A Multi-size Superpixel Approach for Salient Object Detection based on Multivariate Normal Distribution Estimation

Lei Zhu, Dominik A. Klein, Simone Frintrop, Zhiguo Cao*, and Armin B. Cremers

***Abstract*—This article presents a new method for salient object detection based on a sophisticated appearance comparison of multi-size superpixels. Those superpixels are modeled by multivariate normal distributions in CIE-Lab color space, which are estimated from the pixels they comprise. This fitting facilitates an efficient application of the Wasserstein distance on the Euclidean norm ($\mathcal{W}_2$) to measure perceptual similarity between elements. Saliency is computed in two ways: on the one hand, we compute global saliency by probabilistically grouping visually similar superpixels into clusters and rate their compactness. On the other hand, we use the same distance measure to determine local center-surround contrasts between superpixels. Then, an innovative locally constrained random walk technique that considers local similarity between elements balances the saliency ratings inside probable objects and background. The results of our experiments show the robustness and efficiency of our approach against 11 recently published state-of-the-art saliency detection methods on five widely used benchmark datasets.**

***Index Terms*—Saliency detection, Multi-size superpixels, Wasserstein distance, Center-surround contrasts, Cluster compactness, Random walk.**

EDICS Category: 5. SMR-HPM, 2. SMR-SMD, 4. SMR-Rep, 33. ARS-IIU, 8. TEC-MRS

I. Introduction

HUMAN vision is usually capable of locating the most salient parts of the scene with a selective attention mechanism [1]. From the perspective of computer vision, salient region detection is still challenging since the human attention system has not been fully understood. However, an important attribute which makes a region salient is that it stands out from its surroundings in one or more visual features. In recent years, saliency detection has become a major research area and many computational attention systems have been built during the last decade that are based on this center-surround concept [2]. Applications of saliency detection include object detection [3], [4], image retrieval [5], [6], image and video compression [7], [8], as well as image segmentation [9], [10].

Lei Zhu and Zhiguo Cao are with the National Key Lab of Science and Technology on Multi-spectral Information Processing, School of Automation, Huazhong University of Science and Technology, 430074 Wuhan, China. e-mail: (zhulei.iprai@gmail.com, zgcao@mail.hust.edu.cn). Corresponding author is Zhiguo Cao.

Dominik A. Klein, Simone Frintrop, and Armin B. Cremers are with the Institute of Computer Science III, University of Bonn, 53117 Bonn, Germany. e-mail: ({kleind, frintrop, abc}@iai.uni-bonn.de).

The classical approaches for saliency computation stem from the simulation of human attention mechanisms. These approaches compute the saliency of a pixel as the difference of a center and a surround regions, both of which are centered at the pixel and can be rectangular or circular [11]–[13]. Therefore, we call these methods the *local saliency approaches*. The selection of surrounding regions is always a difficult problem for pixel-based or region-based methods due to the ambiguity of unknown object scales. A reasonable solution is the multi-scale scheme that computes the center-surround response at several different scales [11], [12], [14], [15]. Some existing approaches also explore the local contrast on single scale. In this case, the surroundings can be chosen as the maximum symmetric surround [16] or regions of the entire image with spatial weighting [17], [18].

Alternative approaches consider the occurrence frequency of certain features in the whole image, i.e., salient objects are more likely belonging to parts with rare observations in the frequency domain [19], [20]. We call these approaches the *global saliency approaches*. Zhai et al. [21] evaluate the pixel-level saliency by contrasting each pixel to all others. Achanta et al. [9] directly assign the salient value of a pixel with the difference from the average color. By abstracting the color components, the global contrast is efficiently computed in [17] at pixel level. Global saliency is also investigated via the visual organization rule, which can be computationally transformed into rating the color distribution [22].

Different from the methods based on the local or global contrast, some researchers work on the priors regarding the potential positions of foreground and background mathematically or empirically. Gopalakrishnan et al. [23] represent an image as a graph and search the most salient nodes and the background nodes using the random walk technique. By analyzing photographic images, Wei et al. [24] found that pixels located on four boundaries of an image contain the background attributes and validated this prior on two popular datasets. Recently, the assumption of boundary prior was investigated in several graph-based saliency models [25]–[28] and achieved impressive results.

In this work, a new segment-based saliency detection method is proposed. We mainly address two problems that are seldom discussed in previous work:

1) Saliency models which take color information as the primary feature often simply compute the region contrast as the Euclidean distance between the average colors of regions or as the histogram-based contrast. The former is efficient and

reasonable especially when regions are organized as superpixels. However, it might be imprecise when large regions are considered. Conversely, the histogram-based contrast is more precise in this case but still suffers from parameter problems such as the number of bins and the selection of metric space.

Instead, we represent the color appearance of superpixels by multivariate normal distributions. This bases on the assumption that the color occurrences of the pixels in each region follow a multivariate normal distribution. This assumption is especially well suited for superpixels since the clustered pixels have similar properties in the selected feature space. The difference between two superpixels is measured with the Wasserstein distance on the Euclidean norm ($\mathcal{W}_2$ distance), which was firstly introduced to compute the pixel-based saliency in our previous work [29]. Additionally, we also propose a fast algorithm to compute the $\mathcal{W}_2$ distance on N-d (N $\leq$ 3) normal distributions.

2) Holding a uniform saliency distribution of an object interior is difficult in the local saliency computation that is based on the center-surround principle. Typically, this problem can be alleviated by combining multi-layer saliency maps or, smoothing the single layer saliency map at pixel level [18]. We propose a locally constrained random walk procedure to directly refine the local saliency map at region level and achieve a more balanced rating inside of probable proto-objects. On the one hand, this approach can improve the final combination results. On the other hand, compared to the Gaussian weight-based up-sampling [18], it avoids spreading the error of saliency assignment to the background regions when inappropriate Gaussian weights for controlling the sensitivity to color and position are selected.

Thus, in a nutshell, the main contributions of this paper are

- A new representation of superpixels by multivariate normal distributions and their comparison with the Wasserstein distance, which is consistently used throughout the approach for local as well as global saliency computation. It enables to combine the advantage of the rich information of probability distributions to represent feature statistics with a computationally efficient method for representation and comparison.
- A novel saliency flow method, which is a locally constrained random walk procedure to refine the local saliency map. It achieves a more balanced rating inside of probable proto-objects and improves the performance significantly.

II. Related Work

The detection of visual saliency is one of the two aspects of human visual attention: bottom-up and top-down attention [1], [30]. Bottom-up attention relates to the detection of salient regions in the perceptual data by purely analyzing this data without any additional information. Top-down attention on the other hand considers prior knowledge about a target, the context, or the mental state of the agent. While top-down attention is an important aspect in human attention, prior knowledge is not always available and many computational methods profit from purely determining the bottom-up saliency. Among these application areas are object detection and segmentation, that we will consider here. Thus, we concentrate on the following approaches that deal with bottom-up saliency detection.

