Semantic Prior Analysis for Salient Object Detection

Tam V. Nguyen, *Senior Member, IEEE,* Khanh Nguyen, Thanh-Toan Do

***Abstract*—Salient object detection aims to detect the main objects in the given image. In this paper, we proposed an approach that integrates semantic priors into the salient object detection process. The method first obtains an explicit saliency map that is refined by the explicit semantic priors learned from data. Then an implicit saliency map is constructed using a trained model that maps the implicit semantic priors embedded into superpixel features with the saliency values. Next, the fusion saliency map is computed by adaptively fusing both the explicit and implicit semantic maps. The final saliency map is eventually computed via the post-processing refinement step. Experimental results have demonstrated the effectiveness of the proposed method, particularly, it achieves competitive performance with the state-of-the-art baselines on three challenging datasets, namely, ECSSD, HKUIS, and iCoSeg.**

***Index Terms*—Salient Object Detection, Semantic Priors, Deep Networks.**

I. Introduction

The ultimate goal of salient object detection is to determine the salient objects which attract the attention of humans on the input image. This research problem has grown more and more popular from neuroscience to computer vision community. It also has been successfully adopted in many applications, for instance, image classification [1], video classification [2], attention re-targeting [3], image resizing [4], and targeted advertisement [5]. More applications of visual saliency can be found in the extensive survey [6].

There exist many efforts towards highly accurate salient object predictors, from handcrafted features such as global contrast [7], local contrast [8], [9], to image patches [10], [11]. Witnessing the advancements in the field, our motivation stems from a simple research question "why this object is considered more salient than other objects in the same image". In fact, the saliency values of the salient objects are decided by humans via ground-truth annotation. We start this work by looking back to the annotation process in the literature. From the dawn of the research problem, the aforementioned question was out of attention since the early datasets, *i.e.*, MSRA Salient Object Database [12] or MSRA1000 [7], were collected under a simple setting, namely containing images with one object. In that setting, the sole object is unarguably considered as the salient object. The challenging question is becoming critical when more complicated saliency datasets, ECSSD [13], [14] and HKUIS [15] are later introduced with one or multiple objects in an image with cluttered background. This leads us to analyze the difference between two main ground-truth collection methods in saliency detection, namely, the procedure of human fixation collection and the process of salient object labeling. In the former procedure, the ground-truth is captured as the fixation points and saccades when a viewer is displayed a stimulus (in the image form) within a few seconds with no task given. During such a short timespan, the viewer is not able to look at the whole image, instead, he/she is only able to fixate to certain image locations that instantly draw his/her attention. For the latter process, the ground-truth is in the form of object mask(s) labeled by averaging the input of many annotators. Undoubtedly the annotators were given much longer time to mark the pixels belonging to the salient object(s). In case of multiple objects, the annotator naturally identifies the *semantic label* of each object in the image and then decides which object should be marked as salient. This inspires us to connect the problem of salient object detection with the semantic segmentation research, *i.e.*, we proposed a framework to leverage the explicit and implicit maps for saliency detection. In the latter semantic segmentation problem, the semantic label of each single pixel is assigned based on a trained model (semantic parser) which maps the features of the pixel/superpixel with a particular semantic class label [16], [17], [18], [19].

Recently, along with the broad application of deep learning in semantic segmentation, deep models, *i.e.*, Convolutional Neural Network (CNN), and later Fully Convolutional Network (FCN), have been successfully adopted to produce more robust features than handcrafted ones for salient object detection. In particular, deep networks [20], [15], [21], [22] achieve substantially better results than previous state of the art. However, these works mainly focus on either changing the training data (transfer learning), or stacking more network layers. These works actually achieve a higher accuracy, however, the impact of the semantic information is not adequately studied. Thus, in this paper, we explicitly study the impact of semantic information into the problem of salient object detection. In particular, we propose the so-called *semantic priors* to construct the explicit and implicit semantic saliency maps in order to produce a high quality salient object detector. Note that this paper extends the previously proposed framework and provides additional insights, analysis, and evaluation introduced in our previous work [23]. The main contributions of this paper can be summarized as follows.

T. V. Nguyen is with the Department of Computer Science, University of Dayton, Dayton, OH, 45469 USA. Email: tamnguyen@udayton.edu.

K. Nguyen is with University of Information Technology. Email:khanhnd@uit.edu.vn

T. Do is with the Department of Computer Science, University of Liverpool, United Kingdom. Email:thanh-toan.do@liverpool.ac.uk

Manuscript received XXXX XX, XXXX; revised XXXX XX, XXXX.

- The semantic information is harnessed into the salient object detection process. We form the semantic information as semantic priors which perform competitively with state-of-the-art baselines.
- We demonstrate that our work is a general framework which can easily adopt state-of-the-art semantic parsers. In addition, our framework is integrated with the refinement step in order to recover the missing parts of salient objects.
- In addition, the proposed method is able to boost performance of the existing deep learning approaches to a higher bound via the saliency aggregation.
- Last but not least, we analyze the effectiveness of the proposed framework, *i.e.*, the effectiveness of different component semantic-driven saliency maps, and the failure cases in order to pave way to the future work.

The remainder of this paper is organized as follows. We briefly review and analyze related works in Section II. Then, we present in details our proposed framework in Section III. Next, we report the experimental results and discussions in Section IV. Finally, Section V presents the conclusions and future work.

II. Literature Review

In this section, we first review the state-of-the-art semantic parsers including the non-parametric methods and the recently proposed deep learning approaches. We then review various salient object detection models.

A. Semantic Parsers

Most of the early semantic parsers follow the non-parametric label transfer mechanism [16], [24]. Liu *et al.* [16] introduced an image parser based on the dense correspondence across images. Given a test image, the semantic label is then assigned to a pixel according to the majority of the reference pixels from the k similar images in the training set. However, referencing via pixel-wise correspondence is very complex and time-consuming. Thus, Tighe *et al.* [24] transferred labels at the superpixel level. The semantic label is assigned to a superpixel according to the majority of the reference superpixels from the k similar images. Later, Eigen and Fergus [25] further improved [24] by learning different weights to each feature descriptors in order to minimize the classification error. Yang *et al.* [26] drew attention to rare class exemplars. Another approach [27] adopted deep superpixel's features and thus achieved better superpixel classification results.

Recently, there emerges a deep learning method, namely fully convolutional network (FCN) [18], which refines the popular Convolutional Neural Networks (CNN) [28] with upsampling layers in order to predict the input pixel with a semantic class label. Zheng *et al.* [19] improve FCN by considering more factors such as probabilistic graphical models, *i.e.*, Conditional Random Field (CRF). As shown in [18], [19], the deep learning based methods outperform other non-parametric based ones. Most recently, He *et al.* [29] proposed the Mask RCNN method to tackle the semantic segmentation tasks. Mask RCNN performs the instance segmentation from the initial bounding boxes of detected objects.

B. Salient Object Detection

In the literature, the saliency detection approaches can be classified into low-level stimuli-driven attention and machine learning-based ones. The early **low-level stimuli-driven attention** approaches [30], [31] focused on the contrast of low-level features such as color, intensity, or orientation. Since human visual system is very sensitive to color, many approaches use local or global color contrast. Cheng *et al.* [32], [8] proposed a method to measure the local contrast of each image superpixel. Meanwhile, Achanta *et al.* [7] proposed a global contrast method to detect the salient object by computing color dissimilarities to the mean image color. There also exist various patch-based methods which estimate dissimilarity among image patches [10], [9]. Goferman *et al.* [10] found that the salient objects contain the images patches that are most different from others. Perazzi *et al.* [9] further embedded the spatial distribution to ensure that the salient objects are compact. Li et al. [33] modelled an image as a hypergraph that utilizes a set of hyperedges to capture the contextual properties of image pixels or regions. As a result, they cast the problem of salient object detection to the problem of finding salient vertices and hyperedges in the hypergraph. Nguyen *et al.* [36], [34] augmented the objectness hypotheses [35] with the compactness constraint in order to identify the salient objects. As discussed in [37], the resulting saliency maps of these methods are blurry with loss in details. In addition, they are likely to highlight image edges and noise.

