Saliency Detection via Combining Region-Level and Pixel-Level Predictions with CNNs

Youbao Tang and Xiangqian Wu(✉)

Harbin Institute of Technology, Harbin 150001, China
{tangyoubao,xqwu}@hit.edu.cn

Abstract. This paper proposes a novel saliency detection method by combining region-level saliency estimation and pixel-level saliency prediction with CNNs (denoted as CRPSD). For pixel-level saliency prediction, a fully convolutional neural network (called pixel-level CNN) is constructed by modifying the VGGNet architecture to perform multi-scale feature learning, based on which an image-to-image prediction is conducted to accomplish the pixel-level saliency detection. For region-level saliency estimation, an adaptive superpixel based region generation technique is first designed to partition an image into regions, based on which the region-level saliency is estimated by using a CNN model (called region-level CNN). The pixel-level and region-level saliencies are fused to form the final salient map by using another CNN (called fusion CNN). And the pixel-level CNN and fusion CNN are jointly learned. Extensive quantitative and qualitative experiments on four public benchmark datasets demonstrate that the proposed method greatly outperforms the state-of-the-art saliency detection approaches.

Keywords: Saliency detection · Convolutional neural network · Region-level saliency estimation · Pixel-level saliency prediction · Saliency fusion

1 Introduction

Visual saliency detection, which is an important and challenging task in computer vision, aims to highlight the most important object regions in an image. Numerous image processing applications incorporate the visual saliency to improve their performance, such as image segmentation [1] and cropping [2], object detection [3], and image retrieval [4], etc.

The main task of saliency detection is to extract discriminative features to represent the properties of pixels or regions and use machine learning algorithms to compute salient scores to measure their importances. A large number of saliency detection approaches [5–36] have been proposed by exploiting different salient cues recently. They can be roughly categorized as pixel based approaches and region based approaches. For the pixel based approaches, the local and global features, including edges [5], color difference [36], spatial information [6], distance transformation [30], and so on, are extracted from pixels for saliency detection.

B. Leibe et al. (Eds.): ECCV 2016, Part VIII, LNCS 9912, pp. 809–825, 2016.
DOI: 10.1007/978-3-319-46484-8_49

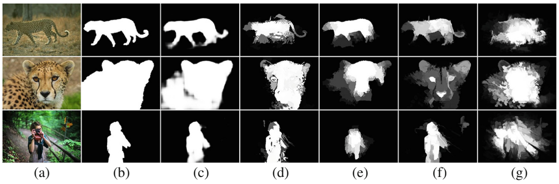


Fig. 1. Three examples of saliency detection results estimated by the proposed method and the state-of-the-art approaches. (a) The input images. (b) The ground truths. (c) The salient maps detected by the proposed method. (d)-(g) The salient maps detected by the state-of-the-art approaches MC [26], MDF [21], LEGS [28], and MB+ [30].

Generally, these approaches highlight high contrast edges instead of the salient objects, or get low contrast salient maps. That is because the extracted features are unable to capture the high-level and multi-scale information of pixels. As we know that convolutional neural network (CNN) is powerful for high-level and multi-scale feature learning and has been successfully used in many applications of computer vision, such as semantic segmentation [37,38], edge detection [39,40], etc. This work will employ CNN for pixel-level saliency detection.

For the region based approaches, they first segment an image into a number of regions, and then many different kinds of hand-designed features [7–10, 17,18,23,25,27,32–35] and CNN based features [21,26,28] are extracted to compute the salienies from these regions. Compared with the pixel based approaches, these regions based approaches are more effective to detect the saliency since more sophisticated and discriminative features can be extracted from regions. The approaches based on CNN learned features have gotten better performance than the ones based on hand-designed features. That is because CNN is able to extract more robust and discriminative features with considering the global context information of regions. Therefore, this work also employs CNN for region-level saliency estimation. Recently, the best region based saliency detection approach proposed by Zhao et al. [26] extracts superpixels as regions, then estimates the saliency for each superpixel based on CNN. In their work, an inevitable problem is that it is hard to decide the number of superpixels. If there are too few superpixels, the regions belonging to salient objects may be under-segmented. If there are too many superpixels, the regions belonging to saliency objects or backgrounds may be over-segmented, which may cause that the saliencies are not uniform in salient objects or backgrounds, and the superpixels around the boundaries of background and salient objects may get wrong saliencies. Furthermore, the number of superpixels should be different according to the complexity of images. In this paper, we follow their work and propose an adaptive superpixel based region generation technique, which can automatically determine the number of generated regions for different images to solve the above-mentioned problems and improve the performance of saliency detection.

Since pixel-level and region-level saliency detection approaches make use of different information of images, these two salient maps are complementary.

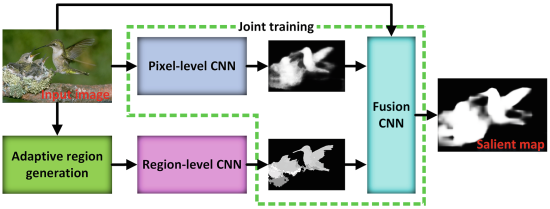


Fig. 2. The framework of the proposed method.

Hence, we propose a CNN network to fuse the pixel-level and the region-level saliencies to improve the performance. Figure 1 shows some results of the proposed method, which are very close to the ground truths.