A. Pixel-based Saliency

The local contrast principle assumes that the more different an image region is compared to its local surround the more salient it is. One of the first pixel-based methods to detect saliency in a biologically motivated way was introduced by Itti et al. [11]. Their *Neuromorphic Vision Toolkit (iNVT)* computes the center-surround contrast at different levels in DoG scale space and searches the local maximum responses with a Winner-Take-All network. Harel et al. [31] extend the approach of Itti by evaluating the saliency with a graph-based method. In a recent approach, Goferman et al. [32] follow several basic principles of human attention and assume that the patches which are distinctive in colors or patterns are salient. The algorithm proposed by Achanta et al. [16] produces an original scale saliency map which can keep the boundaries of salient objects by accumulating the information of the surrounding area of each pixel. Milanfar and Peyman [33] compute the center-surround contrast of each pixel using a kind of local structure called *LSK* which is robust to the noise and variation of luminance. The approach of Liu et al. [14] combines local, regional, and global features in a CRF based framework. Li et al. [34] propose a method using the conditional entropy under the distortion to measure the visual saliency, which is also a center-surrounding scheme.

The pure global approaches assume that the more infrequent features occur in the whole image, the more salient they are. In [19] and [35], Hou et al. assign higher saliency values to those pixels which have higher response to the rare magnitudes in amplitude spectrum, and identify others as the redundant components. However, Guo et al. [20] found the image's phase spectrum is more essential than the amplitude spectrum to obtain the saliency map. Achanta et al. [9] also assume that the background has lower frequencies, and directly compared each pixel with the entire image in color space.

The global principle only works well if the background is free of uncommon feature manifestations. On the other hand, the local contrast principle involves the difficulty to estimate the scale of a salient object. To avoid this problem, such methods usually define several ranges of a pixel's neighborhood or construct a multi-level scale space of the original image. However, these local methods suffer more from the boundary blurring problem, since on unsuitable scales the foreground/background relation cannot be clearly decided.

B. Segment-based Saliency

Segment-based methods take homogeneous regions as the basic element rather than pixels. Cheng et al. [17] segment the image into regions with the algorithm proposed in [36], and obtain the saliency map by computing the distance between histograms which are generated by mapping the color vectors of each region into a 3D space. The same pre-segmentation method was also used in [13] and [37]. Instead of computing the dissimilarity between regions directly, Wei et al. [13]

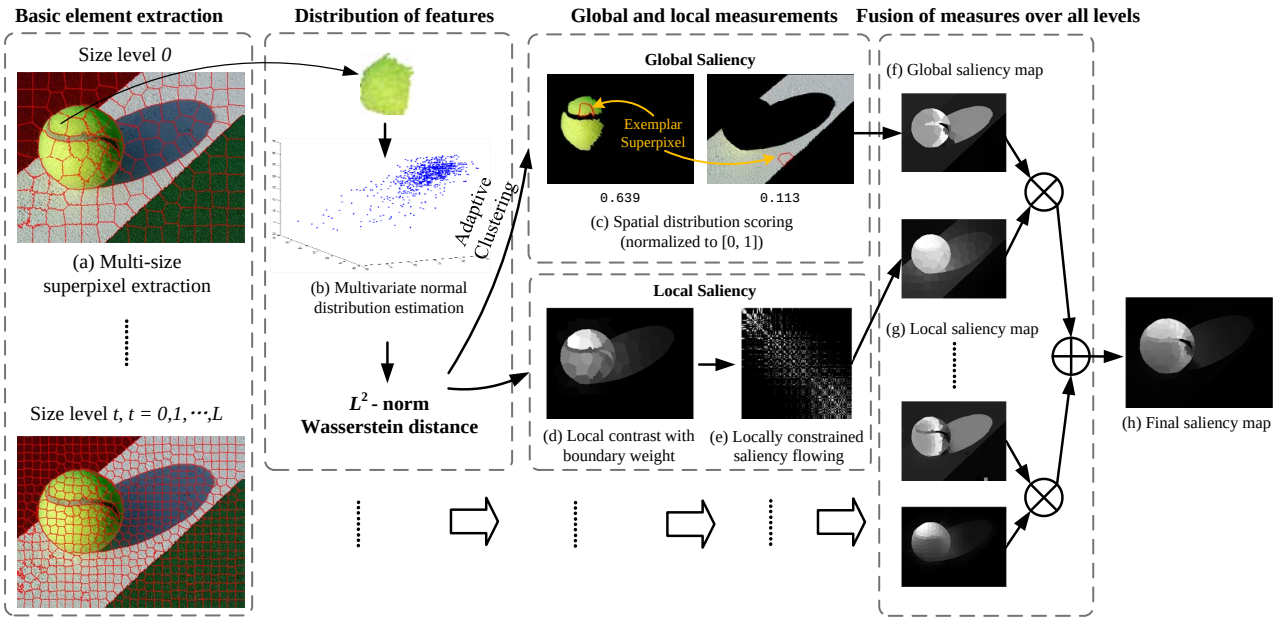


Fig. 1. The overall algorithm flowchart of out method. The structure of the algorithm is exemplarily presented for two scales. (a): each region surrounded by the red curves refers to one superpixel. (b): each superpixel is represented by the multivariate normal distribution estimated from its pixels. Based on the L^2-norm Wasserstein distance between every pair of superpixels, the local and global saliency are obtained. (c): *global saliency* computation: superpixels are clustered according to their color similarity and exemplar superpixels (cluster centers) are determined. The two images show exemplarily two of the clusters, the corresponding exemplar superpixels, and the cluster saliency scores that measure the spatial distribution of a cluster. (d) and (e): the *local saliency* is computed by a local contrast approach based on superpixels, which is further refined by a saliency flow step. (h): the final saliency map is obtained by fusing the global and local saliency maps ((f) and (g), respectively) over all scales.

obtain the saliency of an image window by computing the cost of composing the window from the remaining image parts. Park et al. [37] merge regions with their neighbors repeatedly according to the similarity of their HSV histograms, and update the saliency of joint regions in every combination. Ren et al. [38] first extract the superpixels from the image which are further clustered with GMM, and use the PageRank algorithm to obtain the superpixel-level saliency. Perazzi et al. [18] obtain the region-level saliency map by measuring the color uniqueness and spatial distribution of each extracted superpixel. A finer pixel-level saliency map is produced by comparing each pixel with all superpixels in both color space and location. Wei et al. [24] firstly proposed the background prior which assumes that the boundaries of an image can effectively represent the background components. Following this idea, Yang et al. [25] consider saliency detection as a graph-based ranking problem and use the label propagation to determine the region-level saliency. A similar graph model is employed in [26], which casts saliency detection into a random walk problem in the absorbing Markov chain.