Regarding the **machine learning-based** approaches, Liu *et al.* [12] trained a CRF model to predict saliency from the combination of multiscale contrast, center-surround histogram, and color spatial distribution. Jiang *et al.* [11], [38] proposed the discriminative regional feature integration-based method (DRFI) which maps the regional features with the saliency values. Meanwhile, Kim *et al.* [39] proposed detecting salient objects by learning the combinational weights of color coefficients in the high-dimensional color transform. However, the predicted saliency maps do not totally focus on the salient objects. Instead, they also highlight adjacent regions of salient objects. For saliency fusion, Le Meur *et al.* [40] combined the output of top 2 saliency models to improve the performance. Meanwhile, Nguyen *et al.* [41] fused the saliency maps from different saliency detectors based on the image visual similarity.

Recently, the salient object detectors are outnumbered by the **deep learning**-based approaches. Wang *et al.* [20] introduced deep networks for saliency detection via local estimation and global search. Meanwhile, Li *et al.* [15] detected visual saliency based on multiscale deep features. In another approach, Li *et al.* [21] proposed multi-task deep neural network model to tackle the problem. Wang *et al.* [22] presented a novel method for detecting saliency with Recurrent Fully Convolutional Networks. Wei et al. [42] propose a co-saliency deep model based on a fully convolutional network with group input and group output. Wang *et al.* [43] trained a classifier and a subspace projection to rank object proposals based on R-CNN features. Zhang *et al.* [44] aggregated multi-level convolutional features to achieve the state-of-the-art performance.

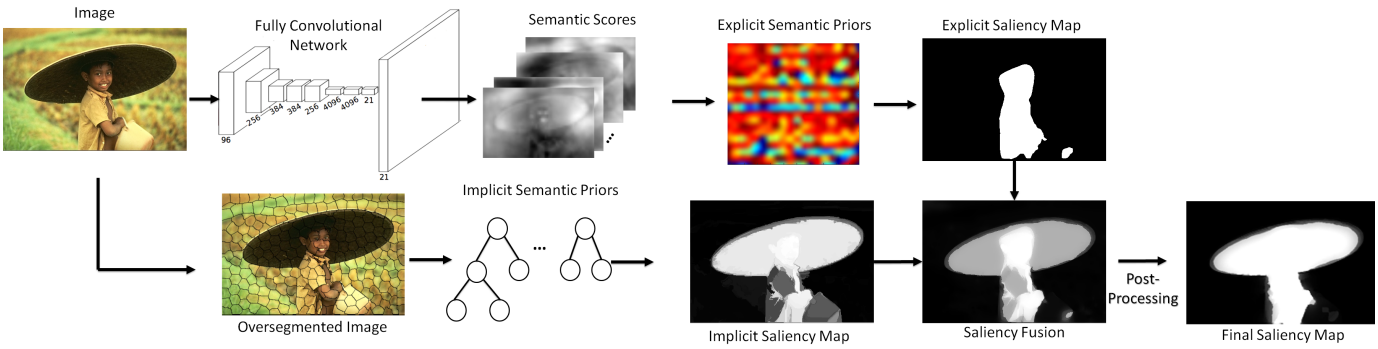


Fig. 1. The flowchart of the Semantic Priors (SP) based salient object detection framework with Fully Convolutional Network [18] used as the semantic parser: semantic scores from the semantic parser (Section III-A), the explicit map computation (Section III-B), the implicit map computation (Section III-C), adaptive saliency fusion (Section III-D), and post-processing step (Section III-E).

Luo *et al.* [45] added a boundary loss term to the typical cross entropy loss via the Mumford-Shah function in an end-to-end deep neural net framework. Hou *et al.* [46] proposed short connections between shallower and deeper side-output layers with a fully connected CRF for correcting wrong predictions. Hu et al. [47] introduced a deep network learning a level set function for salient objects. Although the most recent salient object detection methods [44], [15], [20], [21], [22], [45], [46], [47] utilize the deep networks to achieve a high accuracy rate, they do not clearly study the impact of semantic information. Therefore, in this paper, we explicitly investigate the importance of semantic information as the semantic priors into the task of salient object detection.

III. Proposed Framework

In this section, we introduce the our salient object detection framework in details. Figure 1 illustrates the flowchart of the proposed framework.

A. Semantic Extraction

As aforementioned, salient object detection and semantic segmentation are highly correlated but essentially different in the sense that salient object detection aims at distinguishing salient objects (foreground objects) from background, whereas semantic segmentation focuses on separating objects of different semantic classes. Here, SP can be considered as a general framework in the sense that we can employ any semantic parser. From the literature review, the end-to-end deep networks achieve the top performance in the semantic segmentation task. Therefore, we consider integrating the semantic parser such as the end-to-end deep networks, *i.e.*, [18], [19] into our proposed framework. In this context, "end-to-end" means that a complete semantic map $\mathbb{C}$ can be output from raw image pixels fed directly to the deep networks. In particular, we obtain the response score $\mathbb{C}_{x,y}$ for each single pixel (x, y) as below.

$$\mathbb{C}_{x,y} = \{\mathbb{C}^1_{x,y}, \mathbb{C}^2_{x,y}, \cdots, \mathbb{C}^{n_c}_{x,y}\}, \tag{1}$$

where $\mathbb{C}^1_{x,y}, \mathbb{C}^2_{x,y}, \cdots, \mathbb{C}^{n_c}_{x,y}$ indicate the likelihood that the pixel (x, y) belongs to the listed n_c semantic classes. Given an input image with size $h \times w$, the dimensionality of $\mathbb{C}$ is $h \times w \times n_c$.

B. Explicit Saliency Map

The explicit saliency map aims to capture the human commonsense on detecting salient objects. In other words, it targets to learn the preference of humans over different semantic classes such as 'person', 'car', or 'horse'. In particular, we aim to investigate which class is favoured by humans if there exist more than two semantic classes in the input image. From the response map $\mathbb{C}$ obtained from the previous step, we compute the class label $L_{x,y}$ of each single pixel (x, y) as:

$$\mathbb{L}_{x,y} = \arg\max \mathbb{C}_{x,y}. \tag{2}$$

$\mathbb{L}_{x,y}$ will be the index of the semantic class assigned to pixel (x, y).

In the training phase, given a ground-truth map $\mathbb{G}$, the density of each semantic class k in the input image is calculated by:

$$p_k = \frac{\sum_{x,y}(\mathbb{L}_{x,y} = k) \times \mathbb{G}_{x,y}}{\sum_{x,y}(\mathbb{L}_{x,y} = k)}, \tag{3}$$

where $(\mathbb{L}_{x,y} = k)$ is a boolean comparison which validates whether the assigned class index $\mathbb{L}_{x,y}$ equals k.

We define the *explicit semantic priors* as the accumulation of co-occurrence saliency pairwise of all classes. The explicit semantic priors of two classes k and t is defined as below:

$$sp^{Explicit}_{k,t} = \frac{\sum_{i=1}^{n_t} p^i_k \theta^i_{k,t}}{\sum_{i=1}^{n_t} \theta^i_{k,t} + \epsilon}, \tag{4}$$

where n_t is the number of images in the training set, ϵ is inserted to avoid the division by zero, and the pairwise value $\theta_{g,t}$ of any semantic class pair k and t is computed as below.

$$\theta_{k,t} = \begin{cases} 1 & , \exists \mathbb{L}_{x',y'} = k \wedge \mathbb{L}_{x'',y''} = t \\ 0 & , \text{otherwise} \end{cases}. \tag{5}$$

Note that we extract the co-occurrence saliency pairwise of one semantic class and other $n_c - 1$ classes from the training data.

