To appear in
ICASSP 2010

VISUAL SALIENCY FOR AUTOMATIC TARGET DETECTION, BOUNDARY DETECTION, AND IMAGE QUALITY ASSESSMENT

Hae Jong Seo and Peyman Milanfar

Electrical Engineering Department
University of California at Santa Cruz
{rokaf,milanfar}@soe.ucsc.edu

ABSTRACT

We present a visual saliency detection method and its applications. The proposed method does not require prior knowledge (learning) or any pre-processing step. Local visual descriptors which measure the likeness of a pixel to its surroundings are computed from an input image. *Self-resemblance* measured between local features results in a scalar map where each pixel indicates the statistical likelihood of saliency. Promising experimental results are illustrated for three applications: automatic target detection, boundary detection, and image quality assessment.

Index Terms— Visual Saliency, Automatic Target Detection, Image Quality Assessment, Boundary Detection

1. INTRODUCTION

The human visual system has a remarkable ability to automatically attend to only salient regions, known as focus of attention (FOA) in complex scenes. This ability enables us to allocate limited perceptual and cognitive resources on task-relevant visual input. It is well known that FOA also plays an important role when humans perceive visual quality of images. The goal of machine vision systems is to predict and mimic the human visual system. For this reason, visual saliency detection has been of great research interest [1, 2, 3] in recent years.

Analysis of visual attention has benefited a wide range of applications such as object and action recognition, image quality assessment and more. Gao et al. [4] used discriminative saliency detection for visual recognition and showed good performance on PASCAL 2006 dataset. Saliency-based space-time feature points have been successfully employed for action recognition by Rapantzikos et al. [5]. Ma and Zhang [6] showed performance improvement by simply applying saliency based weights to local structural similarity (SSIM) [4]. Most existing saliency detection methods are based on parametric models [2, 3] which use Gabor or difference of Gaussian filter responses and fit a conditional PDF of filter responses to a multivariate exponential distribution.

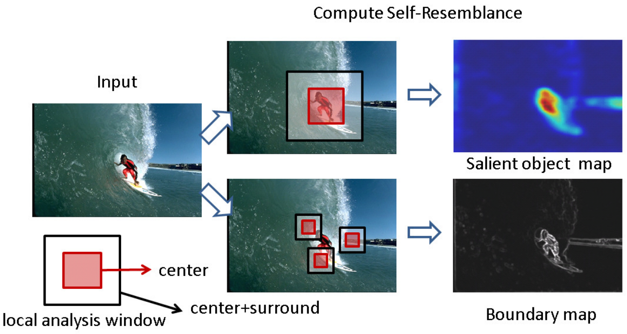

Fig. 1. Self-resemblance reveals a salient object or object's boundary according to the size of local analysis window.

In our previous work [7], we proposed a bottom-up saliency detection method based on a local *self-resemblance* measure. A nonparametric kernel density estimation for local non-linear features results in a scalar value at each pixel, indicating likelihood of saliency. As for local features, we employ local steering kernels which, fundamentally differ from conventional filter responses, but capture the underlying local geometric structure even in the presence of significant distortions. Our computational saliency model exhibits state-of-the art performance on the challenging set of data [1].

In this paper, we address three applications: 1) automatic target detection, 2) boundary detection, and 3) no-reference image quality metric, where our computational model for visual saliency can be applied to. In the following section, we first review saliency detection by self-resemblance and reveal the relation between boundary detection and saliency detection in the proposed framework.

