Static and space-time visual saliency detection by self-resemblance

Hae Jong Seo

Electrical Engineering Department, University of California, Santa Cruz, Santa Cruz, CA, USA

Peyman Milanfar

Electrical Engineering Department, University of California, Santa Cruz, Santa Cruz, CA, USA

We present a novel unified framework for both static and space-time saliency detection. Our method is a bottom-up approach and computes so-called local regression kernels (i.e., local descriptors) from the given image (or a video), which measure the likeness of a pixel (or voxel) to its surroundings. Visual saliency is then computed using the said "self-resemblance" measure. The framework results in a saliency map where each pixel (or voxel) indicates the statistical likelihood of saliency of a feature matrix given its surrounding feature matrices. As a similarity measure, matrix cosine similarity (a generalization of cosine similarity) is employed. State of the art performance is demonstrated on commonly used human eye fixation data (static scenes (N. Bruce & J. Tsotsos, 2006) and dynamic scenes (L. Itti & P. Baldi, 2006)) and some psychological patterns.

Keywords: saliency, attention, eye movements, computational modeling

Citation: Seo, H. J., & Milanfar, P. (2009). Static and space-time visual saliency detection by self-resemblance. *Journal of Vision, 9*(12):15, 1–27, http://journalofvision.org/9/12/15/, doi:10.1167/9.12.15.

Introduction

The human visual system has a remarkable ability to automatically attend to only salient locations in static and dynamic scenes (Chun & Wolfe, 2001; Itti & Koch, 2001; Yarbus, 1967). This ability enables us to allocate limited perceptual and cognitive resources on task-relevant visual input. In machine vision system, a flood of visual information fed into the system needs to be efficiently scanned in advance for relevance. In this paper, we propose a computational model for selective visual attention, otherwise known as visual saliency. In recent years, visual saliency detection has been of great research interest (Bruce & Tsotsos, 2006; Gao, Mahadevan, & Vasconcelos, 2008; Gao & Vasconcelos, 2005; Hou & Zhang, 2008a; Hou & Zhang, 2008b; Itti, Koch, & Niebur, 1998; Kanan, Tong, Zhang, & Cottrell, 2009; Marat et al., 2009; Torralba, Fergus, & Freeman, 2008; Zhang, Tong, & Cottrell, 2009; Zhang, Tong, Marks, Shan, & Cottrell, 2008). Analysis of visual attention has benefited a wide range of applications such as object detection, action detection, video summarization (Marat, Guironnet, & Pellerin, 2007), image quality assessment (Ma & Zhang, 2008; Niassi, LeMeur, Lecallet, & Barba, 2007) and more. There are two types of computational models for saliency according to what the model is driven by: a bottom-up saliency (Bruce & Tsotsos, 2006; Gao et al., 2008; Hou & Zhang, 2008a; Hou & Zhang, 2008b; Itti et al., 1998; Marat et al., 2009; Zhang et al., 2009; Zhang et al., 2008) and a top-down saliency (Gao & Vasconcelos, 2005; Kanan et al., 2009; Torralba et al., 2008). As opposed to bottom-up saliency algorithms that are fast and driven by low-level features, top-down saliency algorithms are slower and task-driven. In general, the plausibility of bottom-up saliency models is examined in terms of predicting eye movement data made by human observers in a task designed to minimize the role of top-down factors. Although some progress has been made by parametric saliency models (Gao et al., 2008; Itti et al., 1998; Torralba et al., 2008; Zhang et al., 2008) in predicting fixation patterns and visual search, there is significant room to further improve the accuracy. In this paper, we develop a nonparametric bottom-up visual saliency method which exhibits the state-of-the art performance. The problem of interest addressed is bottom-up saliency which can be described as follow: Given an image or a video, we are interested in accurately detecting salient objects or actions from the data without any background knowledge. To accomplish this task, we propose to use, as features, so-called *local steering kernels* and *space-time local steering kernels* which capture local data structure exceedingly well. Our approach is motivated by a probabilistic framework, which is based on a nonparametric estimate of the likelihood of saliency. As we describe below, this boils down to the local calculation of a "self-resemblance" map, which measures the similarity of a feature matrix at a pixel of interest to its neighboring feature matrices.

Previous work

Itti et al. (1998) introduced a saliency model which was biologically inspired. Specifically, they proposed to use a

doi: 10.1167/9.12.15 Received May 13, 2009; published November 20, 2009 ISSN 1534-7362 © ARVO

set of feature maps from three complementary channels as intensity, color, and orientation. The normalized feature maps from each channel were then linearly combined to generate the overall saliency map. Even though this model has been shown to be successful in predicting human fixations, it is somewhat ad-hoc in that there is no objective function to be optimized and many parameters must be tuned by hand. With the proliferation of eye-tracking data, a number of researchers have recently attempted to address the question of what attracts human visual attention by being more mathematically and statistically precise (Bruce & Tsotsos, 2006; Gao et al., 2008; Gao & Vasconcelos, 2004; Gao & Vasconcelos, 2005; Hou & Zhang, 2008a; Itti & Baldi, 2006; Zhang et al., 2008).

Bruce and Tsotsos (2006) modeled bottom-up saliency as the maximum information sampled from an image. More specifically, saliency is computed as Shannon's self-information $-\log p(\mathbf{f})$, where $\mathbf{f}$ is a local visual feature vector (i.e., derived from independent component analysis (ICA) performed on a large sample of small RGB patches in the image.) The probability density function is estimated based on a Gaussian kernel density estimate in a neural circuit.

Gao et al. (2008), Gao and Vasconcelos (2004), and Gao and Vasconcelos (2005) proposed a unified framework for top-down and bottom-up saliency as a classification problem with the objective being the minimization of classification error. They first applied this framework to object detection (Gao & Vasconcelos, 2005) in which a set of features are selected such that a class of interest is best discriminated from all other classes, and saliency is defined as the weighted sum of features that are salient for that class. In Gao et al. (2008), they defined bottom-up saliency using the idea that pixel locations are salient if they are distinguished from their surroundings. They used difference of Gaussians (DoG) filters and Gabor filters, measuring the saliency of a point as the Kullback–Leibler (KL) divergence between the histogram of filter responses at the point and the histogram of filter responses in the surrounding region. Mahadevan and Vasconcelos (2008) applied this bottom-up saliency to background subtraction in highly dynamic scenes.

Oliva, Torralba, Castelhano, and Henderson (2003) and Torralba et al. (2008) proposed a Bayesian framework for the task of visual search (i.e., whether a target is present or not.) They modeled bottom-up saliency as $\frac{1}{p(\mathbf{f}|\mathbf{f}_G)}$ where $\mathbf{f}_G$ represents a global feature that summarizes the appearance of the scene, and approximated this conditional probability density function by fitting to a multivariate exponential distribution. Zhang et al. (2008) also proposed saliency detection using natural statistics (SUN) based on a similar Bayesian framework to estimate the probability of a target at every location. They also claimed that their saliency measure emerges from the use of Shannon's self-information under certain assumptions. They used ICA features as similarly done in Bruce and Tsotsos (2006), but their method differs from Bruce and Tsotsos (2006) in that natural image statistics were applied to determine the density function of ICA features. Itti and Baldi (2006) proposed so-called "Bayesian Surprise" and extended it to the video case (Itti & Baldi, 2005). They measured KL-divergence between a prior distribution and posterior distribution as a measure of saliency.

For saliency detection in video, Marat et al. (2009) proposed a space-time saliency detection algorithm inspired by the human visual system. They fused a static saliency map and a dynamic saliency map to generate the space-time saliency map. Gao et al. (2008) adopted a dynamic texture model using a Kalman filter in order to capture the motion patterns even in the case that the scene is itself dynamic. Zhang et al. (2009) extended their SUN framework to a dynamic scene by introducing temporal filter (Difference of Exponential:DoE) and fitting a generalized Gaussian distribution to the estimated distribution for each filter response.

Most of the methods (Gao et al., 2008; Itti et al., 1998; Oliva et al., 2003; Zhang et al., 2009) based on Gabor or DoG filter responses require many design parameters such as the number of filters, type of filters, choice of the non-linearities, and a proper normalization scheme. These methods tend to emphasize textured areas as being salient regardless of their context. In order to deal with these problems, Bruce and Tsotsos (2006) and Zhang et al. (2008) adopted non-linear features that model complex cells or neurons in higher levels of the visual system. Kienzle, Wichmann, Scholkopf, and Franz (2007) further proposed to learn a visual saliency model directly from human eyetracking data using a support vector machine (SVM).

Different from traditional image statistical models, a spectral residual approach based on the Fourier transform was recently proposed by Hou and Zhang (2008b). The spectral residual approach does not rely on parameters and detects saliency rapidly. In this approach, the difference between the log spectrum of an image and its smoothed version is the spectral residual of the image. However, Guo, Ma, and Zhang (2008) claimed that what plays an important role for saliency detection is not spectral residual, but the image's phase spectrum. Recently, Hou and Zhang (2008a) proposed a dynamic visual attention model by setting up an objective function to maximize the entropy of the sampled visual features based on the incremental coding length.

Overview of the proposed approach

In this paper, our contributions to the saliency detection task are three-fold. First we propose to use local regression kernels as features which, fundamentally differ from conventional filter responses, but capture the underlying local structure of the data exceedingly well, even in the presence of significant distortions. Second, instead of using parametric models, we propose to use a non-parametric kernel density estimation for such features,

which results in a saliency map constructed from a local "self-resemblance" measure, indicating likelihood of saliency. Lastly, we provide a simple, but powerful unified framework for both static and space-time saliency detection. These contributions, which we will highlight at the end of this section, are evaluated in Experimental results section in terms of predicting human eye fixation data in both commonly used image (Bruce & Tsotsos, 2006) and video (Itti & Baldi, 2006) data sets. The original motivation behind these contributions is the earlier work on adaptive kernel regression for image and video reconstruction (Takeda, Farsiu, & Milanfar, 2007; Takeda, Milanfar, Protter, & Elad, 2009) and nonparametric object detection (Seo & Milanfar, 2009a) and action recognition (available online from http://users.soe.ucsc.edu/~milanfar/publications; Seo & Milanfar, under review).

As similarly done in Gao et al. (2008), we measure saliency at a pixel in terms of how much it stands out from its surroundings. To formalize saliency at each pixel, we let the binary random variable y_i denote whether a pixel position $\mathbf{x}_i = [\mathbf{x}_1; \mathbf{x}_2]_i^T$ is salient or not as follows:

$$y_i = \begin{cases} 1, & \text{if } \mathbf{x}_i \text{ is salient,} \\ 0, & \text{otherwise,} \end{cases} \tag{1}$$

where $i = 1,\ldots, M$, and M is the total number of pixels in the image. Motivated by the approach in Zhang et al. (2008) and Oliva et al. (2003), we define saliency at pixel position $\mathbf{x}_i$ as a posterior probability $Pr(y_i = 1|\mathbf{F})$ as follows:

$$S_i = Pr(y_i = 1|\mathbf{F}), \tag{2}$$

where the feature matrix, $\mathbf{F}_i = [\mathbf{f}_i^1,\ldots, \mathbf{f}_i^L]$ at 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. (Note that if $L = 1$, we use a single feature vector. Using a feature matrix consisting of a set of feature vectors provides more discriminative power than using a single feature vector as also pointed out in Wu and Nevatia, 2007 and Bregonzio, Gong, & Xiang, 2009.) In turn, the larger collection of features $\mathbf{F} = [\mathbf{F}_1,\ldots, \mathbf{F}_N]$ is a matrix containing features not only from the center, but also a surrounding region (what we call a center + surround region; see Figure 2). N is the number of feature matrices in the center + surround region. Using Bayes' theorem, Equation 2 can be written as

$$S_i = Pr(y_i = 1|\mathbf{F}) = \frac{p(\mathbf{F}|y_i = 1)Pr(y_i = 1)}{p(\mathbf{F})}. \tag{3}$$

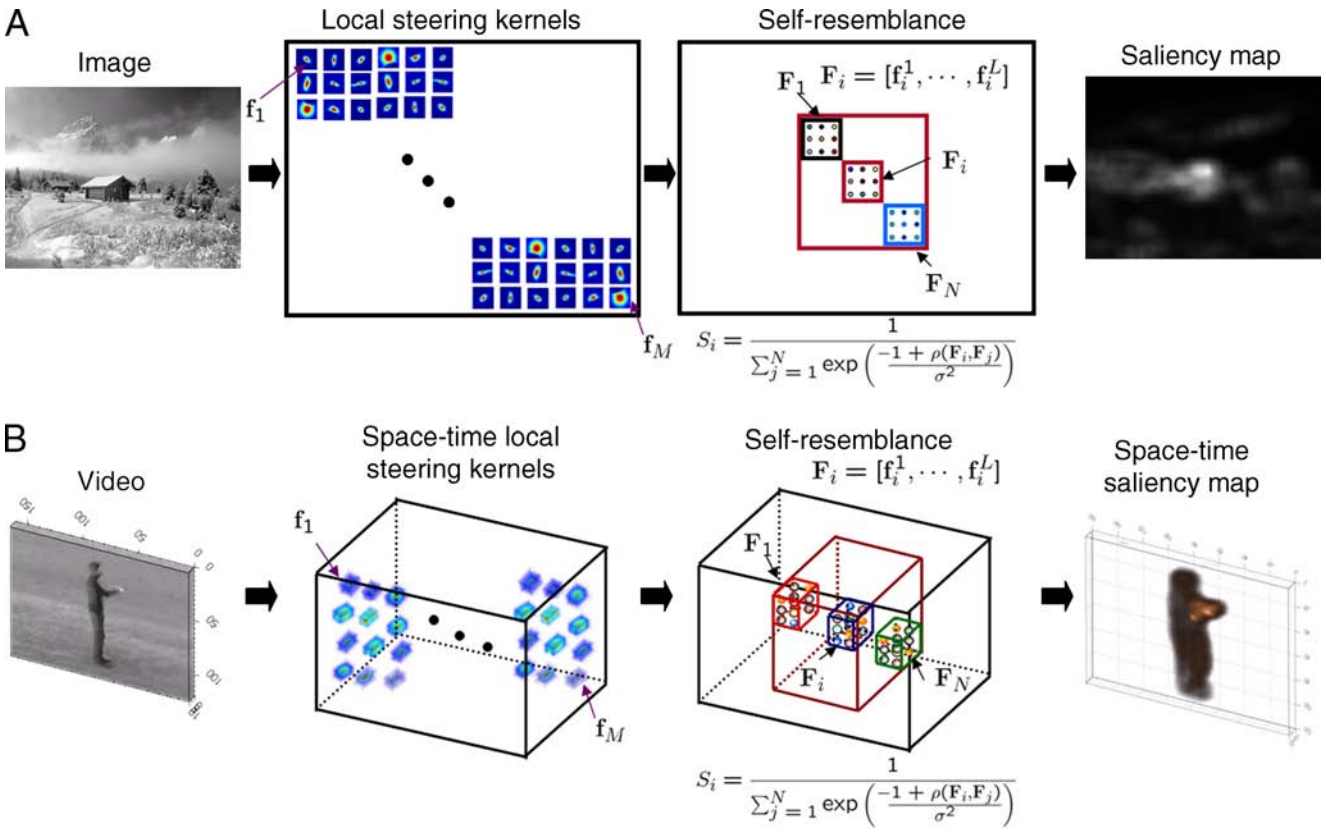

Figure 1. Graphical overview of saliency detection system: (A) static saliency map, (B) space-time saliency map. Note that the number of neighboring features N in (B) is obtained from a space-time neighborhood.

By assuming 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 $p(\mathbf{F}|y_i = 1)$.

Since we do not know the conditional probability density $p(\mathbf{F}|y_i = 1)$, we need to estimate it. It is worth noting that Gao et al. (2008) and Zhang et al. (2008) fit the marginal density of local feature vectors $p(\mathbf{f})$ to a generalized Gaussian distribution. However, in this paper, we approximate the conditional density function $p(\mathbf{F}|y_i = 1)$ based on nonparametric kernel density estimation which will be explained in detail in Saliency by self-resemblance section.

