Co-saliency Detection Based on Siamese Network

Zhengchao Lei, Weiyan Chai, Sanyuan Zhao(✉), Hongmei Song, and Fengxia Li

Beijing Laboratory of Intelligent Information Technology, School of Computer Science, Beijing Institute of Technology, Beijing 100081, People's Republic of China
zhaosanyuan@bit.edu.cn

Abstract. Saliency detection in images attracts much research attention for its usage in numerous multimedia applications. Beside on the detection within the single image, co-saliency has been developed rapidly by detecting the same foreground objects in different images and trying to further promote the performance of object detection. This paper we propose a co-saliency detection method based on Siamese Network. By using Siamese Network, we get the similarity matrix of each image in superpixels. Guided by the single image saliency map, each saliency value, saliency score matrix is obtained to generate the multi image saliency map. Our final saliency map is a linear combination of these two saliency maps. The experiments show that our method performs better than other state-of-arts methods.

Keywords: Co-saliency detection · Siamese network
Feature extraction

1 Introduction

The purpose of saliency detection is to find the regions that humans are most interested in. Following this idea, the images or videos can be compressed more effectively and accurately. In this paper, we mainly focus on the saliency detection area instead of the network problem. In the past decades, saliency prediction models have changed remarkably and applied to many fields, such as image search, redirection and segmentation. There are two main methods for detecting salient area, one is global detecting, the other is local detecting, which are similar to methods in image processing.

Local saliency detection method mainly focuses on the different features that appear in the adjacent local region. Nowadays there exist many mature models for detecting local salient regions, including visual saliency detection based on graph model, saliency based on feature learning, and interpolation method of local region. Local saliency detection initially utilizes image pyramids to generate color and orientation features, which means the images will be sampled by stages. [1] calculated the color difference of each pixel around the center in the fixed adjacent region and extracted local saliency map. The most important part

L. Zhu and S. Zhong (Eds.): MSN 2017, CCIS 747, pp. 99–109, 2018.
https://doi.org/10.1007/978-981-10-8890-2_8

of the local sampling method is to determine the size of sampling area. Besides, the edge information with high contrast is not necessarily in foreground. [2] adopted the multilevel saliency detection method which divided image into multi layers from depth, and then integrated the results with graph models. Global saliency detection method starts from the global characteristics of images. Generally, image frequency or spatial domain computing methods are adopted. [3] introduced a method to subtract the mean value in color space through lowpass filter. [4] extended histograms to three dimensional color space. However Both of these methods were mere to find the pixels that stand out in the whole image instead of considering the spatial distribution. [5] tried to describe the spatial distribution of colors, which may be difficult for similar elements both in the foreground and background.

Both local and global saliency detection methods are single image saliency detection, while Co-saliency detection aims at extracting common objects from a set of images. The extraction of same objects in a set of images is one of the most important application in computer vision and has been widely used in image matching, pattern recognition and synergetic recognition. [6] gave a more precise definition to co-saliency detection. Meanwhile a new method was introduced that co-saliency detection contains single image saliency map and multi images saliency maps which were generated by super pixel segmentation and feature extraction. By using mathematical method, the two results are joined together to obtain the final saliency map. And this approach has been continuously improved. In 2015, [7] tried to eliminate the common background except for extracting common foreground in multiple images. The single image and multi image saliency detection now are known as Intrasaliency and Intersaliency.

Co-segmentation is also closely related to co-saliency detection. The purpose of co-segmentation is to segment the same area from a set of images, which was originally used as histogram matching method. At present, there are two types of co-segmentation. One is collaborative segmentation based on graph models and the problem is converted into maximum flow problem. The other is cooperative segmentation method based on clustering problem. The results of saliency detection are often used to conduct the co-segmentation problem. [8] proposed a co-segmentation method based on cosine similarity and guided by salient features. They optimized the clustering step, and made the algorithm perform better with faster speed.

