feedback module (AFMs) to better learn the details structure of salient objects. BASNet [6], proposed by Qin *et al.*, also aims to address the boundary detection loss of salient object detection. These early methods include a few convolution layers and use several fully connected layers to produce the final output.

In summary, those methods can accurately generate the overall salient map, but the details prediction and inference time consumption are still improvable in the future.

3. DAANet

In this section, we first introduce the overall pipeline of DAANet, as shown in Figure 1 and detailedly present each module afterward.

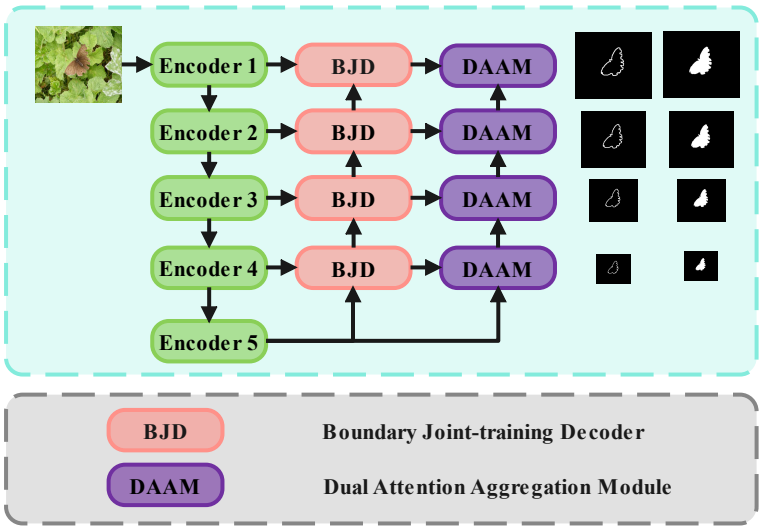

Figure 1: The overall pipeline of DAANet with boundary-joint training. The procedure between the output of model and the segmentation mask has been omitted in this figure.

3.1. Overall Pipeline

DAANet uses an FPN-based encoder-decoder structure. The encoders support various backbone networks, including ResNet50, VGG16, and MobileNetV2. The decoders can be a series of DAAMs and if the boundary-joint training technique is enabled, the network will additionally contain a series of BJDs that are responsible for the prediction of the salient boundary. The ground-truth of salient boundary generation will be introduced in Section 3.4. As for the training technique, we apply multi-stage supervision to enable the supervision of low-level features. We use a hybrid loss to train our model which is a combination of IOU loss and BCE loss.

3.2. Encoder Networks

As mentioned in previous sections, DAANet is a UNet and FPN-liked encoder-decoder network. This fundamental structure allows us to support and switch different encode networks. In our implementation, we support ResNet50[2], VGG16[1], and MobileNetV2[24] for training and inference. The outputs of each encoding stage will be fed into decoders.

The original ResNet-50 network that was designed for ImageNet has five down-sample stages which reduce the size of the feature map to 1/8 of the input image before fully-connected layers, but our DAANet only needs 4 down-sample stages to produce the four different outputs while the size of the smallest feature map is 32x32. To resolve this issue, we modify several parameters of the two convolution layers in the last encoding stage. The first convolution layer mentioned above refers to the second convolution layer in the first bottle-neck block of the last encoding stage whose new dilation is $(2, 2)$, the new padding values are $(1, 1)$, and the stride values are changes to $(1, 1)$. The next convolution layer refers to the first convolution layer in the down-sample sub-block of the same bottle-neck block whose strides are changed to $(1, 1)$. The VGG16 for our DAANet has the same problem and we solve this issue by removing the last max-pooling layer. As for the MobileNetV2, we reduce the stride of a convolution layer to $(1, 1)$.

3.3. Dual Attention Aggregation Module (DAAM)

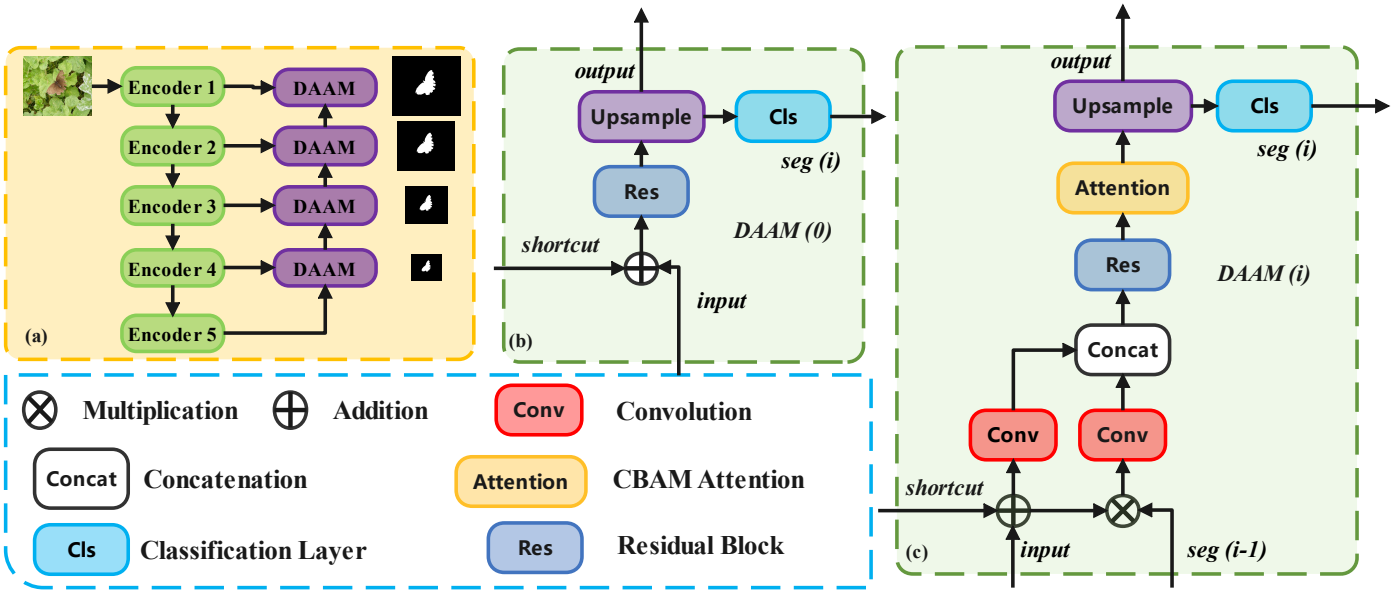

Figure 2: The architectures of DAAM and overall structure without boundary-joint training. (a) presents the overall pipeline without boundary-joint training. (b) presents the DAAM structure in the first decoding stage (c) presents the DAAM without boundary-joint training.

