Fast and Efficient Saliency Detection Using Sparse Sampling and Kernel Density Estimation

Hamed Rezazadegan Tavakoli, Esa Rahtu, and Janne Heikkilä

Machine Vision Group, Department of Electrical and Information Engineering,
University of Oulu, Finland
{hamed.rezazadegan,esa.rahtu,janne.heikkila}@ee.oulu.fi
http://www.ee.oulu.fi/mvg

Abstract. Salient region detection has gained a great deal of attention in computer vision. It is useful for applications such as adaptive video/image compression, image segmentation, anomaly detection, image retrieval, etc. In this paper, we study saliency detection using a center-surround approach. The proposed method is based on estimating saliency of local feature contrast in a Bayesian framework. The distributions needed are estimated particularly using sparse sampling and kernel density estimation. Furthermore, the nature of method implicitly considers what refereed to as center bias in literature. Proposed method was evaluated on a publicly available data set which contains human eye fixation as ground-truth. The results indicate more than 5% improvement over state-of-the-art methods. Moreover, the method is fast enough to run in real-time.

Keywords: Saliency detection, discriminant center-surround, eye-fixation.

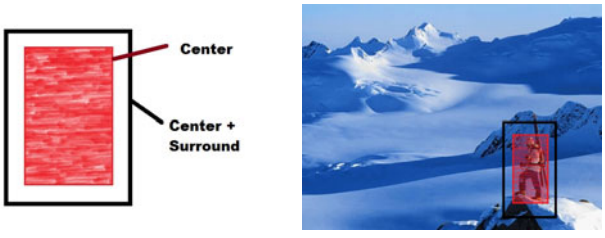
1 Introduction

Saliency detection in images and videos was introduced to computer vision in late 90s. One of the classical, most well-known papers is the one published by Itti et al. [1] in 1998. Their approach is based on extracting early visual features (e.g. colors, orientations, edges, ...) and fusing them into a saliency map in a three-step process.

Saliency detection has two key aspects: biological and computational. From the biological point of view, we can categorize saliency detection methods into top-down, bottom-up, and hybrid classes. In the top-down approach, it is assumed that the process of finding salient regions is controlled by high-level intelligence in brain. The main idea of bottom-up approach is that the process of saliency detection is an uncontrolled action on the shoulder of eye's receptors. Hybrid aspect believes in parallel and complementary role of top-down and bottom-up approaches.

Considering the computational view, we grouped saliency detection into different paradigms. One well-known class of algorithms is the center-surround technique. In this paradigm, the hypothesis is that there exists a local window divided into a center and a surround; and the center contains an object. Figure 1

A. Heyden and F. Kahl (Eds.): SCIA 2011, LNCS 6688, pp. 666–675, 2011.



(a) (b)

Fig. 1. An example showing center surround concept

shows this concept. Achanta et al. [2] provides us such a sample. They measured the color difference of a center pixel and average color in its immediate surrounding. Seo and Milanfar [3] used Local Steering Kernel (LSK) response as a feature and applied Parzen window density estimation to estimate the probability of having an object in each local window. Rahtu et al. [4] employed histogram estimation over contrast features in a window.

Frequency domain methods can be considered another category. Examples of such techniques can be found in [5,6,7]. Hou and Zhang [6] proposed a method based on relating extracted spectral residual features of an image in the spectral domain to the spatial domain. In [5] phase spectrum of quaternion Fourier transform is utilized to compute saliency. Achanta et al. [7] introduced a technique which relies on reinforcement of regions with more information.

Another class of algorithms relies on information theory concepts. In [8] a technique based on self-information is introduced to compute the likelihood of having a salient region. Lin et al. [9] employed local entropy to detect a salient region of an image. Mahadevan and Vasconcelos [10] utilized Kullback-Leibler divergence to measure mutual information to compute saliency.

In this paper, we introduce a method which belongs to the center-surround category. The major difference between the proposed technique and other similar methods is that it uses sparse sampling and kernel density estimation to build the saliency map. Also, proposed method's nature implicitly includes center bias. The method is tested on a publicly available data set. Finally, it is shown that the proposed method is fast and accurate.

2 Saliency Measurement

In this section, general Bayesian framework toward a center-surround approach is initially discussed. Afterwards, basics of proposed method are explained. It is followed by introducing the multi-scale extension. Finally, a brief explanation about implementation and algorithm parameters is provided.

