Weakly Supervised Saliency Detection Based on Image-Level Labels and Superpixel Blocks (Post-print)

Authors: Tan Taizhe, Xuan Kangxi, Zeng Qunsheng

Date: 2018-12-13T00:00:00+00:00

Abstract

To address the costly problem of obtaining training datasets, we propose a novel weakly supervised method for image saliency detection that utilizes only image-level labels during network model training. The method comprises two stages: in the first stage, a classification model is trained based on image-level labels to obtain a foreground inference map; in the second stage, the original image is processed into superpixels and fused with the foreground inference map obtained in stage one, thereby refining the boundaries of salient objects. The algorithm leverages existing large-scale training sets and image-level labels without resorting to pixel-level labels, thus reducing the annotation workload. Experimental results on four public benchmark datasets demonstrate that the performance significantly outperforms unsupervised models and also exhibits certain advantages when compared to fully supervised models.

Full Text

Preamble

Vol. 37 No. 2
Application Research of Computers
ChinaXiv Cooperative Journal

Weakly Supervised Saliency Detection Based on Image-Level Labels and Superpixel Blocks

Tan Taizhe[1],[2], Xuan Kangxi[1]†, Zeng Qunsheng[1]
(1. College of Computer Science and Technology, Guangdong University of Technology, Guangzhou 510006, China;
2. Heyuan Guanggong Collaborative Innovation Research Institute, Heyuan, Guangdong 517000, China)

Abstract: To address the high cost of obtaining training datasets, this paper proposes a novel weakly supervised method for image saliency detection that uses only image-level labels during network training. The method consists of two stages. In the first stage, a classification model is trained using image-level labels to obtain a foreground inference map. In the second stage, the original image is processed into superpixel blocks and fused with the foreground inference map from the first stage to refine salient object boundaries. The algorithm leverages existing large-scale training sets with image-level labels without using pixel-level annotations, thereby thereby reducing annotation workload. Experimental results on four public benchmark datasets demonstrate that the proposed method significantly outperforms unsupervised models and shows certain advantages over fully supervised models.

Keywords: deep learning; weak supervision; saliency detection; superpixel

0 Introduction

Visual saliency is a fundamental research problem in neuroscience and psychology that investigates how the human visual system selects regions of interest from complex scenes. Humans can accurately and rapidly identify objects or regions of interest, a phenomenon known as focal or salient attention. Attention is considered partially automatic, bottom-up, and memory-free, driven by prominent stimuli. It can also be guided by relatively slow, top-down, memory-dependent mechanisms. For instance, when viewing faces, familiar faces may capture attention first. Reliable visual saliency estimation enables appropriate image processing even without prior knowledge, making it a crucial step in many computer vision tasks.

In recent years, Convolutional Neural Networks (CNNs) have achieved remarkable progress in computer vision, sparking a surge in pixel-level annotated sample-based saliency detection [**?**]. Compared with unsupervised methods [**?**, **?**], DNNs based on fully supervised learning more effectively capture semantically salient foreground regions and produce accurate results in complex scenes. However, given the data-hungry nature of DNNs, their superior performance heavily relies on large-scale datasets with pixel-level annotations. Such annotation work is extremely tedious, making precisely annotated training sets scarce and expensive.

To alleviate the need for large-scale pixel-level annotations, this paper investigates weakly supervised methods using image-level labels to train saliency detectors. Image-level labels indicate object categories present in an image and are much easier to collect than pixel-wise annotations. Moreover, image-level labels provide category information about primary objects that are likely salient foregrounds. Recent work [**?**, **?**] has demonstrated that DNNs trained with only image-level labels can also provide object location information. Therefore,

weakly supervised methods that train DNNs using only image-level labels for salient object detection are feasible and effective.

Although DNNs can extract foreground targets that are visually apparent, the boundaries remain blurry because pixels around edges concentrate in similar receptive fields. Consequently, boundary refinement of saliency maps is necessary. This work is therefore divided into two stages: pre-training with image-level labels and boundary refinement combined with superpixel blocks.