Before we begin a more detailed description, it is worthwhile to highlight some aspects of our proposed framework. While the state-of-the art methods (Bruce & Tsotsos, 2006; Gao et al., 2008; Itti & Baldi, 2006; Zhang et al., 2008) are related to our method, their approaches fundamentally differ from ours in the following respects: 1) While they use Gabor filters, DoG filters, or ICA to derive features, we propose to use local steering kernels (LSK) which are highly nonlinear but stable in the presence of uncertainty in the data (Takeda et al., 2007). In addition, normalized local steering kernels provide a certain invariance as shown in Figures 4 and 15. 2) As opposed to Gao et al. (2008) and Zhang et al. (2008) which model marginal densities of band-pass features as a generalized Gaussian distribution, we estimate the conditional probability density $p(\mathbf{F}|y_i = 1)$ using nonparametric kernel density estimation (see Figure 6). 3) While Itti and Baldi (2006) computed, as a measure of saliency, KL-divergence between a prior and a posterior distribution, we explicitly estimate the likelihood function directly using nonparametric kernel density estimation. 4) Our space-time saliency detection method does not require explicit motion estimation. 5) The proposed unified framework can handle both static and space-time saliency detection. Figure 1 shows an overview of our proposed framework for saliency detection. To summarize the operation of the overall algorithm, we first compute the normalized local steering kernels (space-time local steering kernels) from the given image (video) I and vectorize them as $\mathbf{f}$'s. Then, we identify features $\mathbf{F}_i$ centered at a pixel of interest $\mathbf{x}_i$, and a set of feature matrices $\mathbf{F}_j$ in a center + surrounding region and compute the self-resemblance measure (see Equations 16 and 17). The final saliency map is given as a density map as shown in Figure 1. A shorter version of this paper (available online from http://users.soe.ucsc.edu/~milanfar/publications) can be found in the proceeding of IEEE Conference on Computer Vision and Pattern Recognition, 1st International Workshop on Visual Scene Understanding (ViSU09) (Seo & Milanfar, 2009b).

In the next section, we provide further technical details about the steps outlined above. In Experimental results section, we demonstrate the performance of the system with experimental results, and we conclude this paper in Conclusion and future work section.

Technical details

Local regression kernel as a feature

Local steering kernel (2-D LSK)

The key idea behind local steering kernels 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

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

where $l \in \{1,\ldots, P\}$, P is the number of pixels in a local window; h is a global smoothing parameter (This parameter is set to 1 and fixed for the all experiments.) The matrix $\mathbf{C}_l$ is a covariance matrix estimated from a collection of spatial gradient vectors within the local analysis window around a position $\mathbf{x}_l = [\mathbf{x}_1; \mathbf{x}_2]_l^T$. More

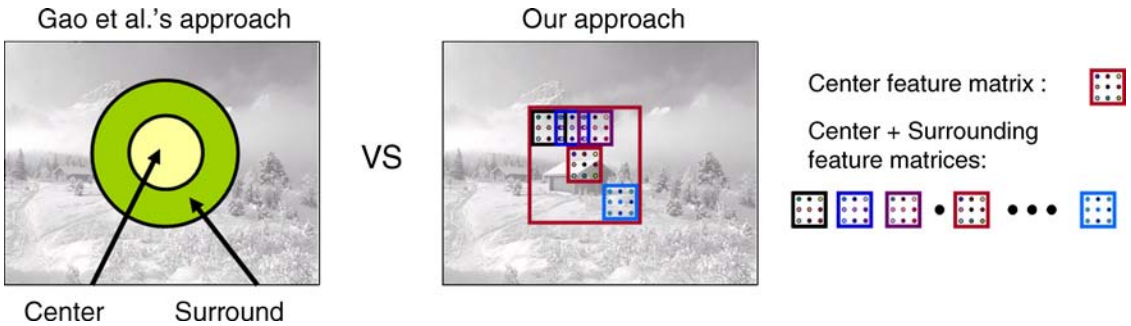

Figure 2. Illustration of difference between Gao et al.'s (2008) approach and our approach about a center-surround definition.

specifically, the covariance matrix $\mathbf{C}_l$ can be first naively estimated as $\mathbf{J}_l^T\mathbf{J}_l$ with

$$\mathbf{J}_l = \begin{bmatrix} z_{x_1}(\mathbf{x}_1), & z_{x_2}(\mathbf{x}_1) \\ \vdots & \vdots \\ z_{x_1}(\mathbf{x}_P), & z_{x_2}(\mathbf{x}_P) \end{bmatrix}, \tag{5}$$

where $z_{x_1}(\cdot)$ and $z_{x_2}(\cdot)$ are the first derivatives along x_1-; and x_2-axes. For the sake of robustness, we compute a more stable estimate of $\mathbf{C}_l$ by invoking the singular value decomposition (SVD) of $\mathbf{J}_l$ with regularization as (Takeda et al., 2007; Seo & Milanfar, 2009a)

$$\mathbf{C}_l = \gamma \sum_{q=1}^{2} a_q^2 \mathbf{v}_q \mathbf{v}_q^T \in \mathbb{R}^{(2\times 2)}, \tag{6}$$

with

$$a_1 = \frac{s_1 + \lambda'}{s_2 + \lambda'}, \quad a_2 = \frac{s_2 + \lambda'}{s_1 + \lambda'}, \quad \gamma = \left(\frac{s_1 s_2 + \lambda''}{P}\right)^{\alpha}, \tag{7}$$

where λ' and λ'' are parameters (λ' and λ'' are set to 1 and 10^{-7} respectively, and they are fixed for all experiments.) that dampen the noise effect and keep the denominators of a_q's from being zero, and α is a parameter (α is set to 0.008 and fixed for all experiments.) that restricts γ. The singular values (s_1, s_2) and the singular vectors ($\mathbf{v}_1$, $\mathbf{v}_2$) are given by the compact SVD of $\mathbf{J}_l = \mathbf{U}_l\mathbf{S}_l\mathbf{V}_l^T = \mathbf{U}_l \text{diag}[s_1, s_2]_l[\mathbf{v}_1, \mathbf{v}_2]_l^T$. Figure 3 illustrates that how covariance matrices and LSK values are computed in an edge region.

Space-time local steering kernel (3-D LSK)

Now, we introduce the time axis to the data model so that $\mathbf{x}_l = [x_1, x_2, t]_l^T$: x_1 and x_2 are the spatial coordinates, t is the temporal coordinate. The approach is fundamentally the same as in 2-D. Again, the covariance matrix $\mathbf{C}_l$ can be naively estimated as $\mathbf{J}_l^T\mathbf{J}_l$ with

$$\mathbf{J}_l = \begin{bmatrix} z_{x_1}(\mathbf{x}_1), & z_{x_2}(\mathbf{x}_1), & z_t(\mathbf{x}_1) \\ \vdots & \vdots & \vdots \\ z_{x_1}(\mathbf{x}_P), & z_{x_2}(\mathbf{x}_P), & z_t(\mathbf{x}_P) \end{bmatrix}, \tag{8}$$

where $z_{x_1}(\cdot)$, $z_{x_2}(\cdot)$, and $z_t(\cdot)$ are the first derivatives along x_1-, x_2-, and t-axes, and P is the total number of samples in a *space-time* local analysis window (or cube) around a sample position at $\mathbf{x}_i$. As similarly done in 2-D case, $\mathbf{C}_l$ is estimated by invoking the singular value decomposition (SVD) of $\mathbf{J}_l$ with regularization as (Takeda et al., 2009):

$$\mathbf{C}_l = \gamma \sum_{q=1}^{3} a_q^2 \mathbf{v}_q \mathbf{v}_q^T \in \mathbb{R}^{(3\times 3)}, \tag{9}$$

with

$$a_1 = \frac{s_1 + \lambda'}{\sqrt{s_2 s_3} + \lambda'}, \quad a_2 = \frac{s_2 + \lambda'}{\sqrt{s_1 s_3} + \lambda'},$$
$$a_3 = \frac{s_3 + \lambda'}{\sqrt{s_1 s_2} + \lambda'}, \quad \gamma = \left(\frac{s_1 s_2 s_3 + \lambda''}{P}\right)^{\alpha}, \tag{10}$$

where λ' and λ'' are parameters that dampen the noise effect and restrict γ and the denominators of a_q's from being zero. As mentioned earlier, the singular values (s_1, s_2, and s_3) and the singular vectors ($\mathbf{v}_1$, $\mathbf{v}_2$, and $\mathbf{v}_3$) are given by the compact SVD of $\mathbf{J}_l = \mathbf{U}_l\mathbf{S}_l\mathbf{V}_l^T = \mathbf{U}_l \text{diag}[s_1, s_2, s_3]_l[\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3]_l^T$.

The covariance matrix $\mathbf{C}_l$ modifies the shape and size of the local kernel in a way which robustly encodes the space-time local geometric structures present in the video

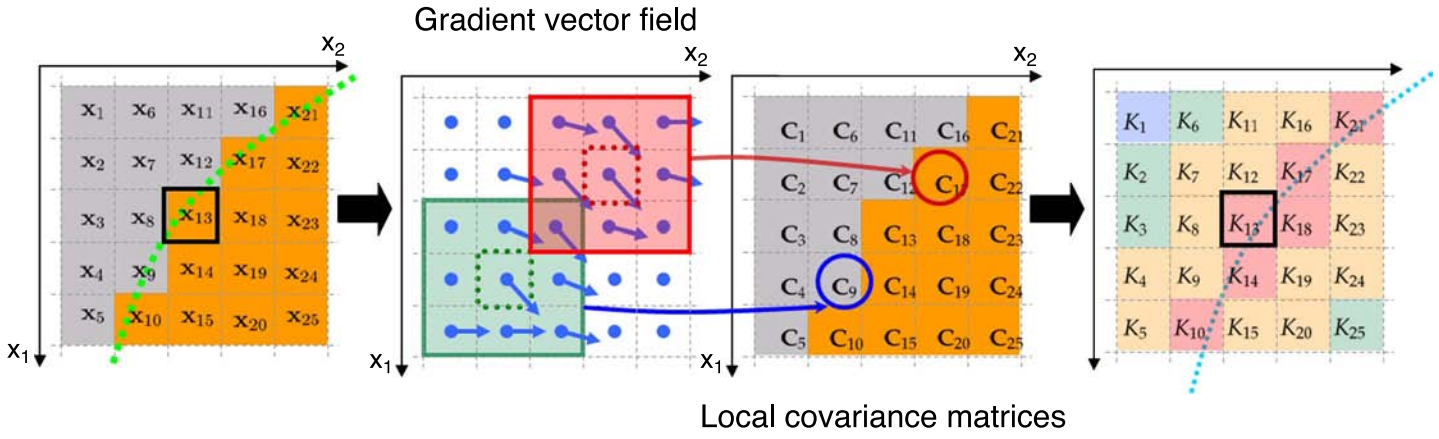

Figure 3. Graphical description of how LSK values centered at pixel of interest x_{13} are computed in an edge region. Note that each pixel location has its own C computed from gradient vector field within a local window.

(see Figure 4b for an example). Similarly to 2D case, 3-D LSKs are formed as follows:

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

$$\mathbf{C}_l \in \mathbb{R}^{(3\times 3)}. \tag{11}$$

In the 3-D case, orientation information captured in 3-D LSK contains the motion information implicitly (Takeda et al., 2009). It is worth noting that a significant strength of using this implicit framework (as opposed to the direct use of estimated motion vectors) is the flexibility it provides in terms of smoothly and adaptively changing the parameters defined by the singular values in Equation 10. This flexibility allows the accommodation of even complex motions, so long as their magnitudes are not excessively large.

Figure 4 illustrates how well local steering kernels capture both 2-D and 3-D local underlying geometric structure. As we can see from Equations 4 and 11, the values of the kernel K are based on the covariance matrices $\mathbf{C}_l$ along with their spatial locations $\mathbf{x}_l$. Intuitively, $\mathbf{C}_l$'s in the local analysis window are similar to one another in the flat region. Therefore, only spatial locations affect the kernel shape, which looks more or less symmetric or isotropic in the flat region. On the other hand, in the edge region, the kernel size and shape depend on both $\mathbf{C}_l$ and its spatial location $\mathbf{x}_l$ in the local window. Thus, high values in the kernel are yielded along the edge region whereas the rest of kernel values are near zero. For a more in depth analysis of local steering kernels, we refer the interested reader to Takeda et al. (2007) and Takeda et al. (2009).

In what follows, at a position $\mathbf{x}_i$, we will essentially be using (a normalized version of) the function $K(\mathbf{x}_l - \mathbf{x}_i)$. To be more specific, the local steering kernel function $K(\mathbf{x}_l - \mathbf{x}_i)$ is calculated at every pixel location and normalized as follows

$$W_i = \frac{K(\mathbf{x}_l - \mathbf{x}_i)}{\sum_{l=1}^{P} K(\mathbf{x}_l - \mathbf{x}_i)}, \quad i = 1, \cdots, M. \tag{12}$$

Thanks to an accurate estimation of the local covariance matrix $\mathbf{C}_l$, LSK features are robust against signal uncertainty such as presence of noise. In addition, the normalized version of LSKs provides certain invariance to illumination changes as shown in Figure 5.

As mentioned earlier, what we do next is to construct the feature matrix $\mathbf{F}_i$ by using $\mathbf{f}$'s which are a vectorized

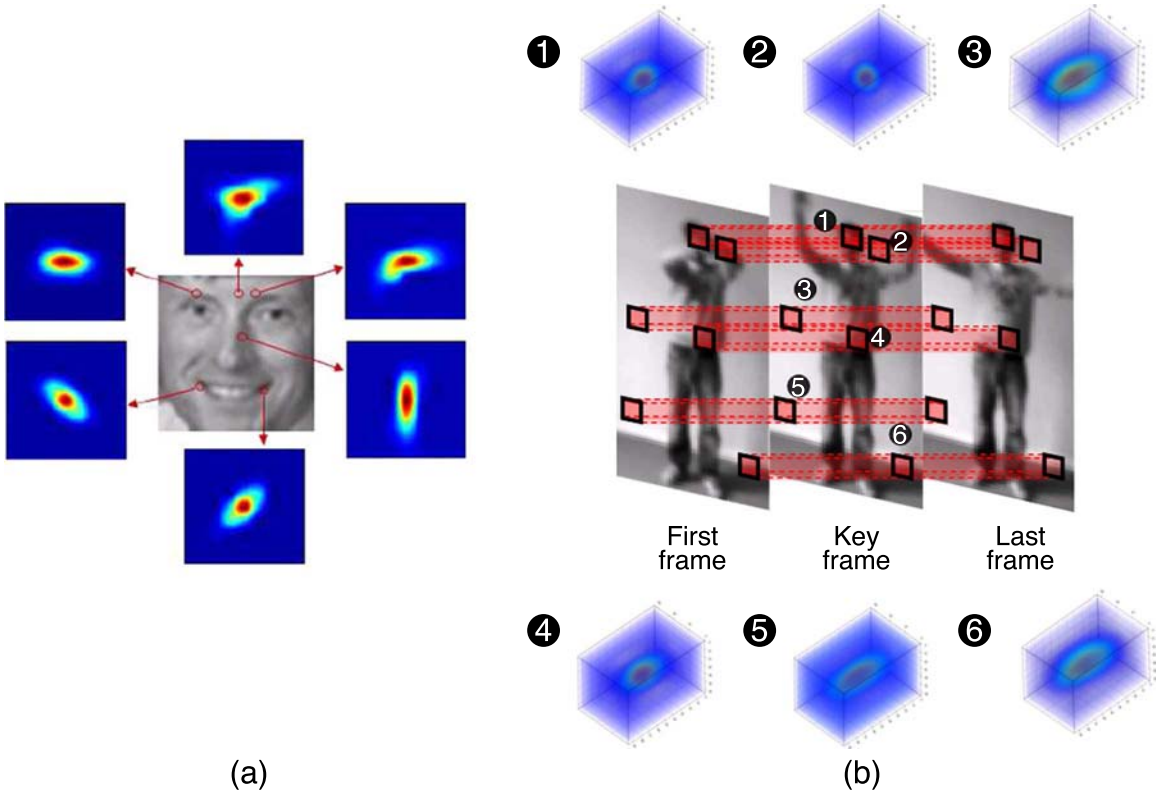

Figure 4. (a) Examples of 2-D LSK in various regions. (b) Examples of space-time local steering kernel (3-D LSK) in various regions. Note that key frame means the frame where the center of 3-D LSK is located.