In the testing phase, given a test image, the explicit saliency value of each single pixel (x, y) is computed as:

$$S_{x,y}^{Explicit} = \sum_{k=1}^{n_c} \sum_{t=1}^{n_c} (\mathbb{L}_{x,y} = k) \times \theta_{k,t} \times sp_{k,t}^{Explicit}. \quad (6)$$

C. Implicit Saliency Map

The previously computed explicit saliency map theoretically performs well in case the semantic labels of the detected objects present in the predefined class labels. Obviously, the explicit saliency map fails in case the expected salient objects are not in the n_c class labels. Therefore, we propose an additional map, namely implicit saliency map, which can uncover the salient objects not belonging to the listed semantic classes. To this end, we oversegment the input image into non-overlapping superpixels and extract the superpixel features. Unlike other methods which rely on regional superpixel features, here, we take the semantic information into account. In particular, we are interested in embedding the semantic-driven features into the superpixel features. Therefore, other than the off-the-shelf superpixel features, we also integrate two new features for each image superpixel, namely, local-and-global semantic features. The local semantic feature of each image superpixel q is defined as: $sp_1 = \frac{\sum_{x,y} \mathbb{C}_{x,y} \times (idx(x,y)=q)}{\sum_{x,y} (idx(x,y)=q)}$,, where $idx(x, y)$ is a function which returns the superpixel index of pixel (x, y). Note that $\mathbb{C}$ can be obtained via a semantic parser as mentioned in the Equation 1. Meanwhile, the global semantic feature is defined as:$sp_2 = \frac{\sum_{x,y} \mathbb{C}_{x,y}}{h \times w}$.

The semantic features $sp^{Implicit} = \{sp_1, sp_2\}$ are finally combined with other superpixel features. We consider the semantic features here as the *implicit semantic priors* since they implicitly affect the mapping of the superpixel features and saliency scores. Then, we train a regressor rf to estimate the saliency values of the aforementioned extracted superpixel features. In particular, we adopt the random forest regressor which reportedly performs well in [11].

In the training phase, we extract a set of n_r superpixels $\{\{r_1, sp_1^{Implicit}\}, \{r_2, sp_2^{Implicit}\}, \cdots, \{r_{n_r}, sp_{n_r}^{Implicit}\}\}$ from the training image set. For the training ground-truth label, we compute the corresponding saliency scores $\{s_1, s_2, \cdots, s_{n_r}\}$ via the following rule: if the number of pixels (in the superpixel) belonging to the salient object or the background exceeds 80% of the number of the pixels in the superpixel, its saliency value is set as 1 or 0, respectively.

In the testing phase, we first oversegment the input image into superpixels and extract the corresponding superpixel features. The implicit saliency value of each test image superpixel q is then computed by feeding the extracted features into the trained regressor rf:

$$S_q^{Implicit} = rf(\{r_q, sp_q^{Implicit}\}). \quad (7)$$

This process is run on all superpixels in order to form the implicit saliency map $S^{Implicit}$.

D. Saliency Fusion

Given an input image with a size $h \times w$, we compute the two aforementioned maps, explicit and implicit saliency maps. Then, the two maps are fused as follows. We scale the implicit saliency map $S^{Implicit}$, and the explicit saliency map $S^{Explicit}$, to the range [0..1]. Then we adaptively fuse these maps to compute a saliency value S^{Fusion} for each pixel:

$$S^{Fusion} = \alpha S^{Explicit} + (1 - \alpha) S^{Implicit}, \quad (8)$$

where the weight α, measuring how large the semantic pixels occupied in the image, is set as $\frac{\sum_{x,y} S_{x,y}^{Implicit}}{h \times w}$.

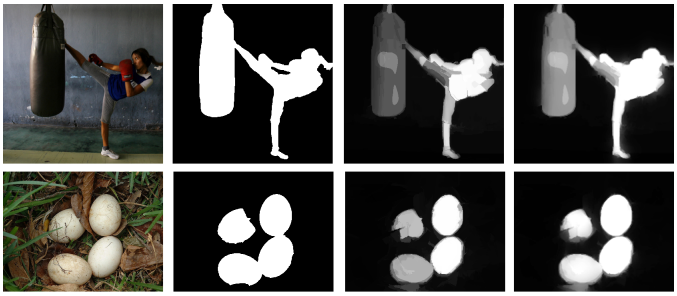


Fig. 2. The illustration of the post-processing step in the SP framework. From left to right: the original image, the ground truth map, the fused saliency map, and the final saliency map after the post-processing step. The final results recover and highlight some missing parts of the salient objects with sharp boundaries.

MS-COCO

PASCAL VOC						
Aeroplane	Dining table	Truck	Handbag	Tennis racket	Carrot	Microwave
Bicycle	Dog	Traffic light	Tie	Wine glass	Hot dog	Oven
Bird	Horse	Fire hydrant	Suitcase	Cup	Pizza	Toaster
Boat	Motorbike	Stop sign	Frisbee	Fork	Donut	Sink
Bottle	Person	Parking meter	Skis	Knife	Cake	Refrigerator
Bus	Potted plant	Bench	Snowboard	Spoon	Bed	Book
Car	Sheep	Elephant	Sports ball	Bowl	Toilet	Clock
Cat	Sofa (Couch)	Bear	Kite	Banana	Laptop	Vase
Chair	Train	Zebra	Baseball bat	Apple	Mouse	Scissors
Cow	TV monitor	Giraffe	Baseball glove	Sandwich	Remote	Teddy Bear
		Backpack	Skateboard	Orange	Keyboard	Hair Drier
		Umbrella	Surfboard	Broccoli	Cell Phone	Toothbrush

Fig. 3. The list of semantic classes in both PASCAL VOC and MS-COCO datasets.

E. Post-processing Refinement

We observe that the fused saliency map does not entirely cover the salient object(s) due to the imperfect oversegmentation. Therefore, we perform the post-processing step to compensate the superpixel imperfection. The dilated saliency value of each pixel (x, y) is computed as below:

$$S_{x,y} = \sigma_{spatial}(x, y) \times \sigma_{color}(c_{x,y}) \times \tilde{S}_{x,y}^{Fusion}. \quad (9)$$

Note that the former term $\sigma_{spatial}$ is used to simulate the central bias (the objects near the center of the image tend to be more salient than others) discussed in [48], [49], and the latter term σ_{color} is used to recover pixels with dominant color in the foreground region. In particular, $\sigma_{spatial}(x, y) = \exp(-\frac{\|[x,y]-[x_{center},y_{center}]\|_2}{\|[x_{center},y_{center}]\|_2})$ is the exponential function of the normalized distance from the pixel to the image center. Meanwhile, $\sigma_{color}(x, y) = \frac{n_f(c_{x,y})}{n_f}$, where n_f is the number of foreground pixels whose saliency values are larger than the mean value of the fused saliency map, and $n_f(c_{x,y})$ is the number of foreground pixels containing the color $c_{x,y}$ in Lab

TABLE I
THE DETAILED DESCRIPTORS OF THE SUPERPIXEL FEATURES

Feature Descriptors	Dimensionality
The average normalized coordinates	2
The bounding box location	4
The aspect ratio of the bounding box	1
The normalized perimeter	1
The normalized area	1
The normalized area of the neighbor superpixels	1
The variances of the RGB values	3
The variances of the Lab values	3
The variances of the HSV values	3
Textons [50]	15
The local semantic features sp_1	n_c
The global semantic features sp_2	n_c

color space. Note that Lab color space is found effective in salient object detection [7], [9]. In addition, $\tilde{S}^{Fusion}$ is the thresholded fused saliency map where saliency values smaller than the mean value of the fused saliency are set to 0.