In the first stage, to address the detail loss caused by pooling layers, we pretrain a Fully Convolutional Network (FCN) using image-level labels. By altering the convolution kernel stride instead of using pooling layers, we obtain multi-scale salient features. In the second stage, inspired by [**?**], we propose a novel method for joint refinement of convolutional features and superpixel boundaries. First, feature maps from the first stage are integrated to obtain their feature boundaries (FB). Then, the original image is processed into superpixels to obtain superpixel boundaries (SPB). The FB is adjusted according to SPB to achieve boundary refinement.

To reduce the need for large-scale pixel-level annotations, this paper makes two main contributions. First, it provides a new direction for weakly supervised saliency detection that uses only existing large-scale image-level labels, significantly reducing annotation workload. Second, it proposes a novel boundary refinement method that better utilizes detailed information from the original image, compensating for the blurry boundaries of CNNs and further improving detection accuracy.

---

1 Related Work

Many traditional image processing algorithms such as CRFs [**?**], random forests [**?**], and SVM [**?**] have been successfully applied to saliency detection. These methods aim to capture contextual image information by finding graph structures and use classifiers to label entities like superpixels [**?**]. Jiang et al. [**?**] formulated saliency detection as a regression problem, using supervised learning to map feature vectors of superpixel regions to saliency scores after superpixel processing, which are then integrated into a saliency map. Li et al. [**?**] trained an SVM to detect salient objects while using super-edge segmentation and multi-scale methods for post-processing.

However, DNN-based methods have demonstrated significant advantages in saliency detection. FCN-based saliency detection methods [**?**, **?**] offer competitive performance in both accuracy and speed. Wang et al. [**?**] predicted saliency maps by integrating local estimation and global search. Reference [**?**] proposed a two-stage deep network that first generates a coarse map and then progressively refines it using another network. However, training these models requires massive pixel-level annotations, which is extremely costly. As a representa-

tive work in weakly supervised detection, [**?**] proposed a Foreground Inference Network (FIN) with iterative conditional random fields, achieving performance significantly better than unsupervised methods and even surpassing some fully supervised algorithms. In summary, combining DNN models with image-level labels for weakly supervised saliency detection represents a promising new direction for addressing saliency problems.

---

2 Weakly Supervised Saliency Detection

CNNs for image-level label prediction typically consist of a series of convolutional layers followed by several fully connected layers. Let I represent a training image and $l \in \{1, 2, \ldots, N\}$ represent its corresponding category label. The CNN takes I as input and produces an N-dimensional score vector Y after a series of computations, where the index of the maximum value in Y indicates the image category. During training, the CNN minimizes a loss function L to measure prediction accuracy. Although CNNs are trained with image-level labels, recent experiments have shown that high-level convolutional layers can act as detectors to capture and identify object parts. However, the location information encoded in convolutional layers cannot be transferred to fully connected layers.

Therefore, for multi-label recognition tasks, Jonathan Long et al. proposed Fully Convolutional Networks (FCN) to preserve object location information. Given an input image of size $H \times W$ and using pixel-level annotations as supervision, the trained model outputs N channels of $H \times W$ score maps, where each channel represents a category and the values at corresponding points represent the probability that the image pixel belongs to that category. However, saliency maps extracted by FCN have very blurry edges. Therefore, in weakly supervised saliency detection, we modify the final output layer to an $N \times 1 \times 1$ score map, where these N values represent the probability that the image belongs to each category. We then integrate high-level convolutional feature maps to obtain a foreground inference map, which undergoes further post-processing.

2.1 Foreground Inference Map

During FCN model training with image-level labels as supervision signals, convolutional kernels capture object regions in the input image, with each channel corresponding to a feature of the object. In saliency detection tasks, we are not concerned with object categories but aim to discover all salient object regions. To obtain such category-agnostic saliency maps, one could sum all channel feature maps at the same scale and then map them to color values between 0-255 for visualization [**?**]. However, this approach has a drawback: responses from salient object parts can be suppressed by high-response regions in other channels, resulting in saliency maps with either significant background noise or non-uniform highlighting of salient regions.