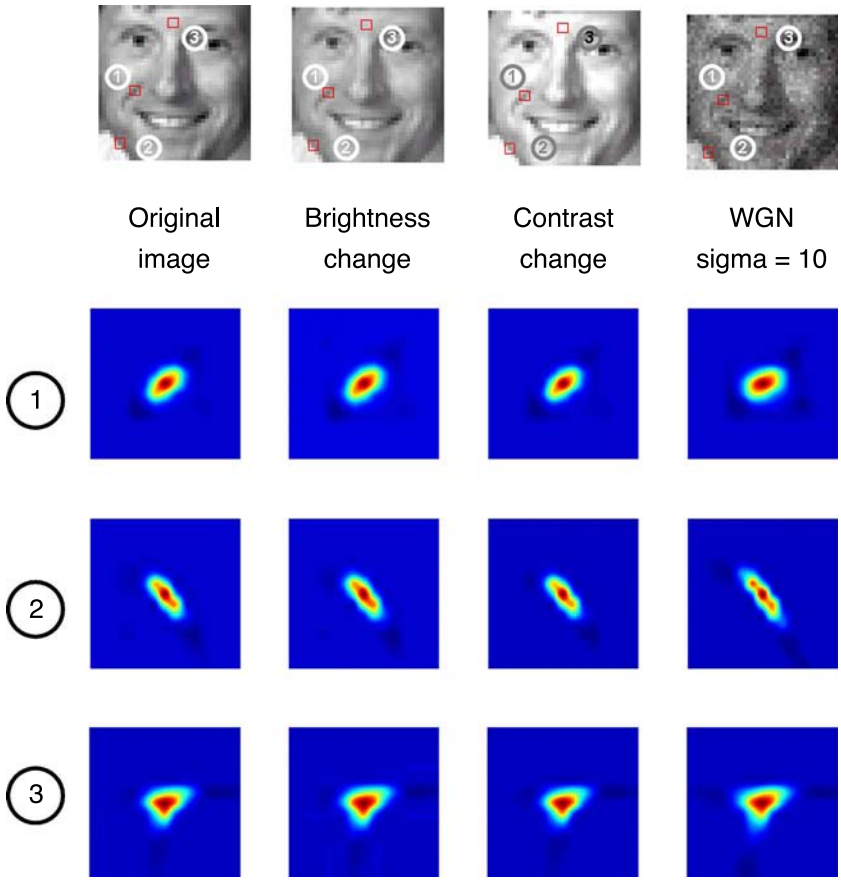

Figure 5. Invariance and robustness of LSK weights *W*in various challenging conditions. Note that WGN means White Gaussian Noise.

version of W's. More specifically, we collect $\mathbf{f}_i^j$ in a local window (say, 3×3) centered at the pixel of interest $\mathbf{x}_i$, where $j = 1,\ldots, 9$. Then, in a larger window (say, 5×5) also centered at $\mathbf{x}_i$, center + surround feature matrices $\mathbf{F}= [\mathbf{F}_1,\ldots, \mathbf{F}_{25}]$ are obtained. In the following section, we explain how we nonparametrically estimate the conditional probability density $p(\mathbf{F}|y_i = 1)$ discussed in Overview of the proposed approach section.

Saliency by self-resemblance

As we alluded to in Overview of the proposed approach section, saliency at a pixel $\mathbf{x}_i$ is measured using the conditional density of the feature matrix at that position: $S_i = p(\mathbf{F}|y_i = 1)$. Hence, the task at hand is to estimate $p(\mathbf{F}|y_i = 1)$ over $i = 1,\ldots, M$. In general, the Parzen density estimator is a simple and generally accurate nonparametric density estimation method (Silverman, 1986). However, in higher dimensions and with an expected long-tail distribution, the Parzen density estimator with an isotropic kernel is not the most appropriate tool (Bengio, Larochelle, & Vincent, 2005; Brox, Rosenhahn, & Cremers, 2007; Vincent & Bengio, 2003). As explained earlier, the LSK features tend to generically come from long-tailed distributions, and as such, there are generally no tight clusters in the feature space. When we estimate a probability density at a particular feature point, for instance $\mathbf{F}_i = [\mathbf{f}_i^1,\ldots, \mathbf{f}_i^L]$ (where $L$is the number of vectorized LSKs ($\mathbf{f}$'s) employed in the feature matrix), the isotropic kernel centered on that feature point will spread its density mass equally along all the feature space directions, thus giving too much emphasis to irrelevant regions of space and too little along the manifold. Earlier studies (Bengio et al., 2005; Brox et al., 2007; Vincent & Bengio, 2003) also pointed out this problem. This motivates us to use *a locally data-adaptive kernel density estimator*. 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:

$$S_i = \hat{p}(\mathbf{F}|y_i = 1) = \frac{G_i(\bar{\mathbf{F}}_i - \bar{\mathbf{F}}_i)}{\sum_{j=1}^{N} G_i(\bar{\mathbf{F}}_i - \bar{\mathbf{F}}_j)}, \quad (13)$$

Inspired by earlier works such as Fu and Huang (2008), Fu, Yan, and Huang (2008), Ma, Lao, Takikawa, and Kawade (2007) and Seo and Milanfar (2009a) that have

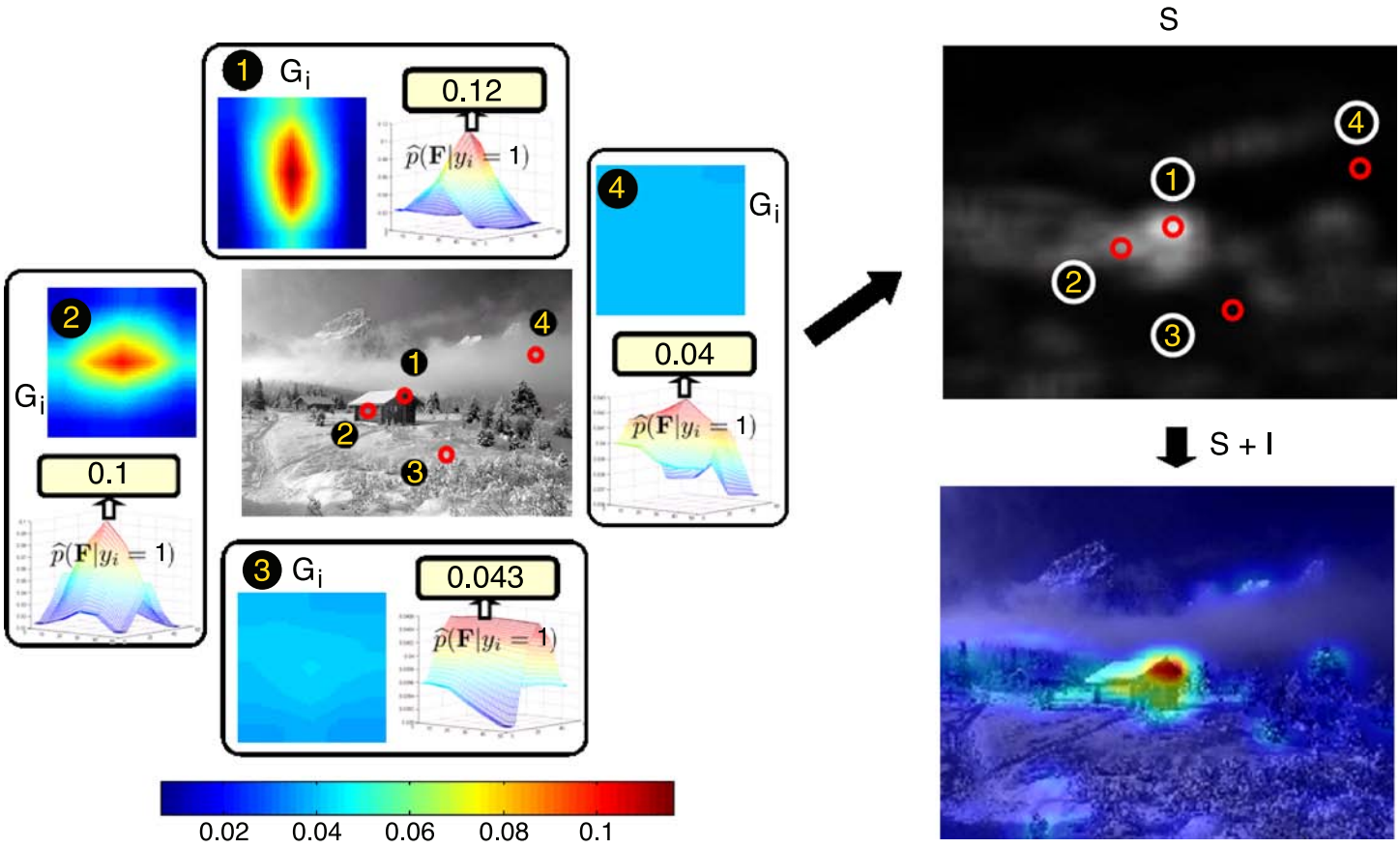

Figure 6. Example of saliency computation in natural gray-scale image. The estimated probability density $\hat{p}(\mathbf{F}|y_i = 1)$ at the point 1 (0.12) is much higher than ones (0.043) and (0.04) at the point 3 and point 4, which depicts that the point 1 is more salient than point 3 and point 4. Note that red values in saliency map represent higher saliency, while blue values mean lower saliency.

shown the effectiveness of correlation-based similarity, the kernel function G_i in Equation 13 can be defined by using the concept of matrix cosine similarity (Seo & Milanfar, 2009a) as follows:

$$G_i(\bar{\mathbf{F}}_i - \bar{\mathbf{F}}_j) = \exp\left(\frac{-\|\bar{\mathbf{F}}_i - \bar{\mathbf{F}}_j\|_F^2}{2\sigma^2}\right) = \exp\left(\frac{-1 + \rho(\mathbf{F}_i, \mathbf{F}_j)}{\sigma^2}\right), \quad j = 1, \cdots, N, \tag{14}$$

where $\bar{\mathbf{F}}_i = \frac{1}{\|\mathbf{F}_i\|_F}[\mathbf{f}_i^1, \ldots \mathbf{f}_i^L]$ and $\bar{\mathbf{F}}_j = \frac{1}{\|\mathbf{F}_j\|_F}[\mathbf{f}_j^1, \ldots \mathbf{f}_j^L]$, $\|\cdot\|_F$ is the Frobenius norm, and σ is a parameter (This parameter is set to 0.07 and fixed for all the experiments.) controlling the fall-off of weights. Here, $\rho(\mathbf{F}_i, \mathbf{F}_j)$ is the "Matrix Cosine Similarity (MCS)" between two feature matrices $\mathbf{F}_i$, $\mathbf{F}_j$ and is defined as the "Frobenius inner product" between two normalized matrices ($\rho(\mathbf{F}_i, \mathbf{F}_j) = \langle \mathbf{F}_i, \mathbf{F}_j \rangle_F = \text{trace}\left(\frac{\mathbf{F}_i^T \mathbf{F}_j}{\|\mathbf{F}_i\|_F \|\mathbf{F}_j\|_F}\right) \in [-1, 1]$.) This matrix cosine similarity can be rewritten as a weighted sum of the vector cosine similarities (Fu & Huang, 2008; Fu et al., 2008; Ma et al., 2007) $\rho(\mathbf{f}_i, \mathbf{f}_j)$ between each pair of corresponding feature vectors (i.e., columns) in $\mathbf{F}_i$, $\mathbf{F}_j$ as follows:

$$\rho_i = \sum_{\ell=1}^{L} \frac{\mathbf{f}_i^{\ell T} \mathbf{f}_j^{\ell}}{\|\mathbf{F}_i\|_F \|\mathbf{F}_j\|_F} = \sum_{\ell=1}^{L} \rho\left(\mathbf{f}_i^{\ell}, \mathbf{f}_j^{\ell}\right) \frac{\|\mathbf{f}_i^{\ell}\| \|\mathbf{f}_j^{\ell}\|}{\|\mathbf{F}_i\|_F \|\mathbf{F}_j\|_F}. \tag{15}$$

The weights are represented as the product of $\frac{\|\mathbf{f}_i^{\ell}\|}{\|\mathbf{F}_i\|_F}$ and $\frac{\|\mathbf{f}_j^{\ell}\|}{\|\mathbf{F}_j\|_F}$ which indicate the relative importance of each feature in the feature sets $\mathbf{F}_i$, $\mathbf{F}_j$. This measure not only generalizes the cosine similarity, but also overcomes the disadvantages of the conventional Euclidean distance which is sensitive to outliers (This measure can be efficiently computed by column-stacking the matrices $\mathbf{F}_i$, $\mathbf{F}_j$ and simply computing the cosine similarity between two long column vectors.) By inserting Equation 14 into Equation 13, S_i can be rewritten as follows:

$$S_i = \frac{1}{\sum_{j=1}^{N} \exp\left(\frac{-1 + \rho(\mathbf{F}_i, \mathbf{F}_j)}{\sigma^2}\right)}. \tag{16}$$

Figure 6 describes what normalized kernel functions G_i look like in various regions of a natural image. As shown in Figure 6, at $\mathbf{x}_i$ (that is, $S_i = \hat{p}(\mathbf{F}|y_i = 1)$) can be explained by the peak value of the normalized weight function G_i which contains contributions from all the surrounding feature matrices. In other words, $\hat{p}(\mathbf{F}|y_i = 1)$ reveals how salient $\mathbf{F}_i$ is given all the features $\mathbf{F}_j$'s in a neighborhood.

Handling color images

Up to now, we only dealt with saliency detection in a grayscale image. If we have color input data, we need an approach to integrate saliency information from all color channels. To avoid some drawbacks of earlier methods (Itti et al., 1998; Meur, Callet, & Barba, 2007), we do not combine saliency maps from each color channel linearly and directly. Instead we utilize the idea of matrix cosine similarity. More specifically, we first identify feature matrices from each color channel c_1, c_2, c_3 as $\mathbf{F}_i^{c1}$, $\mathbf{F}_i^{c2}$, $\mathbf{F}_i^{c3}$ as shown in Figure 7. By collecting them as a larger matrix $\mathbb{F}_i = [\mathbf{F}_i^{c1}, \mathbf{F}_i^{c2}, \mathbf{F}_i^{c3}]$, we can apply matrix cosine similarity between $\mathbb{F}_i$ and $\mathbb{F}_j$. Then, the saliency map from color channels can be analogously defined as follows:

$$S_i = \hat{p}(\mathbb{F}|y_i = 1) = \frac{1}{\sum_{j=1}^{N} \exp\left(\frac{-1 + \rho(\mathbb{F}_i, \mathbb{F}_j)}{\sigma^2}\right)}. \quad (17)$$

In order to verify that this idea allows us to achieve a consistent result and leads us to a better performance than using fusion methods, we have compared three different color spaces; namely opponent color channels (vande Sande, Gevers, & Snoek, 2008), CIE L*a*b* (Seo & Milanfar, 2009a; Shechtman & Irani, 2007) channels, and I R-G B-Y channels (Zhang et al., 2008) (opponent color space has proven to be superior to RGB, HSV, normalized RGB, and more in the task of object and scene recognition (vande Sande et al., 2008). Shechtman and Irani (2007) and Seo and Milanfar (2009a) showed that CIE L*a*b* performs well in the task of object detection.)