III. Multi-size Superpixel-based Saliency Detection

We propose a superpixel based method for bottom-up detection of salient image regions. An image is segmented into a compound of visually homogeneous regions at different scale levels for representing its fine details as well as large scale structures. On each scale, two complementary approaches for the determination of saliency are employed separately: 1) In a global way, we measure the spatial compactness of similar-looking parts. Superpixels are more salient if they form a more coherent cluster within the image when categorized by their color appearances. 2) In a local way, we compute the center-surround contrast at the superpixel level. The more a superpixel differs from its surrounding ones, the more salient it is. Local contrast approaches usually grasp every pop-out region whose scale fits the current center-surround structure. That is, isolated background regions with an appropriate scale are also emphasized. In our work, the boundary prior [24] is used to eliminate the highlighted background regions. Furthermore, the local saliency map is refined by a locally constrained random walk procedure that dilutes saliency in the background and likewise balances it inside potential objects.

We assume that the appearance of pixels grouped into one superpixel is well expressed by the associated ML-estimate of a multivariate normal distribution in CIE-Lab space. This representation enables to efficiently measure visual difference/similarity between superpixels using the Wasserstein distance on the Euclidean norm [29]. Figure 1 shows a flowchart of our system.

A. Multi-size Superpixel Extraction

We use the SLIC superpixel extraction method introduced in [39], which divides an image into adjacent segments of about the same size containing as homogeneous colors inside as possible. For a given number of superpixels, the image is initially segmented into regularly sized grid cells. Then, iterative *K-Means* clustering is performed on a feature space that combines CIE-Lab colors and pixel locations. This clustering of nearby, similar-looking pixels refines the cells into superpixels. As mentioned in Section I, we extract superpixels at multi-

size levels. This is achieved by repeating the SLIC algorithm with different numbers of desired clusters, thus initializing with a coarser or finer grid. In our method, we increase the number of superpixels in steps of factor 2 between scale levels. The images in Figure 2a show examples of segmentation results with different superpixel sizes. Notice that we ensure a minimal cell size of 100 pixels when initializing the grid, because with less pixels it becomes increasingly unlikely to get meaningful appearance distributions. In our experiments, we analyzed $L = 3$ scale levels.

B. Superpixel Representation and Superpixel Contrast

We express the occurrence of low-level features in each superpixel by means of multivariate normal distributions. As argued in Section I, the unimodal distribution assumption is appropriate for superpixels. Different to [29], who splits the feature space into a one-dimensional lightness plus a two-dimensional color distribution, we directly use the original three dimensions of CIE-Lab color space. For conversion from RGB web images, we assume the D65 standard illuminant to be most suitable. For the notations in the following sections, the i^{th} superpixel of scale t forms a set:

$$\mathcal{S}_i^t = \left\{ \mathcal{N}_{\mathcal{S}}(\mu, \Sigma), \vec{c}_{\mathcal{S}} = \begin{pmatrix} x \\ y \end{pmatrix} \right\}_i^t \tag{1}$$

comprised of its feature distribution $\mathcal{N}_{\mathcal{S}i}^t$ and spatial center $\vec{c}_{\mathcal{S}i}^{\,t}$ in image coordinates.[1]

Several measuring techniques for distribution contrasts such as the KL-divergence [12], the Conditional Entropy [34] and the Bhattacharyya distance [40] have been employed in previous methods to identify local differences. Recently, Klein and Frintrop [29] applied the Wasserstein distance on the L^2-norm between feature distributions gathered from Gaussian weighted, local integration windows. We continue this idea, but instead, employ the Wasserstein metric to score contrasts between superpixels. The Wasserstein distance on the Euclidean norm in real-valued vector space is defined as

$$\mathcal{W}_2(\chi, \upsilon) = \left(\inf_{\gamma \in \Gamma(\chi,\upsilon)} \int_{\mathbb{R}^n \times \mathbb{R}^n} ||X - Y||^2 \, \mathrm{d}\gamma(X, Y) \right)^{\frac{1}{2}}, \tag{2}$$

where χ and υ are probability measures on the metric space $(\mathbb{R}^n, L^2)$ and $\Gamma(\chi, \upsilon)$ denotes the set of all measures on $\mathbb{R}^n \times \mathbb{R}^n$ with marginals χ and υ. Briefly worded, the Wasserstein distance represents the minimum cost of transforming one distribution into another, taking into account not only the individual difference in each point of the underlying metric space, but also how far one has to shift probability masses. In machine vision, the discretized $\mathcal{W}_1$ distance is also well known as *Earth Mover's Distance* and widely used to compare histograms.

The calculation of Eq. (2) is very demanding for arbitrary, continuous distributions, but thankfully can be solved to a more facile term in case of normal distributions. As introduced in [41][2], an explicit solution for multivariate normal distributions $\mathcal{N}_1(\mu_1, \Sigma_1)$ and $\mathcal{N}_2(\mu_2, \Sigma_2)$ is

$$\begin{aligned} \mathcal{W}_2(\mathcal{N}_1, \mathcal{N}_2) &= \left(||\mu_1 - \mu_2||^2 + \operatorname{tr}\left(\Sigma_1 + \Sigma_2 - 2\sqrt{\Sigma_1 \Sigma_2} \right) \right)^{\frac{1}{2}} \\ &= \left(||\mu_1 - \mu_2||^2 + \operatorname{tr}(\Sigma_1) + \operatorname{tr}(\Sigma_2) - 2 \operatorname{tr}\left(\sqrt{\Sigma_1 \Sigma_2} \right) \right)^{\frac{1}{2}}. \end{aligned} \tag{3}$$

In general, there is no explicit formula to obtain the square root of an arbitrary $n \times n$ matrix for $n > 2$, which would lead to an iterative algorithm for determining $\sqrt{\Sigma_1 \Sigma_2}$ in Eq. (3). However, noticing the relationship between the trace and the eigenvalues of a matrix, the trace of $\sqrt{\Sigma_1 \Sigma_2}$ can be represented as

$$\operatorname{tr}\left(\sqrt{\Sigma_1 \Sigma_2} \right) = \sum_{k=1}^{n} \lambda_{\Sigma_1 \Sigma_2}(k)^{\frac{1}{2}}, \tag{4}$$

where $\lambda_{\Sigma_1 \Sigma_2}(k)$ is the k^{th} eigenvalue of $\Sigma_1 \Sigma_2$.