Figure 2 shows the framework of proposed method, which consists of three stages, i.e. pixel-level saliency prediction, region-level saliency estimation, and the salient map fusion. For pixel-level saliency prediction, a pixel-level CNN is constructed by modifying the VGGNet [41] and finetuning from the pre-trained VGGNet model for pixel-level saliency prediction. For region-level saliency estimation, the input image is first segmented into a number of regions by using an adaptive superpixel based region generation technique. Then for each region, a salient score is estimated based on a region-level CNN. For salient map fusion, the pixel-level and region-level salient maps are fused to form the final salient map by using a fusion CNN which is jointly trained with the pixel-level CNN.

The main contributions of this paper are summarized as follows. (1) A novel multiple CNN framework is proposed to extract and combine pixel and region information of images for saliency detection. (2) A pixel-level CNN is devised for pixel-level saliency prediction. (3) An adaptive region generation technique is developed to generate regions and based on which a region-level CNN is used for region-level saliency estimation. (4) A fusion-level CNN is proposed to fuse the pixel-level and region-level saliencies.

2 Pixel-Level Saliency Prediction

CNN has achieved a great success in various applications of computer vision, such as classification and segmentation. Here, we proposed a CNN (denoted as pixel-level CNN) to predict the saliency for each pixel. Pixel-level CNN takes the original image as the input and the salient map as the output. To get an accurate saliency prediction, the CNN architecture should be deep and have multi-scale stages with different strides, so as to learn discriminative and multi-scale features for pixels. Training such a deep network from scratch is difficult when the training samples is not enough. However, there are several networks which have achieved the state-of-the-art results in the ImageNet challenge, such as VGGNet [41] and GoogleNet [42]. So it is an effective way to use these excellent models trained on the large-scale dataset as the pre-trained model for finetuning.

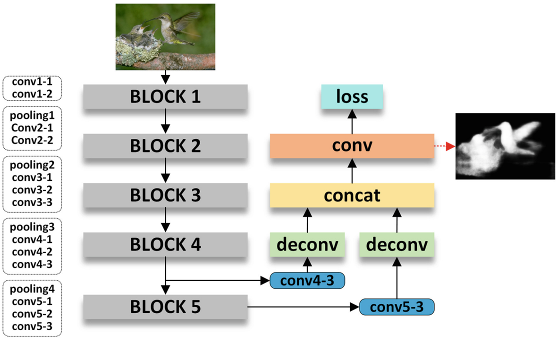


Fig. 3. The architecture of the pixel-level CNN network.

In this work, we construct a deep CNN architecture based on VGGNet for pixel-level saliency prediction. The VGGNet consists of six blocks. The first five blocks contain convolutional layers and pooling layers, as shown in Fig. 3. The last block contains one pooling layer and two fully connected layer, which are used to form the final feature vector for image classification. While for saliency prediction, we need to modify the VGGNet to extract dense pixel-level features. Therefore, the last block is removed in this work. There are two main reasons for this modification. The first one is that the fully connected layers cost much time and memory during training and testing. The second one is that the output of the last pooling layer is too small compared with the original image, which will reduce the accuracy of fullsize prediction. In order to capture the multi-scale information, we combine the outputs of the last two blocks of the modified VGGNet for the multi-scale feature learning. The benefits of doing such combination is two-fold. The first one is that the receptive field size becomes larger when the output size of blocks becomes smaller. Therefore, the output combination of multiple blocks can automatically learn the multi-scale features. The second one is that the shallow blocks mainly learn the local features, such as edges and parts of objects, which are not very useful for saliency detection since we hope to capture the global information of whole salient objects. Therefore, the outputs of the last two blocks are combined for multi-scale feature learning.

Since the output sizes of the last two blocks are different and smaller than the size of the input image. To make the whole CNN network automatically learn the multi-scale features for pixel-level saliency prediction, we first perform the deconvolutional operation for the outputs of the last two blocks to make them have the same size with the input image, and concatenate them in the channel direction. Then a convolutional kernel with size of 1 × 1 is used to map the concatenation feature maps into a probability map, in which larger values mean more saliencies. For testing, the probability map actually is a salient map of the input image. For training, a loss function is needed to compute the errors between the probability map and the ground truth. For most of the images, the numbers of salient and non-salient pixels are heavily imbalanced. Therefore,

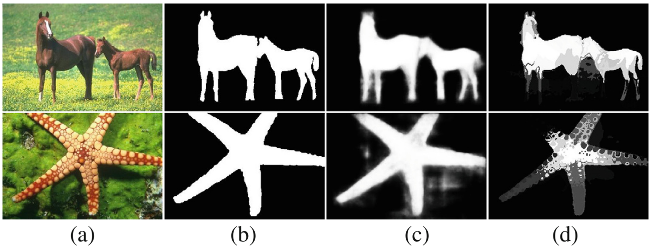


Fig. 4. Examples of pixel-level saliency prediction results. (a) Original images. (b) Ground truths. (c) Pixel-level saliency prediction results. (d) Salient maps estimated by the state-of-the-art approach MC [26].

given an image X and its ground truth Y, a cross-entropy loss function is used to balance the loss between salient and non-salient classes as follows:

$$L\left(\mathbf{W}\right) = -\alpha \sum_{i=1}^{|Y_+|} \log P\left(y_i = 1|X, \mathbf{W}\right) - (1-\alpha) \sum_{i=1}^{|Y_-|} \log P\left(y_i = 0|X, \mathbf{W}\right) \quad (1)$$

where $\alpha = |Y_-| / \left(|Y_+| + |Y_-|\right)$, $|Y_+|$ and $|Y_-|$ mean the number of salient pixels and non-salient pixels in ground truth, and $\mathbf{W}$ denotes the parameters of all network layers. Here and now, the whole pixel-level CNN architecture is constructed as shown in Fig. 3. The standard stochastic gradient descent algorithm is used to minimize the above loss function during training. After training, given an image, we can use the trained CNN model to predict a pixel-level salient map. Figure 4 shows two examples of pixel-level saliency prediction results.