It is easy to see that both single image saliency and co-saliency detection are based on human cognition. Salient object is the region that is quite different from the adjacent area, or the common object repeated in a number of images. Therefore, the artificial intelligence based methods are suitable for saliency detection. This paper we introduce deep learning method assisting us to accomplish co-saliency detection. In 2015, [9] utilized deep convolution neural network(CNN) to learn and express high-level features to detect the saliency region. [7] extended to deep learning framework SDAE. Most of the co-segmentation and co-saliency detection methods are mainly achieved by unsupervised learning, such as clustering and so on. However, both of the deep neural network based method are time and resource consuming because of the training steps. Therefore, we proposed a new siamese network based method to reduce the computing cost in train step and obtained a better performance in co-saliency detection. Our approach is summarized below. 1. A traditional method is used to generate the single image saliency map. 2. By utilizing the deep learning network-Siamese Network to train the feature difference model, we obtain the similarity parameter matrix of each superpixel between interimages. 3. The multi-image saliency map can be achieved by based on the vote of similarity matrix and the score of single image saliency map. 4. In the final step, we linearly combine the single image saliency map with multi image saliency map to obtain the final saliency map. Note that the siamese network in our work is to generate the feature differences between the input images instead of extracting features.

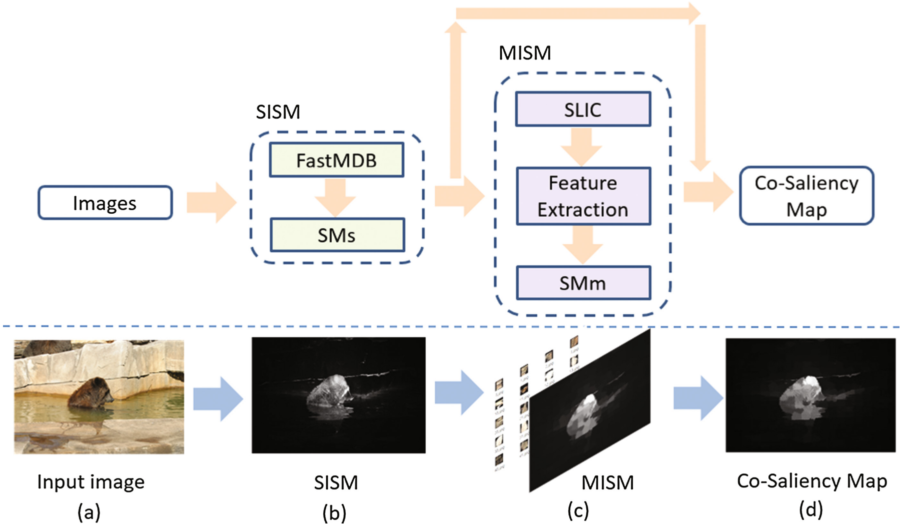

Fig. 1. Illustration of our algorithm

2 Related Work

2.1 Single Image Saliency Detection

There exist lots of saliency detection method for single image [4,5,10,11]. Taking into account time consuming and performance, we choose Fast-MDB [11] as our single image detection approach. Minimum Barrier Distance(MDB) [10] is a fast algorithm designed to generate the saliency map of an image. Barrier connectivity information has always been one of the effective methods for saliency generation. However, this step is also the bottleneck of methods. The MBD algorithm omitted the regional extraction process and improved the original method. At the same time, Fast-MBD algorithm provided an approximate approach based on MBD algorithm, which generated saliency map in milliseconds.

2.2 Siamese Network

The image similarity plays an important role in computer vision and has been applied in the field of image super resolution, dynamic structure, object recognition, image classification. Variety of methods have been explored to determine whether two images are the same. In mathematical, Hu invariant moments are widely used to extract features that insensitive to translation, rotation and scaling in images, and already available in fingerprint matching. Another important method to match images is SIFT.

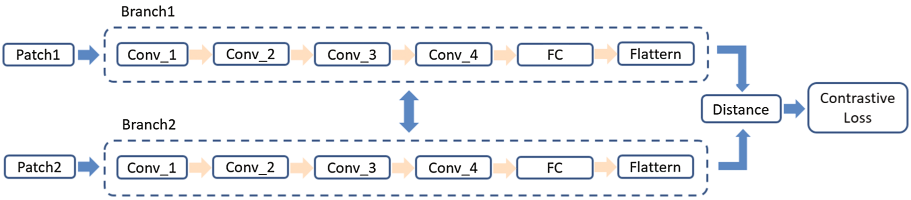

Fig. 2. Siamese network used in our method

Recent years, with the rapid development of artificial intelligence techniques, many researchers tried to solve this problem with neural network. [12] explored to describe the feature through AlexNet network, and they found that deep learning exceed the performance of SIFT algorithm in many cases. [13] verified deep learning method on some data sets. The method achieved good performance only in particular cases. [14] constructed a model to compare images through CNN. In this paper, we introduced siamese network for matching image patches in feature extraction.