2. SALIENCY BY SELF-RESEMBLANCE

Given an image I, we measure saliency at a pixel in terms of how much it stands out from its surroundings [7]. To formalize saliency at each pixel, we let the binary random variable y_i equal to 1 if a pixel position $\mathbf{x}_i = [x_1, x_2]_i^T$ is salient or 0 where $i = 1, \cdots, M$ (where M is the total number of pixels

This work was supported by AFOSR Grants FA9550-07-1-0365.

in the image.) Saliency at pixel position $\mathbf{x}_i$ is defined as a posterior probability as follows:

$$S_i = Pr(y_i = 1|\mathbf{F}), \tag{1}$$

where the feature matrix, $\mathbf{F}_i = [\mathbf{f}_i^1, \cdots, \mathbf{f}_i^L]$ at a pixel of interest $\mathbf{x}_i$ (what we call a center feature) contains a set of feature vectors ($\mathbf{f}_i$) in a local neighborhood where L is the number of features in that neighborhood. In turn, the larger collection of features $\mathbf{F} = [\mathbf{F}_1, \cdots, \mathbf{F}_N]$ is a matrix containing features not only from the center, but also a surrounding region (what we call a center+surround region; see Fig. 1.) N is the number of features in the center+surround region.

Using Bayes' theorem with assumptions that 1) a-priori, every pixel is considered to be equally likely to be salient; and 2) $p(\mathbf{F})$ are uniform over features, the saliency we defined boils down to the conditional probability density $p(\mathbf{F}|y_i = 1)$ which can be approximated by using nonparametric kernel density estimation [8]. More specifically, we define the conditional probability density $p(\mathbf{F}|y_i = 1)$ at $\mathbf{x}_i$ as a center value of a normalized adaptive kernel (weight function) $G(\cdot)$ computed in the center+surround region as follows:

$$S_i = \widehat{p}(\mathbf{F}|y_i = 1) = \frac{G_i(\mathbf{F}_i, \mathbf{F}_i)}{\sum_{j=1}^{N} G_i(\mathbf{F}_i, \mathbf{F}_j)}, \tag{2}$$

where the kernel function $G_i(\mathbf{F}_i, \mathbf{F}_j) = \exp\left(\frac{-1+\rho(\mathbf{F}_i,\mathbf{F}_j)}{\sigma^2}\right)$ and σ is a parameter controlling the fall-off of weights. Here, $\rho(\mathbf{F}_i, \mathbf{F}_j)$ is called *Matrix Cosine Similarity* which is a generalized version of vector cosine similarity to the matrix case. By inserting G into (2), S_i can be rewritten as follows:

$$S_i = \frac{1}{\sum_{j=1}^{N} \exp\left(\frac{-1+\rho(\mathbf{F}_i,\mathbf{F}_j)}{\sigma^2}\right)}, \tag{3}$$

where the denominator is called *self-resemblance*. S_i reveals how salient the center feature $\mathbf{F}_i$ is given all the features $\mathbf{F}_j$'s in its neighborhood. We refer the reader to [7, 9] for more detail.

As shown in Fig. 1, a size of local analysis window determines an output of our saliency detection system. For instance, if we keep the local window size relatively small, saliency at fine scale (here, boundary and corner of objects) will be captured. Conversely, a large size of the local window allows us to generate salient object maps.

2.1. Local steering kernels as features

Local steering kernels (LSK) are employed as features, which fundamentally differ from image patches or conventional filter responses. LSKs have successfully been used for image restoration [10] and object and action detection [11, 12]. The key idea behind LSKs is to robustly obtain the local structure of images by analyzing the radiometric (pixel value) differences based on estimated gradients, and use this structure information to determine the shape and size of a canonical kernel. The local steering kernel is modeled as

$$K(\mathbf{C}_l, \mathbf{x}_l, \mathbf{x}_i) = \frac{\sqrt{\det(\mathbf{C}_l)}}{h^2} \exp\left\{\frac{(\mathbf{x}_l - \mathbf{x}_i)^T \mathbf{C}_l (\mathbf{x}_l - \mathbf{x}_i)}{-2h^2}\right\}, \tag{4}$$

where $l \in \{1, \cdots, P\}$, P is the number of pixels in a local window; h is a global smoothing parameter. The matrix $\mathbf{C}_l \in \mathbb{R}^{2\times2}$ is a covariance matrix estimated from a collection of spatial gradient vectors within the local window around a position $\mathbf{x}_l$. Fig. 2 (Top) illustrates that how covariance matrices and LSK values are computed in an edge region.