3 Region-Level Saliency Estimation

Inspired by the successful application of CNN in salient object detection [21,26, 28], all of which are based on regions (e.g. superpixels [26] and multi-scale regions [21]), this work also employs CNN for the region-level saliency estimation.

3.1 Adaptive Region Generation

During the region-level saliency estimation, the first step is to generate a number of regions from the input image. Wang et al. [28] use the regions in sliding windows to estimate their saliencies, which may result in the salient object and background in the same sliding window having the same saliency. Li et al. [21] use multi-scale hierarchical regions, which consumes much time to perform the region segmentation and some generated regions are under-segmented. Zhao et al. [26] use superpixels as the regions to estimate their saliencies, which is difficult to decide the number of superpixels. If there are too few superpixels, the regions belonging to salient objects may be under-segmented. If there are too many superpixels, the regions belonging to saliency objects or backgrounds

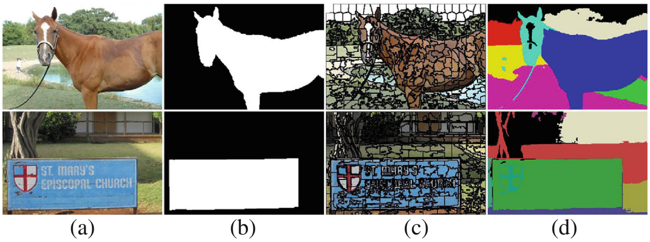


Fig. 5. Examples of our adaptive region generation technique. (a) Original images. (b) Ground truths. (c) Superpixel segmentation results. (d) Region generation results.

may be over-segmented. Both over-segmentation and under-segmentation may make the saliencies are not uniform in salient objects or backgrounds. Different images should be segmented into different number of superpixels because of their different properties.

Since the superpixels based approach [26] gets the state-of-the-art performance, this work proposes an adaptive region generation technique based on this approach to segment the images and solve the above mentioned problems.

Given an input image I, it is first over-segmented into n superpixels by using SLIC algorithm [43]. Here, we set $n = 300$ with considering both of effectiveness and efficiency. Then for each superpixel, a simple feature vector including its average colors in L*a*b color space and average spatial coordinates is computed. Then a graph-based agglomerative clustering algorithm (called Graph Degree Linkage) [44], which takes the superpixel as nodes and assigns each node with k edges whose weights are computed according to the Euclidean distances between the feature vectors of the current node and its k nearest neighbor nodes, is used to cluster the superpixels into different regions. The clustering process is stopped when the least affinity between two clusters is larger than a given threshold t. Therefore, for different images, the numbers of clustered regions are different and are much less than n. The superpixels which are adjacent and have similar colors are usually clustered into the same regions. The whole clustering process has two important parameters k and t, which are set as $k = 15$ and $t = -0.04$ through experiments in this work. Figure 5 shows two examples of region generation results.

3.2 Region Saliency Estimation

After obtaining the regions, the next step is to estimate the regions saliencies. This work employs CNN for region-level saliency estimation. The Clarifai model [45], which is the winning model in the classification task of ImageNet 2013, is used as our CNN model as done by [26]. It contains five convolutional layers and two fully connected layers. For more detail information about this model, please refer to the reference [45]. In this work, we use the CNN model provided by the authors of [26] as the pre-trained model and finetune for the region-level saliency estimation.

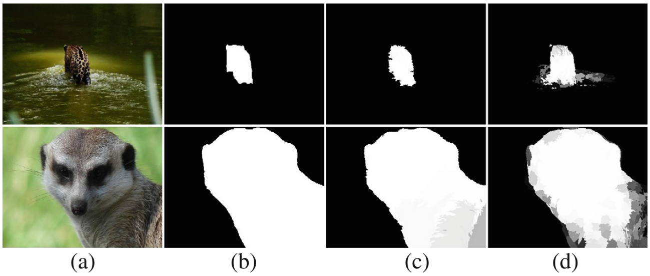


Fig. 6. Examples of region-level saliency estimation results. (a) Original images. (b) Ground truths. (c) Salient maps estimated by the proposed region-level saliency estimation method. (d) Salient maps estimated by superpixel based region saliency estimation method.

In [26], the region in a superpixel-centered large context window is resized and fed into the CNN model to estimate the saliency of current superpixel. If we follow the same way except using region-centered instead of superpixel-centered, a problem will be introduced, that is some background regions may have large saliencies, because the centers of some background regions may belong to or close to the salient objects. To solve this problem, we randomly choose m superpixels around the centerline of each region at first. Then we set these m superpixels centers as the windows centers to construct m large context windows including the full image as done by [26]. We choose superpixels around the regions centerline to make the windows centers far away from the regions boundaries as much as possible, and the constructed windows from different regions are different as much as possible. Here, we set $m = 5$ if the number of superpixels in a region is larger than 5. Otherwise, we set m as the number of superpixels. Through experiments, we find that the performances of saliency detection vary little when $m > 5$.

For each region, we can construct m window images and feed them into the CNN model to obtain m saliencies. In this work, the mean saliency is computed as the regions saliency due to its robustness to noises. Compared with the superpixel-centered saliency estimation approach, the proposed region-level saliency estimation method has three advantages described as follows. (1) More efficiency, because the constructed images are much less than the superpixels. (2) Less boundary effect, which is that the salient regions around the boundaries of salient objects and backgrounds may have small saliencies while the background regions around the boundaries may have large saliencies, as shown in Fig. 6. (3) More uniform salient map, since the pixels in a region are assigned the same salient values, as shown in Fig. 6.