Figure 8 compares saliency maps using simple normalized summation of saliency maps from different channels as compared to using matrix cosine similarity. It is clearly seen that using matrix cosine similarity provides

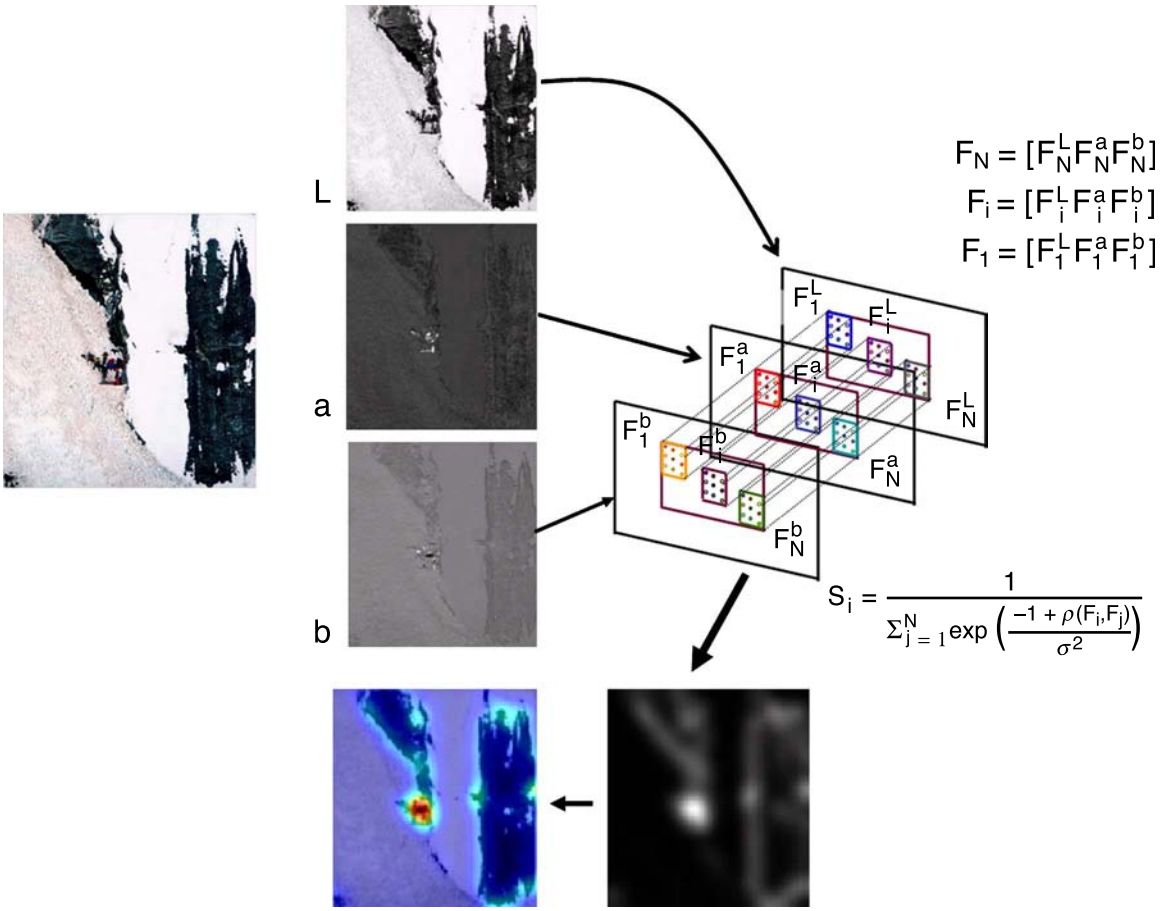

Figure 7. As an example of saliency detection in a color image (in this case, CIE L*a*b*), we show how saliency is computed using matrix cosine similarity.

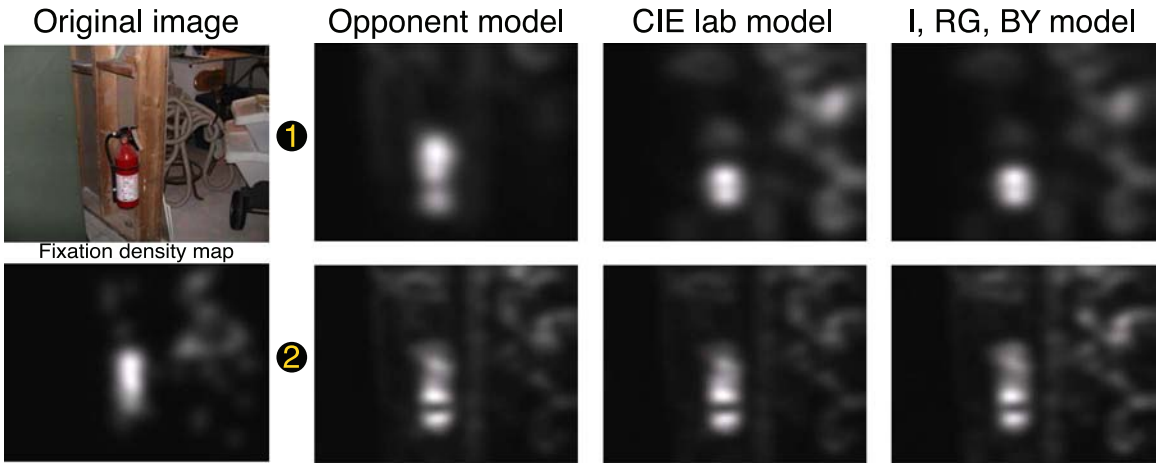

Figure 8. Comparisons between (1) Simple normalized summation and (2) The use of matrix cosine similarity without any fusion in three different color spaces. Simple normalized summation method tends to be dominated by a particular chrominance information. It is clearly shown that using matrix cosine similarity provides consistent results than the simple normalized summation fusion method.

consistent results regardless of color spaces and helps to avoid some drawbacks of fusion-based methods. To summarize, the overall pseudo-code for the algorithm is given in Algorithm 1.

I: input image or video, *P*: size of local steering kernel (LSK) or 3-D LSK window, *h*: a global smoothing parameter for LSK, *L*: number of LSK or 3-D LSK used in the feature matrix, *N*: size of a center + surrounding region for computing self-resemblance, σ: a parameter controlling fall-off of weights for computing self-resemblance.

Stage 1: Compute Features

if *I* is an image **then**

Compute the normalized LSK W_i and vectorize it to $\mathbf{f}_i$, where $i = 1,\ldots, M$.

else

Compute the normalized 3-D LSK W_i and vectorize it to $\mathbf{f}_i$, where $i = 1,\ldots, M$.

end if

Stage 2: Compute Self-Resemblance

for $i = 1,\ldots, M$ **do**

if *I* is a grayscale image (or video) **then**

Identify feature matrices $\mathbf{F}_i$, $\mathbf{F}_j$ in a local neighborhood.

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

else

Identify feature matrices $\mathbf{F}_i = [\mathbf{F}_i^{c1}, \mathbf{F}_i^{c3}, \mathbf{F}_i^{c3}]$ and $\mathrm{F}_j = [\mathbf{F}_j^{c1}, \mathbf{F}_j^{c3}, \mathbf{F}_j^{c3}]$ in a local neighborhood from three color channels.

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

end if

end for

Output: Saliency map S_i, $i = 1,\ldots, M$

Algorithm 1. Visual saliency detection algorithm.

Experimental results

In this section, we demonstrate the performance of the proposed method with comprehensive experiments in terms of 1) interest region detection; 2) prediction of human fixation data; and 3) performance on psychological patterns. Comparison is made with other state-of-the-art methods both quantitatively and qualitatively.

Interest region detection

Detecting proto-objects in images

In order to efficiently compute the saliency map, we downsample an image I to an appropriate coarse scale (64×64) (changing the scale leads to a different result in the saliency map. Assume that we use a 3×3 LSK and

Figure 9. Some examples of proto-objects detection in face images (http://www.facedetection.com/facedetection/datasets.htm).

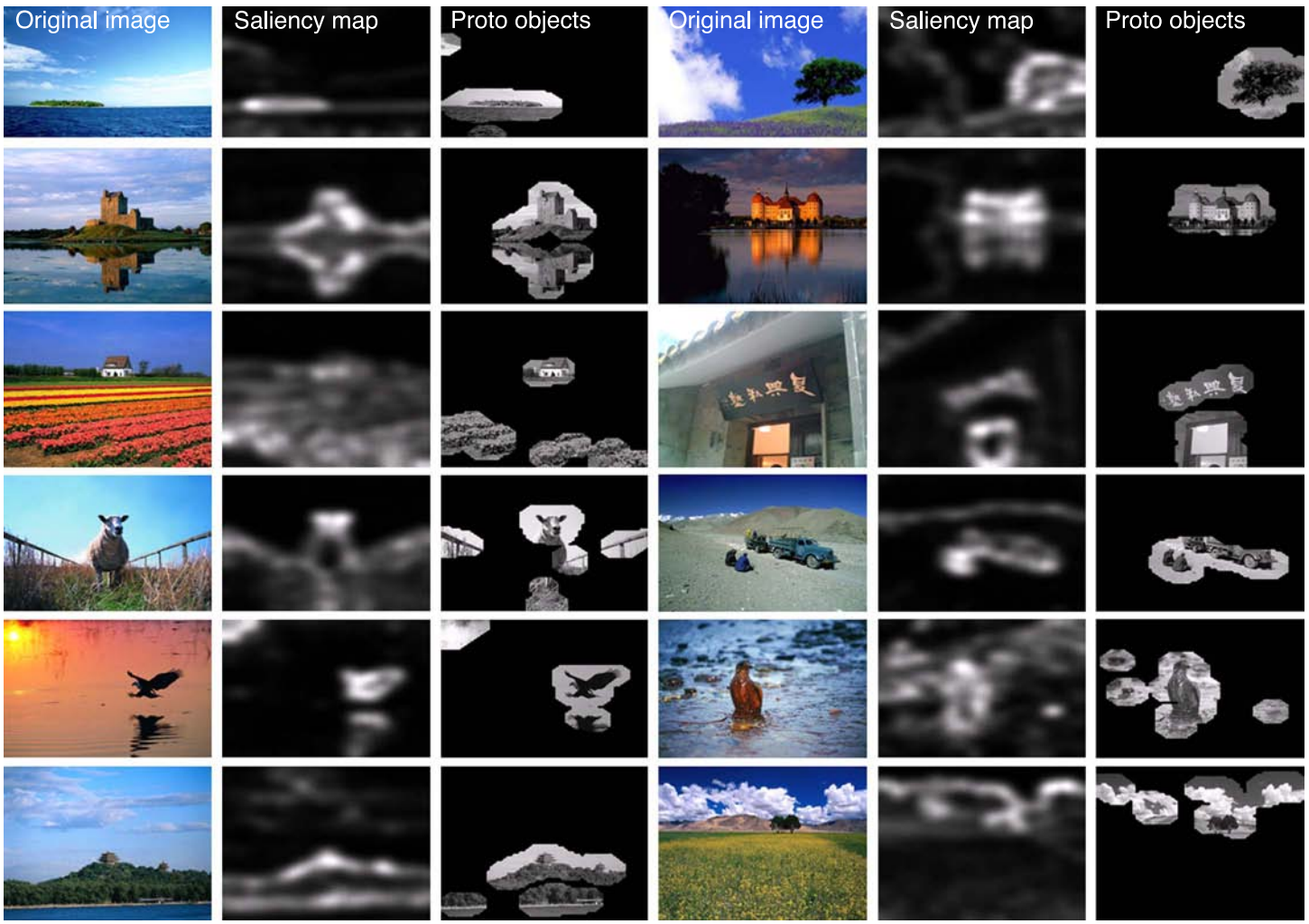

Figure 10. Some examples of proto-objects detection in natural scene images (Hou & Zhang, 2008b).

5 × 5 local analysis window for **F**. If the visual search is performed at a fine scale, finer detail will be captured as salient whereas at a coarser scale, larger objects will be considered to be salient. As expected, computing saliency map at a finer scale takes longer. In fact, we have tried to combine saliency maps from multi-scale, but this idea did not improve performance even at the expense of time complexity. This brings up an interesting question worth considering for future research; namely; what is the optimal resolution for saliency detection? Clearly, higher resolution images do not imply better saliency maps.) 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 LSK used in the feature matrix $\mathbf{F}_i$ is set to 9. For all the experiments, the smoothing parameter h for computing LSK was set to 1 and the fall-off parameter $\sigma$for computing self-resemblance was set to 0.07. We obtained an overall saliency map by using CIE L*a*b* color space throughout all the experiments. A typical run time takes about 1 second at scale (64 × 64) on an Intel Pentium 4, 2.66 GHz core 2 PC with 2 GB RAM.

From the point of view of object detection, saliency maps can explicitly represent proto-objects. We use the idea of non-parametric significance testing to detect proto-objects. Namely, we compute an empirical PDF from all the saliency values and set a threshold so as to achieve, for instance, a 95% significance level in deciding whether the given saliency values are in the extreme (right) tails of the empirical PDF. The approach is based on the assumption that in the image, a salient object is a relatively rare object and thus results in values which are in the tails of the distribution of saliency values. After making a binary object map by thresholding the saliency map, a morphological filter is applied. More specifically, we dilate the binary object map with a disk shape of size 5 × 5. Proto-objects are extracted from corresponding locations of the original image. Multiple objects can be extracted sequentially. Figure 9 shows that the proposed method works well in detecting proto-objects in the images which contain a group of people in a complicated cluttered background. In order to quantitatively evaluate the performance of our method in terms of finding proto-objects, we also tested our method on Hou and Zhang's data set (Hou & Zhang 2008b). This data set contains 62 natural scene images and ground truth images ($\mathcal{G}$) labeled by 4 naive human subjects. Figure 10 also illustrates that our method accurately detects salient objects in natural scenes (Hou & Zhang, 2008b). For the sake of completeness, we compute the *Hit Rate*(HR) and the *False Alarm Rate* (FAR) as follows:

$$HR = E\left(\prod_k \mathcal{G}_i^k \times O_i\right), \tag{18}$$

$$FAR = E\left(\prod_k (1 - \mathcal{G}_i^k) \times O_i\right), \tag{19}$$

where O is a proto-objects map, k is the image index. From Table 1, we observe that our method overall outperforms Hou and Zhang's method (downloadable from http://bcmi.sjtu.edu.cn/~houxiaodi/) (Hou & Zhang, 2008b) and Itti's method (downloadable from http://www.saliencytoolbox.net/) (Itti et al., 1998).

	Our method	Hou and Zhang (2008b)	Itti et al. (1998)
HR	**0.5933**	0.4309	0.2482
Fixed FAR	0.1433	0.1433	0.1433
Fixed HR	0.5076	0.5076	0.5076
FAR	**0.1048**	0.1688	0.2931

Table 1. Performance comparison of the methods on finding proto-objects in Hou and Zhang's data set (Hou & Zhang, 2008b). We compare HR and FAR of three methods at a fixed FAR and a fixed HR respectively.

Detecting actions in videos

The goal of action recognition is to classify a given action query into one of several pre-specified categories. Here, a query video may include a complex background which deteriorates recognition accuracy. In order to deal with this problem, it is necessary to have a procedure which automatically segments from the query video a small cube that only contains a valid action. Space-time saliency can provide such a mechanism. In order to compute the space-time saliency map, we only use the illumination channel because color information does not play a vital role in detecting motion saliency. We downsample each frame of input video I to a coarse spatial scale (64 × 64) in order to reduce the time-complexity (we do not downsample the video in the time domain.) We then compute 3-D LSK of size 3 × 3 × 3 as features and generate feature matrices $\mathbf{F}_i$ in a (3 × 3 × 7) local space-time neighborhood. The number of 3-D LSK used in the feature matrix $\mathbf{F}_i$ is set to 1 for time efficiency. The procedure for detecting space-time proto-objects and the rest of parameters remain the same as in the 2-D case. A typical run of space-time saliency detection takes about 52 seconds on 50 frames of a video at spatial scale (64 × 64) on an Intel Pentium 4, 2.66 GHz core 2 PC with 2 GB RAM.