To address this issue, we add a branch during FCN training to automatically generate a Foreground Inference Map (FIM). This branch also consists of a series of convolutional layers and a sigmoid layer. The output feature map F has only one channel with values in $[0, 1]$, representing the saliency of corresponding pixels. Overall, given an image X, the model generates C feature maps $S(n \times n)$ and one foreground inference map F, which are substituted into the following formula:

$$S_{out} = \sum_{k=1}^{C} (S_k \odot F) \oplus S_k$$

where S_k represents the k-th channel of feature map S, $\odot$ represents element-wise multiplication between S_k and F, and $\oplus$ represents the integrated feature map passed to the next layer. This approach leverages high responses in each channel of feature map S without mutual suppression, enabling the FIM generation to undergo a continuous learning and training process [**?**, **?**].

Considering that FCN trained with specific image-level labels struggles to cover categories not in the training set, we apply the masking operation in Equation (1) to intermediate feature maps rather than the final layer. Since intermediate feature maps do not directly correspond to image categories but instead extract specific structures and textures, these representations are more general. Consequently, the generated FIM can better capture new categories not seen during training, improving model robustness.

[Figure 1: see original paper] shows the network structure proposed in this paper. In the first stage, FCN (1)-(5) is trained to generate a Foreground Inference Map (FIM). In the second stage, the FIM undergoes edge refinement using superpixel block saliency maps (6) to generate the final saliency map (7).

2.2 Pre-training Based on Image-Level Labels

This section formally introduces the first stage of the weakly supervised saliency detection method. The network is trained using the ImageNet dataset, which includes 1000 object categories with 1000 images per category.

As discussed above, since FIM generation is closely related to FCN training, they can be jointly trained with shared convolutional features. Specifically, we design a shared network branch after the 16-layer VGG network [**?**], which consists of 13 convolutional layers connected by ReLU nonlinear functions and 4 max-pooling layers. The FIM branch network is computed through one convolutional layer, one BN layer [**?**], and one Sigmoid layer, then serves as a mask integrated with FCN to obtain new feature maps. Finally, fully connected layers generate a 1000-dimensional object category score vector, which is converted to category probabilities using the Softmax function.

For each generated saliency map, larger values indicate higher foreground likelihood. Through extensive observation, we can infer an explicit association between foreground pixels and semantic objects. Since each simple image is accompanied by semantic labels, we can easily assign corresponding image-level labels to foreground candidate pixels. We then propose a multi-label cross-entropy loss function to train the segmentation network under saliency map supervision.

Given a training set containing N training samples $\{X_i, Y_i\}_{i=1}^{N}$, the loss function is minimized to achieve model convergence:

$$L(\theta) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C}[y_{ij}\log(p_{ij}) + (1-y_{ij})\log(1-p_{ij})] + \lambda\|F\|_1 + \eta\|\theta\|_2^2$$

where θ represents network parameters. The first and second terms constitute cross-entropy loss ensuring prediction accuracy, the third term is L1 regularization for FIM, and the last term is network decay. Based on experience, λ and η are set to 7×10^{-5} and 5×10^{-5}, respectively. Shared layer weights are initialized using the pre-trained VGG model [**?**], while other layers are randomly initialized using the method from [**?**]. All input images are normalized to a fixed resolution of 224×224, while FIM resolution is 56×56 and upsampled to 224×224 via bilinear interpolation. To accelerate convergence of the loss function, we employ Stochastic Gradient Descent (SGD).

2.3 Edge Detection Based on Superpixel Blocks

After DNN training, we obtain the FIM. However, as previously mentioned, FIM edges are relatively blurry, requiring further post-processing to refine contours.

Contrast is an important parameter for evaluating human vision. Since salient objects have different contrast from their surroundings and human visual cells are more sensitive to image edges, we determine image edges through contrast calculation and segment images into superpixel blocks. Traditional image processing methods operate based on three attributes: color, texture, and shape [**?**, **?**]. These techniques have been successfully applied in various domains. However, these attributes cannot provide high-level image understanding because humans typically do not comprehend images from color, texture, or shape alone, but rather based on the interconnections among these three attributes. In other words, the saliency of target objects depends on their uniqueness relative to the surrounding environment.