Figure 2 demonstrates the siamese network structure used in our method. The network contains two branches and the input is image pair with same height, width and channel. Each branch consists of 4 convolution layers, 1 fully convolution layer and flattern layer. The kernel size for each convolution layer is 7×7, 5×5, 3×3, 1×1. The contrastive loss function is defined as the euclidean distance of two branch output. Branches and last output in our network can be considered as the descriptor and similarity function respectively. In traditional methods, each image was represented with the same feature, that is why branches in our network share the same weight. At the same time this will reduce the training cost significantly. We utilize the network to obtain an image comparison model to distinguish the same patches among images.

In addition to the siamese network, there are some similar networks can be used for comparison of images in experiments. The pseudo siamese network is the same as the siamese network in structure. The only difference between them is that the pseudo network does not share weight. In this way, the computation time of pseudo siamese and siamese is the same in the forward process. However in learning process, pseudo siamese should learn much more weight and need more time. This is another reason why we set the same weight of branches in our network.

3 Proposed Method

The procedure of the proposed framework is summarized in Fig. 1. The input of our network is the superpixels of images divided by [3]. Firstly, We obtain the single image saliency map through Fast-MDB algorithm in SISM step. At the beginning of MISM step, an image comparison model based on siamese network is trained. By utilizing this model, we obtain the similarity parameter matrix of each superpixel between interimage and intraimage. The multi image saliency map is generated combining the similarity matrix with scores of single image saliency map. The final saliency map is the linear combination of single and multi image saliency map.

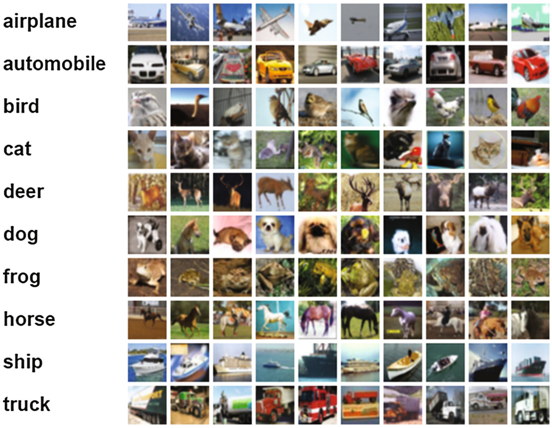

Fig. 3. Cifar-10 dataset for training

3.1 Multi Images Saliency Detection

Network Training. Because of the particularity of siamese network structure, the loss function and training step of siamese network are different from those of the ordinary single branch deep learning network. The training of siamese network generally uses contrast loss as the loss function. Similar to classification, Siamese network mainly focuses on the difference of image features instead of the classification score. As shown in Fig. 2, the distance function in our network is defined as:

$$D_W(\overrightarrow{X_1}, \overrightarrow{X_2}) = \left\| G_W(\overrightarrow{X_1}) - G_W(\overrightarrow{X_2}) \right\|_2 \tag{1}$$

$\overrightarrow{X_1}$ and $\overrightarrow{X_2}$ is the input pair of our network, and G_W means the operation done in each branch. In fact, the distance function is the Euclidean distance of two network outputs. The loss function is defined as follows.

$$L = (1 - Y)L_S(D_w) + Y L_D(D_w) \tag{2}$$

Where Y represents the label of the input pair. For example, $Y = 1$ indicates that two inputs are the same, otherwise $Y = 0$. Obviously, L_S means the partial loss for the same inputs and L_D represents for the different. For L_S and L_D, it is necessary to ensure that L is smaller when the inputs are the same and larger when different after training. After experiments, we obtain our loss function:

$$L = \frac{1}{2}(1 - Y)(D_W)^2 + \frac{1}{2}Y[\max(0, m - D_W)]^2 \tag{3}$$

We adopt Cifar-10 as our training dataset as shown in Fig. 3. Cifar-10 has a number of advantages: the regular images and label format; easy to read; large number of images in each class for training. Compared to MNIST, image in Cifar-10 is smaller enough with 32 height and 32 width, and this will reduce the complexity of training. In the training step, we first select an image from each class as one input, and select another image from the same class while setting the label $Y = 1$. At the same time, we select another image from the different class and set the $Y = 0$. After numbers of iterations, the image comparison model will be obtained.