2.1 Bayesian Center-Surround

Let us assume that we have an image I. We define each pixel as $x = (\bar{x}, f)$ where $\bar{x}$ is the coordinate of pixel x in image I, and f is a feature vector for each

coordinate. So, f can be a gray-scale value, a color vector, or any other desired feature (e.g., LBP, Gabor, SIFT, LSK, ...).

Suppose, there exists a binary random variable H_x that defines pixel saliency. It is defined as follows:

$$H_x = \begin{cases} 1, & \text{if } x \text{ is salient} \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

The saliency of pixel x can be computed using $P(H_x = 1|f) = P(1|f)$. It can be expanded using the Bayes rule as follows:

$$\frac{P(f|1)P(1)}{P(f|1)P(1) + P(f|0)P(0)}. \tag{2}$$

In the center-surround approach, we have a window W divided into a surround B and center K where the hypothesis is that K contains an object. In fact, pixels in K contribute to $P(f|1)$, and pixels in B contribute to $P(f|0)$. Having a sliding window W, we can sweep the whole image and calculate the saliency value locally.

The difference between center-surround methods is the way they deal with $P(1|f)$. For instance, Rahtu et al. [11] estimate (2), by approximating both $P(f|1)$ and $P(f|0)$ using histogram approximation over pixels' color values. Moreover, they assume $P(0)$ and $P(1)$ are constant. Seo and Milanfar [3] suppose $P(1|f) \propto P(f|1)$ and apply Parzen window estimation over LSK features to approximate $P(f|1)$.

2.2 Defining Saliency Measure

We define saliency measure for x belonging to center utilizing $P(1|f, \bar{x})$. Applying Bayes' theorem, we can write:

$$P(1|f, \bar{x}) = \frac{P(f|\bar{x}, 1)P(1|\bar{x})}{P(f|\bar{x})}. \tag{3}$$

This can be further expanded to:

$$P(1|f, \bar{x}) = \frac{P(f|\bar{x}, 1)P(1|\bar{x})}{P(f|\bar{x}, 1)P(1|\bar{x}) + P(f|\bar{x}, 0)P(0|\bar{x})}. \tag{4}$$

Computing (4) require the estimation of $P(f|\bar{x}, 1)$ and $P(f|\bar{x}, 0)$, which can be done in several ways. For instance in order to estimate $P(f|1)$ and $P(f|0)$, in [12,13,14] a generalized Gaussian model is used, in [3] Parzen window estimation was adapted, and Histogram estimation is applied in [11]. We adapt kernel density estimation method to compute feature distribution. As a result, we can write:

$$P(1|f, \bar{x}) = \frac{\frac{1}{m}\sum_{i=1}^{m} \mathcal{G}\left(f - f_{\bar{x}_{Ki}}\right) P(1|\bar{x})}{\frac{1}{m}\sum_{i=1}^{m} \mathcal{G}\left(f - f_{\bar{x}_{Ki}}\right) P(1|\bar{x}) + \frac{1}{n}\sum_{i=1}^{n} \mathcal{G}\left(f - f_{\bar{x}_{Bi}}\right) P(0|\bar{x})}, \tag{5}$$

where n and m are the number of samples, $x_{Bi} = (\bar{x}_{Bi}, f_{\bar{x}_{Bi}})$ is the i_{th} sample belonging to B and $x_{Ki} = (\bar{x}_{Ki}, f_{\bar{x}_{Ki}})$ is the i_{th} sample belonging to K, and $\mathcal{G}(.)$ is a Gaussian kernel.

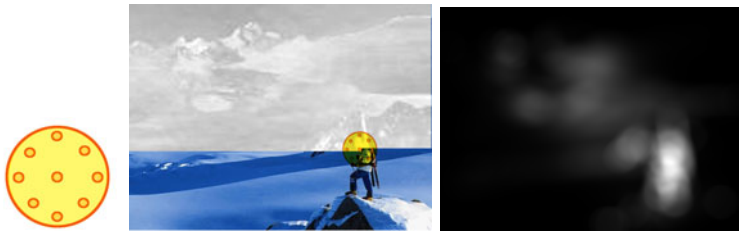
Since we plan to compute saliency of pixel x belonging to center, (5) can be simplified by selecting K as small as a pixel. In that case, we have

$$\mathcal{G}(f - f_{\bar{x}_{Ki}}) = \frac{1}{\sqrt{2\pi}\sigma_1} exp(-\frac{\|f - f\|^2}{2\sigma_1^2}) = \frac{1}{\sqrt{2\pi}\sigma_1}, \tag{6}$$