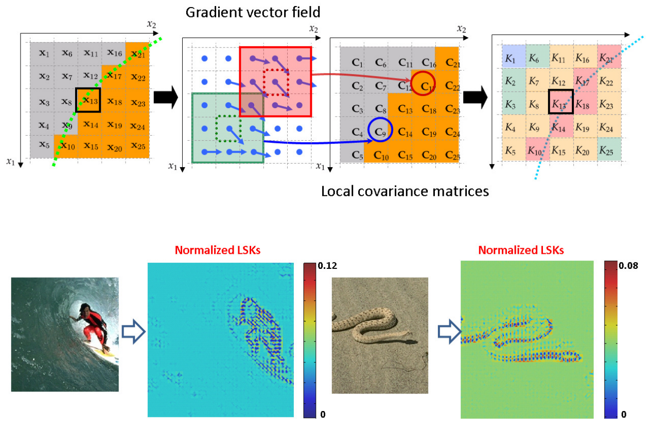
Gradient vector field
Local covariance matrices
x_1
x_2
$\mathbf{x}_1$ $\mathbf{x}_6$ $\mathbf{x}_{11}$ $\mathbf{x}_{16}$ $\mathbf{x}_{21}$
$\mathbf{x}_2$ $\mathbf{x}_7$ $\mathbf{x}_{12}$ $\mathbf{x}_{17}$ $\mathbf{x}_{22}$
$\mathbf{x}_3$ $\mathbf{x}_8$ $\mathbf{x}_{13}$ $\mathbf{x}_{18}$ $\mathbf{x}_{23}$
$\mathbf{x}_4$ $\mathbf{x}_9$ $\mathbf{x}_{14}$ $\mathbf{x}_{19}$ $\mathbf{x}_{24}$
$\mathbf{x}_5$ $\mathbf{x}_{10}$ $\mathbf{x}_{15}$ $\mathbf{x}_{20}$ $\mathbf{x}_{25}$
$\mathbf{C}_1$ $\mathbf{C}_6$ $\mathbf{C}_{11}$ $\mathbf{C}_{16}$ $\mathbf{C}_{21}$
$\mathbf{C}_2$ $\mathbf{C}_7$ $\mathbf{C}_{12}$ $\mathbf{C}_{17}$ $\mathbf{C}_{22}$
$\mathbf{C}_3$ $\mathbf{C}_8$ $\mathbf{C}_{13}$ $\mathbf{C}_{18}$ $\mathbf{C}_{23}$
$\mathbf{C}_4$ $\mathbf{C}_9$ $\mathbf{C}_{14}$ $\mathbf{C}_{19}$ $\mathbf{C}_{24}$
$\mathbf{C}_5$ $\mathbf{C}_{10}$ $\mathbf{C}_{15}$ $\mathbf{C}_{20}$ $\mathbf{C}_{25}$
K_1 K_6 K_{11} K_{16} K_{21}
K_2 K_7 K_{12} K_{17} K_{22}
K_3 K_8 K_{13} K_{18} K_{23}
K_4 K_9 K_{14} K_{19} K_{24}
K_5 K_{10} K_{15} K_{20} K_{25}
Normalized LSKs
0.12
0
Normalized LSKs
0.08
0

Fig. 2. Top: Graphical description of how LSK values centered at pixel of interest $\mathbf{x}_{13}$ are computed in an edge region. Bottom: Examples of LSKs: for graphical description, we only computed LSKs at non-overlapping 3×3 patch, even though we compute LSKs densely in practice.

In what follows, at a position $\mathbf{x}_i$, we will essentially be using (a normalized version of) the function K. LSK features are robust against signal uncertainty such as presence of noise and the normalized version of LSKs provide certain invariance to illumination changes [11]. Fig. 2 (Bottom) illustrates normalized LSKs computed from two images.