The salient object detection task requires the model to segment the object with the highest probability to be the foreground object, whose mechanism is much similar to the attention technique which motivates us to integrate channels and spatial attention module (CBAM) [5] into our decoder, shown as Figure 3. To enhance the attention, we consider the output of a DAAM as a pseudo-attention which will be multiplied with the addition of a shortcut and input. Those two attention constructs dual attention.

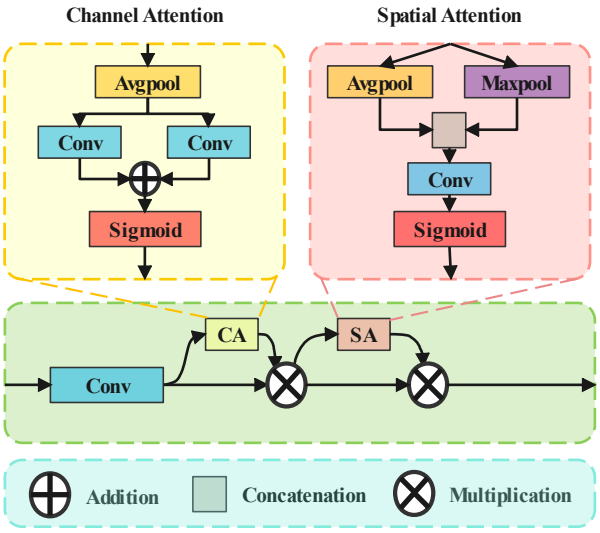

Figure 3: The structure of modified CBAM [5]

In each DAAM block, the shortcut, and the output together with the salient map of the previous layer will be accepted as inputs while its output will be sent to the next DAAM block. To better exploit and recover highly-compressed features in the overall-structure perspective, we consider the salient map from the previous layer as pseudo-attention weight. Firstly, we adopt an element-wise multiplication between the addition result and the saliency map of the previous DAAM block which constructed a rough weighted feature. Secondly, the additional results and the weighted feature will be concatenated after a Conv2d-Bn-ReLU group while this group of layers will reduce the channels to half of its inputs. After that, the features will go through a residual block which refers to the original bottle-neck design, and then it will be fed into CBAM to apply more accurate attention. The penult operation is an up-sample operation with the stride equal to 2. Finally, there will be two branches of outputs and one of which will go through a single convolution layer to build the prediction logits.

We also need to remind that the structure is shown in Figure 2 (b) indicates the first DAAM in decoders which only has a residual block, an up-sample layer, and a convolution layer which results from the low-resolution of features and it is hard to capture the spatial structure of the salient object and it is no need to apply the dual attention.

$$z_i = Concat(Conv(a_i), Conv(a_i \odot seg_{i-1})) \tag{1}$$

$$a_i = output_{i-1} + shortcut_i \tag{2}$$

As for the boundary-joint DAAM, we consider the salient boundary prediction from the previous layer as an additional input of DAAM, which is shown in Figure 2(c), and this process can be formulated as:

$$z_i = Concat(Conv(a_i), Conv(a_i \odot seg_{i-1}), e_i) \tag{3}$$

$$a_i = output_{i-1} + shortcut_i \tag{4}$$

$$e_i = Conv(\sigma(edge_{i-1})) \tag{5}$$

where z_i indicates the concatenation output, seg_i and $edge_i$ are the saliency map and boundary of i th decoding stage, $\odot$ is the Hadamard operator, and σ denotes the sigmoid activation function.

We observe that the original CBAM is designed for the backbone network and our design needs to aggregate it in decoders, the results show that the original CBAM may result in a significant decrease in MAE which motivates us to make some modifications. We modify it by removing the max-pooling layer in the channel attention module, as shown in Figure 3. The channel attention module of CBAM generates attention through a global average pooling followed by two paralleled convolution layers and a sigmoid function. The spatial attention module retrieves the attention map by passing the input feature into the max-pooling branch and average-pooling branch, concatenating them, and followed by a convolution layer with sigmoid activation.

As for our lightweight approach with MobileNetV2 [24] backbone, we replace the original residual block with the inverted residual [24], which can significantly decrease the parameter amount by using a series of depth-wise convolution.

3.4. Boundary Ground Truth Generation

Considering the boundary ground truth is not provided by the training dataset, we use the Prewitt operator to implement a simple boundary extraction function to extract the accurate boundary from provided saliency map ground-truth, four samples have been shown in Figure 4, while the Prewitt operator is shown as follows:

$$kernel_x = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}, kernel_y = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$$

Extraction procedure can be formulated as:

$$prewitt_x = Conv(img, kernel_x) \tag{6}$$

$$prewitt_y = Conv(img, kernel_y) \tag{7}$$

$$prewitt = \frac{prewitt_x + prewitt_y}{2} \tag{8}$$

where $prewitt_x$ and, $prewitt_y$ are the outputs of 2D convolution with $kernel_x$ and $kernel_y$, $prewitt$ indicates the extracted boundary.

3.5. Boundary-Joint training Decoder (BJD)

The boundary-joint training decoder is designed for multi-task training configuration which makes DAANet support the training of prediction for both salient objects and salient boundaries at the same time. The architecture of BJD is a simple straight-forward design, which concatenates the two inputs and goes through a residual block followed by an up-sample layer, as shown in Figure 5(b). The Figure 5(a) shows that the output of a BJD will be fed into both the next BJD block and the DAAM at the same level.