Based on observations, salient objects have three distinct features that enable shape feature calculation: (a) salient objects are always significantly different from their surroundings; (b) salient objects are almost always located near the image center; (c) salient object boundaries are perfectly closed.

The first feature is based on bottom-up visual stimuli, extensively studied in [?, ?]. The second is a location priority feature—human attention always focuses first on the image center before diverging outward [?]. The third feature, proposed in [?, ?], indicates that salient targets are usually clustered rather than scattered throughout the image, and humans observe object edges before the brain combines them to form objects.

Our goal is to segment images into closed contours whose boundaries contain salient object boundaries. First, images are segmented via edge detection to generate superpixel blocks, then saliency is calculated based on superpixel images. References [?, ?] scale original images to smaller sizes to reduce computation. Our method produces far fewer superpixel blocks than pixels, thereby reducing computational load while generating full-resolution edge maps.

For image segmentation, we consider two aspects: color distance and spatial distance. Using the method from [?], we divide images into several regions and evaluate each region's color-based saliency according to color contrast. Based on features one and two, if a region (superpixel block) is significantly different from its surrounding context and relatively close to the image center, it is more likely to be salient. Meanwhile, for superpixel blocks belonging to the same category, their similarity remains high regardless of spatial distance, yet distant similar superpixel blocks cannot significantly enhance each other's saliency.

In summary, for an image containing N superpixel blocks, we first minimize the following loss function [?] to achieve model convergence:

$$E(L) = \sum_i D(l_i) + \sum_{i,j} V(l_i, l_j)$$

The saliency of the center superpixel block is calculated as:

$$S(p) = \sum_{q \neq p} \exp\left(-\frac{d_{color}(p,q)}{w_c} - \frac{d_{position}(p,q)}{w_p}\right)$$

where d_{color} and $d_{position}$ represent color distance and spatial distance between superpixel blocks p and q, respectively, and w_c and w_p are hyperparameters controlling the strength of color and spatial distances.

For non-center superpixel blocks, based on feature 2, we additionally incorporate the influence of center superpixel block saliency:

$$S'(p) = S(p) \cdot \exp\left(-\frac{d_{position}(p,c)}{\sigma}\right)$$

where c refers to the center superpixel block. Through these formulas, we obtain the saliency of each superpixel block in the image.

2.4 Joint Boundary Refinement of FIM and Superpixel Blocks

After pre-training in Section 2.2, the generated FIM has captured foreground regions. As known, FIM contains substantial edge information. By setting a threshold, we generate a binarized FIM, as shown in Figure 1. During binarization, each image uses a different threshold—i.e., the threshold is not fixed. We first denoise the image to remove high-response noise points in FIM, then compute the image histogram to replace low-frequency pixels with similar ones [**?**], and finally take the median between maximum and minimum pixel values as the threshold. However, object boundaries remain discontinuous, requiring connectivity processing.

Based on the third feature of salient objects—perfectly closed boundaries—we first find the longest edge in FIM by computing connected component sizes, take one endpoint as the starting point, and search for other nearby edges following gradient priority rules. As shown in Figure 2, endpoint A's extension trend is downward, so it prioritizes searching for unconnected edges below. After finding endpoint B that needs connection, point A extends toward B based on the saliency score map obtained in Section 2.3.

[Figure 2: see original paper] illustrates the FIM edge refinement process. Figure (a) shows the superpixel saliency map, (b) shows the binarized foreground inference map FIM, (c) shows the edge-refined FIM where red lines represent edges extended based on superpixel saliency, and (d) shows the final saliency segmentation result.

---

3 Experiments

For hyperparameter design, our method is implemented based on TensorFlow with weight decay set to 0.001 and momentum set to 0.9. During FIM binarization, pixels with frequency less than 10 are replaced by similar points. Pixels in FIM with saliency greater than the threshold are set to 255, otherwise 0.