where σ_1 is standard deviation. Afterwards, we assume that only a few samples from B, which are scattered uniformly on a hypothetical circle with radius r contribute to $P(f|\bar{x}, 0)$. In fact, by substituting (6) into (5) and knowing the fact that $P(0|\bar{x}) + P(1|\bar{x}) = 1$, we can write:

$$P_r^n(1|f, \bar{x}) = 1 \Bigg/ \left(1 + \frac{\sigma_1(1 - P(1|\bar{x}))}{n\sigma_0 P(1|\bar{x})} \sum_{i=1}^{n} exp\left(\frac{\|f - f_{\bar{x}_{Bi,r}}\|^2}{2\sigma_0^2}\right)\right), \tag{7}$$

where σ_1 and σ_0 are standard deviations, n is the number of samples form B and r shows the radius at which samples will be taken. Figure 2 illustrates an example of such a central pixel and sample pixels around it. This sparse sampling reduces the number of operations required to estimate $P(f|\bar{x}, 0)$ and increases computation speed.



(a) (b) (c)

Fig. 2. (a) A pixel and its selected surrounding samples in a window, (b) Procedure of applying a window, (c) A sample saliency map obtained, using proposed method

In order to approximate the distribution $P(1|\bar{x})$, we compute average fixation map over a training set. Also, to avoid zero value we biased the obtained probability by adding $b = 0.1$. We further smooth the estimated distribution using a Gaussian kernel of size 30×30 and $\sigma = 20$. Figure 3 shows the probability obtained.

We define saliency in terms of sampling circle radius and number of samples as follows:

$$S_r^n(x) = \mathcal{A}_c * [P_r^n(1|f, \bar{x})]^\alpha, \tag{8}$$

where $\mathcal{A}_c$ is a circular averaging filter, $*$ is convolution operator, $P_r^n(1|f, \bar{x})$ is calculated using (7), and $\alpha \geq 1$ is an attenuation factor which emphasizes the effect of high probability areas.

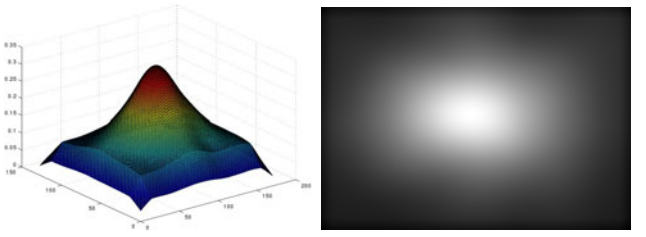


Fig. 3. Estimated $P(1|\bar{x})$. It is generated by low pass filtering an average fixation map obtained from several fixation maps.

Center bias. There exists evidence that human eyes fixates mostly on the center of image [15]. This is because most of the human taken photos are taken in such a way to have the subject in the center of the image. This knowledge can be used to improve saliency detection performance. Saliency detection benefits from center bias by giving more weight to the center. For instance, Judd et al. [15] applied an arbitrary Gaussian blob on the center of the image to centrally bias their technique. Yang et al. [14] learned the central bias by learning a normal Bivariate Gaussian over the eye fixation distribution.

In our method $P(1|\bar{x})$ gives weight to the positions more probable to observe an object. However, since we learn it from human taken photos, it can be considered equivalent to center bias in this case. Studying figure 3 conveys the same concept.

2.3 Multi-scale Measure

Many techniques apply the multi-scale approach toward saliency. The reason for such an approach is that each image may consist of objects of different sizes. Generally multi-scale property is achieved by changing the size of W. In order to make our approach multi-scale, it is only needed to change the radius and number of samples. We refer to the radius as "size scale" denoted by r and to the number of samples as "precision scale" denoted by n. Computing saliency of a pixel at different scales we take the average over all scales:

$$S(x) = \frac{1}{M} \sum_{i=1}^{M} S_{r_i}^{n_i}(x), \tag{9}$$

where M is the number of scales, $S_{r_i}^{n_i}(x)$ is the i_{th} saliency map calculated at a different scale using (8).

2.4 Implementation

In our implementation, we used CIELab color vector as feature. So for any pixel $x = (\bar{x}, f)$, we have $f = [L(\bar{x}), a(\bar{x}), b(\bar{x})]$ where $L(\bar{x})$, $a(\bar{x})$ and $b(\bar{x})$ are CIELab values at $\bar{x}$. In order to reduce the effect of noise, we employed a low-pass filter. For this purpose, we used a Gaussian kernel of size 9×9 with standard deviation

$\sigma = 0.5$. We also normalized all the images to 171×128 to make easier the process of images with different sizes.