Then, we apply the edge-preserving filters in [51] on the dilated saliency map. This aims to expand more pixels of salient objects while preserving edges. The resulting pixel-level saliency map recovers many missing parts of the salient objects but it may have an arbitrary scale. Therefore, in the final step, we rescale the saliency map S to the range [0..1] or to contain at least 10% saliency pixels. Figure 2 illustrates the results of the refinement step. The post-processing results actually recover and highlight some missing parts of the salient objects.

F. Implementation Details

For the implementation, we adopt the superpixel features (*e.g.*, the normalized area/perimeter/aspect ratio of the superpixel bounding box, the variances/means of color histograms (RGB, Lab, HSV), Textons [50], Local Binary Patterns [52], *etc.*) as listed in Table I. We consider various semantic parsers, namely, FCN [18] with 3 coarse-to-fine settings, 32s, 16s, 8s, FCN-CRF [19], and Mask RCNN [29] to perform the semantic segmentation for the input image. In particular, we utilize the FCNs and FCN-CRF models trained from the PASCAL VOC 2007 dataset [53] with 20 semantic classes and Mask RCNN trained on MS-COCO [54] with 80 semantic classes (and an additional 'unlabeled' class for a pixel is not classified as any aforementioned semantic classes). Figure 3 shows the list of semantic classes in both datasets. It is worth noting that all classes of PASCAL VOC are fully included in MS-COCO.

We trained the proposed framework on HKUIS dataset [15] (training part). For the image over-segmentation, we adopt SLIC method [55]. We set the number of superpixels to 200 which is a trade-off between the fine over-segmentation and the running time.

IV. EXPERIMENTAL RESULTS

A. Experimental Settings

1) Benchmark Datasets: For the evaluation, we compare the performances of our framework with previous baseline algorithms on three public benchmark datasets: ECSSD [13], iCoSeg [56], HKUIS [15] (testing part).

The ECSSD dataset contains $1,000$ images in each of which a complex background is presented. This dataset is introduced to overcome the simple setting in MSRA1000 [7] or MSRA Salient Object Database [12].

The HKUIS dataset has $5,447$ images in two sets, namely, training set and testing set with $4,000$ and $1,447$ images, respectively.

The iCoSeg dataset consists of 643 images. The dataset is collected under the setting where users first decide what foreground is, and then guide the co-segmentation algorithm [56] via scribbles.

Note that each image in all three datasets consists of single or multiple salient objects.

2) Evaluation Metrics: We evaluate the performance using Precision-Recall Curve (PRC), F-measure, and Mean Absolute Error (MAE).

The first evaluation metric is computed based on the overlapping area between obtained results and provided ground-truth. Using a fixed threshold between 0 and 255, the scores of $(Precision, Recall)$ pairs are computed and then combined to form a PRC. We also use the adaptive threshold proposed by [7], defined as twice the mean value of the saliency map S. The second metric, $F-measure$, is a balanced measurement between $Precision$ and $Recall$:

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

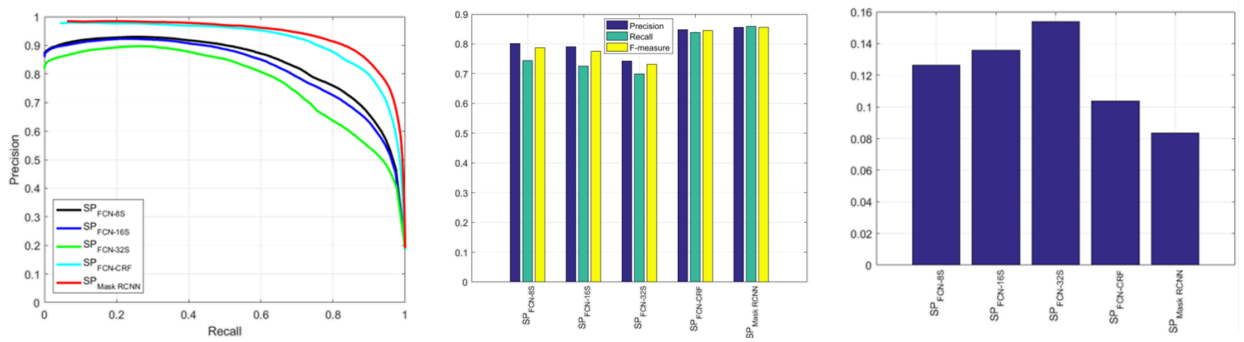
We use β^2 = 0.3 as suggested in [7], [9] to put an emphasis on precision. For the third evaluation, we compute the mean absolute error (MAE) between the predicted saliency map S and the binary ground truth G as:

$$MAE = \frac{1}{h \times w} \sum_{x,y} |S_{x,y} - G_{x,y}|. \tag{11}$$

B. Effectiveness of Semantic Parsers

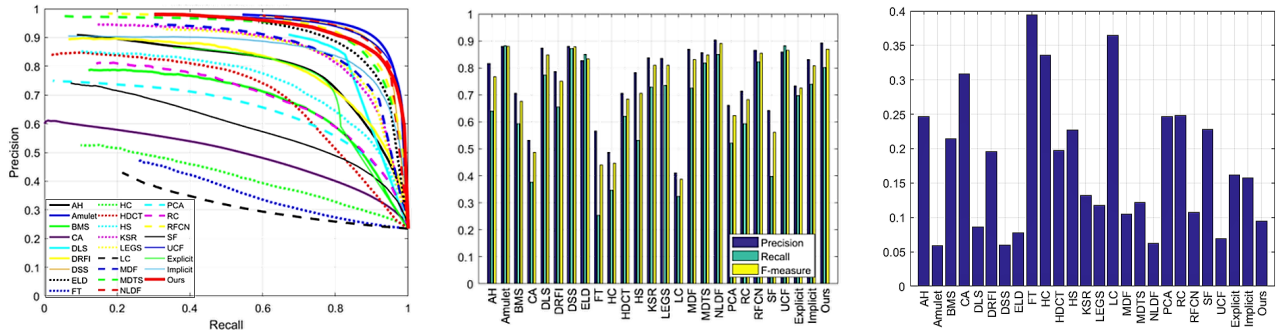
We first evaluate different semantic parsers for our proposed SP framework. As mentioned, we investigate different methods, *i.e.*, Fully Convolutional Network (FCN) [18] with three settings 'FCN-8S', 'FCN-16S', 'FCN-32S', FCN-CRF [19], and Mask RCNN [29]. Note that FCN and FCN-CRF yield the semantic probability maps for 20 semantic classes. Meanwhile, Mask RCNN produces the bounding boxes with the corresponding semantic class and the segmented object. We concatenate those segmented objects to form the final semantic probability maps. FCN and FCN-CRF are trained on PASCAL dataset whereas Mask RCNN is trained on MS-COCO dataset.

We sequentially integrate each parser into our framework and train the corresponding framework on HKUIS-training set. Then, we evaluate the performance of different parsers on HKUIS-testing set. The performance of different parsers is shown in Figure 4. Following [13], we first use a Precision-Recall Curve. Regarding the FCN family, the finer segmentation (8s) outperforms other coarser versions, *i.e.*, 16s and 32s. Meanwhile, FCN-CRF outperforms FCN thanks to the



(a) Fixed threshold (b) Adaptive threshold (c) Mean absolute error

Fig. 4. The performance comparison of our proposed method with different semantic parsers on HKUIS-testing set [15]: (a) the average precision recall curve with fixed thresholds, (b) the average precision recall by adaptive thresholding, and (c) the mean absolute error.