Considering a $n = 3$ dimensional space such as CIE-Lab, given a 3×3 matrix A, its characteristic polynomial can be represented as

$$\begin{aligned} &\det(\lambda_A I - A) = \\ &\lambda_A^3 - \lambda_A^2 \operatorname{tr}(A) - \frac{1}{2} \lambda_A \big(\operatorname{tr}(A^2) - \operatorname{tr}^2(A) \big) - \det(A), \end{aligned} \tag{5}$$

where λ_A is an eigenvalue of A. λ_A can be directly determined using a trigonometric solution introduced in [43] by making an affine change from A to B as

$$A = pB + qI. \tag{6}$$

Thereby, B is a matrix with the same eigenvectors as A

$$\forall p \in \mathbb{R}_{\setminus 0}, q \in \mathbb{R} \Rightarrow \vec{v}_A = \vec{v}_B, \tag{7}$$

thus from the definition of eigenvalues it holds that

$$\overset{\text{Def., Eqs. (6), (7)}}{\Longleftrightarrow} \lambda_A = p \cdot \lambda_B + q, \tag{8}$$

where λ_B is an eigenvalue of B.

Choosing $p = \sqrt{\operatorname{tr}((A - qI)^2/6)}$ and $q = \operatorname{tr}(A)/3$ [3] as well as considering Eq. (5) to Eq. (8), the characteristic equation of B can be simplified to

$$\det(\lambda_B I - B) = \lambda_B^3 - 3\lambda_B - \det(B) = 0. \tag{9}$$

By directly solving Eq. (9), one can get all three eigenvalues of B as

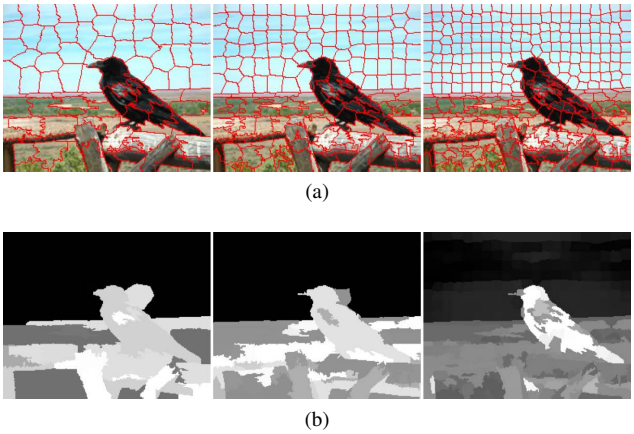
$$\lambda_B(k) = 2 \cos\left(\frac{1}{3} \arccos\left(\frac{\det(B)}{2} \right) + \frac{2k\pi}{3} \right), \tag{10}$$

where $\lambda_B(k)$ is the k^{th} eigenvalue of B with $k = 0, 1, 2$. Thus, Eq. (3) can be applied to quickly compute meaningful

[1] Note that $\mathcal{N}_{\mathcal{S}}$ denotes the normal distribution representing a superpixel, while $\mathcal{N}_{\mathcal{C}}$, that will be introduced in Section III-C, denotes the normal distribution representing a cluster.

[2] A slightly different term was later introduced in [42], claiming that Eq. (3) is only valid in case of commuting covariances. However, we could show that both solutions are equivalent, because $\Sigma_1 \Sigma_2 = \sqrt{\Sigma_1}(\sqrt{\Sigma_1}\Sigma_2)$ has the same characteristic polynomial as $(\sqrt{\Sigma_1}\Sigma_2)\sqrt{\Sigma_1}$, thus has the same eigenvalues.

[3] This choice guarantees the validity of Eqs. (6) and (8) also in the special case $p = 0$, since this would imply $A = qI$, thus it has a triple eigenvalue $\lambda_A = q = \operatorname{tr}(qI)/3$.



(a) (b)

Fig. 2. An example of multi-size superpixel segmentation and the corresponding global saliency maps. (a): from left to right, the initial grid area in superpixel extraction decreases approximately in steps of 2. (b): the images are the obtained global saliency maps corresponding to each scale in (a).

appearance distances between two superpixels using Eqs. (4), (6), (8), and (10).

In the following, we use the $\mathcal{W}_2$ distance coherently in the different aspects of saliency computation: it first serves as a similarity measure in the clustering approach for global saliency computation (Section III-C), second, it measures the local contrast of a superpixel to its neighbors (Section III-D), and third, it provides the similarity metric required for the random walk process that enables the saliency flow computation introduced in Section III-E.

C. Global Saliency: The Spatial Distribution of Colors

In natural scenes, the colors of regions belonging to the background are usually more spatially scattered in the whole image than in salient regions. In other words, the more the color is spread, the less salient it is [22]. To determine the spatial spreading, a further clustering is needed. This is computed much more efficiently on superpixels than would be possible on pixel level, since there are much less elements. Thereby, the spatial distribution of colors can be estimated in terms of a higher cluster-of-superpixels level by comparing the spatial intra-cluster distances. GMM method is widely used to represent the probabilities of color appearance, such as in [14], [38], [44]. However, it may be inappropriate to assign a fixed number of clusters for different images, since this should depend on the image complexity. e.g., a cluttered scene has much more dominant colors than one showing a monotonous background. We employ the APC algorithm (Affinity Propagation Clustering) introduced in [45] to identify clusters. Here, it is not necessary to initialize the cluster centers as well as the number of clusters.

APC is based on the similarities between elements (superpixels). It tries to minimize squared errors, thus in our method, we use $-(\mathcal{W}_2(\mathcal{N}_{si}^t, \mathcal{N}_{sj}^t))^2$ obtained by Eq. (3) between each pair of superpixels $\mathcal{S}_i^t$ and $\mathcal{S}_j^t$. Figure 1(c) shows exemplarily two identified clusters. Compatible to superpixels, the k^{th} cluster on scale t forms a set

$$\mathcal{C}_k^t = \left\{\mathcal{N}_c(\mu, \Sigma), \vec{c}_c\right\}_k^t. \tag{11}$$

APC selects so called *exemplar* superpixels to become cluster centers. Thus, we define the cluster appearance model $\mathcal{N}_c$ to equal the one of its corresponding exemplar superpixel. The spatial center of a cluster in image coordinates is computed from a linear combination of superpixel centers weighted by their cluster membership probability:

$$\vec{c}_c{}_k^t = \frac{\sum_{i=1}^{\mathrm{M}(t)} P_{\mathrm{g}}(\mathcal{C}_k^t|\mathcal{S}_i^t) \cdot \vec{c}_s{}_i^t}{\sum_{i=1}^{\mathrm{M}(t)} P_{\mathrm{g}}(\mathcal{C}_k^t|\mathcal{S}_i^t)}, \tag{12}$$

where $\mathrm{M}(t)$ denotes the number of superpixels on scale t.