In experiments, we compare our method with nine models: MBS [**?**], wCtr [**?**], MR [**?**], BSCA [**?**], WSS [**?**], RFCN [**?**], DCL [**?**], DS [**?**], and MC [**?**]. Since saliency detection is a relatively new vision problem with limited published datasets, we select four public datasets for testing:

1. **SED** [**?**]: Contains 100 images, one category per image.
2. **ECSSD** [**?**]: Contains 1000 structurally complex images with multiple categories per image.
3. **MSRA-B** [**?**]: Contains 5000 images across 200+ categories.
4. **PASCAL-S** [**?**]: 850 carefully selected images from the PASCAL VOC dataset in complex environments.

For comparison, we introduce F_β as the evaluation metric, as shown in Figure 3 [Figure 3: see original paper]. The final saliency maps are binarized and

compared with pixel-level ground truth annotations to obtain precision and recall values. The F_β score for each dataset is derived from averaged precision and recall values across all images, defined as:

$$F_\beta = \frac{(1+\beta^2) \cdot \text{Precision} \cdot \text{Recall}}{\beta^2 \cdot \text{Precision} + \text{Recall}}$$

where β^2 is set to 0.3. We demonstrate our method's effectiveness by comparing it with state-of-the-art methods.

[Figure 3: see original paper] shows precision-recall curves. Table 1 presents comparative results with four state-of-the-art unsupervised algorithms, one weakly supervised algorithm, and four fully supervised algorithms. F_β measures detection performance, with bold indicating the best result on each dataset.

As shown in Table 1, since our method extracts features with purposeful supervision signals for continuous optimization, it consistently outperforms unsupervised methods. Despite not using expensive pixel-level annotations, our final results also show advantages over fully supervised methods. Moreover, while most fully supervised saliency detection datasets contain many images but fewer than 300 categories, our method is trained on ImageNet, extracting correspondingly richer category features and thus achieving better robustness.

---

4 Conclusion

This paper proposes a weakly supervised saliency detection method based on image-level labels. The method comprises two stages: In the first stage, we add a novel layer to FCN that learns to predict image-level labels to generate a Foreground Inference Map (FIM). In the second stage, we process the input image into superpixels based on three features of salient objects combined with contextual information, calculate each superpixel block's saliency, and refine FIM edges based on superpixel saliency. Evaluation on benchmark datasets validates our method's effectiveness.

Future work will explore more advanced weakly supervised methods to further improve detection accuracy and generalization capability.

References

[17] Jiang Huaizu, Wang Jingdong, Yuan Zejian, et al. Salient object detection: a discriminative regional feature integration approach [J]. International Journal of Computer Vision, 2017, 123(2): 251-268.

[18] Li Xi, Li Yao, Shen Chunhua, et al. Contextual hypergraph modeling for salient object detection [C]//Proc of IEEE International Conference on Computer Vision. Washington DC: IEEE Computer Society, 2014: 332-337.

[1] Zhao Rui, Ouyang Wanli, Li Hongsheng, et al. Saliency detection by multi-context deep learning [C]//Computer Vision and Pattern Recognition. IEEE, 2015: 1265-1274.

[2] Li Guanbin, Yu Yizhou. Visual saliency based on multiscale deep features [C]//Computer Vision and Pattern Recognition. IEEE, 2015: 5455-5463.

[3] Wang Linzhao, Wang Lijun, Lu Huchuan, et al. Saliency detection with recurrent fully convolutional networks [C]//Proc of European Conference on Computer Vision. Berlin: Springer, 2016: 825-841.

[4] Zhang Jianming, Stan Sclaroff, Lin Zhe, et al. Minimum barrier salient object detection at 80 fps [C]//Proc of IEEE International Conference on Computer Vision. Washington DC: IEEE Computer Society, 2015: 1404-1411.

[5] Kong Yuqiu, Wang Lijun, Liu Xiuping, et al. Pattern mining saliency detection [M]//Computer Vision. Springer International Publishing, 2016: 301-317.

[6] Long Jonathan, Zhang Ning, Trevor Darrell. Do convnets learn correspondence? [C]//Advances in Neural Information Processing Systems. 2014: 1601-1609.