(a) Fixed threshold (b) Adaptive threshold (c) Mean absolute error

Fig. 5. The performance comparison of our proposed method with baselines on ECSSD dataset [13]: (a) the average precision recall curve with fixed thresholds, (b) the precision, recall, and F-measure under adaptive thresholding, and (c) the mean absolute error.

CRF refinement step. And $\mathrm{SP}_{\mathrm{MaskRCNN}}$ slightly improves $\mathrm{SP}_{\mathrm{FCN-CRF}}$ and reaches the highest Precision-Recall rate over all other versions. Regarding MAE, as shown in Figure 4c, the implementation of SP with Mask RCNN also achieves the lowest MAE. In other words, the saliency maps from $SP_{MaskRCNN}$ are closer to the ground truth map than others. Therefore, we adopt Mask RCNN as the semantic parser in our proposed framework for the successive experiments.

C. Comparison to State-of-the-art Methods

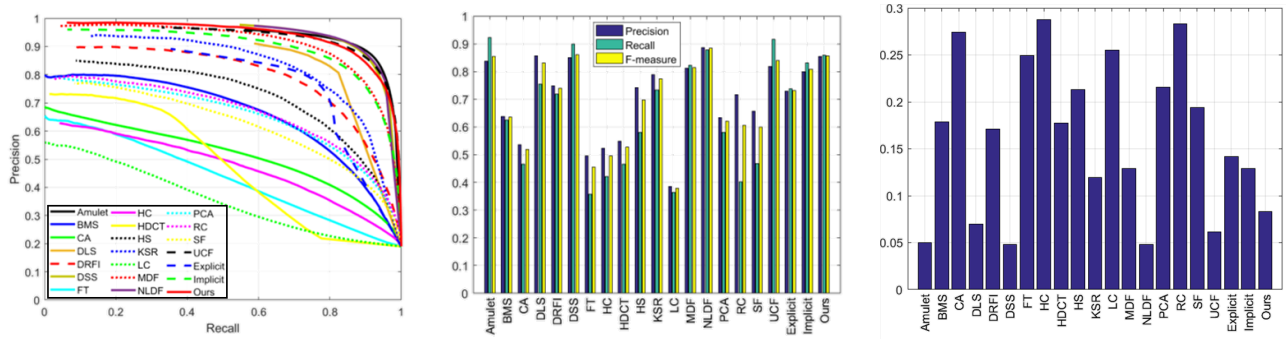
In this subsection, we evaluate our proposed SP framework with state-of-the-art methods on three challenging datasets, ECSSD, HKUIS, and iCoSeg.

1) Performance on ECSSD dataset: We compare our work with 23 state-of-the-art methods by running the approaches' publicly available source code or pre-computed results: augmented hypotheses (AH [34]), aggregating multi-level convolutional features (Amulet [44]), boolean map saliency (BMS [57]), context-aware saliency (CA [10]), deep level sets (DLS [47]), discriminative regional feature integration (DRFI [11]), deeply supervised salient object detection with short connection (DSS [46]), deep saliency with encoded low level distance map (ELD [58]) frequency-tuned saliency (FT [7]), global contrast saliency (HC and RC [32]), high-dimensional color transform (HDCT [39]), hierarchical saliency (HS [13]), kernelized subspace ranking (KSR [43]), spatial temporal cues (LC [59]), local estimation and global search (LEGS [20]), multiscale deep features (MDF [15]), multi-task deep saliency (MTDS [21]), non-local deep features (NLDF [45]), principal component analysis (PCA [60]), recurrent fully convolutional networks (RFCN [22]), saliency filters (SF [9]), uncertain convolutional features (UCF [61]). Among them, Amulet, DLS, DSS, ELD, KSR, LEGS, MDF, NLDF, MTDS, RFCN, and UCF are deep learning based methods.

As shown in Figure 5a, the machine learning-based methods outperform other low-level stimuli-driven attention baselines. In addition, the deep learning-based methods surpass the handcrafted feature-based methods [11], [39]. Meanwhile, our proposed method tops all handcrafted feature based baselines and achieves a competitive performance to deep learning methods. In particular, our method surpasses most of deep learning methods such as RFCN, ELD, KSR, LEGS. Meanwhile, the proposed method is slightly inferior to the most recent deep models such as Amulet and NLDF. Likewise, our method also obtains the good performance in terms of F-measure in the adaptive threshold setting (shown in Figure 5b). As shown in Figure 5c, our work achieves a good performance in terms of MAE (0.086). In Figure 6, we compare the saliency maps resulted from our method and baselines. Our results are clearly close to ground truth and cover the salient objects.

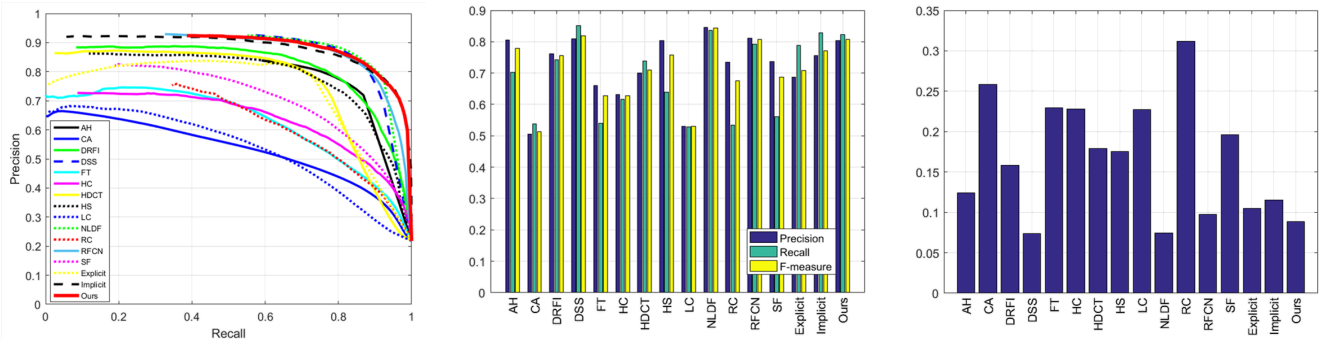
2) Performance on HKUIS dataset: We have 18 baselines running on this relatively new dataset. We first evaluate our method using a Precision-Recall Curve which is shown in

Fig. 6. The visual comparison of different salient object detection baselines. From left to right: (a) Original images, (b) ground truth, (c) our SP method, saliency baselines with (d) boolean map (BMS), (e) context-aware (CA), (f) discriminative regional feature integration (DRFI), (g) frequency-tuned (FT), (h) high-dimensional color transform (HDCT), (i) local estimation and global search (LEGS), (j) multiscale deep features (MDF), (k) multi-task deep saliency (MTDS), (l) principal component analysis (PCA), (m) contrast (RC), (n) saliency filters (SF).



(a) Fixed threshold (b) Adaptive threshold (c) Mean absolute error

Fig. 7. The performance comparison of our proposed method with baselines on HKUIS dataset [15]: (a) the average precision recall curve with fixed thresholds, (b) the precision, recall, and F-measure under adaptive thresholding, and (c) the mean absolute error.



(a) Fixed threshold (b) Adaptive threshold (c) Mean absolute error

Fig. 8. The performance comparison of our proposed method with baselines on iCoSeg dataset [56]: (a) the average precision recall curve with fixed thresholds, (b) the precision, recall, and F-measure under adaptive thresholding, and (c) the mean absolute error.