The structure which is built with backbone encoders and BJDs is a U-Net shaped design, and the fusion of encoded feature map and up-sampled output is completed by concatenation instead of using the addition operation mentioned in Section 3.3. The motivation of this idea is original U-Net is used to perform medical image processing, especially in cell

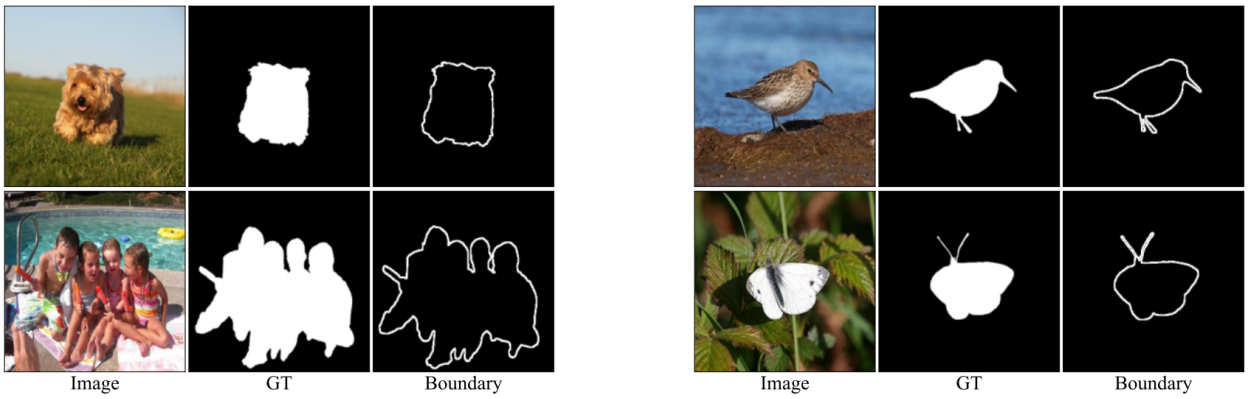

Figure 4: Sample of boundary ground-truth

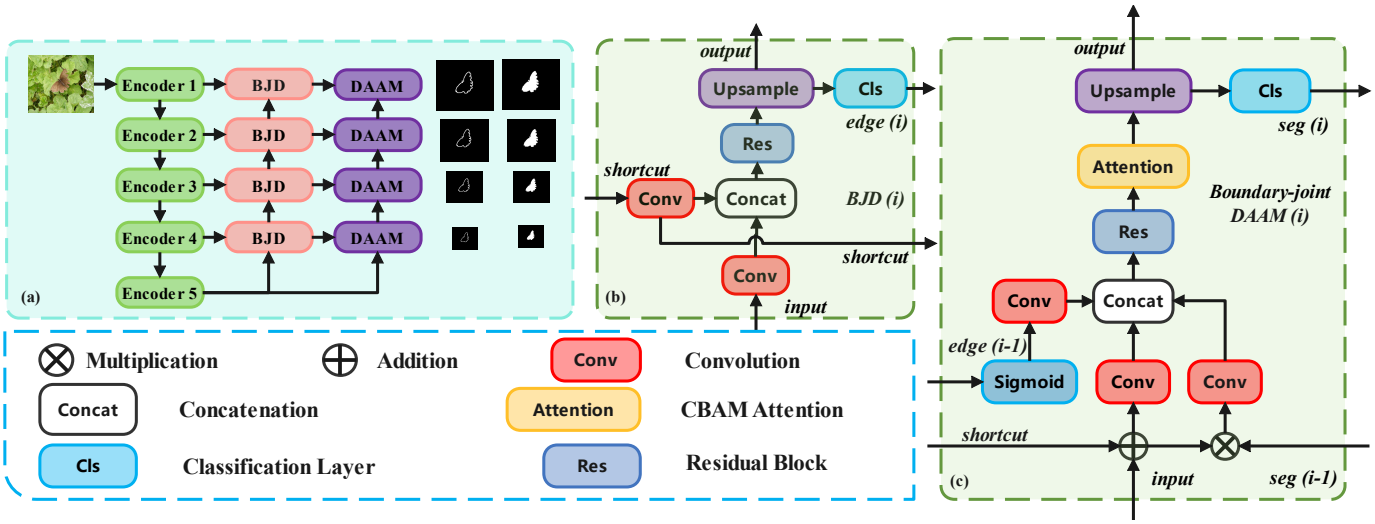

Figure 5: The structures of overall pipeline with DAAM and BJD, BJD and boundary-joint DAAM. (a) represent the overall pipeline with both DAAM and BJD. (b) is the structure of BJD. (c) represent the boundary-joint DAAM which have an additional output

boundary segmentation which means the performance of using U-Net liked structure in boundary detection tasks will bring us better results. Besides, the output of the up-sample layer will go through a Sigmoid activation to generate a prediction for multistage supervision and boundary-joint DAAM. In boundary-joint DAAM, as shown in Figure 5 (c), the output channels of multiplication results will be reduced to 1/4 of the original channels which are different from DAAM without boundary-joint training, while the channels of the boundary input will be increased to a suitable number from one.

3.6. Loss function

The loss functions we used in training are binary cross entropy (BCE) and the intersection of union (IOU) loss. The BCE loss is widely used in regression and binary classification tasks and the IOU loss cab add additional constrain to the supervision of overall structure, while those objective functions are shown below:

$$L_{BCE} = \frac{1}{NHW}\sum_{n=1}^{N}\sum_{i,j} -t_{i,j}\ln(p_{i,j}) + (1-t_{i,j})\ln(1-p_{i,j}) \tag{9}$$

$$L_{IOU} = 1 - \frac{1}{N}\sum_{n=1}^{N}\frac{\sum_{i,j}(t_{i,j}\times p_{i,j})}{\sum_{i,j}(t_{i,j}+p_{i,j}-t_{i,j}p_{i,j})} \tag{10}$$

To guarantee the convergence of loss and reduce the training iterations, we use multistage supervision for training and the total objective function without boundary-joint training can be given as:

$$L_{total} = \sum_{i=1}^{4}\beta_i(\alpha_1 L_{BCE} + \alpha_2 L_{IOU}) \tag{11}$$

if we consider training with the salient boundary at the same time, another loss function will be included:

$$L_{total} = \sum_{i=1}^{4}\beta_i(\alpha_1 L_{BCE_1} + \alpha_2 L_{BCE_2} + \alpha_3 L_{IOU}) \tag{12}$$

where t and p are target and prediction, n is the number of samples, m indicates the number of model outputs, α is the weight to balance two different objective functions, while β is used to balance the contribution of outputs with different resolution, 256x256, 128x128, 64x64, and 32x32.