We applied three different size scales with fixed precision scales. The parameters were $r = [13, 25, 38]$, $n = [8, 8, 8]$, $\sigma_1 = [1, 1, 1]$, and $\sigma_0 = [10, 10, 10]$. The attenuation parameter α was set to 25, and an averaging disk filter of radius 10 was applied. All the tests were performed by means of MATLAB R 2010a on a machine with Intel 6600 CPU running at 2.4 GHz clock, and 2GB RAM.

3 Experiments

This section is dedicated to the evaluation of proposed saliency detection technique. We compare proposed technique with Achanta et al. [7], Achanta et al. [2], Zhang et al.[1] [13], Seo and Milanfar[2] [3], Goferman et al.[3] [16], Rahtu et al.[4] [11], Bruce and Tsotsos[5][8], and Hou et al.[6] [6]. We use available public codes for all the methods except for [2,7]. The parameters used for each algorithm are the same as reported in the original paper. We provide qualitative and quantitative tests to show the pros and cons of each technique. Also, we evaluate running time of each method.

We used data set released by Bruce and Tsotsos in [8]. The data set consists of 120 images of size 681×511 and eye fixation maps. Almost all of images are composed of everyday life situations which makes the data set a difficult one. We divide the data set into the train and test sets containing 80 and 40 images, respectively. $P(1|\bar{x})$ was obtained using the training set.

3.1 Quantitative Analysis

Receiver Operating Characteristic (ROC) curve is a method of evaluating saliency maps using eye fixation density maps [8,15,14]. In this method, a threshold is varied over saliency map; and number of fixated and non-fixated points are counted at each threshold value. Amount of true positive rate and false positive rate are obtained by comparing the results with a reference saliency map. These values will build the ROC curve.

In order to perform quantitative analysis, we use a similar approach as in [8]. In fact, we moved the threshold value from zero to maximum pixel value, and computed true positive and false positive values. Eventually, reported the average value over all images. Figure 4 depicts the resulting curves. Table 1 reports the Area Under the ROC (AUC). As it can be seen from both Figure 4 and Table 1, proposed method outperforms all the other methods with a considerable margin.

[1] http://cseweb.ucsd.edu/~l6zhang/

[2] http://users.soe.ucsc.edu/~rokaf/SaliencyDetection.html

[3] http://webee.technion.ac.il/labs/cgm/Computer-Graphics-Multimedia/Software/Saliency/Saliency.html

[4] http://www.ee.oulu.fi/mvg/page/saliency

[5] www.cse.yorku.ca/~neil

[6] http://www.its.caltech.edu/~xhou/projects/spectralResidual/spectralresidual.html

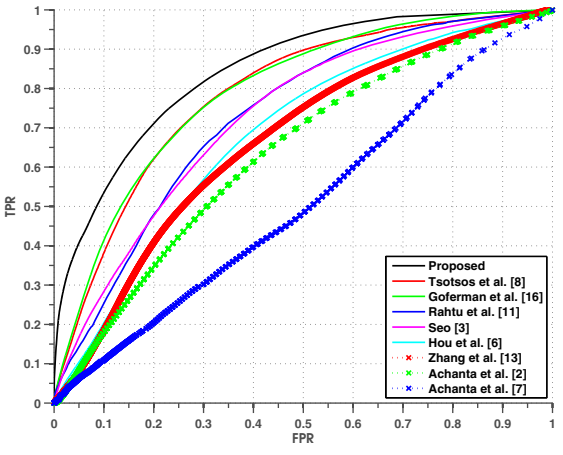


Fig. 4. Comparing the performance of proposed method and other state-of-the-art techniques in terms of Receiver Operating Characteristic (ROC) curve

Table 1. Comparison of different methods in terms of Area Under the Curve (AUC). Mean ± standard deviation is reported.