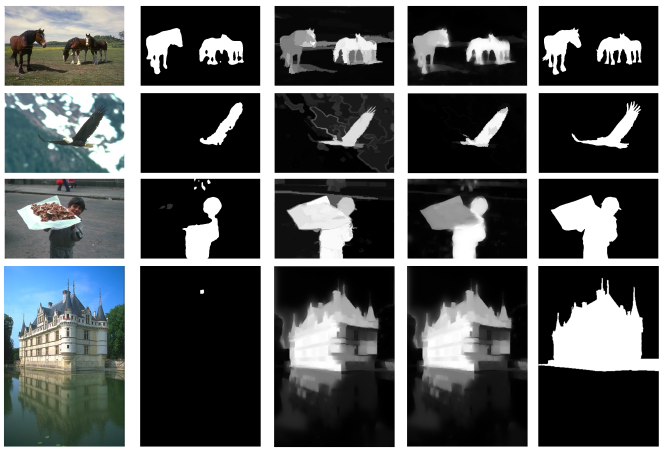


Fig. 9. The examples of the mutual collaboration of the explicit and implicit saliency maps. From left to right: the original image, the explicit saliency map, the implicit saliency map, our final saliency map, the ground-truth map. The two maps are complementary to each other. On the one hand, the explicit map removes the noise in the background from the implicit map in the first two rows. Meanwhile, the implicit map recovers the *tray* and the *building* in the last two rows. Note that the *tray* the and *building* are not included in either PASCAL VOC or MS-COCO semantic classes mentioned in Section III-F.

Figure 7a, b. We observe the similar pattern as in the previous subsection where the learning-based methods outperform the low-level feature baselines. Our method achieves competitive results to baselines. In particular, under adaptive threshold, our method tops all baselines in terms of precision and F-measure. As shown in Figure 7c, our method achieves the good performance in terms of MAE (< 0.1).

3) Performance on iCoSeg dataset: We finally evaluate the proposed framework on iCoSeg dataset. This dataset contains images with one or multiple salient objects. We compare our SP method with other 13 methods. As shown in Figure 8a and 8b, our work achieves the similar Precision-Recall rate as NLDF [45] and DSS [46], and high F-measure over baselines. Note that the baselines perform inconsistently on all three evaluation metrics. For example, HDCT [39] and DRFI [11] perform better over AH [34] on PR curve, however, AH achieves a better MAE rate. In addition, the deep learning methods such as NLDF and DSS do not achieve the high

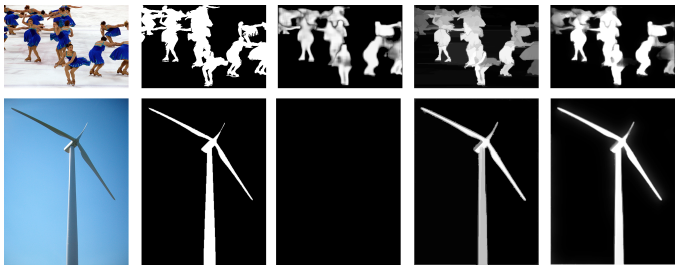


Fig. 10. From left to right: the original image, the ground truth map, the Explicit map, the Implicit map, and our final result. The Implicit map performs well due to the nature of the dataset.

performance as expected. This phenomenon will be discussed later in Section IV-F.

D. Impact of Explicit and Implicit Saliency Maps

We also evaluate the individual components in our system, namely, the explicit saliency map (Explicit), and the implicit saliency map (Implicit), in the three benchmarking datasets, ECSSD, HKUIS, and iCoSeg. As shown in Fig. 5, Fig. 7, and Fig. 8, the two components generally achieve the acceptable performance (in terms of precision, recall, F-measure and MAE) which is comparable to other baselines. Generally, the Explicit map outperforms Implicit map in terms of MAE, whereas Implicit map achieves a better performance in terms of F-measure. Therefore, the semantic information cannot be directly used for saliency detection and it is non-trivial to directly exploit the output of the semantic segmentation task for the salient object detection.

As shown in the previous subsection, the two individual maps are later fused and refined to produce the final saliency maps which surpass all baselines in three benchmark datasets. That combination demonstrates that these individual maps complement each other in our unified framework. Fig. 9 visualizes examples of the mutual collaboration of the explicit and implicit saliency maps.

In iCoSeg dataset, we notice that Implicit map performs really well and combines with Explicit map to reach a small gain. It can be explained by the nature of the dataset since many images in iCoSeg contain clean and clear background,

TABLE II
RUNTIME COMPARISON OF DIFFERENT METHODS.

Method	AH	RC	SF	DSS	NLDF	CA	DRFI	RFCN	Ours
Time (s)	0.07	0.25	0.15	0.07	0.88	51.2	10.0	5.19	4.75
Code	C++	C++	C++	Python-Caffe	Python-Tensorflow	Matlab	Matlab	Matlab	Matlab

Fig. 11. From left to right: the original image, our saliency prediction, and ground truth map. There exists no clear explanation that the football player in the red jersey is more salient than the footballers in white (the top row). Likewise, the pyramid at the background is more salient than the horse riding man in the foreground (the bottom row).

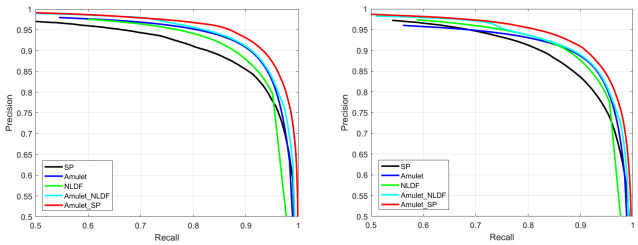
which makes the Implicit map well focuses on the main salient objects. Figure 10 illustrates some examples of iCoSeg dataset.

E. Time Efficiency

It is also worth investigating the time efficiency of different methods. In Table II, we compare the average running time for a typical 300×400 image of our approach to other methods. The average time is taken on a PC with Intel i7 2.6 GHz CPU and 8GB RAM with our unoptimized Matlab code. Actually there are three prominent types of implementations, MATLAB, C++, and Python with deep learning frameworks such as Caffe or Tensorflow. Basically, C++ implementation, *i.e.*, AH, RC, or SF, runs faster than the Matlab based code, *i.e.*, CA, DRFI. The CA method [10] is the slowest one because it requires an exhaustive nearest-neighbor search among patches. Meanwhile, deep learning-based methods are efficient since those methods are performed on an end-to-end manner. Our method is able to run faster than other Matlab based implementations. Our procedure spends most of the computation time on semantic segmentation and extracting regional features.

F. Failure Cases

We observe that the adaptive F-measure on iCoSeg benchmark is lower than those of ECSSD and HKUIS datasets



(a) PR curve on ECSSD (b) PR curve on HKUIS

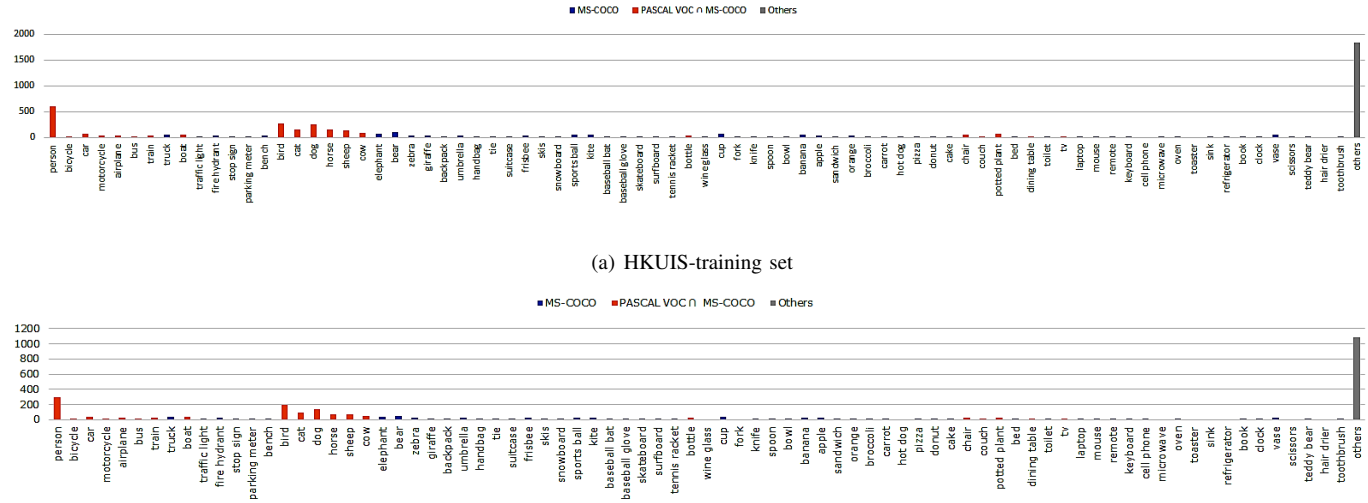
Fig. 12. The performance comparison of different saliency aggregations.