[7] Zhou Bolei, Aditya Khosla, Antonio Lapedriza, et al. Learning deep features for discriminative localization [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. Washington DC: IEEE Computer Society, 2015: 2921-2929.

[8] Jiang Huaizi, Wang Jingdong, Yuan Zejian, et al. Automatic salient object segmentation based on context and shape prior [C]//Proc of British Machine Vision Conference. 2011.

[9] Liu Tie, Sun Jian, Zheng Nanning, et al. Learning to detect a salient object [J]. IEEE Trans on Pattern Anal Mach Intell, 2011, 33(2): 353-367.

[10] Jiang Huaizu, Wang Jingdong, Yuan Zejian, et al. Salient object detection: a discriminative regional feature integration approach [J]. International Journal of Computer Vision, 2017, 123(2): 251-268.

[11] Jiwhan Kim, Han Dongyoon, Tai Yu-Wing, et al. Salient region detection via high-dimensional color transform [J]. IEEE Trans on Image Processing, 2015, 25(1): 9-23.

[12] Li Yin, Hou Xiaodi, Christof Koch, et al. The secrets of salient object segmentation [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2014: 280-287.

[13] Lu Song, Vijay Mahadevan, Nuno Vasconcelos. Learning optimal seeds for diffusion-based salient object detection [C]//Proc of IEEE Conference on

Computer Vision and Pattern Recognition. Washington DC: IEEE Computer Society, 2014: 2790-2797.

[14] Joseph Tighe, Marc Niethammer, Svetlana Lazebnik. Scene parsing with object instances and occlusion ordering [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2014: 3748-3755.

[15] Guo Ruiqi, Derek Hoiem. Labeling complete surfaces in scene understanding [J]. International Journal of Computer Vision, 2015, 112(2): 172-187.

[16] Nasim Souly, Mubarak Shah. Scene labeling using sparse precision matrix [C]//Computer Vision and Pattern Recognition. IEEE, 2016: 2240-2248.

[19] Jason Kuen, Wang Zhehua, Wang Gang. Recurrent attentional networks for saliency detection [C]//Computer Vision and Pattern Recognition. IEEE, 2016: 3668-3677.

[20] Pedro F. Felzenszwalb, Daniel P. Huttenlocher. Distance transforms of sampled functions [J]. Theory of Computing, 2004, 8(19): 415-428.

[21] Wang Lijun, Lu Huchuan, Ruan Xiang, et al. Deep networks for saliency detection via local estimation and global search [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. Washington DC: IEEE Computer Society, 2015: 3183-3192.

[22] Li Guanbin, Yu Yizhou. Deep Contrast learning for salient object detection [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. Washington DC: IEEE Computer Society, 2016: 478-486.

[23] Wang Lijun, Lu Huchuan, Wang Yifan, et al. Learning to detect salient objects with image-level supervision [C]//Computer Vision and Pattern Recognition. IEEE, 2017: 3796-3805.

[24] Zeiler M D, Fergus R. Visualizing and understanding convolutional networks [C]//Proc of European Conference on Computer Vision. Berlin: Springer, 2014: 818-833.

[25] Kelvin Xu, Jimmy Ba, Ryan Kiros, et al. Show, attend and tell: neural image caption generation with visual attention [J]. Computer Science, 2015: 2048-2057.

[26] Dai Jifeng, He Kaiming, Sun Jian. Convolutional feature masking for joint object and stuff segmentation [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2015: 3992-4000.

[27] Karen Simonyan, Andrew Zisserman. Very deep convolutional networks for large-scale image recognition [J]. Computer Science, 2014.

[28] Sergey Ioffe, Christian Szegedy. Batch normalization: accelerating deep network training by reducing internal covariate shift [J]. 2015: 448-456.

[29] He Kaiming, Zhang Xiangyu, Ren Shaoqig, et al. Delving deep into rectifiers: surpassing human-level performance on imagenet classification [C]//Proc

of IEEE International Conference on Computer Vision. Washington DC: IEEE Computer Society, 2015: 1026-1034.