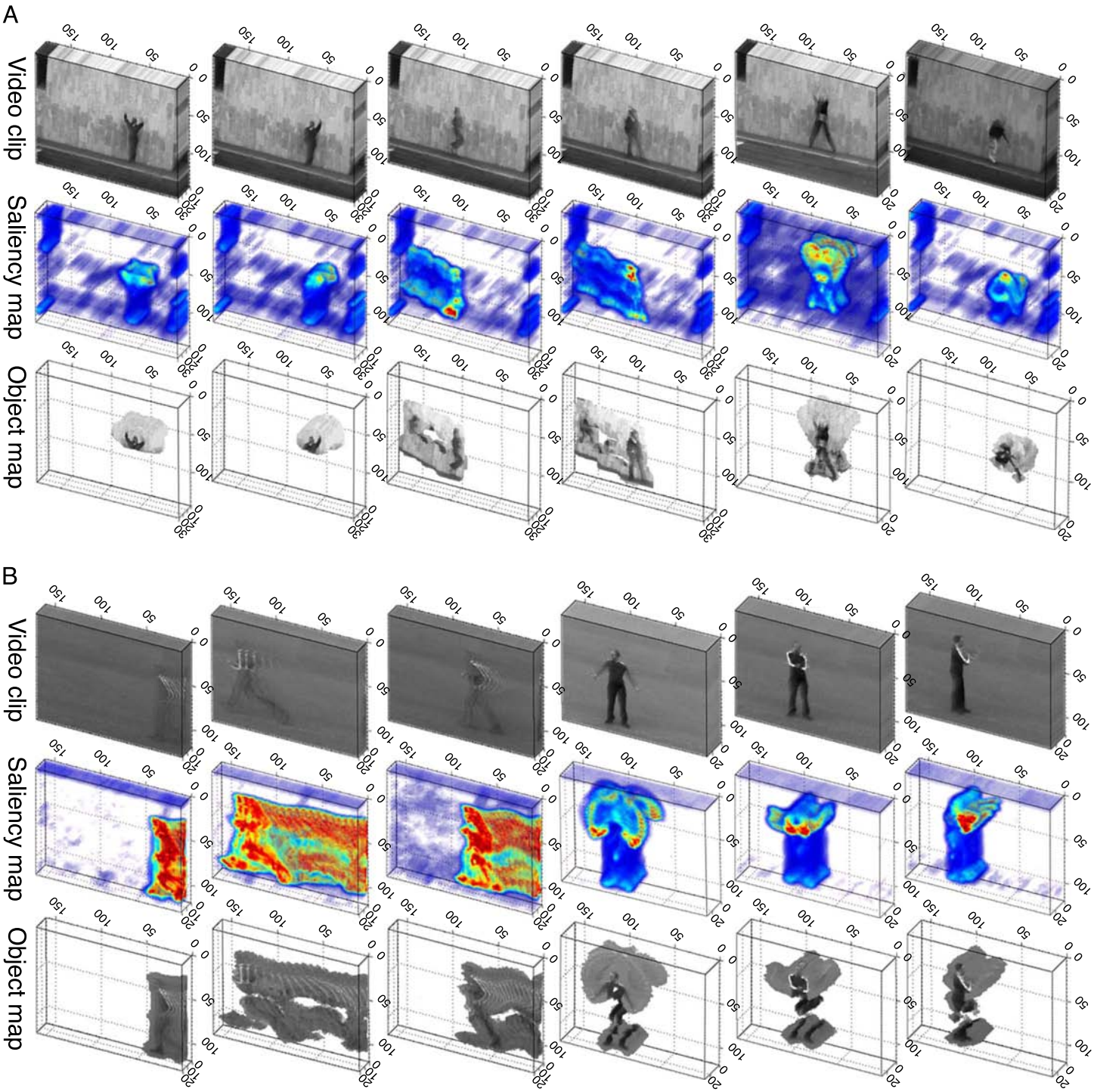

Figure 11. Some examples of detecting salient human actions in the video: (a) the Weizmann data set (Gorelick et al., 2007) and (b) the KTH data set (Schuldt et al., 2004).

Figure 11 shows that the proposed space-time saliency detection method successfully detects only salient human actions in both the Weizmann data set (Gorelick, Blank, Shechtman, Irani, & Basri, 2007) and the KTH data set (Schuldt, Laptev, & Caputo, 2004). Our method is also robust to the presence of fast camera zoom in and out as shown in Figure 12 where a man is performing a boxing action while a camera zoom is activated.

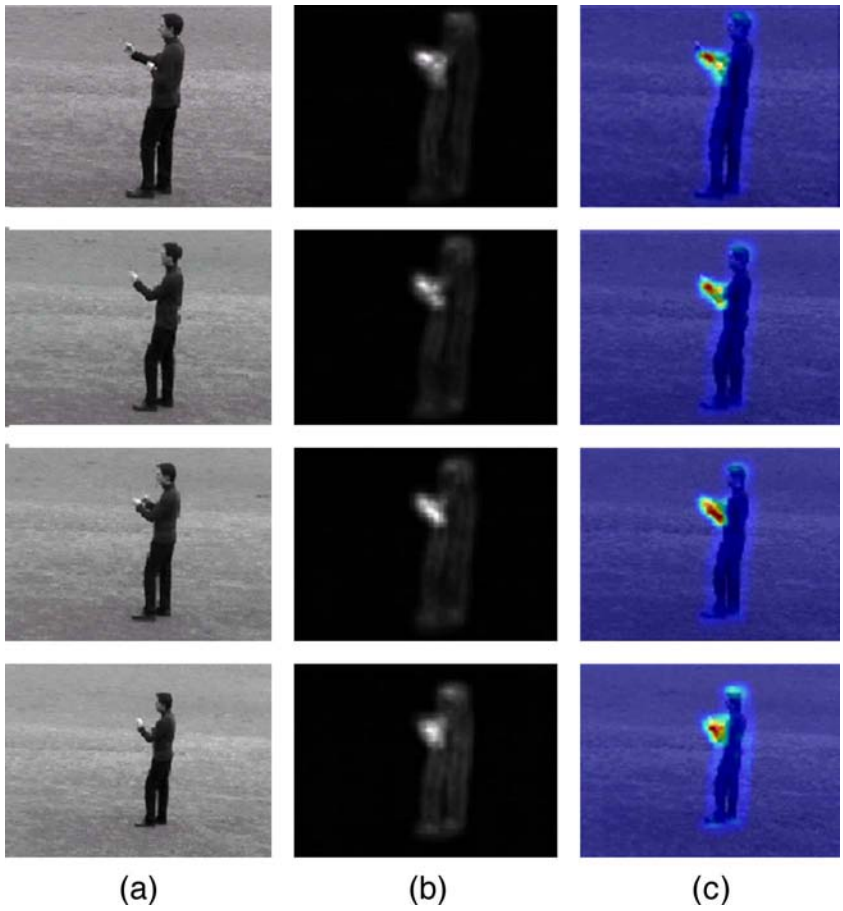

Figure 12. Space-time saliency detection even in the presence of fast camera zoom-in. Note that a man is performing a boxing action while a camera zoom is activated.

Predicting human visual fixation data

Static images

In this section, we used an image database and its corresponding fixation data collected by Bruce and Tsotsos (2006) as a benchmark for quantitative performance analysis and comparison. This data set contains eye fixation records from 20 subjects for a total of 120 images of size 681 × 511. The parameter settings are the same as explained in Interest region detection section. Some visual results of our model are compared with state-of-the-art methods in Figure 13. As opposed to Bruce's method (Bruce & Tsotsos, 2006) which is quite sensitive to textured regions, and SUN (Zhang et al., 2008) which is somewhat better in this respect, the proposed method is much less sensitive to background texture. To compare the methods quantitatively, we also computed the area under receiver operating characteristic (ROC) curve, and KL-divergence by following the experimental protocol of Zhang et al. (2008). In Zhang et al. (2008), Zhang et al. pointed out that the data set collected by Bruce and Tsotsos (2006) is center-biased and the methods by Itti et al. (1998), Bruce and Tsotsos (2006) and Gao et al. (2008) are all corrupted by edge effects which resulted in relatively higher performance than they should have (see Figure 14.). We compare our model against Itti et al. (1998) (downloadable from http://ilab.usc.edu/toolkit/home.shtml), Bruce and Tsotsos (2006) (downloadable from http://www-sop.inria.fr/members/Neil.Bruce/#SOURCECODE), Gao et al. (2008), and SUN (downloadable from http://www.roboticinsect.net/index.htm) (Zhang et al., 2008). For the evaluation of the algorithm, we used the same procedure as in Zhang et al. (2008). More specifically, we first compute true positives from the saliency maps based on the human eye fixation points. In order to calculate false positives from the saliency maps, we use the human fixation points from other images by permuting the order of images. This permutation of images is repeated 100 times. Each time, we compute KL-divergence between the histograms of true positives and false positives and average them over 100 trials. When it comes to calculating the area under the ROC curve, we compute detection rates and false alarm rates by thresholding histograms of true positives and false positives at each stage of shuffling. The final ROC area shown in Table 2 is the average value over 100

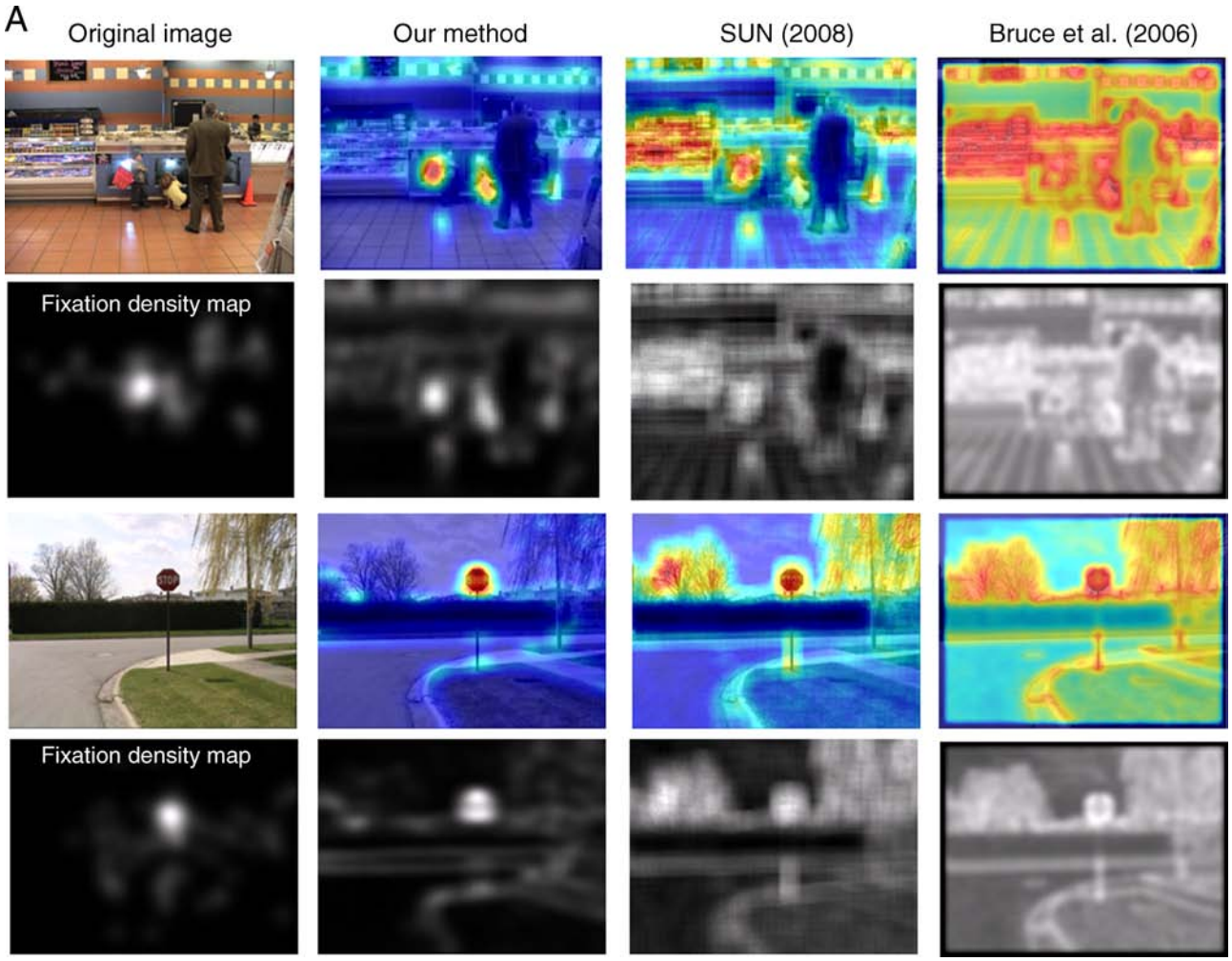

Figure 13. Examples of saliency maps with comparison to the state-of-the-art methods. Human fixation density maps are derived from human eye fixation data and are shown right below the original images. Visually, our method outperforms other state-of-the-art methods.