4 Salient Map Fusion

Given an input RGB image, the proposed saliency detection method efficiently produces two salient maps, i.e. region-level salient map and the pixel-level salient

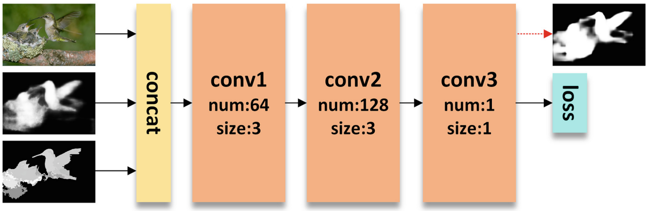


Fig. 7. The architecture of the fusion CNN network.

map. These two salient maps are computed by using different information of images, hence they are complementary and can be fused to further improve the performance.

There are many fusion strategies, such as establishing some measures to select a better individual salient map [11] or combining salient maps with weighted values [7]. They don't use the information of all salient maps or only linearly combine them. In this work, we sufficiently dig their complementary information with a nonlinear manner to improve the performance by using a CNN network. The CNN network contains one concatenation layer, three convolutional layers, and a loss layer, as shown in Fig. 7. The input image and its two salient maps are concatenated into a 5-channel image, and then through three convolutional layers whose configures are given in Fig. 7. For testing, the output of the last convolutional layer is the prediction salient map. For training, the loss layer is used to compute the errors between the output of the last convolutional layer and the ground truth with the cross-entropy loss function described before. It is needed to be noticed that the original image also is used for fusion except two salient maps. That's because richer information of original images is incorporated to correct some errors which cannot be solved by only using the salient maps.

The fusion CNN network can be trained separately. But as we know that joint training multiple sub-networks can gain the performance improvement. In this work, the region-level salient estimation needs to generate a number of regions at the beginning and the region-level CNN has a big different with the pixel-level CNN and fusion CNN. So it is hard to treat all of these three CNN network as an end-to-end network for joint training. Finally, the region-level CNN is trained alone, and after that, the pixel-level CNN and fusion CNN are jointly trained to get the final salient map as shown in Fig. 2. Based on the final salient maps, some post-processings, such as fully connected CRF [46], can be used to further improve the performance. But in this work, to focus on the performance of saliency detection models, we don't conduct any post-processing.

5 Experiments

5.1 Implementation

We use the popular Caffe library [47] to implement the proposed saliency detection framework. The THUS-10000 dataset [34] contains 10,000 images and their corresponding ground truths, which is used for CNN model training. For the region-level CNN network training, we use the Clarifai model trained by [26] as the pre-trained model to finetune on the training dataset. Before joint training the pixel-level CNN and fusion CNN network, we separately train them to get the initial models. For the pixel-level CNN network, since it is a fully convolutional network, arbitrary images don't need to be resized. And the weights of the first five blocks of VGGNet model trained on ImageNet are used to do the weight initialization, based on which the modified VGGNet is finetuned for pixel-level saliency prediction. For the fusion CNN network, we train the model from scratch. After obtaining the initial models of pixel-level and fusion CNN network, we use the weights of these models as weight initialization of the joint CNN network and use the training dataset to do the end-to-end training. The above training process costs about 49 h for 30,000 iterations on a PC with an Intel i7-4790k CPU, a TESLA k40c GPU, and 32 G RAM. For testing on an image with the size of 300×400, the region-level saliency estimation takes about 0.5 s, the process of pixel-level saliency prediction and saliency fusion takes about 0.38 s. Therefore, the whole process time of our saliency detection method is about 0.88 s.

5.2 Datasets and Evaluation Criteria

Datasets. We evaluate the proposed method on four standard benchmark datasets: SED [48], ECSSD [7], PASCAL-S [19], and HKU-IS [21].

SED [48] contains 200 images with one or two salient object, in which objects have largely different sizes and locations. This dataset is the combination of SED1 and SED2 dataset.

ECSSD [7] contains 1,000 images with complex backgrounds, which makes the detection tasks much more challenging.

PASCAL-S [19] is constructed on the validation set of the PASCAL VOC 2012 segmentation challenge. This dataset contains 850 natural images with multiple complex objects and cluttered backgrounds. The PASCAL-S data set is arguably one of the most challenging saliency data sets without various design biases (e.g., center bias and color contrast bias).

HKU-IS [21] contains 4447 challenging images, which is newly developed by considering at least one of the following criteria: (1) there are multiple disconnected salient objects, (2) at least one of the salient objects touches the image boundary, (3) the color contrast (the minimum Chi-square distance between the color histograms of any salient object and its surrounding regions) is less than 0.7.

All datasets provide the corresponding ground truths in the form of accurate pixel-wise human-marked labels for salient regions.

Evaluation Criteria. The standard precision-recall (PR) curves are used for performance evaluation. Precision corresponds to the percentage of salient pixels correctly assigned, while recall corresponds to the fraction of detected salient pixels in relation to the ground truth number of salient pixels. The PR curves are obtained by binarizing the saliency map in the range of 0 and 255. The F-measure (F_β) is the overall performance measurement computed by the weighted harmonic of precision and recall:

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

where we set $\beta^2 = 0.3$, as done by other approaches.

The mean absolute error (MAE), which is the average per-pixel difference between the ground truth GT and the saliency map S, is also evaluated. Here, GT and S are normalized to the interval [0, 1]. MAE is defined as

$$MAE = \frac{\sum_{x=1}^{W} \sum_{y=1}^{H} \left| S\left(x,y\right) - GT\left(x,y\right) \right|}{W \times H} \tag{3}$$

where W and H are the width and height of the image.