(0.81 vs. 0.87 and 0.86). Therefore, we take a closer look at iCoSeg dataset for failure cases. We observe that iCoseg contains multiple objects and the ground truth map is very subjective. Figure 11 demonstrates some failure cases. There is no obvious explanation that the footballer in the red jersey is salient while the other footballers are not (in the first row). Also, in the second row, the pyramid at the background is salient whereas the riding man in the foreground is not. It may refer to the top-down saliency where the ground-truth is provided from the annotators performing a certain task, *i.e.*, annotating or searching certain object such as pyramids or footballers wearing the red jersey.

G. Discussions

First, we would like to highlight the main objective of our work which aims to analyze and integrate the semantic information into the salient object detection problem. In addition, our method is much simpler than aforementioned deep learning models. Indeed, the contemporary deep network models for salient object detection are complicated and perform in multi-level, multi-scale in order to improve the performance. For example, Amulet [44], aggregates multi-level convolutional feature to achieve the state-of-the-art performance.

As discussed in [40], the saliency aggregation of the top 2 saliency models leads the improvement of both, for example, in terms of precision and recall rates. Therefore, we are interested in aggregating different models, namely, Amulet [44] + NLDF [45], and Amulet + SP (ours), by averaging their saliency maps. As shown in Figure 12, the combination of Amulet and NLDF slightly improves the performance of Amulet in the two datasets, ECSSD and HKUIS. Meanwhile, the combination of Amulet and our proposed method (SP) further improves the performance of Amulet to even a larger margin in terms of precision and recall rates. This explicitly demonstrates the usefulness of semantic information in the task of salient object detection. On other words, this illustrates that the semantic priors are beneficial to the deep learning methods. Although our method



(a) HKUIS-training set

(b) HKUIS-testing set

Fig. 13. The number of images in HKUIS dataset with salient objects belonging to each semantic MS-COCO class.

is favorably comparable with recent state-of-the-art methods, combining our method with these state-of-the-arts via saliency aggregation significantly boosts their performance. This may suggest the future improvement to the existing state-of-the-art deep learning methods.

Last but not least, there is a research question "how many semantic classes are enough?". In order to answer the arisen question, we are interested in investigating the overlapping classes between MS-COCO, PASCAL VOC and saliency detection datasets. In particular, we manually compute the appearance of different semantic classes in HKUIS training set and testing set. For each semantic class, we count how many images containing the salient objects belonging to that class. Figure 13 shows the number of images with salient objects belonging to each semantic class. We can see that the salient objects do not equally distribute among 80 semantic classes. As a closer look, the salient objects from 20 common classes (PASCAL VOC and MS-COCO) outnumber others in the total 80 semantic classes (MS-COCO). Many classes, i.e., *hot dog*, *sink*, *toaster* are not even found or labeled as salient in both training and testing sets. Meanwhile, the classes not listed in the 80 classes (marked as *Others*) appear in more than 1800 and 1000 images in the HKUIS training set and the testing set, respectively. Obviously, the Explicit map alone cannot detect these "*Others*" classes. The addition of the Implicit map is helpful to recognize the *Others* class. From the experiment in Section IV-B, we can see that the usage of MS-COCO (80 classes) leads to a small increment to the usage of PASCAL VOC (20 classes). The reason is that the 20 classes in PASCAL are principal classes such as *person*, *car*, *bicycle*, *dog*, and *cat*. Only few of additional 60 classes in MS-COCO can be helpful such as *bear*, *elephant*, and *truck*. This demonstrates that the importance of semantic classes is more significant than the number of classes.

V. Conclusions and Future Work

We have presented a framework for saliency detection via semantic priors (SP). Our proposed framework consists of several novel technical elements, including: (a) It takes the semantic segmentation output into consideration in order to detect *salient objects*; (b) Two individual maps, namely, the explicit saliency map and the implicit saliency map, are extracted from the semantic priors. These two maps are adaptively fused together incorporating with post-processing to yield a highly accurate saliency map. A distinguishable feature of the paper is that, unlike related approaches on deep network-based saliency models, here the semantic information is explicitly studied. According to the experimental results provided, this leaded to a good performance in terms of mean absolute errors and precision and recall rates.

For future work, we aim to explore salient object detection with top-down cues as discussed in Section IV-F and Section IV-G. Since the experimental results show that SP is a general framework which can exploit any semantic parser, more advanced semantic parsers promise to improve the framework performance in the future.

References

[1] Q. Chen, Z. Song, Y. Hua, Z. Huang, and S. Yan, "Hierarchical matching with side information for image classification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3426–3433.

[2] T. V. Nguyen, Z. Song, and S. Yan, "STAP: Spatial-Temporal Attention-Aware Pooling for Action Recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 1, pp. 77–86, 2015.

[3] T. V. Nguyen, B. Ni, H. Liu, W. Xia, J. Luo, M. S. Kankanhalli, and S. Yan, "Image re-attentionizing," *IEEE Transactions on Multimedia*, vol. 15, no. 8, pp. 1910–1919, 2013.

[4] T. V. Nguyen and G. Gao, "Novel evaluation metrics for seam carving based image retargeting," in *IEEE International Conference on Image Processing*, 2017, pp. 450–454.

[5] T. Mei, L. Li, X. Tian, D. Tao, and C. Ngo, "Pagesense: Toward stylewise contextual advertising via visual analysis of web pages," *IEEE Trans. Circuits Syst. Video Techn.*, vol. 28, no. 1, pp. 254–266, 2018.

[6] T. V. Nguyen, Q. Zhao, and S. Yan, "Attentive systems: A survey," *International Journal of Computer Vision*, vol. 126, no. 1, pp. 86–110, 2018.

[7] R. Achanta, S. S. Hemami, F. J. Estrada, and S. Süsstrunk, "Frequency-tuned salient region detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1597–1604.

[8] M. Cheng, N. J. Mitra, X. Huang, P. H. S. Torr, and S. Hu, "Global contrast based salient region detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 569–582, 2015.

[9] F. Perazzi, P. Krähenbühl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 733–740.

[10] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-aware saliency detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2376–2383.

[11] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li, "Salient object detection: A discriminative regional feature integration approach," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2083–2090.

[12] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H. Shum, "Learning to detect a salient object," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 353–367, 2011.

[13] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1155–1162.

[14] J. Shi, Q. Yan, L. Xu, and J. Jia, "Hierarchical image saliency detection on extended CSSD," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 4, pp. 717–729, 2016.

[15] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5455–5463.

[16] C. Liu, J. Yuen, and A. Torralba, "Nonparametric scene parsing via label transfer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2368–2382, 2011.

[17] J. Tighe and S. Lazebnik, "Finding things: Image parsing with regions and per-exemplar detectors," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3001–3008.

[18] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[19] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. S. Torr, "Conditional random fields as recurrent neural networks," in *International Conference on Computer Vision*, 2015, pp. 1529–1537.