Algorithm	**AUC**
Achanta et al. [7]	0.5024 ± 0.1216
Achanta et al. [2]	0.6447 ± 0.1067
Zhang et al. [13]	0.6719 ± 0.0874
Hou et al. [6]	0.6863 ± 0.1117
Seo and Milanfar [3]	0.7292 ± 0.0972
Rahtu et al. [11]	0.7495 ± 0.0624
Goferman et al. [16]	0.8022 ± 0.0844
Bruce and Tsotsos [8]	0.7971 ± 0.0691
Proposed	0.8614 ± 0.0648

Table 2. Comparison of different methods in terms of running time

Algorithm	**Timing**(msec/pixel)
Achanta et al. [7]	1.25e-3
Achanta et al. [2]	3.71e-3
Zhang et al. [13]	0.20
Hou et al. [6]	6e-3
Seo and Milanfar [3]	0.58
Rahtu et al. [11]	6.5e-2
Goferman et al. [16]	1.43
Bruce and Tsotsos [8]	8.7e-2
Proposed	7.6e-3

Table 2 summarizes the measured running time per pixel. Although the proposed method is not the fastest method in the list, it is the fastest among high-performance methods.

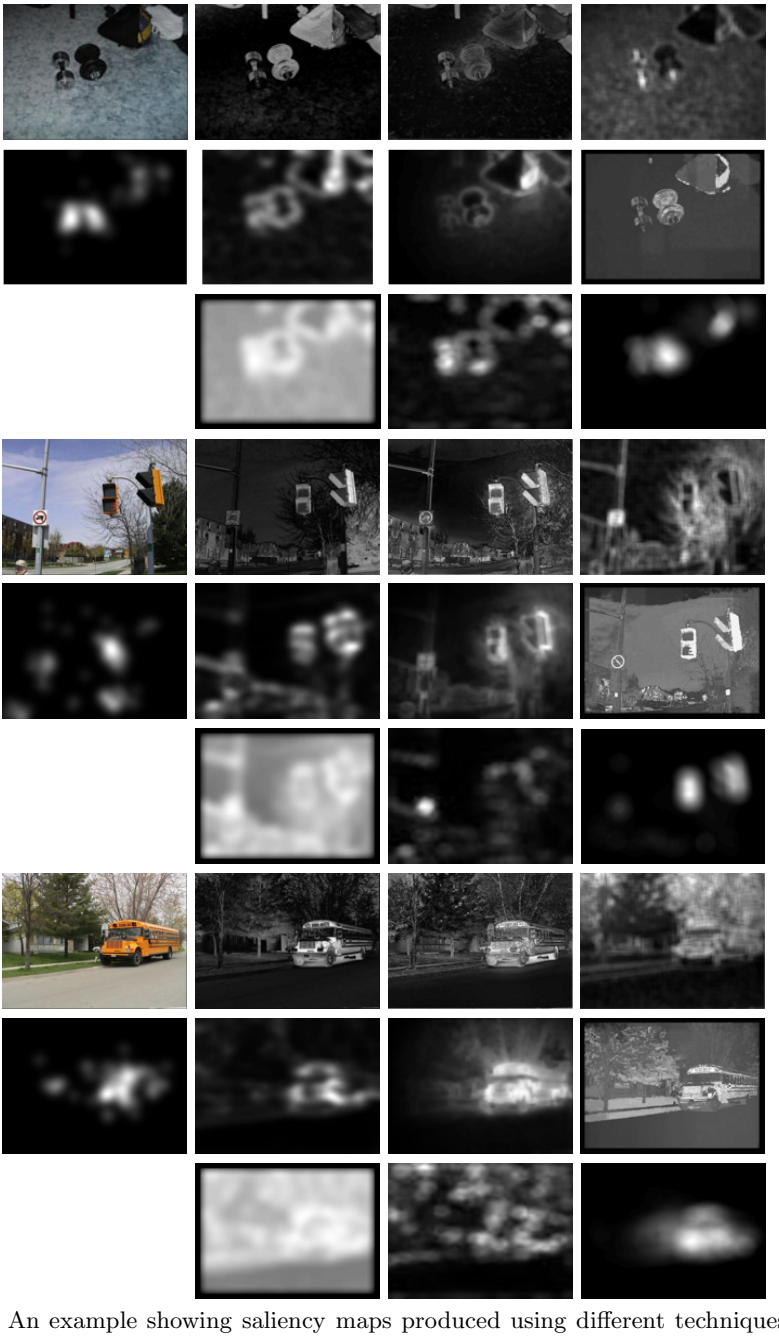


Fig. 5. An example showing saliency maps produced using different techniques. The leftmost column shows the original image and its fixation map. On the right side from left to right and top to bottom results from Achanta et al. [7], Achanta et al. [2], Zhang et al. [13], Seo and Milanfar [3], Goferman et al. [16], Rahtu et al. [11], Bruce and Tsotsos [8], Hou et al. [6], and Proposed method are depicted.