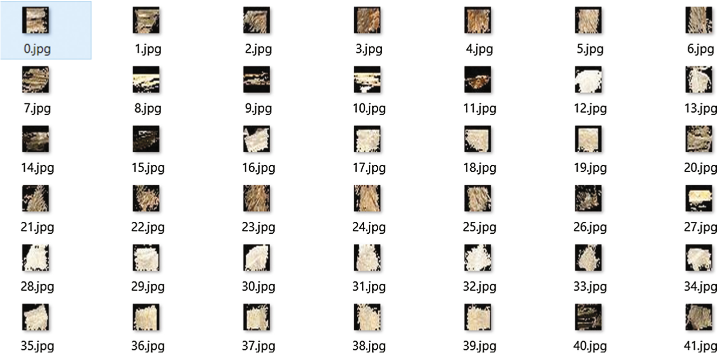

Fig. 4. Result of image preprocessing

Preprocessing. For multi image saliency detection, each image should be divided into a number of superpixels firstly. Each superpixel will be filled to an image patch with black background. Center of the superpixel locates at the center of image patch. Other pixels will be put in the patch according to the original location. Since the generated superpixels vary greatly in sizes, the image patch should be adaptive to cover all the pixels in each superpixel without being too small. In our experiments, we assume that S_k is the quantity of superpixels, and the size of the patch is $A/(\sqrt{S_k} - n)$. While A means the length of original image, and n is an adjustable parameter. This doesn't always get the best result. However it ensures that the computation complexity is still about the linear time of the image size, which is independent of the number of superpixels. According to the size of Cifar-10, each image patch will be reshaped to 32×32 finally and stored in hard drive shown in Fig. 4.

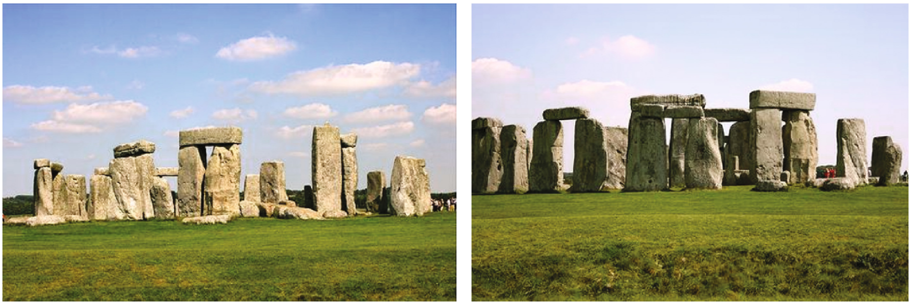

Fig. 5. Result of image preprocessing

Feature Extraction and Comparison. Our proposed method is to use the forward transmission of siamese network to directly replace the feature extraction and matching. For forward transmission, the input is the image patch pairs of the same image generated by preprocessing. By computing the feature differences between one image patch with all other patches in the same image, we can get the similarity of this patch. The other similarities can be computed in the same way. Finally, similarity matrix is obtained for each image. Based on this matrix, We define our score function as

$$Score_c = 0.5 + m - Sigmoid(2/d) \tag{4}$$

d represents the contrastive loss of two input patches. When d is large, which means there is big differences between two image patches, the score is close to 0. Otherwise, the score is close to 1. The score function reflect the negative feedback relationship between distance and score. However, there are some problems in our approach. As shown in Fig. 5, the grassland should be labeled as background. However the similarity matrix treats this area as the most frequently parts and labels foreground. Therefore, we make use of the single image saliency map to guide the generation of multi image saliency map. In our method, we construct a saliency score matrix S_s based on single image saliency map in superpixels. The saliency score matrix is

$$Score_{si} = \frac{\sum_{p_j \in SP_i} SISM(p_j)}{NUM} \tag{5}$$

SISM stands for the saliency value of single image saliency map. p_j is a pixel of image. NUM means the count of pixels for the i-th superpixel SP_i. As a consequence, $Score_{si}$ is the saliency score for the i-th superpixel. The final saliency value R for multi image is summarized

$$R = (Score_c + c_1) \times (Score_{si} + c_2) \tag{6}$$

While c_1 and c_2 are the gaussian noise avoiding too many zeros. Afterward, R is normalized to R_f, where R_f is the final saliency value for multi image.