3.7. Implementation Details

We perform the training on a single NVIDIA Tesla V100-SXM2 16GB GPU. We use the standard train-valid data split provided by original datasets. As for training and validation datasets, we train DAANet on DUTS-TR [25] and validate our DAANet with different configuration on six benchmark datasets, including DUTS-TE [25], SOD [26], HKU-IS [11], ECSSD [27], PASCAL-S [28], and DUT-OMRON [9]. We set the training batch size to 16 and trained 25 epochs for the model with ResNet50 and 30 epochs with other backbones. As for the optimizer, we use SGD with an initialized learning rate of 1e-2, momentum is 0.9, and weight decay is 5e-4. We also use exponential learning-rate decay with gamma equal to 0.85 after each training epoch. We evaluate DAANet on the test set after every 5 epochs. As for loss function, we set $\alpha_1 = \alpha_2 = \alpha_3 = 1$, $\beta_1 = 1$, $\beta_2 = 0.8$, $\beta_3 = \beta_4 = 0.5$ from high resolution output to low resolution output.

4. Experiments

4.1. Datasets

We train our DAANet on DUTS-TR [25] dataset, and evaluate on DUTS-TE [25], SOD [26], HKU-IS [11], ECSSD [27], PASCAL-S [28], and DUT-OMRON [9]. DUTS-TR has 10,553 images pair in total which is widely used for training. DUTS-TE has 5,019 images for testing. SOD include only 300 images while each of them contains complex semantic information and it is the most difficult benchmark for validation. HKU-IS contains 4,447 image pairs. ECSSD contains 1,000 samples for testing. The PASCAL-S dataset has 850 images. DUT-OMRON has 5,168 images for testing.

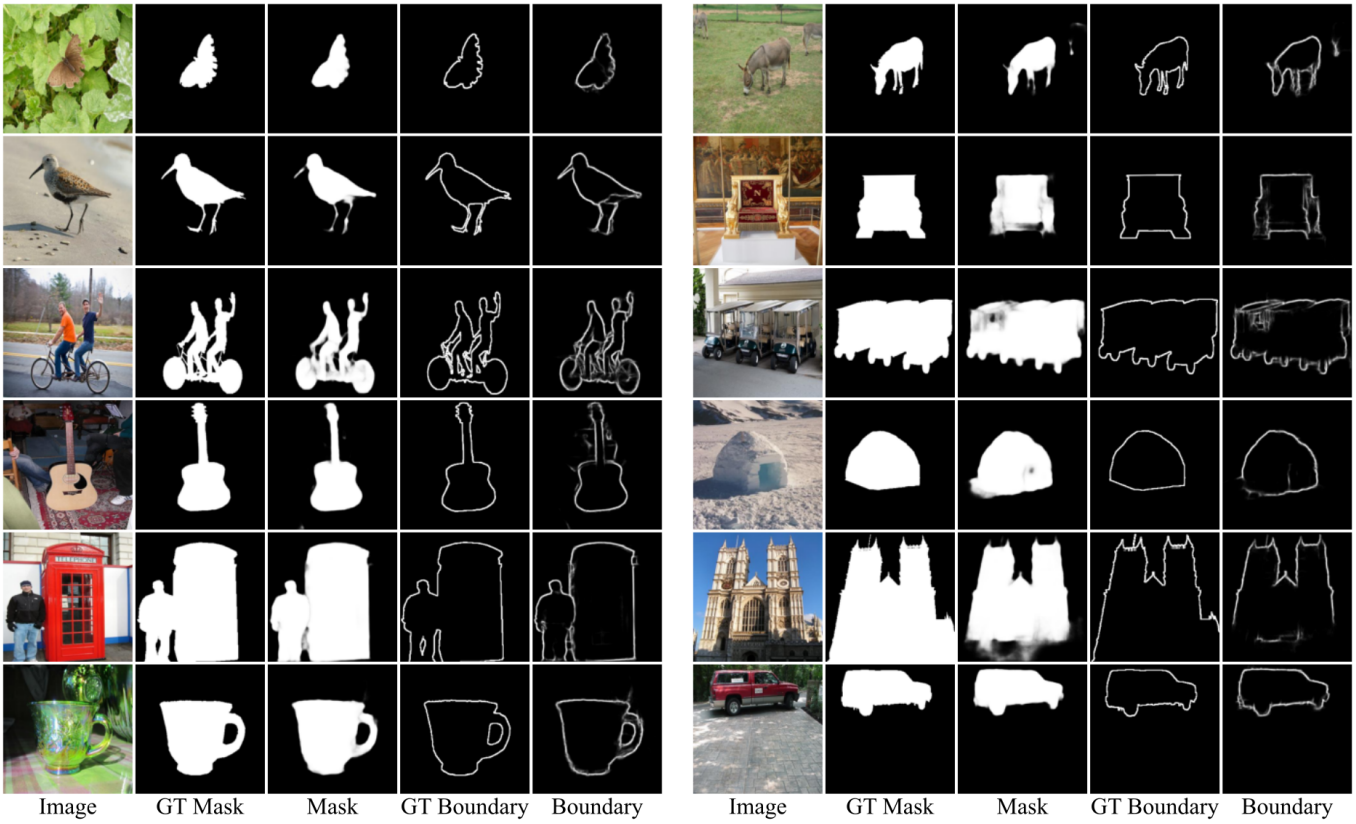

Figure 6: Samples of qualitative results, generated with our DAANet with VGG16 backbone and boundary-joint training