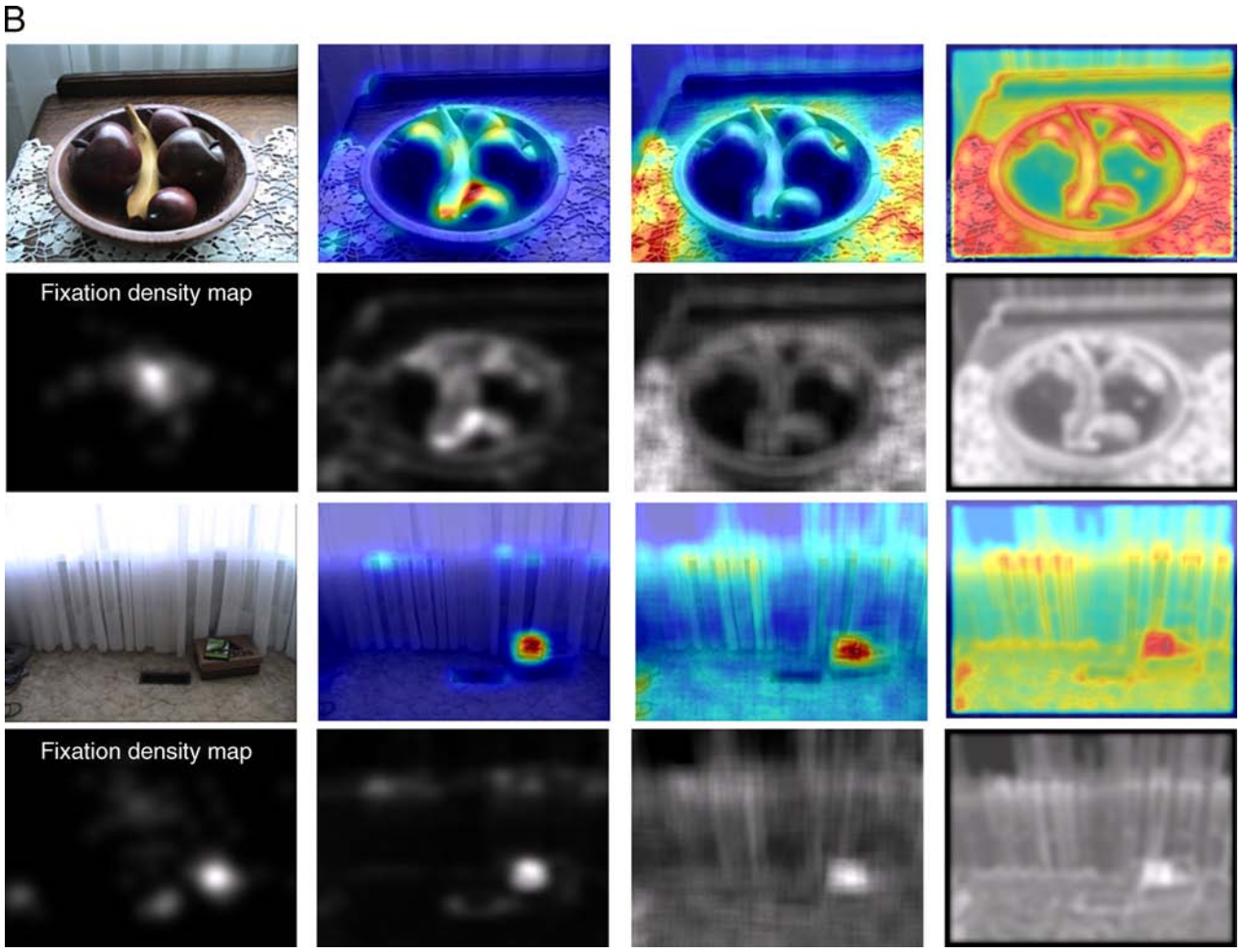

Figure 13. continued

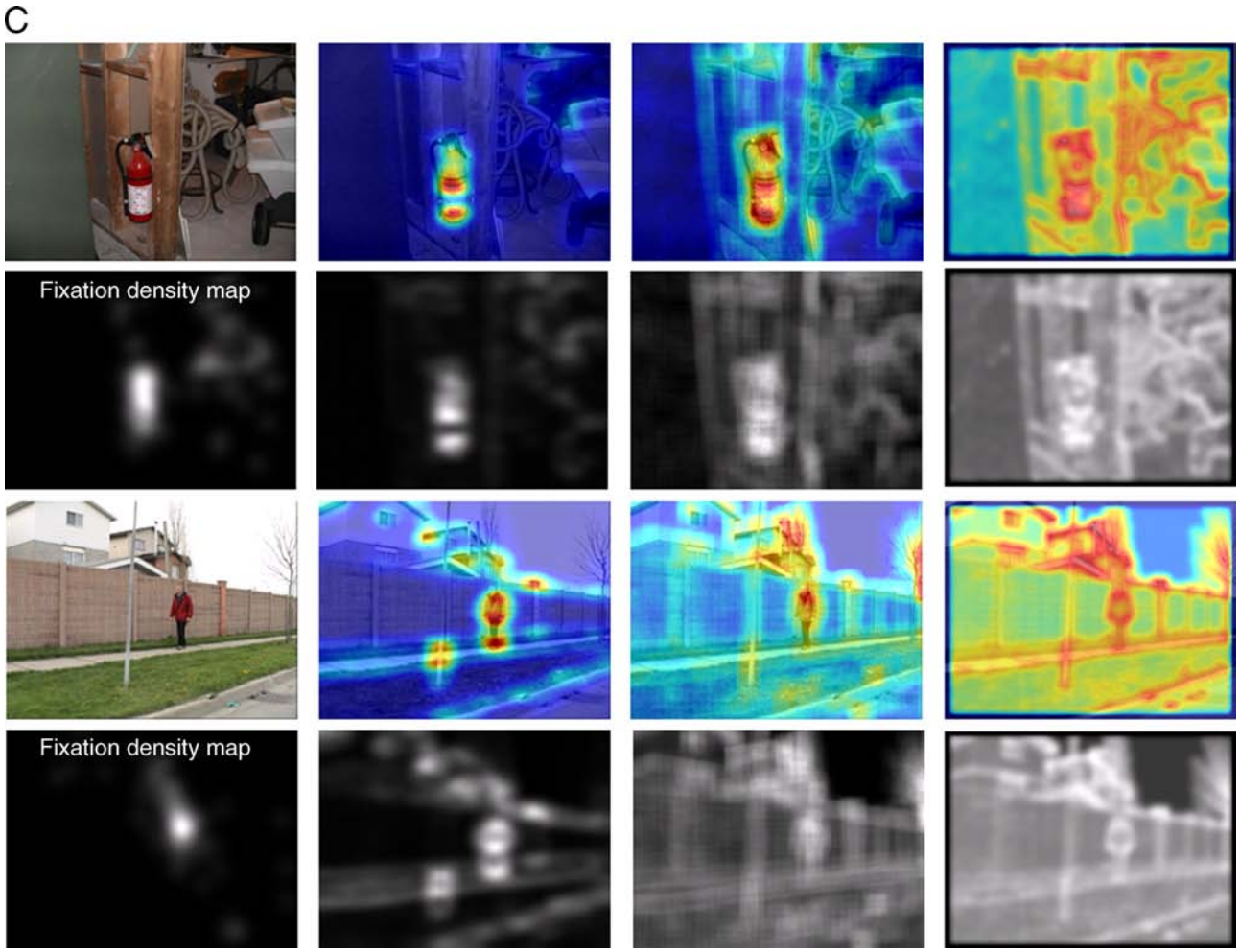

Figure 13. continued

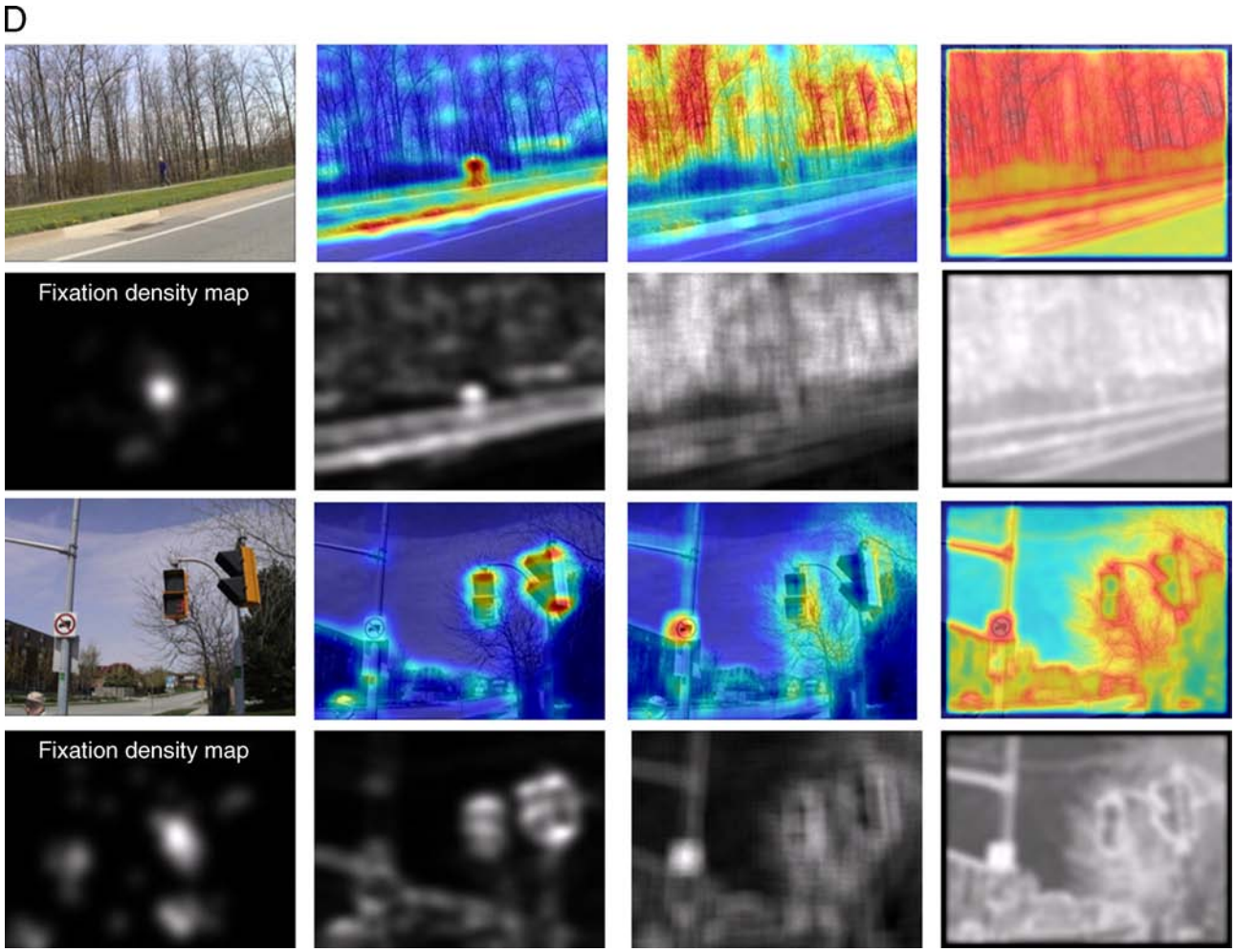

Figure 13. continued

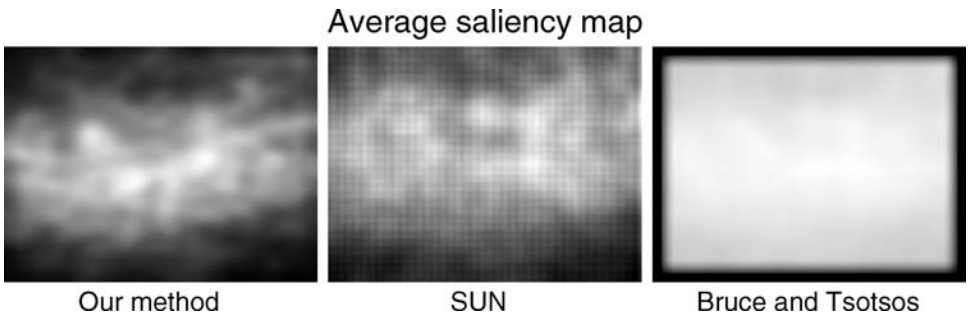

Figure 14. Comparison of average saliency maps on human fixation data by Bruce and Tsotsos (2006). Averages were taken across the saliency maps for a total of 120 color images. Note that Bruce et al.'s method (Bruce & Tsotsos, 2006) exhibits zero values at the image borders while SUN (Zhang et al., 2008) and our method do not have edge effects.

permutations. The mean and the standard errors are also reported in Table 2. Our model outperforms all the other state-of-the-art methods in terms of both KL-divergence and ROC area.

• As we alluded to Local regression kernel as a feature section earlier, our LSK features are robust to the presence of noise and changes in brightness and contrast. Figure 15 well demonstrates that the self-resemblance maps based on LSK features are not influenced by various distortions such as white-color noise, contrast change, and brightness change.

• We further examined how the performance of the proposed method is affected by the choice of parameters such as 1) N: size of center + surrounding region for computing self-resemblance; 2) P: size of LSK; and 3) L: number of LSK used in the feature matrix. As shown in Figure 16, it turns out that as we increase N, the overall performance is improved while increasing P and L rather deteriorates the performance. Overall, the best performance was achieved with the choice of $P = 3 \times 3 = 9$; $L = 3 \times 3 = 9$; and $N = 7 \times 7 = 49$ at the expense of increased runtime.

It is important to note that while the LSK size (P) and the number of LSK (L) determine a feature dimension, the surrounding size (N) affects how many surrounding feature matrices would be compared with the center feature matrix. We do not wish to increase feature dimensions unnecessarily. Instead, we keep the surrounding size large enough so that we could get a reasonable self-resemblance value.

Response to psychological pattern

We also tested our method on psychological patterns. Psychological patterns are widely used in attention experiments not only to explore the mechanism of visual search, but also to test effectiveness of saliency maps (Treisman & Gelade, 1980; Wolfe, 1994). As shown in Figure 17, whereas SUN (Zhang et al., 2008) and Bruce's method (Bruce & Tsotsos, 2006) failed to capture perceptual differences in most cases, Gao's method (Gao et al., 2008) and Spectral Residual (Hou & Zhang, 2008b) tend to capture perceptual organization rather better. Overall, however, the proposed saliency algorithm outperforms other methods in all cases including closure pattern (Figure 17a) and texture segregation (Figure 17b) which seem to be very difficult even for humans to distinguish.

The proposed method also predicts search asymmetry (Treisman & Gelade, 1980) well. As shown in Figure 18, it is evident that our method mimics the human tendency of finding a Q (or a plus) among Os (or dashes) to be easier than finding an O (or a dash) among Qs (pluses).

Dynamic scenes

In this section, we quantitatively evaluate our space-time saliency algorithm on the human fixation video data from Itti et al. (2005). This data set consists of a total of 520 human eye-tracking data traces recorded from 8 distinct subjects watching 50 different videos (TV programs, outdoors, test stimuli, and video games: about 25 minutes of total playtime). Each video has a resolution of size 640 × 480. Eye movement data was collected using an ISCAN RK-464 eye-tracker. For evaluation, two hundred (four subjects × fifty video clips) eye movement traces were used (see Itti & Baldi, 2005 for more details). As similarly done earlier, we computed the area under receiver operating characteristic (ROC) curve, and the KL-divergence. We compare our model against Bayesian Surprise (Itti et al., 1998) and SUNDAy (Zhang et al., 2009). Note that human eye movement data collected by

Model	KL (*SE*)	ROC (*SE*)
Itti et al. (1998)	0.1130 (0.0011)	0.6146 (0.0008)
Bruce and Tsotsos (2006)	0.2029 (0.0017)	0.6727 (0.0008)
Gao et al. (2008)	0.1535 (0.0016)	0.6395 (0.0007)
Zhang et al. (2008)	0.2097 (0.0016)	0.6570 (0.0008)
Hou and Zhang (2008b)	0.2511 (0.0019)	0.6823 (0.0007)
Our method	**0.2779** (0.002)	**0.6896** (0.0007)

Table 2. Performance in predicting human eye fixations when viewing videos color images. *SE* means standard errors.

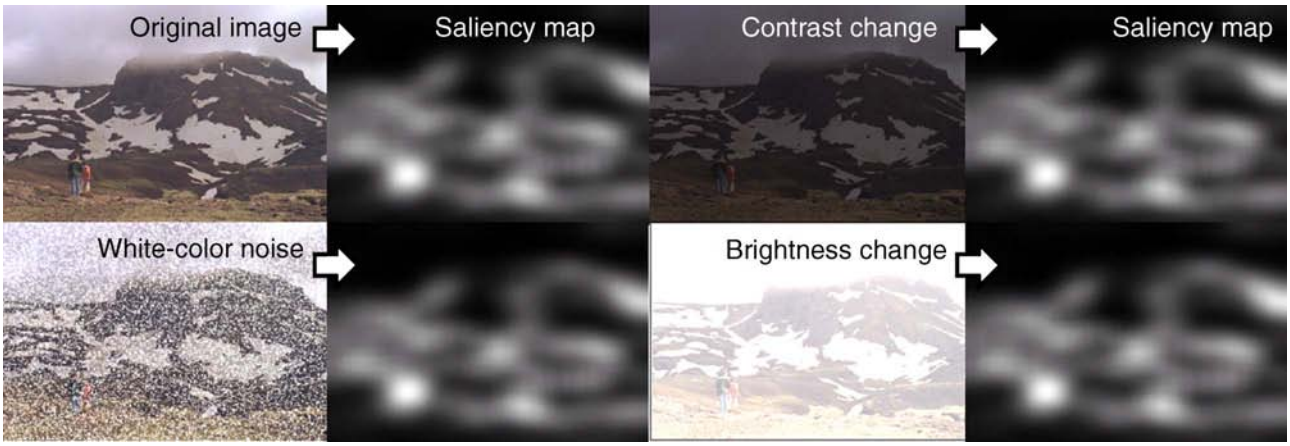

Figure 15. Our saliency model is largely unaffected by various distortions such as white-color noise, brightness change, and contrast change.