As mentioned earlier, the feature matrix $\mathbf{F}_i$ and $\mathbf{F}_j$ are constructed by using $\mathbf{f}$'s which are a normalized and vectorized version of K's. In the following section, we introduce three applications where our visual saliency is successfully applied to.

3. APPLICATIONS

3.1. Automatic target detection

Automatic target detection systems [13] have been developed mainly for military applications to assist or replace human experts whose performance might be inevitably degraded by fatigue following intensive and prolonged surveillance. In some situations, it is even impossible for the human experts to perform the task on site. The development of robust automatic target detection system is considered still challenging due to large variations in scale and rotation, occlusion, cluttered backgrounds, and different geographic and weather conditions. Conventional saliency approaches model a target

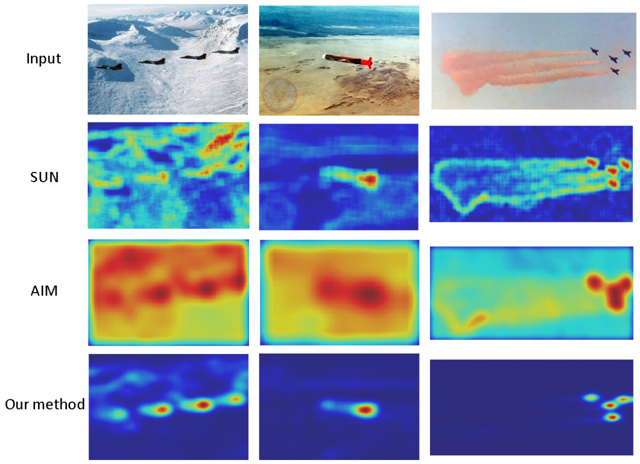

Fig. 3. Automatic target detection. Our saliency map consistently outperforms other state-of-the art methods (SUN [3] and AIM [1])

sought after and require a lot of training (both positive and negative) examples. However, our saliency detection method can automatically detect a salient target without any training.

In order to compute a salient object map, we need to use a large analysis window size. However, for efficiency, we downsample an image I to a coarse scale (i.e., 64×64). We then compute LSK of size 3×3 as features and generate feature matrices $\mathbf{F}_i$ in a 5×5 local neighborhood. The number of LSKs used in the feature matrix $\mathbf{F}_i$ is set to 9. The smoothing parameter h for computing LSK was set to 1 and the falloff parameter σ for computing self-resemblance was set to 0.07 for all the experiments. We obtained an overall salient object map by using CIE L*a*b* color space. Fig. 3 shows a comparison between our method and other state-of-the art methods. It is clear that our method consistently outperforms SUN [3] and AIM [1]. We refer the reader to [7, 9] for a further quantitative performance comparison on the challenging set of data [1].

3.2. Boundary detection

Boundary detection [15] is one of the most studied problems in computer vision, and serves as a basis for higher level applications such as object recognition, segmentation, and tracking. It is known that high-quality boundary detection still remains challenging since low-level cues such as image patch or gradient alone are generally not sufficient, but should be engaged with higher level information or supervised learning. However, it turns out that our visual saliency model can detect object boundary without any higher level information or training.

In order to generate a boundary map, we compute LSK of size 3×3 as features from an image I at its original scale and generate feature matrices $\mathbf{F}_i$ in a 3×3 local neighborhood. The rest of parameter settings remains the same as the case explained in the previous section. Fig. 4 shows boundary maps computed from some images [14]. The proposed boundary detection method tends to be sensitive to highly textured region, which is also the problem of current state-of-the art boundary detection methods. While the method by Maire et al. [16] detects contours first and tries to find junctions by using contour information, our method can automatically find junction and contour at the same time[1].