Note that APC is also employed to group GMMs in [46]. Different from that work, the inherent message exchanged in APC is further explored to facilitate the computation of $P_{\mathrm{g}}(\mathcal{C}_k^t|\mathcal{S}_i^t)$. The membership probability of a superpixel to each cluster can be computed from its visual similarity to the exemplar of that cluster. Converting distances to similarities using Gaussian function has been widely adopted by numerous methods [18], [25], [26], [46], [47]. However, the fall-off rate of the exponential function is often selected empirically. In this section, we take advantage of the messages that are propagated between superpixels for directly determining the membership probabilities [48].

Let $\mathcal{X}_k^t$ denote the exemplar of cluster $\mathcal{C}_k^t$ and then, let $r(\mathcal{S}_i^t, \mathcal{X}_k^t)$ denote the exchanged message named *responsibility* which represents how well-suited superpixel $\mathcal{X}_k^t$ is to serve as the exemplar for superpixel $\mathcal{S}_i^t$. Actually, $r(\mathcal{S}_i^t, \mathcal{X}_k^t)$ implies the logarithmic form of the cluster membership probability [45]. Let $\mathbf{B}^t$ denote the set that is composed of all non-exemplar superpixels. We first normalize all responsibilities between the superpixels in $\mathbf{B}^t$ and exemplar $\mathcal{X}_k^t$ to $[-1, 0]$ (denoting as $\hat{r}(\mathbf{B}^t, \mathcal{X}_k^t)$) then exponentially scale them as

$$\hat{r}_e(\mathcal{B}_i^t, \mathcal{X}_k^t) = \exp\left(\hat{r}(\mathcal{B}_i^t, \mathcal{X}_k^t) \Big/ \mathrm{Var}\left(\hat{r}(\mathbf{B}^t, \mathcal{X}_k^t)\right)\right), \tag{13}$$

where $\hat{r}(\mathcal{B}_i^t, \mathcal{X}_k^t)$ refers to the normalized responsibility between the non-exemplar superpixel $\mathcal{B}_i^t$ and exemplar $\mathcal{X}_k^t$, and $\mathrm{Var}(\cdot)$ refers to the variance. For exemplars, we simply assign their scaled responsibilities as

$$\hat{r}_e(\mathcal{X}_i^t, \mathcal{X}_k^t) = \begin{cases} 1, & \text{if } i = k \\ 0, & \text{otherwise} \end{cases}. \tag{14}$$

Eqs. (13) and (14) construct the scaled responsibilities between all superpixels to each cluster. Then, the intra-cluster probabilities of each superpixel can be computed as

$$P_{\mathrm{g}}(\mathcal{C}_k^t|\mathcal{S}_i^t) = \hat{r}_e(\mathcal{S}_i^t, \mathcal{X}_k^t) \Big/ \sum_{k=1}^{\mathrm{K}(t)} \hat{r}_e(\mathcal{S}_i^t, \mathcal{X}_k^t), \tag{15}$$

where $\mathrm{K}(t)$ is the number of clusters on scale t. Next, we compute the probability of being salient for cluster $\mathcal{C}_k^t$. This probability value is obtained by scoring the relative spatial spreading between the superpixels within the cluster:

$$P_{\mathrm{g}}(sal|\mathcal{C}_k^t) = 1 \Big/ \sum_{j=1}^{\mathrm{K}(t)} \frac{\sum_{i=1}^{\mathrm{M}(t)} P_{\mathrm{g}}(\mathcal{C}_k^t|\mathcal{S}_i^t) \cdot ||\vec{c}_s{}_i^t - \vec{c}_c{}_j^t||^2}{\sum_{i=1}^{\mathrm{M}(t)} P_{\mathrm{g}}(\mathcal{C}_k^t|\mathcal{S}_i^t)}, \tag{16}$$

where $Sal = \{sal, \neg sal\}$ is a binary random variable, indicating whether something is salient, that means, whether

it belongs to the salient object in the image. This cluster saliency score $P_{\mathrm{g}}(sal|\mathcal{C}_k^t)$ is shown for two example clusters in Figure 1(c).

Finally, the global, superpixel-level saliency can be represented as the joint probability of cluster-level saliency and the cluster membership probability of that superpixel:

$$P_{\mathrm{g}}(sal|\mathcal{S}_i^t) = \sum_{k=1}^{\mathrm{K}(t)} P_{\mathrm{g}}(sal|\mathcal{C}_k^t) \cdot P_{\mathrm{g}}(\mathcal{C}_k^t|\mathcal{S}_i^t). \tag{17}$$

On each scale, the global saliency maps are computed by employing Eq. (17) with the aid of Eq. (12) to Eq. (16). Some global saliency maps of single scale levels are shown in Figure 2b. As expected, the global saliency maps exhibit that an object behaves salient in some but not all scales depending on its own size.

D. Local Saliency: The Local Contrast with Boundary Prior

If the salient object is homogeneous inside and shows a distinct color to a relatively clean background, it would often be sufficient to evaluate the superpixel saliency using the global method introduced in the last section. However, it is less effective if the background is cluttered or partially similar to the objects in terms of colors. An alternative way to solve this problem is introducing local information. For the scale t, the local contrast of superpixel $\mathcal{S}_i^t$ can be obtained by accumulating all the appearance distances from other superpixels as

$$P_{\mathrm{l}}(\mathcal{S}_i^t) = \frac{\sum_{j=1}^{\mathrm{M}(t)} \mathcal{W}_2(\mathcal{N}_{\mathcal{S}i}^t, \mathcal{N}_{\mathcal{S}j}^t) \cdot \mathrm{g}(i,j,t)}{\sum_{j=1}^{\mathrm{M}(t)} \mathrm{g}(i,j,t)}, \tag{18}$$

spatially Gaussian weighted with

$$\mathrm{g}(i,j,t) = \exp\left(-\left(\frac{||\vec{c}_{\mathcal{S}i}^{\,t} - \vec{c}_{\mathcal{S}j}^{\,t}||}{\sigma_c(t)}\right)^2\right).$$

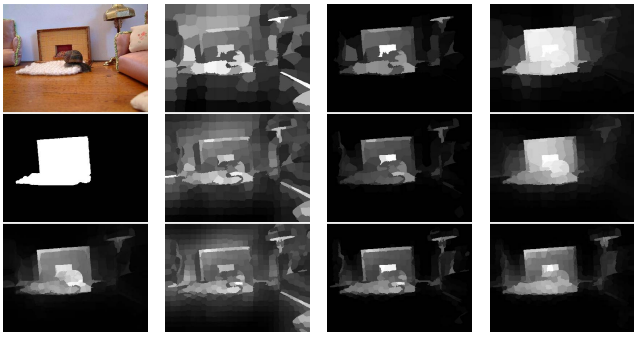
The parameter $\sigma_c(t)$ controls the spatial radius of neighbor range, and a higher value means that the contributing local surround is larger. The dependence on t ensures that the influence of neighbors is scaled proportionally. We choose this value according to the average distance of superpixels which is defined as follows:

$$\sigma_c(t) = \kappa \cdot \frac{\sum_{i=1}^{\mathrm{M}(t)} \sum_{j=1}^{\mathrm{M}(t)} \left(||\vec{c}_{\mathcal{S}i}^{\,t} - \vec{c}_{\mathcal{S}j}^{\,t}|| \cdot a_{i,j;t}\right)}{||A_t||_F}, \tag{19}$$

where $||\cdot||_F$ denotes the Frobenius norm which equals the number of '1' entries. κ is a damping constant and we set $\kappa = 4$ in our experiments. A_t is the symmetric adjacency matrix which uses binary values to indicate the connecting relationship between all superpixels of scale t:

$$a_{i,j;t} = \begin{cases} 1, & \text{if } \mathcal{S}_i^t \text{ is connected to } \mathcal{S}_j^t \\ 0, & \text{otherwise} \end{cases}. \tag{20}$$