4.2. Evaluation Metrics

We use three measurements to evaluate DAANet: mean absolute error (MAE) [29], mean F-measure, and PR-curve. The MAE measures average pixel-wise differences between ground truth and prediction, given a salient ground truth t and a prediction p, the MAE can be defined as:

$$MAE = \frac{1}{NHW} \sum_{n=0}^{N} \sum_{i,j} |t_{i,j} - p_{i,j}| \tag{13}$$

The mean F-measure is a weighted harmonic mean of precision and recall which can be represented as:

$$mF_\beta = \frac{(1+\beta^2) \times Precision \times Recall}{\beta^2 \times Precision + Recall} \tag{14}$$

where β^2 is set to 0.3 which follows the widely used configuration. The PR curve is defined by a series of precision and recall pairs, each of which is calculated under a different threshold. We first cast the model output to 0-255 by multiplying the probability output by 255 and then set 256 threshold from 0 to 255, and then plot the PR-curve with those 256 points.

4.3. Ablation Study

In this section, we validate DAANet with different configuration on the DUTS-TE dataset, and the metrics we used in this experiment is MAE and mean F-measure. We will first describe the ablation of module composition and objective function, then we present the results on different backbone networks.

No.	Backbone	FPN configs			DUTS-TE	
		IOU	DAAM	BJD	mF_β	MAE
1	VGG16				0.748	0.053
2	VGG16	✓			0.777	0.049
3	VGG16	✓	✓		0.793	0.045
4	VGG16	✓	✓	✓	0.794	0.045
5	ResNet50	✓	✓		**0.800**	**0.042**
6	MobileNetV2	✓	✓		0.770	0.051

Table 1: Ablation study on different module composition objective function, and backbone networks. IOU represent the IOU loss.

Module composition and objective function: To show the influence of our proposed module on overall performance we perform experiments on DAAM, BJD, and IOU loss. We take the pure FPN+VGG16 as our baseline model. Then we extend the baseline model with IOU loss, DAAM module, and BJD. As shown in Table 1, the FPN+VGG16 with proposed DAAM can achieve MAE with 0.045 and F-measure equal to 0.794, while FPN+VGG16 with both proposed DAAM and BJD can achieve the same MAE and better F-measure. The results quantitatively proved the effectiveness.

Experiments on different backbones: To analyze the performance of our proposed DAAM on different backbone networks, we perform another two experiments on ResNet50 and MobileNetV2. Table 1 shows that DAAM with ResNet50 can achieve the best performance on both two metrics. The configuration with MobileNetv2 can achieve an MAE of 0.051, but the model size is only 15.8 MB.

4.4. Qualitative Evaluation

We evaluate DAANet qualitatively by comparing the generated salient map and boundary with ground truth, shown in Figure 6, which shows the effectiveness of both the DAAM module and boundary-joint training strategy. We can also see that if a part of the quality of salient map prediction is accurate, the prediction of its corresponding boundary is also accurate and it can be deduced that the prediction of those two tasks are highly coupled which means our design for boundary-joint training is rational. We also compare DAANet (ResNet50 backbone) with seven different approaches, including BASNet [6], PiCANet [30], BMPM [14], R3Net+ [31], PAGRN [22], SRM [32] and DGRL [20]. As shown in Figure 7, the results show that DAANet significantly improves the quality of the generated salient map, while DAANet can capture more details and have fewer wrong predictions than others.

4.5. Quantitative Evaluation

To further analyze the performance DAANet, we perform quantitative evaluation on three configuration of DAANet with 12 state-of-the-art approaches: UCF [33], Amulet [34], DSS [35], PAGRN [22], BMPM [14], AFNet [36], RAS [37], PiCANet [30], DGRL [20], SRM [32], PiCANet-R [30], and BASNet [6]. The three configurations of DAANet are represented as follow:

- **DAANet-A**: VGG16+DAAM+BJD
- **DAANet-B**: ResNet50+DAAM

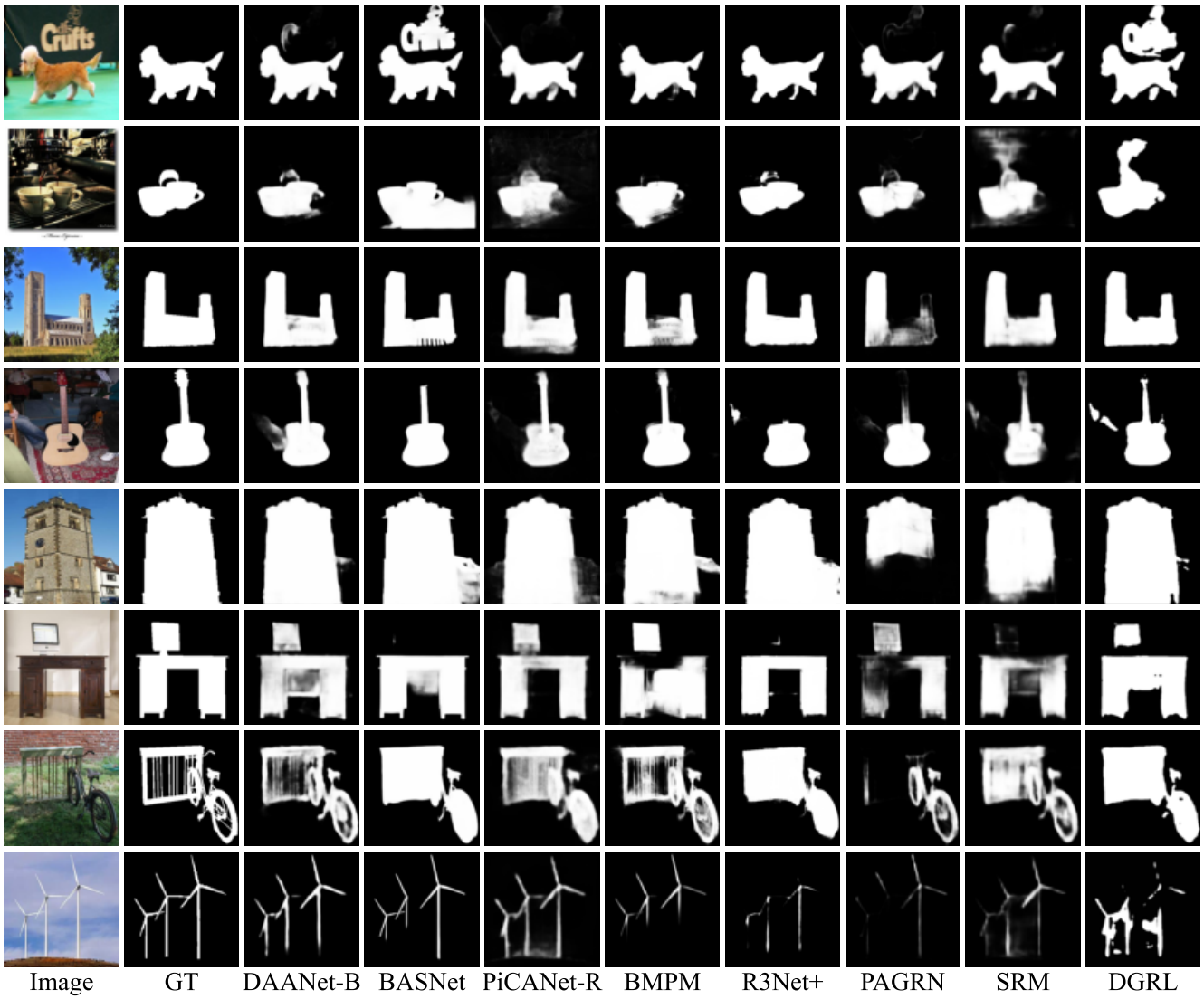

Figure 7: Qualitative results on DUTS-TE dataset with ResNet50 backbone and without BJD (DAANet-B)

- **DAANet-C**: MobileNetV2+DAAM

The evaluation metrics and datasets has mentioned in Section 4.2 and 4.1. As shown in Table 2, **DAANet-B** achieve the best performance on DUTS-TE dataset with $maxF_\beta$ by 0.870, mF_β by 0.801. and MAE by 0.042, which also be in the lead of other models on other five benchmark datasets. We also build light-weight approaches with MobileNetV2 backbone, **DAANet-C**, which only has 15.8 MB of parameters in total, but it can achieve the MAE by 0.051, which proved the effectiveness of our proposed DAAM module. Besides, we evaluate DAANet quantitatively by using the PR curves and F-measure curves, as shown in Figure 8, the area under the curve is larger, and the performance of the model is better. We can see from those figures that our proposed ResNet50+DAAM (**DAANet-B**) has the best performance on most datasets.

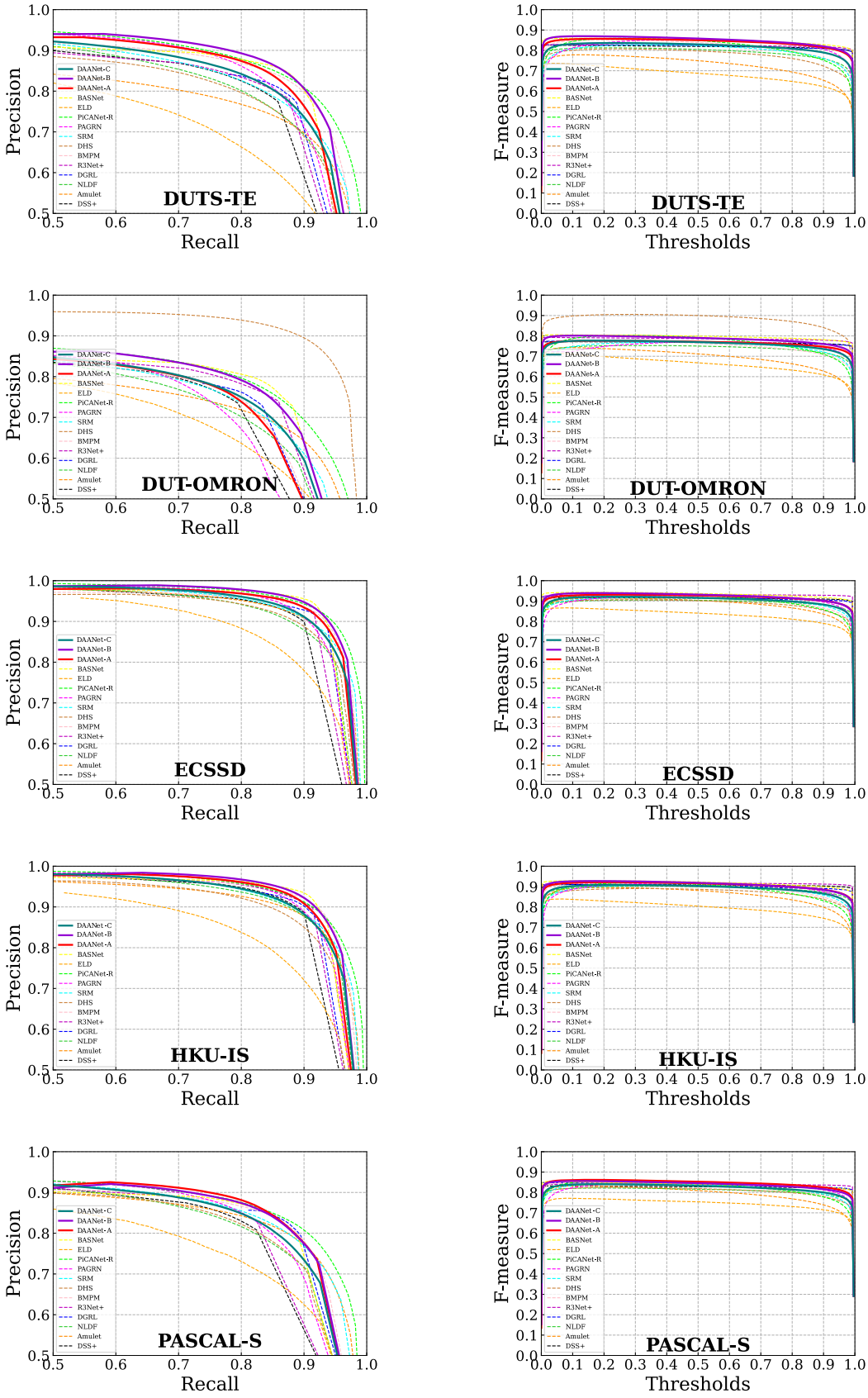

Figure 8: Illustration of PR curves (the first column) and F-measure curves (the second column) on five benchmark datasets