Itti et al. (2005) is also center-biased and Bayesian Surprise (Itti & Baldi, 2005) is corrupted by edge effects which resulted in relatively higher performance than it should have. For the evaluation of the algorithm, we first compute true positives from the saliency maps based on the human eye movement fixation points. In order to calculate false positives from the saliency maps, we use the human fixation points from frames of other videos by permuting the order of video. This permutation of images is repeated 10 times. Each time, we compute KL-divergence between the histograms of true positives and false positives and average them over 10 trials. When it

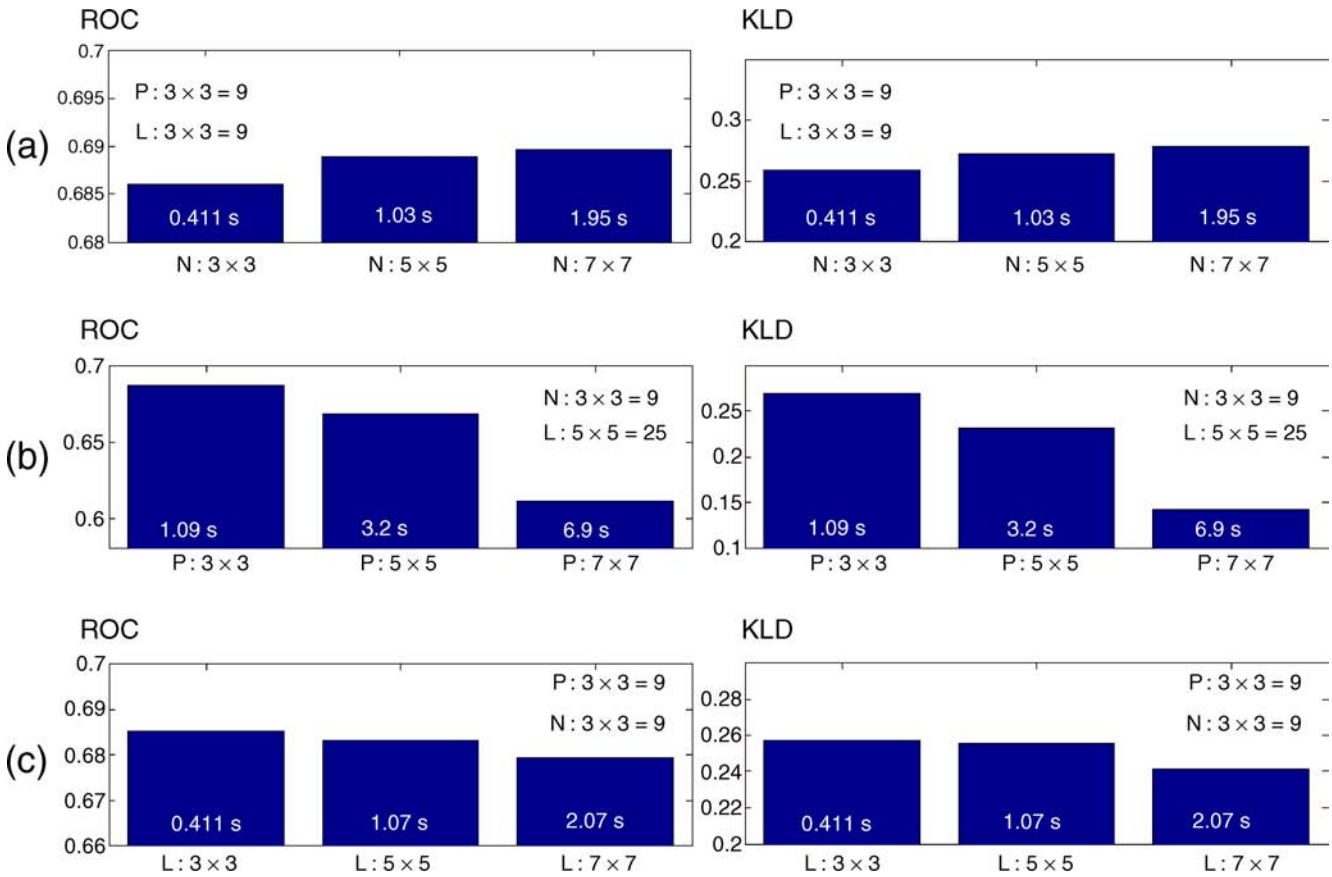

Figure 16. Performance comparison on human fixation data by Bruce and Tsotsos (2006) with respect to the choice of 1) *N*: size of center + surrounding region for computing self-resemblance; 2) *P*: size of LSK; and 3) *L*: number of LSK used in the feature matrix. Run time on one image is shown on top of each bar.

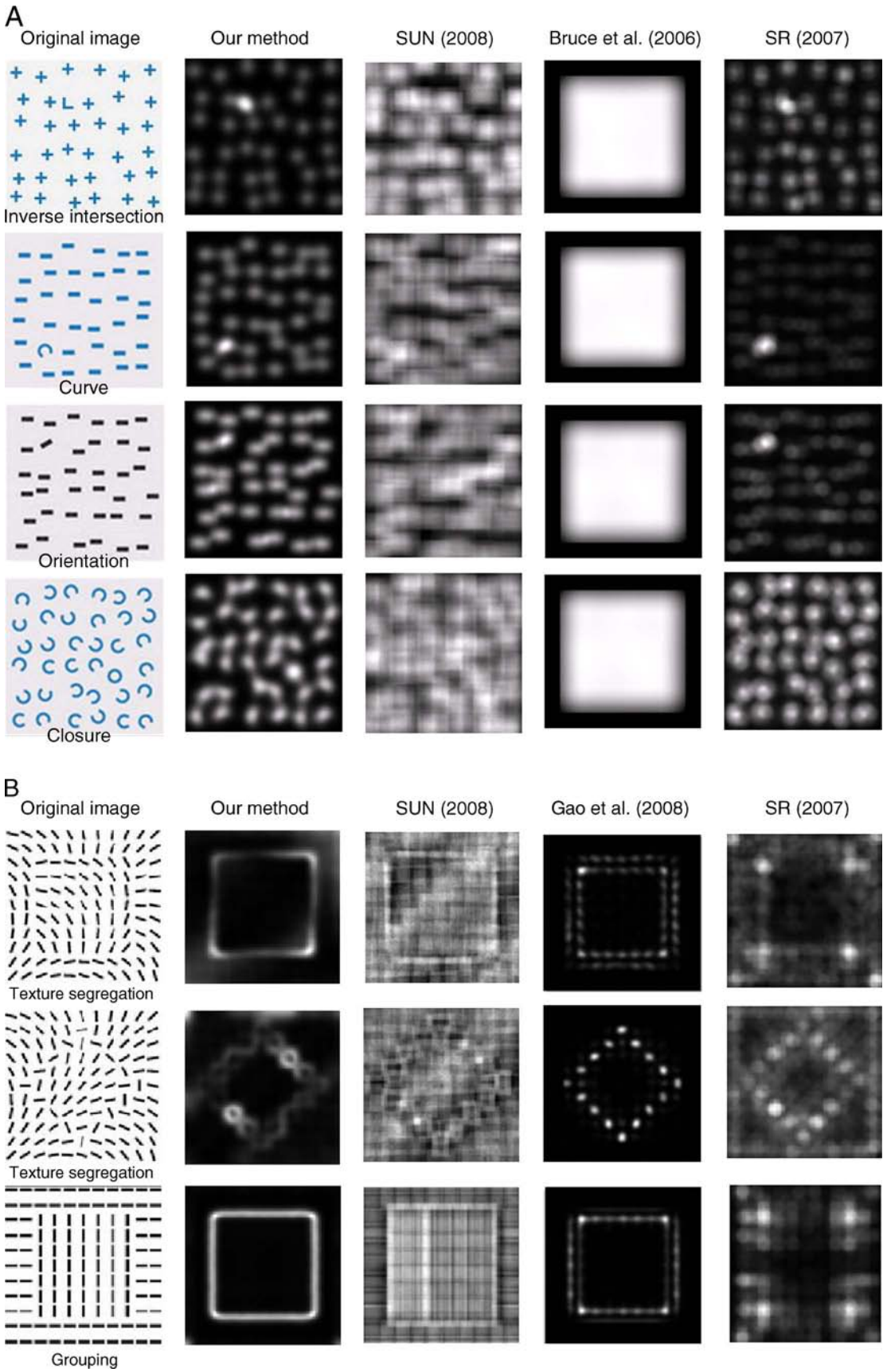

Figure 17. Examples of Saliency map on psychological patterns. (a) Images are from Hou and Zhang (2008b). (b) Images are from Gao et al. (2008).

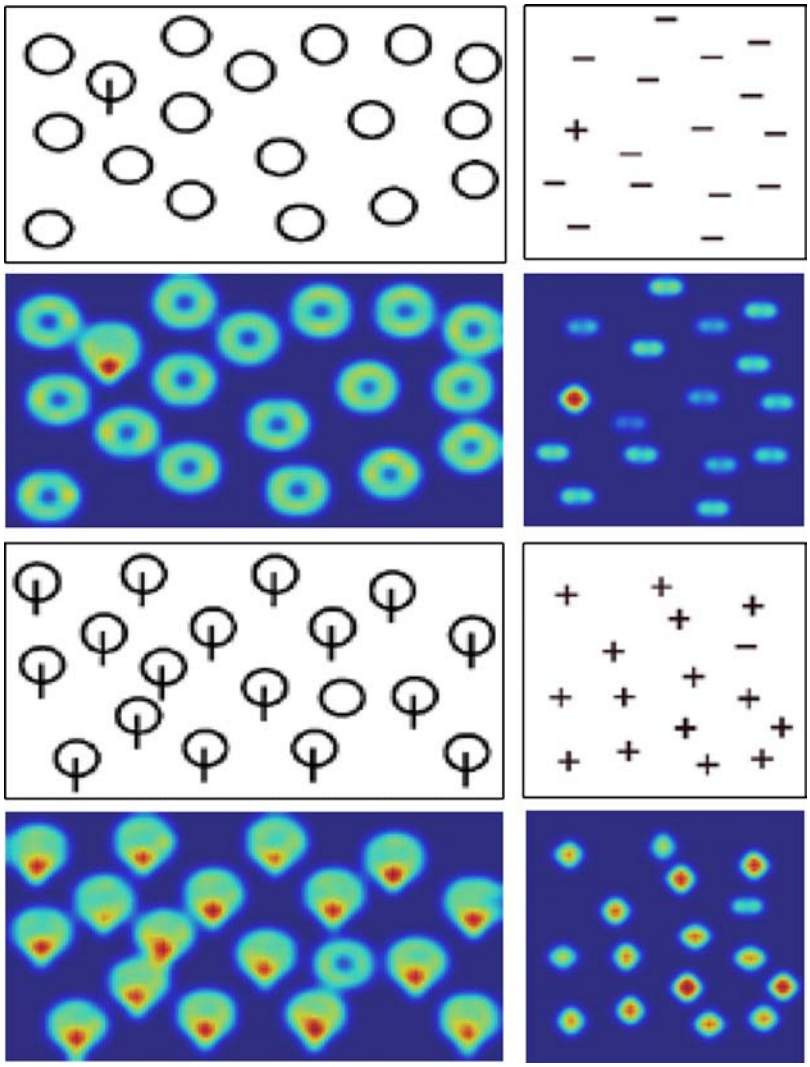

Figure 18. Example of asymmetry. (Top: The task of finding a Q among Os is easier than finding an O among Qs. Bottom: The task of finding a plus among dashes is easier than finding a dash among plus.) This effect demonstrates a specific example of search asymmetry discussed in Treisman and Gelade (1980).

comes to calculating the area under the ROC curve, we compute detection rates and false alarm rates by thresholding histograms of true positives and false positives at each time of shuffling. The mean ROC area and the mean KL-divergence are reported in Table 3. Some visual results of our model are shown in Figure 19. Our model outperforms Bayesian Surprise and SUNDAy in terms of both KL-divergence and ROC area. It seems at first surprising that our KL-divergence value is much higher than Bayesian (Itti & Baldi, 2005) and SUNDAy (Zhang et al., 2009) while there is a rather smaller difference between ROC areas. However, this phenomenon can be explained as follows. While the range of ROC area is limited from 0 to 1, the range of KL-divergence is from 0 to ∞. The KL-divergence is asymptotically related to the probability of detection and false alarm rate and provides an upper bound on the detection performance (Kullback, 1968; Shahram, 2005). Namely, as the number of samples increases, $P_f(1 - P_d) \rightarrow \exp(-\alpha J)$, where α is a constant and J is KL-divergence. Even though there is a large difference between KL-divergence values, the difference in ROC area can be relatively small.

Our model is simple, but very fast and powerful. In terms of time complexity, a typical run time takes about 8 minutes (Zhang et al., 2009 reported that their method runs in

Model	KL (*SE*)	ROC (*SE*)
Bayesian Surprise (Itti & Baldi, 2005)	0.034	0.581
SUNDAy (Zhang et al., 2009)	0.041	0.582
Our method	**0.262** (0.0085)	**0.589** (0.0031)

Table 3. Performance in predicting human eye fixations when viewing videos (Itti & Baldi, 2005).

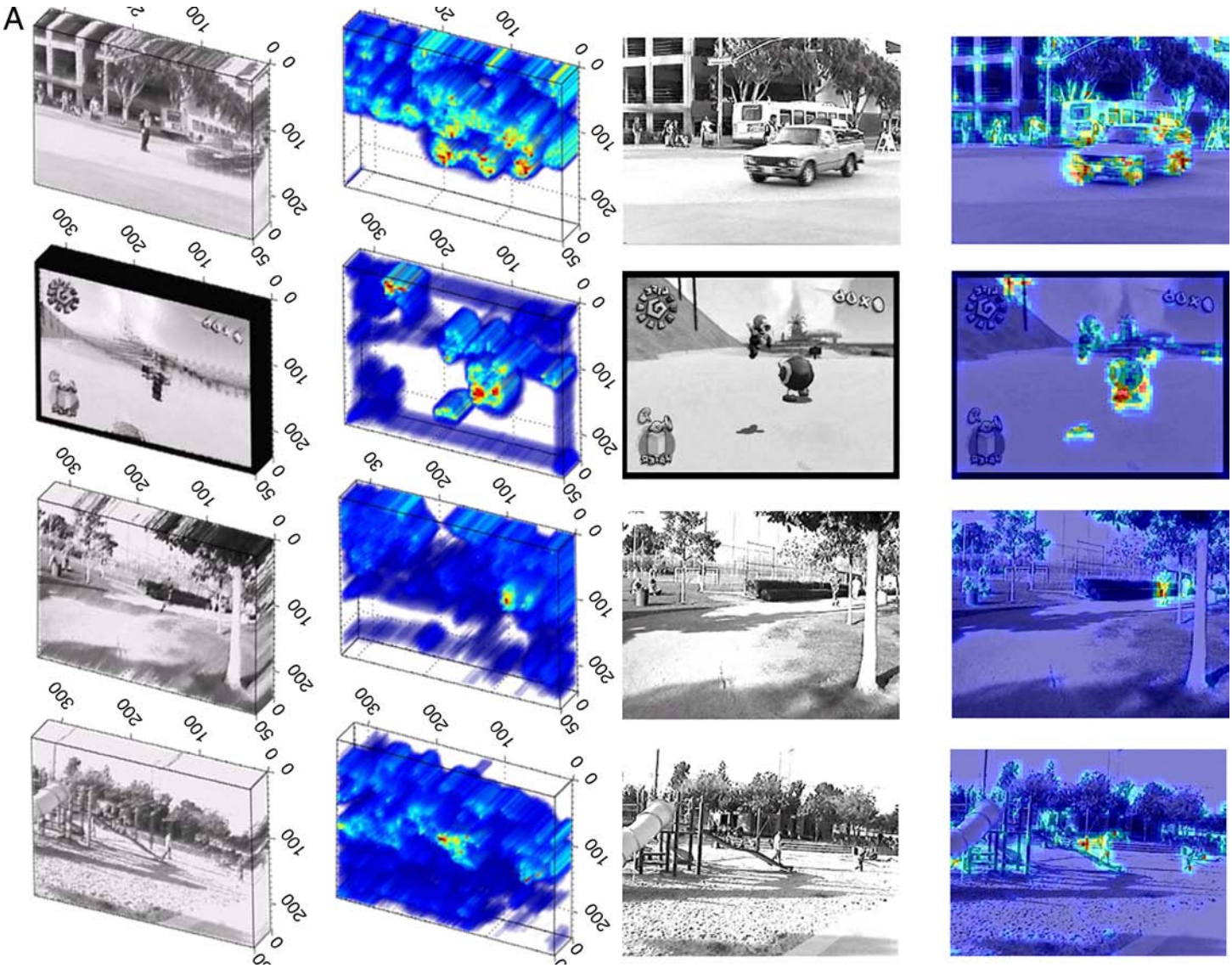

Figure 19. Some results on the video data set (Itti & Baldi, 2005): (a) video clips, (b) space-time saliency map, (c) a frame from (a), (d) a frame superimposed with corresponding saliency map from (b).