As shown in Figure 3b, a local-contrast-based approach usually has two drawbacks: 1) it grasps all pop-out structures with the appropriate scale but doesn't particularly emphasize the most salient objects; and 2) it has difficulties of holding an analogous saliency scoring inside of objects [9].



(a) (b) (c) (d)

Fig. 3. From top to bottom: (a): the original image, ground truth and final saliency map. (b)-(d): the superpixels are extracted at 3 scale layers with descending sizes. (b): the local saliency maps. (c): the local saliency maps weighted by the boundary prior. (d): the refined local saliency map after the saliency flow step at different scales.

As the regions on the boundaries of an image usually contain the characteristics of the background [26], 1) can be effectively solved by weighting each superpixel with respect to the ones that lay on the image boundaries. We simply employ the idea proposed in [24] to get the weights. For facilitating the computation, the $\mathcal{W}_2$ distance is used to measure the region contrast. Let w_i^t denote the boundary weight of superpixel $\mathcal{S}_i^t$, the refined local salient score of $\mathcal{S}_i^t$ is

$$P_{\mathrm{l}}(sal|\mathcal{S}_i^t) = P_{\mathrm{l}}(\mathcal{S}_i^t) \cdot w_i^t. \tag{21}$$

Figure 3c shows the examples of incorporating the boundary prior to refine the local contrast maps in Figure 3b.

E. Saliency Flow: Region-based Smoothing on Single Scale

Improving the local saliency computation by incorporating the boundary prior still remains the problem of holding an analogous saliency scoring inside of objects. As a solution, we propose a procedure called *saliency flow* to refine the local saliency map by balancing the saliency values inside probable proto-objects. Different to [31] and [38], which setup a global random walk process, we use a locally constrained random walk process that the saliency is only allowed to flow between neighboring superpixels.

We construct a weighted graph model $G_t = \{V_t, E_t\}$ with $V_t = \left\{\mathcal{S}_1^t, \ldots, \mathcal{S}_{\mathrm{M}(t)}^t\right\}$ and E_t is the matrix of edge weights based on the $\mathcal{W}_2$ similarity metric and connectivity between superpixels:

$$e_{i,j;t} = \exp\left(-\left(\frac{\mathcal{W}_2(\mathcal{N}_{\mathcal{S}i}^t, \mathcal{N}_{\mathcal{S}j}^t)}{\sigma_d}\right)^2\right) \cdot a_{i,j;t}, \tag{22}$$

where $a_{i,j;t}$ is the adjacency matrix which is defined in Eq. (20). σ_d nonlinearly controls the fall-off rate of the $\mathcal{W}_2$ distances and we adaptively set $\sigma_d = \max_{i,j}\left(\mathcal{W}_2(\mathcal{N}_{\mathcal{S}i}^t, \mathcal{N}_{\mathcal{S}j}^t)\right)/3$ in the experiments. Finally, each row of E_t must be normalized to become a flow probability distribution. In each step of the random work, the saliency of superpixels at both sides of the

boundary between the background and the object attempt to move in two directions: The superpixels at the background side have a high probability to flow to the adjacent regions of background while the superpixels at the object side will flow towards the inside of object. The balanced local saliency after n steps of saliency flow can be expressed in vector and matrix notation as a special kind of power iteration:

$$\begin{aligned} f(0)^t &= \vec{p}^t \\ &= \left(P_{\mathrm{l}}(sal|\mathcal{S}_1^t), \ldots, P_{\mathrm{l}}(sal|\mathcal{S}_{\mathrm{M}(t)}^t)\right)^{\mathsf{T}} \\ f(n)^t &= f(n-1)^t \cdot E_t \\ &= \vec{p}^t \cdot (E_t)^n \end{aligned} \tag{23}$$

We found that the value of n crucially controls the final result. On the one hand, sufficient iterations are required to ensure a smoothed interior of objects. On the other hand, excessive iterations drive an inverse flow from objects to the background. In the extreme case, e.g., $n \to \infty$ produces a uniform saliency distribution that all superpixels have a consistent salient value. We introduce the *'Saliency Exchanged in Grouped Regions'* (SEGR) to evaluate the effect of saliency flow. SEGR must be a descending function of n as saliency flow always tries to balance both sides involved in an exchanging process. Let $f\,(n)_i^t$ denote the salient value of superpixel $\mathcal{S}_i^t$ after n iterations. The average value of SEGR can be measured as

$$\mathrm{T}(n) = \mathrm{mean}\left\{\sum_{i,j}\left|f\,(n)_i^t - f\,(n)_j^t\right| \cdot e_{i,j;t}\right\}, \quad \text{where } \mathcal{S}_i^t, \mathcal{S}_j^t \in \mathcal{C}_k^t. \tag{24}$$

$\mathrm{T}(n)$ counts the absolute saliency mass that is exchanged between neighboring superpixels only when they are in the same cluster identified in our global saliency computation. We observed that $\mathrm{T}(n)$ decreases sharply in the initial rounds of saliency flow and the rate of decline significantly slows down when the inside regions of objects are well smoothed. Therefore, an appropriate value of n can be selected as

$$\underset{n}{\mathrm{argmin}}\left\{\frac{\mathrm{T}(n)}{\mathrm{T}(n-1)} \times 100\%\right\} > c_{\min} \tag{25}$$

where $c_{\min} = 95\%$ was chosen in our experiments. Figure 3d shows the refined local saliency map after our saliency flow operation on different scales.