We also adopt the weighted F_β metric [49] (denoted as wF_β) for evaluation, which suffers less from curve interpolation flaw, improper assumptions about the independence between pixels, and equal importance assignment to all errors. We use the code and the default setting of wF_β provided by the authors of [49].

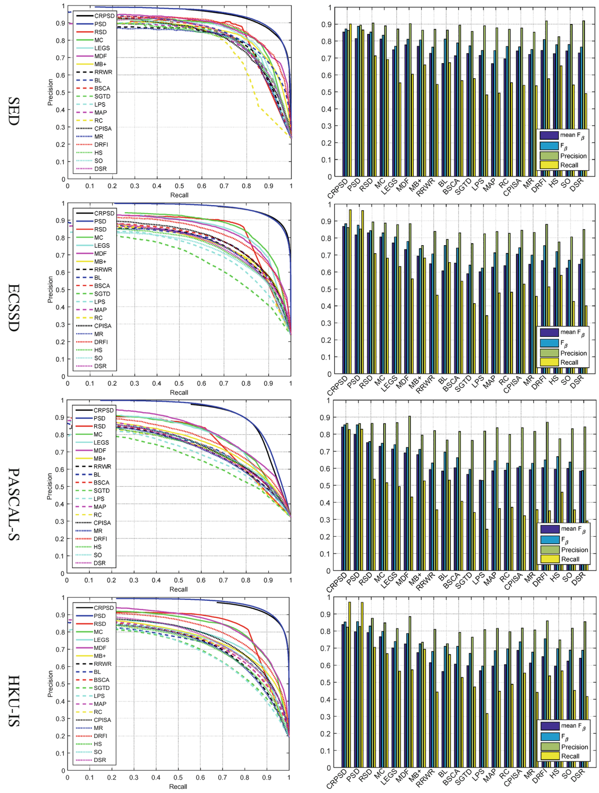
5.3 Performance Comparisons with State-of-the-Art Approaches

We compare the proposed method (denoted as CRPSD) and the two submodules (pixel-level saliency prediction, denoted as PSD, and region-level saliency estimation, denoted as RSD) with seventeen existing state-of-the-art saliency detection approaches on four datasets, including MC [26], MDF [21], LEGS [28], CPISA [31], MB+ [30], SO [17], BSCA [25], DRFI [10], DSR [9], LPS [32], MAP [33], MR [8], RC [34], RRWR [27], SGTD [35], BL [23], and HS [7]. For fair comparison, the source codes of these state-of-the-art approaches released by the authors are used for test with recommended parameter settings in this work.

According to Fig. 8 and Table 1, the proposed method (CRPSD) significantly outperforms all of the state-of-the-art approaches on all test datasets in terms of all evaluation criterions, which convincingly demonstrates the effectiveness of the proposed method. In these four test datasets, the most complex one is PASCAL-S. Therefore, all methods get the worst performance on this dataset. For all datasets, our method gets the largest gain on PASCAL-S dataset compared with the best state-of-the-art approach (MC) or our PSD, which demonstrates that our method can better deal with the complex cases than other approaches.

From the experimental results, three benefits of our method can be confirmed. (1) Although only the submodule region-level saliency estimation is used, it still gets the best performance compared with the state-of-the-art approaches on four datasets. Compared with MC [26], the RSD estimates the region saliency based on the regions generated by the proposed adaptive region generation technique while MC is based on superpixels, and the RSD uses a different strategy to form the context windows. The good performance of the RSD demonstrates the effectiveness of these improvements. (2) The submodule PSD also gets the best performance compared with the state-of-the-art approaches, which validates that the pixel-level CNN modified from VGGNet can well extract the multi-scale deep features for pixels to decide its saliency. (3) The proposed CRPSD by using the fusion network and joint training with the pixel-level CNN network can greatly improve the performance of the submodules, which demonstrates that CRPSD can well dig the complementary information of saliencies estimated by RSD and PSD for saliency detection.

Also, we qualitatively compare the salient maps detected by different approaches, as shown in the first ten rows of Fig. 9. Obviously, the proposed method is able to highlight saliencies of salient objects and suppress the saliencies



(a)

(b)

Fig. 8. Results of all test approaches on four standard benchmark datasets, i.e. SED, ECSSD, PASCAL-S, and HKU-IS. (a) presents the PR curves, (b) presents the mean F_β and the adaptive F_β/precision/recall which are computed from the binary images obtained by using Otsu algorithm on the salient maps.

Table 1. The wF_β and MAE of different saliency detection method on different test datasets (red, blue, and green texts respectively indicate rank 1, 2, and 3).