[20] L. Wang, H. Lu, X. Ruan, and M. Yang, "Deep networks for saliency detection via local estimation and global search," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3183–3192.

[21] X. Li, L. Zhao, L. Wei, M. Yang, F. Wu, Y. Zhuang, H. Ling, and J. Wang, "Deepsaliency: Multi-task deep neural network model for salient object detection," *IEEE Transactions on Image Processing*, vol. 25, no. 8, pp. 3919–3930, 2016.

[22] L. Wang, L. Wang, H. Lu, P. Zhang, and X. Ruan, "Saliency detection with recurrent fully convolutional networks," in *European Conference on Computer Vision*, 2016, pp. 825–841.

[23] T. V. Nguyen and L. Liu, "Salient object detection with semantic priors," in *International Joint Conference on Artificial Intelligence*, 2017, pp. 4499–4505.

[24] J. Tighe and S. Lazebnik, "Superparsing: Scalable nonparametric image parsing with superpixels," in *European Conference on Computer Vision*, 2010, pp. 352–365.

[25] D. Eigen and R. Fergus, "Nonparametric image parsing using adaptive neighbor sets," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2799–2806.

[26] J. Yang, B. Price, S. Cohen, and M.-H. Yang, "Context driven scene parsing with attention to rare classes," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[27] T. V. Nguyen, L. Liu, and K. Nguyen, "Exploiting generic multi-level convolutional neural networks for scene understanding," in *International Conference on Control, Automation, Robotics and Vision*, 2016, pp. 1–6.

[28] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Conference on Neural Information Processing Systems*, 2012, pp. 1106–1114.

[29] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," in *IEEE International Conference on Computer Vision*, 2017, pp. 2980–2988.

[30] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.

[31] C. Koch and S. Ullman, "Shifts in selective visual attention: towards the underlying neural circuitry." *Hum Neurobiol*, 1985.

[32] M. Cheng, G. Zhang, N. J. Mitra, X. Huang, and S. Hu, "Global contrast based salient region detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 409–416.

[33] X. Li, Y. Li, C. Shen, A. R. Dick, and A. van den Hengel, "Contextual hypergraph modeling for salient object detection," in *IEEE International Conference on Computer Vision, ICCV*, 2013, pp. 3328–3335.

[34] T. V. Nguyen and J. Sepulveda, "Salient object detection via augmented hypotheses," in *International Joint Conference on Artificial Intelligence*, 2015, pp. 2176–2182.

[35] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. H. S. Torr, "BING: Binarized normed gradients for objectness estimation at 300fps," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[36] T. V. Nguyen, "Salient object detection via objectness proposals," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2015, pp. 4286–4287.

[37] A. Borji, M. Cheng, H. Jiang, and J. Li, "Salient object detection: A benchmark," *IEEE Trans. Image Processing*, vol. 24, no. 12, pp. 5706–5722, 2015.

[38] J. Wang, H. Jiang, Z. Yuan, M. Cheng, X. Hu, and N. Zheng, "Salient object detection: A discriminative regional feature integration approach," *International Journal of Computer Vision*, vol. 123, no. 2, pp. 251–268, 2017.

[39] J. Kim, D. Han, Y. Tai, and J. Kim, "Salient region detection via high-dimensional color transform," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 883–890.

[40] O. L. Meur and Z. Liu, "Saliency aggregation: Does unity make strength?" in *Asian Conference on Computer Vision*, 2014, pp. 18–32.

[41] T. V. Nguyen and M. S. Kankanhalli, "As-similar-as-possible saliency fusion," *Multimedia Tools Appl.*, vol. 76, no. 8, pp. 10 501–10 519, 2017.

[42] L. Wei, S. Zhao, O. E. F. Bourahla, X. Li, and F. Wu, "Group-wise deep co-saliency detection," in *Proceedings of International Joint Conference on Artificial Intelligence*, 2017, pp. 3041–3047.

[43] T. Wang, L. Zhang, H. Lu, C. Sun, and J. Qi, "Kernelized subspace ranking for saliency detection," in *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII*, 2016, pp. 450–466.

[44] P. Zhang, D. Wang, H. Lu, H. Wang, and X. Ruan, "Amulet: Aggregating multi-level convolutional features for salient object detection," in *IEEE International Conference on Computer Vision*, 2017, pp. 202–211.

[45] Z. Luo, A. K. Mishra, A. Achkar, J. A. Eichel, S. Li, and P. Jodoin, "Non-local deep features for salient object detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6593–6601.

[46] Q. Hou, M. Cheng, X. Hu, A. Borji, Z. Tu, and P. H. S. Torr, "Deeply supervised salient object detection with short connections," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5300–5309.

[47] P. Hu, B. Shuai, J. Liu, and G. Wang, "Deep level sets for salient object detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 540–549.

[48] T. Judd, K. Ehinger, F. Durand, and A. Torralba, "Learning to predict where humans look," in *International Conference on Computer Vision*, 2009, pp. 2106–2113.

[49] C. Lang, T. V. Nguyen, H. Katti, K. Yadati, M. S. Kankanhalli, and S. Yan, "Depth matters: Influence of depth cues on visual saliency," in *European Conference on Computer Vision*, 2012, pp. 101–115.

[50] T. K. Leung and J. Malik, "Representing and recognizing the visual appearance of materials using three-dimensional textons," *International Journal of Computer Vision*, vol. 43, no. 1, pp. 29–44, 2001.

[51] E. S. L. Gastal and M. M. Oliveira, "Domain transform for edge-aware image and video processing," *ACM Trans. Graph.*, vol. 30, no. 4, pp. 69:1–69:12, 2011.

[52] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, 2002.

[53] M. Everingham, L. J. V. Gool, C. K. I. Williams, J. M. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.

[54] T. Lin, M. Maire, S. J. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: common objects in

context," in *European Conference on Computer Vision*, 2014, pp. 740–755.
[55] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.
[56] D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen, "icoseg: Interactive co-segmentation with intelligent scribble guidance," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 3169–3176.
[57] J. Zhang and S. Sclaroff, "Exploiting surroundedness for saliency detection: A boolean map approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 5, pp. 889–902, 2016.
[58] G. Lee, Y. Tai, and J. Kim, "Deep saliency with encoded low level distance map and high level features," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 660–668.
[59] Y. Zhai and M. Shah, "Visual attention detection in video sequences using spatiotemporal cues," in *ACM Multimedia*, 2006, pp. 815–824.
[60] R. Margolin, A. Tal, and L. Zelnik-Manor, "What makes a patch distinct?" in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1139–1146.
[61] P. Zhang, D. Wang, H. Lu, H. Wang, and B. Yin, "Learning uncertain convolutional features for accurate saliency detection," in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 2017, pp. 212–221.

Tam V. Nguyen is an Assistant Professor at Department of Computer Science, University of Dayton. Prior to that, he was a research scientist and principal investigator at ARTIC research centre, Singapore Polytechnic. He was also an adjunct lecturer at National University of Singapore. He received PhD degree in National University of Singapore in 2013. His research topics include computer vision, applied deep learning, multimedia content analysis, and mixed reality. He is an IEEE Senior Member.

Khanh Nguyen is a PhD student at University of Information Technology. Prior to that, he obtained his B.S. degree from University of Science in 2008. His research interests include computer vision, multimedia analysis and deep learning.

Thanh-Toan Do Thanh-Toan Do is currently a Lecturer at the Department of Computer Science, the University of Liverpool (UoL), United Kingdom. He obtained Ph.D. in Computer Science from INRIA, Rennes, France in 2012. Before joining UoL, he was a Research Fellow at the Singapore University of Technology and Design, Singapore (2013 - 2016) and the University of Adelaide, Australia (2016 - 2018). His research interests are Computer Vision and Machine Learning.