3.2 Final Saliency Map

We obtain our final saliency map by linear combination with the single image saliency map and multi image saliency map. The equation is as follows.

$$SS(I_i) = \alpha_1 \cdot S_s(I_i) + \alpha_2 \cdot S_m(I_i) \tag{7}$$

S_s and S_m are the saliency map for single and multi image respectively. α_1 and α_2 stand for the weight. In our experiments, we set $\alpha_1 = 0.2$ and $\alpha_1 = 0.8$ to increase the weight of our multi image saliency map.

4 Experiments and Results

In this section, we describe our evaluation protocol and implementation details, provide exhaustive comparison results over two datasets, analysis the performance our methods.

4.1 Dataset and Metrics Method

We evaluate our method on two public datasets, including iCoseg [15] and MSRC [16]. The iCoseg dataset consists of 38 classes with 643 images. MSRC-v1 contains 9 object classed and 240 images. The results are compared with several state-of-art method, including Kim [17], Jou [18] and Fast-MDB [11]. Each image is segmented into 400 superpixels. We leverage the Precision(P) and Jaccard similarity coefficient(J) as evaluation indicators. P represents the proportion of the correct pixels in the final saliency map. J refers to the similarity between the detection result R and the ground true G, and it can be defined as

$$J = \frac{R \cap G}{R \cup G} \tag{8}$$

4.2 Results

Quantitative Results. Table 1 is the results of iCoseg dataset. As shown our method performs better than Kim and Jou in precision. However, our method does not have the advantage in J. It is worth noting that all the other methods, such as Kim, is actually a method for image cosegmentation. Although it is similar to saliency detection, the segmentation results generally guarantee the edge smoothing, and the final result is strong connectivity. Therefore, J can be higher than saliency method. For our method, loss of some low saliency values and wrong annotations lead to the lower J.

Compared to Fast-MDB, our method works little better in P and J. Nonetheless, our method labels some images that are not detected by Fast-MDB shown in Fig. 6. The two warriors belong to the foreground. However, Fast-MDB only labels some parts of edge while our method annotates the right warrior.

Table 1. Comparison with other methods

	Kim	Jou	Fast-MDB	Ours
P	70.2	70.4	82.8	83.2
J	42.6	39.7	38.7	41.4

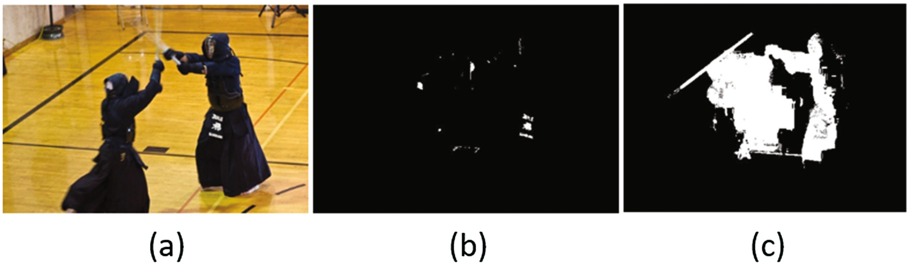

Fig. 6. Results of our method

Visualization. Combined with single and multi image saliency map, our method can correctly locate the approximate salient objects in the image. Figure 7 is an image chosen from iCoseg. (a) is the source image and (b) is the result segmented by [19]. (c) represents the saliency map generated by Fast-MDB. As we can see that some sharp edges, such as the stone crevice, make the map noisy. When the co-saliency method is added, it can better filter the noise below. The multi image saliency map plays an important role in co-saliency detection. The graph neutralizing the bear's own color close to the tree trunk and the obvious shadow can not be removed, which is also related to the limitations of the fast MDB algorithm itself focusing on the center of image.