Method	Year	SED		ECSSD		PASCAL-S		HKU-IS	
		wF_β	MAE	wF_β	MAE	wF_β	MAE	wF_β	MAE
CRPSD	/	0.8292	0.0509	0.8485	0.0455	0.7761	0.0636	0.8209	0.0431
PSD	/	0.7590	0.0758	0.7572	0.0798	0.7113	0.1057	0.7371	0.0693
RSD	/	0.7759	0.0922	0.7569	0.0915	0.6195	0.1338	0.7286	0.0813
MC	CVPR2015	0.7387	0.1032	0.7293	0.1019	0.6064	0.1422	0.6899	0.0914
LEGS	CVPR2015	0.6498	0.1279	0.6722	0.1256	0.5791	0.1593	0.5911	0.1301
MDF	CVPR2015	0.6748	0.1196	0.6194	0.1377	0.5386	0.1633	0.6135	0.1152
MB+	ICCV2015	0.6555	0.1364	0.5632	0.1717	0.5307	0.1964	0.5438	0.1497
RRWR	CVPR2015	0.6117	0.1547	0.5026	0.1850	0.4435	0.2262	0.4592	0.1719
BL	CVPR2015	0.4986	0.1887	0.4615	0.2178	0.4464	0.2478	0.4119	0.2136
BSCA	CVPR2015	0.5671	0.1576	0.5159	0.1832	0.4703	0.2220	0.4643	0.1760
SGTD	TIP2015	0.6216	0.1475	0.4689	0.2007	0.4385	0.2269	0.4785	0.1627
LPS	TIP2015	0.5976	0.1477	0.4585	0.1877	0.3882	0.2162	0.4252	0.1635
MAP	TIP2015	0.5567	0.1621	0.4953	0.1861	0.4361	0.2222	0.4533	0.1717
RC	TPAMI2015	0.5652	0.1588	0.5118	0.1868	0.4694	0.2253	0.4768	0.1714
CPISA	TIP2015	0.6174	0.1474	0.5735	0.1596	0.4478	0.1983	0.5575	0.1374
MR	CVPR2013	0.6052	0.1586	0.4985	0.1875	0.4406	0.2288	0.4556	0.1740
DRFI	CVPR2013	0.6464	0.1360	0.5433	0.1658	0.4817	0.2042	0.5180	0.1444
HS	CVPR2013	0.5828	0.1948	0.4571	0.2283	0.4516	0.2625	0.4213	0.2151
SO	CVPR2014	0.6568	0.1351	0.5134	0.1733	0.4723	0.1986	0.5162	0.1426
DSR	ICCV2013	0.6055	0.1476	0.5162	0.1728	0.4385	0.2043	0.5079	0.1429

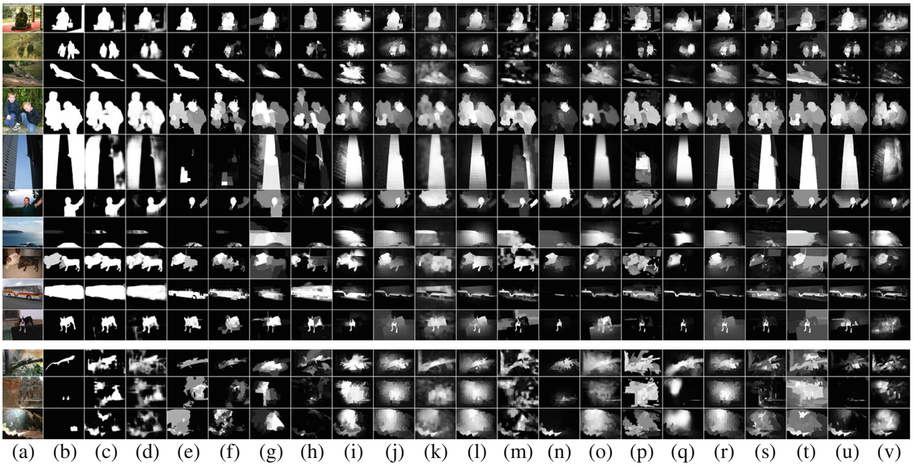


Fig. 9. Visual Comparisons of different saliency detection approaches in various challenging scenarios. (a) Original images, (b) Ground truths, (c) CRPSD, (d) PSD, (e) RSD, (f) MC, (g) LEGS, (h) MDF, (i) MB+, (j) RRWR, (k) BL, (l) BSCA, (m) SGTD, (n) LPS, (o) MAP, (p) RC, (q) CPISA, (r) MR, (s) DRFI, (t) HS, (u) SO, (v) DSR.

of background better than other approaches, and the salient maps of the proposed method are much close to the ground truths in various challenging scenarios.

The last three rows of Fig. 9 show some cases in which the proposed method fails. For example, the colors of salient objects and backgrounds are very similar, the salient objects are too small, and the backgrounds are too complex. In these cases, the other approaches also cannot correctly detect the salient objects and it is not easy to accurately locate the salient objects even for human eyes.

5.4 Performance Comparisons with Baselines

As pixel labeling task, saliency detection and semantic segmentation are very similar. And recently, many CNN models [37,38,50] have been proposed for semantic segmentation. In order to test their performance on saliency detection, the most powerful model of deeplab [50], i.e. the DeepLab-MSc-LargeFOV model (DML), is chosen as a baseline, which is trained on THUS-10000 dataset for saliency detection. And its pretrained DeepLab-LargeFOV-COCO-MSC model (pre-DML) on semantic image segmentation is used as another baseline, which is directly used for saliency detection by summing up the probability predictions across all 20 object classes and using these sumed-up probabilities as a salient map. And to demonstrate the benefit of joint training of our method, we also test the performance of our method with separate training (sep-CRPSD).

Table 2. The wF_β of baselines and our methods on all test datasets.

Method	SED	ECSSD	PASCAL-S	HKU-IS
pre-DML	0.5140	0.6530	0.7322	0.6755
DML	0.7439	0.7482	0.6948	0.7258
sep-CRPSD	0.8109	0.8249	0.7621	0.7942
CRPSD	**0.8292**	**0.8485**	**0.7761**	**0.8209**

Table 2 lists the wF_β of baselines and our methods on all test datasets. According to Table 2, three conclusions can be summarized: (1) The performance of pre-DML is very good on PASCAL-S, while dramatically drops on other datasets. Because many salient objects in other datasets don't belong to the trained classes, and hence are considered as non-salient objects during saliency detection. (2) The DML trained for saliency detection gets better results than pre-DML on all datasets except PASCAL-S, but still much worse than our method, which further demonstrates that our method with multiple CNNs is powerful for saliency detection. (3) Our method with joint training (CRPSD) gets better performance than separate training (sep-CRPSD), which demonstrates the effectiveness of joint training.