Methods	Size(MB)	Training Data		DUTS-TE[25]			HKU-IS[11]			SOD[26]			DUT-OMRON[9]			PASCAL-S[28]			ECSSD[27]		
		Datasets	#Images	$maxF_\beta \uparrow$	$mF_\beta \uparrow$	$MAE \downarrow$	$maxF_\beta \uparrow$	$mF_\beta \uparrow$	$MAE \downarrow$	$maxF_\beta \uparrow$	$mF_\beta \uparrow$	$MAE \downarrow$	$maxF_\beta \uparrow$	$mF_\beta \uparrow$	$MAE \downarrow$	$maxF_\beta \uparrow$	$mF_\beta \uparrow$	$MAE \downarrow$	$maxF_\beta \uparrow$	$mF_\beta \uparrow$	$MAE \downarrow$
VGG backbone																					
UCF[33]	**117.9**	MSRA10K	10,000	0.771	0.629	0.117	0.886	0.808	0.074	0.803	0.699	0.164	0.734	0.613	0.132	0.828	0.706	0.126	0.911	0.840	0.078
Amulet[34]	132.6	MSRA10K	10,000	0.778	0.676	0.085	0.895	0.839	0.052	0.806	0.755	0.141	0.742	0.647	0.098	0.837	0.768	0.098	0.915	0.870	0.059
DSS[35]	447.3	MSRA-B	2,500	0.826	0.791	0.057	0.910	**0.895**	0.041	0.841	0.793	0.121	0.772	0.729	0.066	0.831	-	0.093	0.916	**0.901**	0.053
PAGRN[22]	-	DUTS-TR	10,553	0.855	0.788	0.056	0.918	886	0.048	-	-	-	0.771	0.711	0.071	0.856	**0.807**	0.093	0.927	**0.894**	0.061
BMPM[14]	-	DUTS-TR	10,553	0.851	0.751	0.049	0.921	0.871	0.039	**0.855**	0.763	0.107	0.774	0.692	0.064	0.862	0.769	0.074	0.929	0.869	0.045
AFNet[36]	143.9	DUTS-TR	10,553	**0.862**	**0.797**	**0.046**	**0.923**	0.888	**0.036**	**0.856**	**0.809**	0.109	0.797	**0.738**	0.057	**0.868**	**0.826**	0.071	0.935	**0.908**	0.042
RAS[37]	**81.0**	MSRA-B	2,500	0.831	0.755	0.060	0.913	0.871	0.045	0.850	0.799	0.124	0.786	0.713	0.062	0.837	0.785	0.104	0.921	0.889	0.056
PiCANet[30]	153.3	DUTS-TR	10,553	0.851	0.755	0.054	0.921	0.870	0.042	0.853	0.791	0.102	0.794	0.710	0.068	**0.868**	0.801	0.077	0.931	0.884	0.047
DAANet-A	119.3	DUTS-TR	10,553	0.857	0.794	**0.045**	0.922	0.888	0.036	0.838	0.753	0.114	0.777	0.722	**0.058**	0.860	0.785	**0.072**	0.931	0.878	0.044
ResNet backbone																					
DGRL[20]	648.0	DUTS-TR	10,553	0.829	0.798	0.050	0.921	0.890	0.036	0.845	**0.799**	**0.104**	0.774	0.733	0.062	0.854	**0.825**	**0.072**	0.922	0.906	**0.041**
SRM[32]	213.1	DUTS-TR	10,553	0.827	0.757	0.059	0.906	0.874	0.046	0.843	**0.800**	0.127	0.769	0.707	0.069	0.847	0.801	0.085	0.917	0.892	0.054
PiCANet-R[30]	197.2	DUTS-TR	10,553	**0.860**	0.764	0.051	0.919	0.870	0.043	0.853	0.785	**0.103**	**0.803**	0.717	0.065	0.857	0.792	0.076	**0.935**	0.886	0.046
BASNet[6]	348.5	DUTS-TR	10,553	0.859	**0.796**	0.048	**0.928**	**0.896**	**0.032**	0.851	0.745	0.113	**0.805**	**0.755**	**0.057**	**0.862**	0.779	0.077	**0.942**	0.879	**0.037**
DAANet-B	229.0	DUTS-TR	10,553	**0.870**	**0.801**	**0.042**	**0.927**	**0.892**	**0.034**	**0.860**	0.755	**0.102**	**0.801**	**0.745**	**0.056**	0.856	0.800	**0.071**	**0.939**	0.886	**0.041**
MobileNet backbone																					
DAANet-C	**15.8**	DUTS-TR	10,553	0.834	0.770	0.051	0.908	0.876	0.042	0.835	0.739	0.121	0.777	0.718	0.062	0.841	0.768	0.081	0.921	0.875	0.051

Table 2: Quantitative comparison with DAANets (**DAANet-A**, **DAANet-B**, **DAANet-C**) and 12 other approaches on six benchmark datasets. The metrics including maximum F-measure $maxF_\beta$, mean F-measure mF_β, and MAE. We use different color to label the top three approaches on each benchmark, Red, Green, Blue indicate the best, the second best, and the third best model results, respectively.