F. Feature Fusion and Multi-size Combination

On each scale t, we normalize both global and local saliency maps to the range of $[0, 1]$ and obtain the combined saliency map by superpixel-wise multiplication as follows:

$$\mathrm{s}(\mathcal{S}_i^t) = \frac{P_{\mathrm{g}}(sal|\mathcal{S}_i^t)}{\max_{j=1}^{\mathrm{M}(t)} P_{\mathrm{g}}(sal|\mathcal{S}_j^t)} \cdot \frac{f(n)_i^t}{\max_{j=1}^{\mathrm{M}(t)} f(n)_j^t}, \tag{26}$$

where $P_{\mathrm{g}}(sal|\mathcal{S}_i^t)$ is the global saliency obtained in Eq. (17) and $f(n)_i^t$ is the i^{th} component of the saliency flow result vector obtained in Eq. (23), thus corresponds to the balanced local saliency of superpixel $\mathcal{S}_i^t$. The multiplication is used here because a salient object should be outstanding in both saliency measurements on the same scale [18], [49].

We assign the saliency value $\mathrm{s}(\mathcal{S}_i^t)$ to all pixels of superpixel $\mathcal{S}_i^t$ for generating the pixel-level saliency map. Furthermore, the arithmetic mean of the normalized pixel-level saliency maps over all scales is used to generate the final saliency map as follows:

$$\mathrm{saliency}(x,y) = \frac{1}{L}\sum_{t=1}^{L}\sum_{i=1}^{\mathrm{M}(t)} \begin{cases} \mathrm{s}(\mathcal{S}_i^t), & \text{if } (x,y) \in \mathcal{S}_i^t \\ 0, & \text{otherwise} \end{cases}. \tag{27}$$

IV. Experiments

In this section, we evaluate our salient region detection method. We compare it with 11 most recently proposed state-of-the-art saliency detection approaches: AMC [26], CHM [50], MR [25], HSD [51], SIA [46], BMS [52], STD [53], LMC [47], GS [24], SF [18], RC [17]. All methods are evaluated on images from 5 widely used datasets:

ASD [9]: this dataset contains 1000 images from MSRA dataset and Achanta et al. created accurate binary maps for each image that provide accurate, object-contour-based reference data [9].

MSRA-B [14]: this dataset is an extension of ASD, which contains 5000 images. Each image was originally labeled from 9 users with a bounding box that enclosing the most salient objects [14]. Furthermore, Jiang et al. [27] manually segmented the salient objects and obtained the exact binary ground truth for this dataset.

SOD [54]: this dataset is based on the 300 images Berkeley segmentation dataset [55] and the foreground salient object masks were obtained by several subjects in the work of [54]. However, consistent foreground salient object masks weren't provided in this literature. In our evaluation, we follow the strategy that is introduced in [24] to generate the final ground truth annotations.

SED1 and **SED2** [56]: SED1 is a single object database while two objects exist in each image from SED2. Both datasets contain 100 images which were labeled by several subjects and we consider a pixel salient if it is annotated by all subjects.

Figure 7 shows several examples of saliency maps to enable a visual comparison of the different approaches. Quantitative experiments follow in this section, which is organized as follows. In Section IV-A, the internal baseline methods of our approach are separately evaluated. We segment the saliency maps with fixed thresholds and evaluate them in terms of the precision versus recall measure. In Section IV-B, similar measure is taken to compare our approach with other methods on 5 benchmarks described above. Furthermore, instead of using a constant value, each image is segmented with a threshold dependent on the saliency map. The results are evaluated with the F-measure in Section IV-C. Our saliency detector is employed for facilitating the object segmentation task in Section IV-D. The test images are segmented by the GrabCut algorithm [57] which is initialized by our saliency maps. The comparative results of other methods are also included in this subsection. The whole evaluations as well as the computation complexity of our algorithm are discussed in Section IV-E.

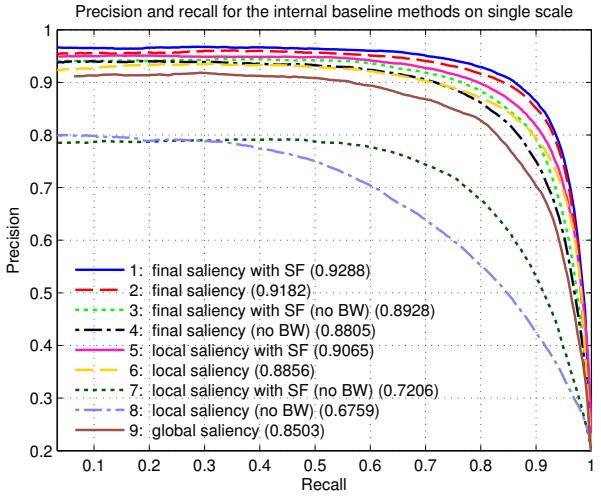


Fig. 4. Evaluations of the baseline methods of our approach on ASD. Values of *Area Under Curve* (AUC) are labeled in the parentheses after their corresponding legends. 'BW' indicates that the local saliency is weighted by incorporating the boundary prior that is described in Section III-D. 'SF' refers to the saliency flow refinement which is introduced in Section III-E.

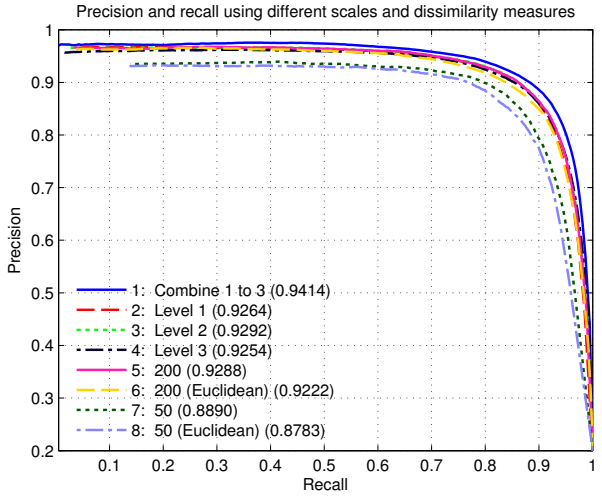


Fig. 5. Evaluations of the impacts induced by using different scales and distance measures on ASD. The scales from level 1 to 3 refer to descending initial sizes of superpixels. The names of legends starting with 50 and 200 refer to the additional layers which respectively segments each image into about 50 and 200 regions. *Euclidean* in the parentheses indicates that the Euclidean distance is used as the basic metric.

A. Evaluations of internal baseline methods

Our approach evaluates two attributes of pre-segmented regions: the global rareness and the local prominence with an effective $\mathcal{W}_2$ distance measure. Both evidences work in different ways independently and usually give prominence to same salient objects while suppress different background regions in an image. In this section we use the precision versus recall (PR) scores to evaluate the individual contribution of each component of our approach. For each sample, 256 binary maps are obtained by segmenting its saliency map with all possible thresholds. Then 256 pairs of precision and recall values are computed by comparing each binary map with the human-masked ground truth. Finally, we average both precision and recall of all images in the database to get an overall evaluation of the selected benchmarks.