Table 3. The mean shuffled-AUC of different fixation prediection methods on test datasets.

Dataset	PSD	Mr-CNN [51]	SDAE [55]	BMS [54]
MIT	**0.7587**	0.7184	0.7095	0.7105
Toronto	**0.7606**	0.7221	0.7230	0.7243

5.5 Performance of Fixation Prediction with Pixel-Level CNN

The model (PSD) for pixel-level saliency prediction also can be used for fixation prediction. To validate its performance for fixation prediction, we use the same experimental setting with Mr-CNN [51] to test our model on MIT [52] and Toronto [53] datasets. The evaluation metric is mean shuffled-AUC [54]. Table 3 lists the experimental results of our model and the other three state-of-the-art fixation prediction approaches on these two datasets. According to Table 3, PSD gets the best performance, which means that our model has powerful ability of fixation prediction. Above experimental results further demonstrate the effectiveness of our pixel-level CNN model.

6 Conclusions

This paper proposes a novel saliency detection method by combining region-level saliency estimation and pixel-level saliency prediction (denoted as CRPSD). A multiple CNN framework, composed of pixel-level CNN, region-level CNN and fusion CNN, is proposed for saliency detection. The pixel-level CNN, which is a modification of VGGNet, can predict the saliency at pixel-level by extracting multi-scale features of images. The region-level CNN can effectively estimate the saliencies of these regions generated by the proposed adaptive region generation technique. The fusion CNN can take full advantage of the original image, the pixel-level and region-level saliencies for final saliency detection. The proposed method can effectively detect the salient maps of images in various scenarios and greatly outperform the state-of-the-art saliency detection approaches.

Acknowledgments. This work was supported by the Natural Science Foundation of China under Grant 61472102. The authors would like to thank the founders of the publicly available datasets and the support of NVIDIA Corporation with the donation of the Tesla K40 GPU used for this research.

References

1. Jung, C., Kim, C.: A unified spectral-domain approach for saliency detection and its application to automatic object segmentation. IEEE Trans. Image Process. **21**(3), 1272–1283 (2012)
2. Rother, C., Bordeaux, L., Hamadi, Y., Blake, A.: Autocollage. ACM Trans. Graph. **25**(3), 847–852 (2006)

3. Luo, P., Tian, Y., Wang, X., Tang, X.: Switchable deep network for pedestrian detection. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 899–906 (2014)
4. Gao, Y., Wang, M., Zha, Z.J., Shen, J., Li, X., Wu, X.: Visual-textual joint relevance learning for tag-based social image search. IEEE Trans. Image Process. **22**(1), 363–376 (2013)
5. Rosin, P.L.: A simple method for detecting salient regions. Pattern Recogn. **42**(11), 2363–2371 (2009)
6. Liu, T., Yuan, Z., Sun, J., Wang, J., Zheng, N., Tang, X., Shum, H.Y.: Learning to detect a salient object. IEEE Trans. Pattern Anal. Mach. Intell. **33**(2), 353–367 (2011)
7. Yan, Q., Xu, L., Shi, J., Jia, J.: Hierarchical saliency detection. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1155–1162 (2013)
8. Yang, C., Zhang, L., Lu, H., Ruan, X., Yang, M.H.: Saliency detection via graph-based manifold ranking. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3166–3173 (2013)
9. Li, X., Lu, H., Zhang, L., Ruan, X., Yang, M.H.: Saliency detection via dense and sparse reconstruction. In: International Conference on Computer Vision, pp. 2976–2983 (2013)
10. Jiang, H., Wang, J., Yuan, Z., Wu, Y., Zheng, N., Li, S.: Salient object detection: A discriminative regional feature integration approach. In: International Conference on Computer Vision, pp. 2083–2090 (2013)
11. Cheng, M.M., Warrell, J., Lin, W.Y., Zheng, S., Vineet, V., Crook, N.: Efficient salient region detection with soft image abstraction. In: International Conference on Computer Vision, pp. 1529–1536 (2013)
12. Zhang, J., Sclaroff, S.: Saliency detection: a boolean map approach. In: International Conference on Computer Vision, pp. 153–160 (2013)
13. Jiang, B., Zhang, L., Lu, H., Yang, C., Yang, M.H.: Saliency detection via absorbing markov chain. In: International Conference on Computer Vision, pp. 1665–1672 (2013)
14. Li, X., Li, Y., Shen, C., Dick, A., Van Den Hengel, A.: Contextual hypergraph modeling for salient object detection. In: International Conference on Computer Vision, pp. 3328–3335 (2013)
15. Liu, R., Cao, J., Lin, Z., Shan, S.: Adaptive partial differential equation learning for visual saliency detection. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3866–3873 (2014)
16. Lu, S., Mahadevan, V., Vasconcelos, N.: Learning optimal seeds for diffusion-based salient object detection. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2790–2797 (2014)
17. Zhu, W., Liang, S., Wei, Y., Sun, J.: Saliency optimization from robust background detection. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2814–2821 (2014)
18. Kim, J., Han, D., Tai, Y.W., Kim, J.: Salient region detection via high-dimensional color transform. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 883–890 (2014)
19. Li, Y., Hou, X., Koch, C., Rehg, J.M., Yuille, A.L.: The secrets of salient object segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 280–287 (2014)
20. Tang, Y., Wu, X., Bu, W.: Saliency detection based on graph-structural agglomerative clustering. In: ACM International Conference on Multimedia, pp. 1083–1086 (2015)