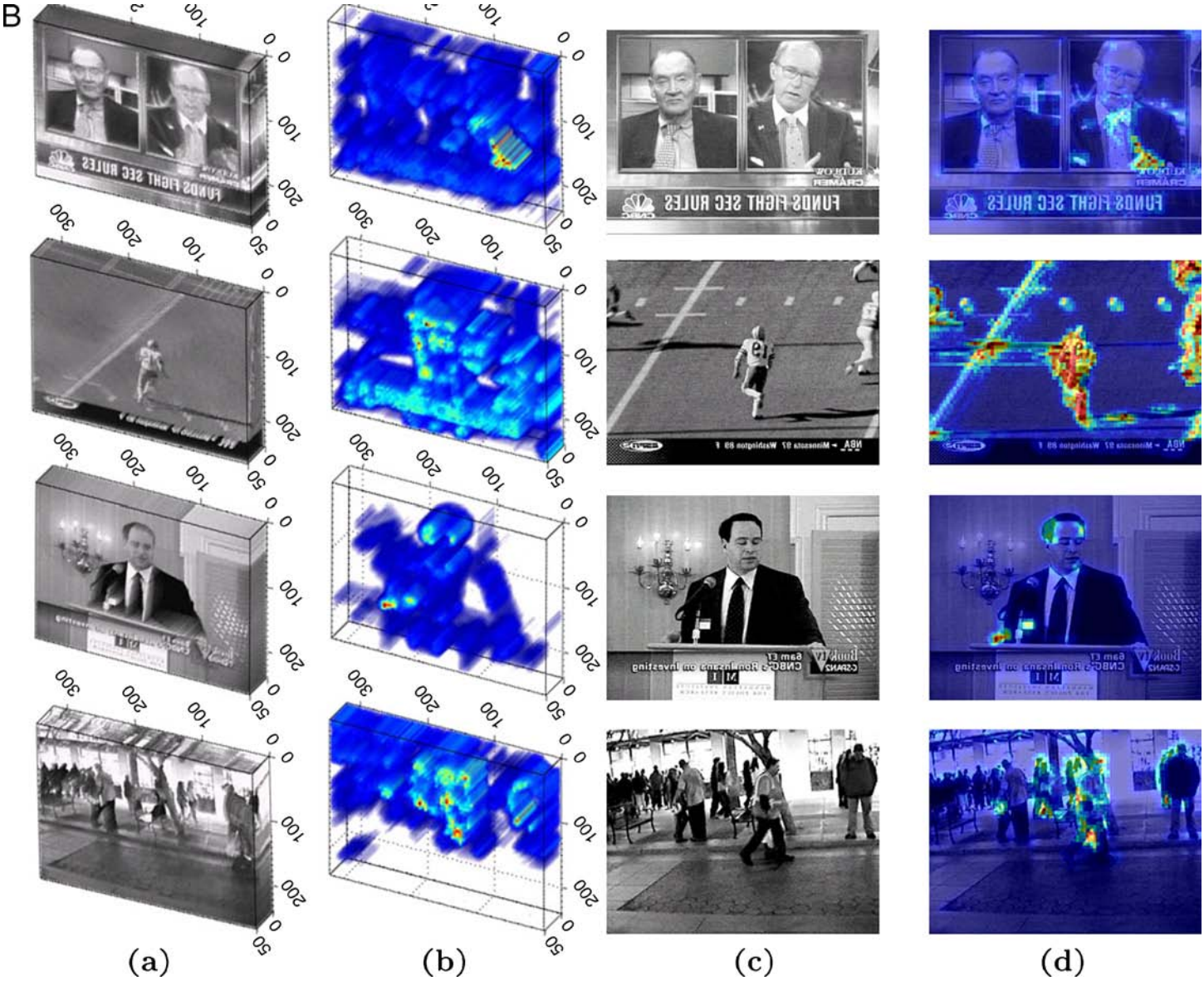

Figure 19. continued

Matlab on a video of about 500 frames in minutes on a Pentium 4, 3.8 GHz dual core PC with 1 GB RAM) on a video of size of 640 × 480 with about 500 frames while Bayesian Surprise requires hours because there are 432,000 distributions that must be updated with each frame.

Discussion

In the previous section, we have provided comprehensive experimental results which show that our method consistently outperforms other state-of-the art methods. It is worth noting that we estimate saliency by using non-parametric density estimation, while other competing methods (Gao et al., 2008; Itti et al., 1998; Torralba et al., 2008; Zhang et al., 2008) focused on fitting the conditional probability density function to a parametric distribution. In other words, we do not assume a distributional form or model for the data. As such, we call our method non-parametric. Even though we have a few parameters such as h, σ, and λ, these parameters are mostly set and fixed for all the experiments. Our model is somewhat similar to Gao et al. (2008) in the sense that a center-surround notion is used to compute saliency. One of the most important factors which makes the proposed method more effective is the use of LSKs as features. LSKs can capture local geometric structure exceedingly well even in the presence of signal uncertainty. In addition, unlike standard fusion methods which linearly and directly combine saliency maps computed from each color channel, we used the matrix cosine similarity to combine information from three color spaces. Our comprehensive experimental results indicate that the self-resemblance measure derived from a locally data-adaptive kernel density estimator is much more effective and simpler than other existing methods and does not require any training. Although our method is built entirely on computational principles, the resulting model structure exhibits considerable agreement with fixation behavior of the human visual system. With very good features like LSKs, the center-surround model is arguably an effective computational model of how the human visual system works.

Conclusion and future work

In this paper, we have proposed a unified framework for both static and space-time saliency detection algorithm by employing 2-D and 3-D *local steering kernels*; and by using a nonparametric kernel density estimation based on (*Matrix Cosine Similarity*) (MCS). The proposed method can automatically detect salient objects in the given image and salient moving objects in videos. The proposed method is practically appealing because it is nonparametric, fast, and robust to uncertainty in the data. Experiments on challenging sets of real-world human fixation data (both images and videos) demonstrated that the proposed saliency detection method achieves a high degree of accuracy and improves upon state-of-the-art methods. Due to its robustness to noise and other systemic perturbations, we also expect the present framework to be quite effective in other applications such as image quality assessment, background subtraction in dynamic scene, and video summarization.

Acknowledgments

The authors would like to thank Neil Bruce and John K. Tsotsos for kindly sharing their human fixation data, Laurent Itti for sharing his eye movement data, and Lingyun Zhang for sharing her Matlab codes and helpful discussion. This work was supported by AFOSR Grant FA 9550-07-01-0365.

Commercial relationships: none.
Corresponding author: Hae Jong Seo.
Email: rokaf@soe.ucsc.edu.
Address: Electrical Engineering Department, University of California, Santa Cruz, 1156 High Street, Santa Cruz, CA 95064, USA.

References

Bengio, Y., Larochelle, H., & Vincent, P. (2005). Non-local manifold parzen windows. In *Advances in Neural Information Processing Systems, 18,* 115–122.

Bregonzio, M., Gong, S., & Xiang, T. (2009). Recognising actions as clouds of space-time interest points. *IEEE Conference on Computer Vision and Pattern Recognition.*

Brox, T., Rosenhahn, B., & Cremers, H.-P. S. D. (2007). Nonparametric density estimation with adaptive anisotropic kernels for human motion tracking. *2nd Workshop on Human Motion, Springer-Verlag Berlin Heidelberg, 4814,* 152–165.

Bruce, N., & Tsotsos, J. (2006). Saliency based on information maximization. In *Advances in Neural Information Processing Systems, 18,* 155–162.

Chun, M., & Wolfe, J. M. (2001). Visual attention. *Blackwell handbook of perception* (pp. 272–310). Oxford, UK: Blackwell Publishers Ltd.

Fu, Y., & Huang, T. S. (2008). Image classification using correlation tensor analysis. *IEEE Transactions on Image Processing, 17,* 226–234. [PubMed]

Fu, Y., Yan, S., & Huang, T. S. (2008). Correlation metric for generalized feature extraction. *IEEE Transactions*

on Pattern Analysis and Machine Intelligence, 30, 2229–2235. [PubMed]

Gao, D., Mahadevan, V., & Vasconcelos, N. (2008). On the plausibility of the discriminant center-surround hypothesis for visual saliency. *Journal of Vision, 8*(7):13, 1–18, http://journalofvision.org/8/7/13/, doi:10.1167/8.7.13. [PubMed] [Article]

Gao, D., & Vasconcelos, N. (2004). Discriminant saliency for visual recognition from cluttered scenes. In *Advances in Neural Information Processing Systems, 17,* 481–488.

Gao, D., & Vasconcelos, N. (2005). Integrated learning of saliency, complex features, and object detectors from cluttered scenes. *IEEE Conference on Computer Vision and Pattern Recognition.*

Gorelick, L., Blank, M., Shechtman, E., Irani, M., & Basri, R. (2007). Actions as space-time shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 29,* 2247–2253. [PubMed]

Guo, C., Ma, Q., & Zhang, L. (2008). Spatio-temporal saliency detection using phase spectrum of quaternion Fourier transform. *IEEE Conference on Computer Vision and Pattern Recognition.*

Hou, X., & Zhang, L. (2008a). Dynamic visual attention: Searching for coding length increments. In *Advances in Neural Information Processing Systems, 21,* 681–688.

Hou, X., & Zhang, L. (2008b). Saliency detection: A spectral residual approach. *IEEE Conference on Computer Vision and Pattern Recognition.*

Itti, L., & Baldi, P. (2005). A principled approach to detecting surprising events in video. *IEEE Conference on Computer Vision and Pattern Recognition, 1,* 631–637.

Itti, L., & Baldi, P. (2006). Bayesian surprise attracts human attention. In *Advances in Neural Information Processing Systems, 18,* 1–8.

Itti, L., & Koch, C. (2001). Computational modeling of visual attention. *Nature Review Neuroscience, 2,* 194–203.

Itti, L., Koch, C., & Niebur, E. (1998). A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 20,* 1254–1259.

Kanan, C., Tong, M., Zhang, L., & Cottrell, G. (2009). SUN: Top-down saliency using natural statistics. *Visual Cognition, 17,* 979–1003.

Kienzle, W., Wichmann, F., Scholkopf, B., & Franz, M. (2007). A nonparametric approach to bottom-up visual saliency. In *Advances in Neural Information Processing Systems, 19,* 689–696.

Kullback, S. (1968). Information theory and statistics. New York: Dover Publications.

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

Ma, Y., Lao, S., Takikawa, E., & Kawade, M. (2007). Discriminant analysis in correlation similarity measure space. *IEEE International Conference on Machine Learning (ICML).*

Mahadevan, V., & Vasconcelos, N. (2008). Background subtraction in highly dynamic scenes. *IEEE Conference on Computer Vision and Pattern Recognition.*

Marat, S., Guironnet, M., & Pellerin, D. (2007). Video summarization using a visual attentional model. *EUSIPCO, EURASIP,* 1784–1788.

Marat, S., Phuoc, T., Granjon, L., Guyader, N., Pellerin, D., & Guerin-Dugue, A. (2009). Modelling spatio-temporal saliency to predict gaze direction for short videos. *International Journal of Computer Vision, 82,* 231–243.

Meur, O., Callet, P. L., & Barba, D. (2007). Predicting visual fixations on video based on low-level visual features. *Vision Research, 47,* 2483–2498. [PubMed]

Niassi, A., LeMeur, O., Lecallet, P., & Barba, D. (2007). Does where you gaze on an image affect your perception of quality? Applying visual attention to image quality metric. *IEEE International Conference on Image Processing.*

Oliva, A., Torralba, A., Castelhano, M., & Henderson, J. (2003). Top-down control of visual attention in object detection. *IEEE International Conference on Image Processing.*

Schuldt, C., Laptev, I., & Caputo, B. (2004). Recognizing human actions: A local SVM approach. *IEEE Conference on Pattern Recognition.*

Seo, H. J., & Milanfar, P. (2009a). Training-free, generic object detection using locally adaptive regression kernels. Accepted for publication in *IEEE Transactions on Pattern Analysis and Machine Intelligence.* Available online at http://ieeexplore.ieee.org/xpl/tocpreprint.jsp?isnumber=4359286&Submit3=View+Articles&punumber=34.

Seo, H. J., & Milanfar, P. (under review). Generic human action recognition from a single example. Submitted to *IEEE Transactions on Pattern Analysis and Machine Intelligence.*

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

Shahram, M. (2005). Statistical and information-theoretic analysis of resolution in imaging and array processing. Ph.D. thesis, University of California, Santa Cruz.

Shechtman, E., & Irani, M. (2007). Matching local self-similarities across images and videos. *IEEE Conference on Computer Vision and Pattern Recognition*.

Silverman, B. (1986). *Density estimation for statistics and data analysis. Monographs on Statistics and Applied Probability 26*. New York: Chapman & Hall.

Takeda, H., Farsiu, S., & Milanfar, P. (2007). Kernel regression for image processing and reconstruction. *IEEE Transactions on Image Processing, 16,* 349–366. [PubMed]

Takeda, H., Milanfar, P., Protter, M., & Elad, M. (2009). Super-resolution without explicit subpixel motion estimation. *IEEE Transactions on Image Processing, 18,* 1958–1975. [PubMed]

Torralba, A., Fergus, R., & Freeman, W. (2008). 80 million tiny images: A large dataset for non-parametric object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 30,* 1958–1970. [PubMed]

Treisman, A., & Gelade, G. (1980). A feature-integration theory of attention. *Cognitive Psychology, 12,* 97–136.

vande Sande, K., Gevers, T., & Snoek, C. (2008). Evaluation of color descriptors for object and scene recognition. *IEEE Conference on Computer Vision and Pattern Recognition*.

Vincent, P., & Bengio, Y. (2003). Manifold parzen windows. In *Advances in Neural Information Processing Systems, 15,* 825–832.

Wolfe, J. (1994). Guided search 2.0: A revised model of guided search. *Psychonomic Bulletin and Review, 1,* 202–238.

Wu, B., & Nevatia, R. (2007). Simultaneous object detection and segmentation by boosting local shape feature based classifier. *IEEE Conference on Computer Vision and Pattern Recognition*.

Yarbus, A. (1967). *Eye movements and vision*. New York: Plenum.

Zhang, L., Tong, M., & Cottrell, G. (2009). SUNDAy: Saliency using natural statistics for dynamic analysis of scenes. In *Proceedings of the 31st Annual Cognitive Science Conference, Amsterdam, Netherlands*. Mahwah: Lawrence Erlbaum.

Zhang, L., Tong, M., Marks, T., Shan, H., & Cottrell, G. (2008). SUN: A Bayesian framework for saliency using natural statistics. *Journal of Vision, 8*(7):32, 1–20, http://journalofvision.org/8/7/32/, doi:10.1167/8.7.32. [PubMed] [Article]