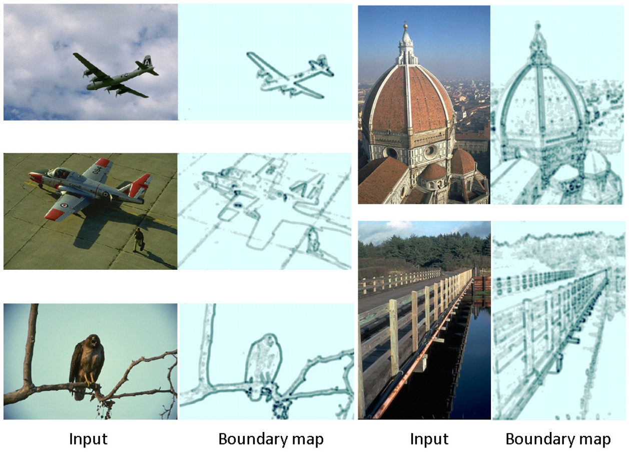

Fig. 4. Boundary detection examples on the Berkeley segmentation data set [14]

3.3. No-reference image quality metric

The goal of object quality metric is to accurately predict the perceived visual quality by human subjects. In general, objective quality metric [17] can be divided into two categories: full-reference and no-reference. Full-reference metrics calculate similarity between the target and reference images. Such measures of similarity include the classical mean-squared error (MSE), peak signal to noise ratio (PSNR), and Structural Similarity (SSIM) [17]. However, in most practical applications the reference image is not available. Therefore, in applications such as denoising, deblurring, and super-resolution, the quality metrics MSE, PSNR, or SSIM can not be directly used to assess the quality of output images or optimize the parameters of algorithms. Recently, Zhu and Milanfar [18] have proposed a no-reference quality metric Q which can automatically tune parameters of state of the art denoising algorithms such as ISKR [10]. Q metric is computed from a given noisy image by dividing it into non-overlapping patches of size 8×8, and discovering anisotropic patches based on so-called local coherence measure [18]. We provide a more reliable anisotropic patch selection method based on boundary detection explained in the earlier section. Since the features (LSKs) are robust to the presence of noise, the resulting

[1] Boundary map values at junction is higher than those at contour.

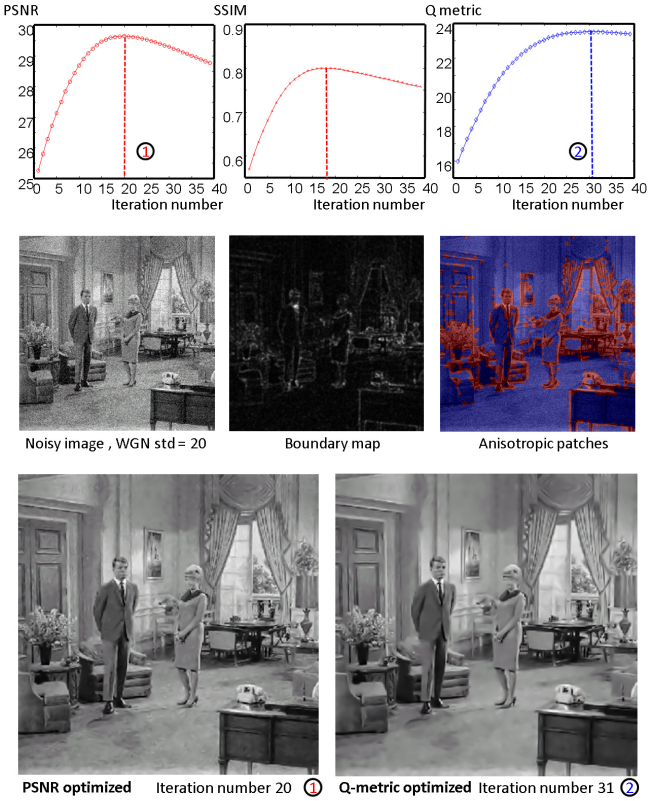

Fig. 5. Top: plots of PSNR,SSIM, and Q metric versus iteration number in ISKR denoising, Middle: a noisy image (std=20), boundary map, selected anisotropic patches (red area), Bottom: optimally denoised images by PSNR and Q