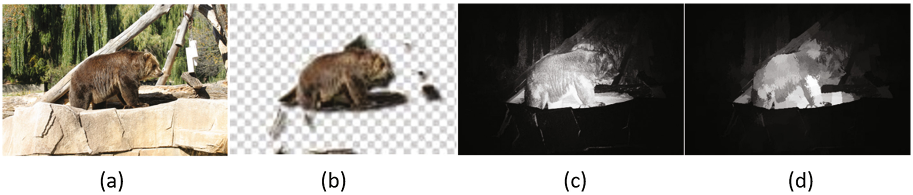

Fig. 7. Visualization of result

5 Conclusion

We proposed a new co-saliency detection method based on siamese network. The proposed method consists of two parts. One is to detect the single image saliency region with traditional method. Another is to generate the multi image saliency map through siamese network. The network is trained to evaluate the difference between two input image features. Therefore, similarity matrix is generated for each image in superpixels. Combing with saliency score matrix Guided by the exist single image saliency map, we can get the multi image saliency map. Afterward, the co-saliency map is the linear combination of single and multi image saliency maps. Finally, we conduct experiments on two public datasets, and the experiments show that our method performs better in metrics P and J. Our method provide an idea of network transmission. We can improve the compression algorithm to keep the saliency area as much as possible while reduce the size of images and videos.

References

1. Ma, Y., Zhang, H.: Contrast-based image attention analysis by using fuzzy growing. In: ACM International Conference on Multimedia, pp. 374–381 (2003)
2. Yan, Q., Xu, L., Shi, J., Jia, J.: Hierarchical saliency detection. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition(CVPR), pp. 1155–1162 (2013)
3. 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)
4. Cheng, M., Zhang, G., Mitra, N., Huang, X., Hu, S.: Global contrast based salient region detection. In: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 37(3), pp. 409–416 (2011)
5. Perazzi, F., Krahenbuhl, P., Pritch, Y., Hornung, A.: Saliency filters: contrast based filtering for salient region detection. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 733–740 (2012)
6. Li, H., Ngan, K.: A co-saliency model of image pairs. IEEE Trans. Image Process. **20**(12), 3365–3375 (2011)
7. Zhang, D., Han, J., Han, J., Shao, L.: Cosaliency detection based on intrasaliency prior transfer and deep intersaliency mining. IEEE Trans. Neural Netw. Learn. Syst. **27**(6), 1163–1176 (2016)
8. Tao, Z., Liu, H., Fu, H., Fu, Y.: Image cosegmentation via saliency-guided constrained clustering with cosine similarity. In: 13th AAAI Conference on Artificial Intelligence (2017)
9. Zhang, D., Han, J., Li, C., Wang, J.: Co-saliency detection via looking deep and wide. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2994–3002 (2015)
10. Strand, R., Ciesielski, K.C., Malmberg, F., Saha, K.: The minimum barrier distance. Comput. Vis. Image Underst. **117**(4), 429–437 (2013)
11. Zhang, J., Sclaroff, S., Lin, Z., Shen, X., Price, B., Mech, R.: Minimum barrier salient object detection at 80 FPS. In: 2015 IEEE International Conference on Image Processing, pp. 1404–1412 (2015)
12. Philipp, F., Alexey, D., Thomas, D.: Descriptor Matching with Convolutional Neural Networks: a Comparison to Sift. Computer Science (2014)
13. Žbontar, J., LeCun, Y.: Computing the stereo matching cost with a convolutional neural network. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1592–1599 (2015)

14. Zagoruyko, S., Komodakis, N.: Learning to compare image patches via convolutional neural networks. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4353–4361 (2015)
15. Dhruv, B., Adarsh, K., Devi, P., Luo, J., Chen, T.: iCoseg: interactive co-segmentation with intelligent scribble guidance. In: 2015 IEEE conference on Computer Vision and Pattern Recognition (CVPR), pp. 3169–3176 (2010)
16. Microsoft Research. http://research.microsoft.com/en-us/projects/objectclassrecognition/
17. Kim, G., Xing, E., Li, F., Kanade, T.: Distributed cosegmentation via submodular optimization on anisotropic diffusion. In: 2011 International Conference on Computer Vision, pp. 169–176 (2011)
18. Joulin, A., Bach, F., Ponce, J.: Multi-class cosegmentation. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 542–549 (2012)
19. Rubinstein, M., Joulin, A., Kopf, J., Liu, C.: Unsupervised joint object discovery and segmentation in internet images. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1939–1946 (2013)