5. Conclusions and Future Work

In this paper, we propose a pipeline with a dual attention aggregation module (DAAM) and boundary-joint training decoder for salient object detection. Our proposed DAANet is an encoder-decoder structure that uses an ImageNet pre-trained backbone as an encoder and adopts the FPN architecture with DAAM modules. The DAAM module can help the basic FPN structure capture more features and details to produce salient masks with higher accuracy. The BJD module can combine the salient boundary detection with the main task and achieve a good performance. We evaluate DAANet by comparing the prediction with different approaches qualitatively and prove the advantages of our method. The quantitative experimental results on six benchmark datasets illustrate that DAANet outperforms the other 12 approaches. Additionally, our lightweight approach can be easily deployed on mobile devices and other embedded artificial intelligence systems to satisfy the real-time inference requirements.

Acknowledgments

This research was conducted with the financial support of Science Foundation Ireland under Grant Agreement No. 13/RC/2106_P2 at the ADAPT SFI Research Centre at University College Dublin. ADAPT, the SFI Research Centre for AI-Driven Digital Content Technology, is funded by Science Foundation Ireland through the SFI Research Centres Programme.

References

[1] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, International Conference on Learning Representations (ICLR) (2014).

[2] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.

[3] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2117–2125.

[4] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), 2015, pp. 234–241.

[5] S. Woo, J. Park, J.-Y. Lee, I. S. Kweon, Cbam: Convolutional block attention module, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 3–19.

[6] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, M. Jagersand, Basnet: Boundary-aware salient object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 7479–7489.

[7] R. S. Srivatsa, R. V. Babu, Salient object detection via objectness measure, in: IEEE International Conference on Image Processing (ICIP), 2015, pp. 4481–4485.

[8] Y. Wei, F. Wen, W. Zhu, J. Sun, Geodesic saliency using background priors, in: Proceedings of the European Conference on Computer Vision (ECCV), 2012, pp. 29–42.

[9] C. Yang, L. Zhang, H. Lu, X. Ruan, M.-H. Yang, Saliency detection via graph-based manifold ranking, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 3166–3173.

[10] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, S.-M. Hu, Global contrast based salient region detection, IEEE Transactions on Pattern Analysis and Machine Intelligence (IEEE T-PAMI) 37 (3) (2014) 569–582.

[11] G. Li, Y. Yu, Visual saliency based on multiscale deep features, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 5455–5463.

[12] R. Zhao, W. Ouyang, H. Li, X. Wang, Saliency detection by multi-context deep learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 1265–1274.

[13] L. Wang, H. Lu, X. Ruan, M.-H. Yang, Deep networks for saliency detection via local estimation and global search, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 3183–3192.

[14] L. Zhang, J. Dai, H. Lu, Y. He, G. Wang, A bi-directional message passing model for salient object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 1741–1750.

[15] X. Hu, L. Zhu, J. Qin, C.-W. Fu, P.-A. Heng, Recurrently aggregating deep features for salient object detection, in: Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), Vol. 32, 2018.

[16] S. Chen, X. Tan, B. Wang, X. Hu, Reverse attention for salient object detection, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 234–250.

[17] X. Li, F. Yang, H. Cheng, W. Liu, D. Shen, Contour knowledge transfer for salient object detection, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 355–370.

[18] P. Zhang, W. Liu, H. Lu, C. Shen, Salient object detection by lossless feature reflection, IEEE Transactions on Image Processing (IEEE TIP) (2018).

[19] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 3431–3440.

[20] T. Wang, L. Zhang, S. Wang, H. Lu, G. Yang, X. Ruan, A. Borji, Detect globally, refine locally: A novel approach to saliency detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 3127–3135.

[21] N. Liu, J. Han, M.-H. Yang, PiCANet: Learning Pixel-wise Contextual Attention for Saliency Detection Supplemental Material, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 3089–3098.

[22] X. Zhang, T. Wang, J. Qi, H. Lu, G. Wang, Progressive attention guided recurrent network for salient object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 714–722.

[23] M. Feng, H. Lu, E. Ding, Attentive feedback network for boundary-aware salient object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 1623–1632.

[24] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.-C. Chen, Mobilenetv2: Inverted residuals and linear bottlenecks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 4510–4520.

[25] L. Wang, H. Lu, Y. Wang, M. Feng, D. Wang, B. Yin, X. Ruan, Learning to detect salient objects with image-level supervision, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 136–145.

[26] V. Movahedi, J. H. Elder, Design and perceptual validation of performance measures for salient object segmentation, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops), 2010, pp. 49–56.

[27] Q. Yan, L. Xu, J. Shi, J. Jia, Hierarchical saliency detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 1155–1162.

[28] Y. Li, X. Hou, C. Koch, J. M. Rehg, A. L. Yuille, The secrets of salient object segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 280–287.

[29] F. Perazzi, P. Krähenbühl, Y. Pritch, A. Hornung, Saliency filters: Contrast based filtering for salient region detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012, pp. 733–740.

[30] N. Liu, J. Han, M.-H. Yang, Picanet: Learning pixel-wise contextual attention for saliency detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 3089–3098.

[31] Z. Deng, X. Hu, L. Zhu, X. Xu, J. Qin, G. Han, P.-A. Heng, R^3Net: Recurrent residual refinement network for saliency detection, in: International Joint Conference on Artificial Intelligence (IJCAI), 2018.

[32] T. Wang, A. Borji, L. Zhang, P. Zhang, H. Lu, A stagewise refinement model for detecting salient objects in images, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, pp. 4019–4028.

[33] P. Zhang, D. Wang, H. Lu, H. Wang, B. Yin, Learning uncertain convolutional features for accurate saliency detection, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, pp. 212–221.

[34] P. Zhang, D. Wang, H. Lu, H. Wang, X. Ruan, Amulet: Aggregating multi-level convolutional features for salient object detection, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, pp. 202–211.

[35] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, P. Torr, Deeply supervised salient object detection with short connections, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 5300–5309.

[36] M. Feng, H. Lu, E. Ding, Attentive Feedback Network for Boundary-aware Salient Object Detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

[37] S. Chen, X. Tan, B. Wang, X. Hu, Reverse Attention for Salient Object Detection, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018.