21. Li, G., Yu, Y.: Visual saliency based on multiscale deep features. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 5455–5463 (2015)
22. Frintrop, S., Werner, T., Martin Garcia, G.: Traditional saliency reloaded: a good old model in new shape. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 82–90 (2015)
23. Tong, N., Lu, H., Ruan, X., Yang, M.H.: Salient object detection via bootstrap learning. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1884–1892 (2015)
24. Gong, C., Tao, D., Liu, W., Maybank, S.J., Fang, M., Fu, K., Yang, J.: Saliency propagation from simple to difficult. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2531–2539 (2015)
25. Qin, Y., Lu, H., Xu, Y., Wang, H.: Saliency detection via cellular automata. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 110–119 (2015)
26. Zhao, R., Ouyang, W., Li, H., Wang, X.: Saliency detection by multi-context deep learning. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1265–1274 (2015)
27. Li, C., Yuan, Y., Cai, W., Xia, Y., Dagan Feng, D.: Robust saliency detection via regularized random walks ranking. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2710–2717 (2015)
28. Wang, L., Lu, H., Ruan, X., Yang, M.H.: Deep networks for saliency detection via local estimation and global search. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3183–3192 (2015)
29. Li, N., Sun, B., Yu, J.: A weighted sparse coding framework for saliency detection. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 5216–5223 (2015)
30. Zhang, J., Sclaroff, S., Lin, Z., Shen, X., Price, B., Mech, R.: Minimum barrier salient object detection at 80 fps. In: International Conference on Computer Vision, pp. 1404–1412 (2015)
31. Wang, K., Lin, L., Lu, J., Li, C., Shi, K.: Pisa: Pixelwise image saliency by aggregating complementary appearance contrast measures with edge-preserving coherence. IEEE Trans. Image Process. **24**(10), 3019–3033 (2015)
32. Li, H., Lu, H., Lin, Z., Shen, X., Price, B.: Inner and inter label propagation: salient object detection in the wild. IEEE Trans. Image Process. **24**(10), 3176–3186 (2015)
33. Sun, J., Lu, H., Liu, X.: Saliency region detection based on markov absorption probabilities. IEEE Trans. Image Process. **24**(5), 1639–1649 (2015)
34. Cheng, M.M., Mitra, N.J., Huang, X., Torr, P.H., Hu, S.M.: Global contrast based salient region detection. IEEE Trans. Pattern Anal. Mach. Intell. **37**(3), 569–582 (2015)
35. Scharfenberger, C., Wong, A., Clausi, D.A.: Structure-guided statistical textural distinctiveness for salient region detection in natural images. IEEE Trans. Image Process. **24**(1), 457–470 (2015)
36. Achanta, R., Hemami, S., Estrada, F., Susstrunk, S.: Frequency-tuned salient region detection. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1597–1604. IEEE (2009)
37. Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C., Torr, P.H.: Conditional random fields as recurrent neural networks. In: International Conference on Computer Vision, pp. 1529–1537 (2015)
38. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)

39. Xie, S., Tu, Z.: Holistically-nested edge detection. In: International Conference on Computer Vision, pp. 1395–1403 (2015)
40. Bertasius, G., Shi, J., Torresani, L.: Deepedge: A multi-scale bifurcated deep network for top-down contour detection. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 4380–4389 (2015)
41. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint (2014). arXiv:1409.1556
42. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9 (2015)
43. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S.: Slic superpixels compared to state-of-the-art superpixel methods. IEEE Trans. Pattern Anal. Mach. Intell. **34**(11), 2274–2282 (2012)
44. Zhang, W., Wang, X., Zhao, D., Tang, X.: Graph degree linkage: agglomerative clustering on a directed graph. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7572, pp. 428–441. Springer, Heidelberg (2012). doi:10.1007/978-3-642-33718-5_31
45. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8689, pp. 818–833. Springer, Heidelberg (2014). doi:10.1007/978-3-319-10590-1_53
46. Koltun, V.: Efficient inference in fully connected crfs with gaussian edge potentials. Neural Inf. Process. Syst. (2011)
47. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. In: ACM International Conference on Multimedia, pp. 675–678. ACM (2014)
48. Alpert, S., Galun, M., Brandt, A., Basri, R.: Image segmentation by probabilistic bottom-up aggregation and cue integration. IEEE Trans. Pattern Anal. Mach. Intell. **34**(2), 315–327 (2012)
49. Margolin, R., Zelnik-Manor, L., Tal, A.: How to evaluate foreground maps? In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255 (2014)
50. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. arXiv preprint (2016). arXiv:1606.00915
51. Liu, N., Han, J., Zhang, D., Wen, S., Liu, T.: Predicting eye fixations using convolutional neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 362–370 (2015)
52. Judd, T., Ehinger, K., Durand, F., Torralba, A.: Learning to predict where humans look. In: International Conference on Computer Vision, pp. 2106–2113. IEEE (2009)
53. Bruce, N.D., Tsotsos, J.K.: Saliency, attention, and visual search: an information theoretic approach. J. Vis. **9**(3), 5–5 (2009)
54. Zhang, J., Sclaroff, S.: Exploiting surroundedness for saliency detection: a Boolean map approach. IEEE Trans. Pattern Anal. Mach. Intell. **38**(5), 889–902 (2016)
55. Han, J., Zhang, D., Wen, S., Guo, L., Liu, T., Li, X.: Two-stage learning to predict human eye fixations via SDAEs. IEEE Trans. Cybern. **46**(2), 487–498 (2016)