boundary map is not sensitive to noise, and thus provides a good way of finding anisotropic patches. To threshold the boundary map, we use the idea of nonparametric significance test. To be more specific, we compute empirical PDF of boundary map values, and set a threshold so as to achieve 70% confidence level. From plots of PSNR, SSIM, Q metric versus iteration number in ISKR [10] in Fig. 5, we can observe that the Q metric with our anisotropic patch selection behave similarly to full-reference metrics PSNR and SSIM. However, we claim that denoising results with optimal iteration number given by the Q metric is more visually pleasing than PSNR and SSIM. As shown in Fig. 5 (Bottom), a denoising result of ISKR selected by PSNR still contain noise and artifact around the door and floor.

4. CONCLUSION

In this paper, we introduced a visual saliency detection for three applications; 1) automatic target detection, 2) boundary detection, and 3) image quality assessment. We presented promising experimental results of each application. The proposed saliency detection method based on self-resemblance is practically appealing since no learning or pre-processing step are required.

5. REFERENCES

[1] N. Bruce and J. Tsotsos, "Saliency based on information maximization," *In Advances in Neural Information Processing Systems (NIPS)*, vol. 18, pp. 155–162, 2006.

[2] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transcations on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 20, pp. 1254–1259, 1998.

[3] L. Zhang, M.H. Tong, T.K. Marks, H. Shan, and G.W. Cottrell, "SUN: A Bayesian framework for saliency using natural statistics," *Journal of Vision*, vol. 8, no. 7, pp. 32,1–20, 2008.

[4] D. Gao, S. Han, and N. Vasconcelos, "Discriminant saliency, the detection of suspicious coincidences, and applications to visual recognition," *IEEE Transcations on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 31, no. 6, pp. 989–1005, 2009.

[5] K. Rapantzikos, Y. Avrithis, and S. Kollias, "Dense saliency-based spationtemporal feature points for action recognition," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[6] Q. Ma and L. Zhang, "Saliency-based image quality assessment criterion," *Advanced Intelligent Computing Theories and Applications. With Aspects of Theoretical and Methodological Issues (LNCS)*, vol. 5226, pp. 1124–1133, 2008.

[7] H. J. Seo and P. Milanfar, "Nonparametric bottom-up saliency detection by self-resemblance," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1st International Workshop on Visual Scene Understanding (ViSU09)*, Apr 2009.

[8] P. Vincent and Y. Bengio, "Manifold parzen windows," *In Advances in Neural Information Processing Systems (NIPS)*, vol. 15, pp. 825–832, 2003.

[9] H. J. Seo and P. Milanfar, "Static and space-time visual saliency detection by self-resemblance," *The Journal of Vision,*, vol. 9(12), no. 15, pp. 1–27, 2009.

[10] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Transactions on Image Processing (TIP)*, vol. 16, no. 2, pp. 349–366, February 2007.

[11] H. J. Seo and P. Milanfar, "Training-free, generic object detection using locally adaptive regression kernels," *To appear in IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010.

[12] H. J. Seo and P. Milanfar, "Generic human action detection from a single example," *IEEE International Conference on Computer Vision(ICCV)*, Sep 2009.

[13] L. A. Chan, S. Z. Der, and N. M. Nasarbadi, "Automatic target detection," *Encyclopedia of Optical Engineering*, vol. 1, pp. 101–113, 2003.

[14] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," *IEEE International Conference on Computer Vision (ICCV)*, 2001.

[15] D. Martin, C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE Transcations on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 26, no. 1, pp. 1–20, 2004.

[16] M. Maire, P. Arbelaez, C. Fowlkes, D. Tal, and J. Malik, "Using contours to detect and localize junctions in natural images," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[17] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Transactions on Image Processing (TIP)*, vol. 13, pp. 600–612, April 2004.

[18] X. Zhu and P. Milanfar, "Automatic parameter selection for denoising algorithms using a no-reference measure of image content," *Submitted to IEEE Transactions on Image Processing (TIP)*, 2009.