[30] Cheng Mingming, Zhang Guoxin, N J Mitra, et al. Global contrast based salient region detection [C]//Computer Vision and Pattern Recognition. IEEE, 2011: 409-416.

[31] Stas Goferman, Lihi Zelnik-manor, Ayellet Tal. Context-Aware Saliency Detection [C]//Computer Vision and Pattern Recognition. IEEE, 2010: 2376-2383.

[32] Hou Xiaodi, Zhang Liqing. Saliency detection: a spectral residual approach [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. Washington DC: IEEE Computer Society, 2007: 1-8.

[33] Laurent Itti, Christof Koch, Ernst Niebur. A model of saliency-based visual attention for rapid scene analysis [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2002, 20(11): 1254-1259.

[34] Vladimir Kolmogorov, Ramin Zabih. What energy functions can be minimized via graph cuts? [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2004, 26(2): 147-159.

[35] Bhattacharya Subhabrata, Sukthankar Rahul, Shah Mubarak. A framework for photo-quality assessment and enhancement based on visual aesthetics [C]//Proc of ACM International Conference on Multimedia. New York: ACM Press, 2010: 271-280.

[36] Ritendra Datta, Dhiraj Joshi, Li Jia, et al. Studying aesthetics in photographic images using a computational approach [C]//Proc of European Conference on Computer Vision. Springer-Verlag, 2006: 288-301.

[37] Luo Yiwen, Tang Xiaoou. Photo and video quality evaluation: focusing on the subject [C]//Proc of European Conference on Computer Vision. Springer-Verlag, 2008: 386-399.

[38] Joachim S. Stahl, Wang Song. Edge grouping combining boundary and region information [J]. IEEE Trans on Image Processing, 2007, 16(10): 2630-2646.

[39] Sara Vicente, Vladimir Kolmogorov, Carsten Rother. Graph cut based image segmentation with connectivity priors [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2008: 1-8.

[40] Pedro F. Felzenszwalb, Daniel P. Huttenlocher. Efficient graph-based image segmentation [J]. International Journal of Computer Vision, 2004, 59(2): 167-181.

[41] Zhang Zhiqi, Cao Yu, Dhaval Salvi, et al. Free-shape subwindow search for object localization [C]//Computer Vision and Pattern Recognition. IEEE, 2010: 1086-1093.

[42] Qin Yao, Lu Huchuan, Xu Yiqun, et al. Saliency detection via cellular automata [C]//Computer Vision and Pattern Recognition. IEEE, 2015: 1100-1109.

[43] Zhang Dingwen, Meng Deyu, Han Junwei. Co-saliency detection via a self-paced multiple-instance learning framework [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2017, 39(5): 865-878.

[44] Philipp Krähenbühl, Vladlen Koltun. Efficient inference in fully connected CRFs with gaussian edge potentials [EB/OL]. 2012. https://arxiv.org/pdf/1210.5644.pdf.

[45] Sharon Alpert, Meirav Galun, Ronen Basri, et al. Image Segmentation by probabilistic bottom-up aggregation and cue integration [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. IEEE Xplore, 2007: 1-8.

[46] Yan Qiong, Xu Li, Shi Jiaya, et al. Hierarchical saliency detection [C]//Computer Vision and Pattern Recognition. IEEE, 2013: 1155-1162.

[47] Zhu Wangjiang, Liang Shuang, Wei Yichen, et al. Saliency optimization from robust background detection [C]//Computer Vision and Pattern Recognition. IEEE, 2014: 2814-2821.

[48] Yang Chuan, Zhang Lihe, Lu Huchuan, et al. Saliency detection via graph-based manifold ranking [C]//Computer Vision and Pattern Recognition. IEEE, 2013: 3166-3173.

[49] Li Xi, Zhao Liming, Wei Lina, et al. Deepsaliency: multi-task deep neural network model for salient object detection [J]. IEEE Trans on Image Processing, 2016, 25(8): 3919.

Note: Figure translations are in progress. See original paper for figures.