3.2 Qualitative Assessment

In order to have better conception, we provide some sample images. Figure 5 shows saliency maps produced by several methods. The Human eye fixation map of each image is also provided for comparison.

4 Conclusion

In this paper, We introduced a new saliency technique based on center-surround approach. We showed that the proposed method can effectively compute the amount of saliency in images. It is fast in comparison to other similar approaches.

We introduced a method which utilizes $P(1|f, \bar{x})$ to measure saliency. We used sparse sampling and kernel density estimation to build the saliency map. The proposed method's nature implicitly includes center bias.

We compared the proposed method with eight state-of-the-art algorithms. We considered running time, and area under the curve in evaluation of methods. The method is the best technique in terms of AUC. Also, it is the fastest accurate method in comparison to other techniques.

References

1. Itti, L., Koch, C., Niebur, E.: A model of saliency-based visual attention for rapid scene analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence 20, 1254–1259 (1998)
2. Achanta, R., Estrada, F.J., Wils, P., Süsstrunk, S.: Salient region detection and segmentation. In: Gasteratos, A., Vincze, M., Tsotsos, J.K. (eds.) ICVS 2008. LNCS, vol. 5008, pp. 66–75. Springer, Heidelberg (2008), http://icvs2008.info/index.htm
3. Seo, H.J., Milanfar, P.: Training-free, generic object detection using locally adaptive regression kernels. IEEE Transactions on Pattern Analysis and Machine Intelligence 32, 1688–1704 (2010)
4. Rahtu, E., Heikkilä, J.: A simple and efficient saliency detector for background subtraction. In: Proc. the 9th IEEE International Workshop on Visual Surveillance (VS 2009), Kyoto, Japan, pp. 1137–1144 (2009), http://www.ee.oulu.fi/mvg/page/saliency
5. Guo, C., Ma, Q., Zhang, L.: Spatio-temporal saliency detection using phase spectrum of quaternion fourier transform. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008, pp. 1–8 (2008), doi:10.1109/CVPR.2008.4587715
6. Hou, X., Zhang, L.: Saliency detection: A spectral residual approach. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2007, pp. 1–8 (2007), doi:10.1109/CVPR.2007.383267
7. Achanta, R., Hemami, S., Estrada, F., Süsstrunk, S.: Frequency-tuned Salient Region Detection. In: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Miami Beach, Florida (2009), http://www.cvpr2009.org/

8. Tsotsos, J.K., Bruce, N.D.B.: Saliency based on information maximization. In: Weiss, Y., Schölkopf, B., Platt, J. (eds.) Advances in Neural Information Processing Systems 18, pp. 155–162. MIT Press, Cambridge (2006)
9. Lin, Y., Fang, B., Tang, Y.: A computational model for saliency maps by using local entropy. In: AAAI Conference on Artificial Intelligence (2010)
10. Mahadevan, V., Vasconcelos, N.: Spatiotemporal saliency in dynamic scenes. IEEE Transactions on Pattern Analysis and Machine Intelligence 32, 171–177 (2010)
11. Rahtu, E., Kannala, J., Salo, M., Heikkilä, J.: Segmenting salient objects from images and videos. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6315, pp. 366–379. Springer, Heidelberg (2010), http://www.ee.oulu.fi/mvg/page/saliency
12. Gao, D., Mahadevan, V., Vasconcelos, N.: On the plausibility of the discriminant center-surround hypothesis for visual saliency. Journal of Vision 8(7) (2008), http://www.journalofvision.org/content/8/7/13.abstract, doi:10.1167/8.7.13
13. Zhang, L., Tong, M.H., Marks, T.K., Shan, H., Cottrell, G.W.: Sun: A bayesian framework for saliency using natural statistics. Journal of Vision 8(7) (2008), http://www.journalofvision.org/content/8/7/32.abstract, doi:10.1167/8.7.32
14. Yang, Y., Song, M., Li, N., Bu, J., Chen, C.: What is the chance of happening: A new way to predict where people look. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6315, pp. 631–643. Springer, Heidelberg (2010), http://dx.doi.org/10.1007/978-3-642-15555-0_46
15. Judd, T., Ehinger, K., Durand, F., Torralba, A.: Learning to predict where humans look. In: IEEE International Conference on Computer Vision, ICCV (2009)
16. Goferman, S., Zelnik-Manor, L., Tal, A.: Context-aware saliency detection. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2376–2383 (2010), doi:10.1109/CVPR.2010.5539929