Figure 4 shows the PR scores that are obtained by independently using each baseline method of our approach to compute the saliency maps of images from ASD. For exploring the contribution of each component more explicitly, the results are produced on single scale that each image is segmented into about 200 regions. It is interesting that the performance of local saliency computation is greatly boosted by incorporating the boundary prior (see the 6^{th} and the 8^{th} curves in Figure 4), which was proved to be a reasonable assumption on this dataset [24]. However, the boundary prior might be more effective in certain applications such as analyzing photographic images than the others. Therefore, the performance of our local saliency computation using simple center-surround cue is also evaluated. As the 3^{rd} curve in Figure 4 illustrating, without the boundary prior, our approach based on single scale also achieves similar performance with several methods that are evaluated in the following section. Additionally, the 7^{th} and the 8^{th} curves in Figure 4 proves that the proposed saliency flow technique also improves the performance significantly. The 5^{th} and the 6^{th} curves show another example. Although the result of local saliency computation with the boundary prior is quite acceptable, it is still promoted by the saliency flow technique considerably.

The most important parameter of our algorithm is the number of scales that is used for the superpixel extraction. As shown in Figure 2b, different color components are emphasized in the global saliency map at different scales. In addition, taking more scales also means a great increase in the computation complexity. As introduced in Section III-A, the restriction of the number of pixels per superpixel in the finest scale ensures a correct estimation of Gaussian distribution. On the other hand, the clustering of color components also requires a proper number of superpixels in the largest scale. Considering the resolution of images in the datasets, in total, 3 scale levels are investigated in our experiments so far, generated by increasing the area of the initial grid in steps of factor 2. In contrast, Figure 5 shows the PR scores of our method using different numbers of superpixels. Our algorithm evaluated individually for each scale achieves similar performance (see the 2^{nd}, the 3^{rd} and the 4^{th} curves). When combining 3 scale levels, our method outperforms each of them across the entire range of recall values.

Many segment-based methods simply use the L^2-norm to measure the pairwise dissimilarities between segments. Representing segments by distributions enables to capture more information about the statistics of the feature distributions. Figure 5 shows several PR curves of our algorithm when replacing the $\mathcal{W}_2$ distance to the Euclidean distance. For exploring the relationship between the impacts induced by different measures and the size of associated regions, two additional layers are designed to respectively segment each image into about 50 and 200 regions. It is clear that there is a drop in PR scores when simplifying the $\mathcal{W}_2$ distance to the Euclidean distance, e.g., about 0.01 off in AUC is found when segmenting each image into 50 regions. Figure 5 also

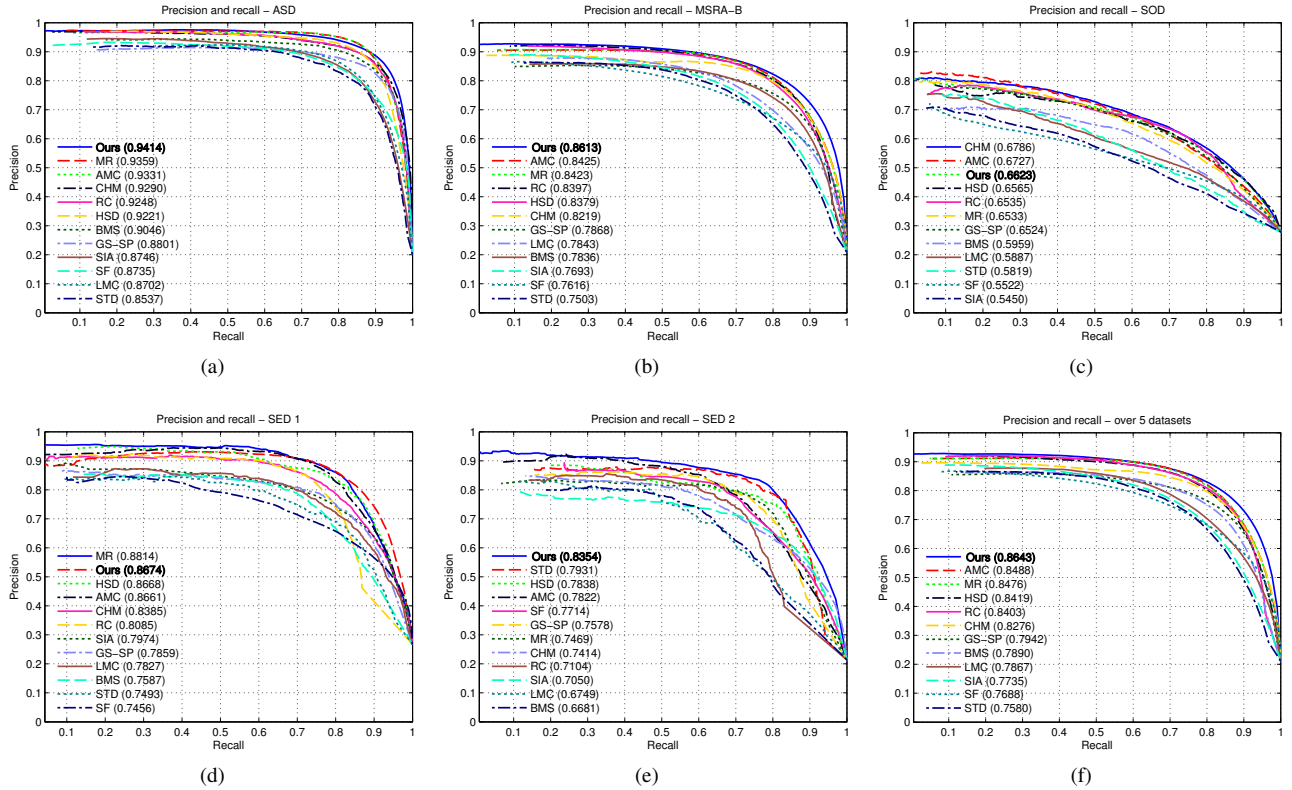


Fig. 6. PR curves for all evaluated methods using fixed thresholds on 5 popular saliency datasets: (a) ASD, (b) MSRA-B, (c) SOD, (d) SED1, (e) SED2, and (f) Overall evaluation on 5 datasets. All methods are sorted by descending AUCs. Please see Figure 7 for the visual comparison of selected sample results of all evaluated methods (this figure is best viewed in color).

illustrates that the impact is decreasing along with reducing the size of regions. However, our experimental results consistently demonstrate that measuring the region contrast based on the $\mathcal{W}_2$ distance always performs better than the one based on the Euclidean distance.

B. Salient region segmentation using fixed thresholds

With the method introduced in Section IV-A for computing the precisions and recalls, Figure 6 compares the PR scores of our method with other algorithms on 5 datasets. It is worth noting that, decreasing the threshold T_f from 255 to 0 corresponds to increasing the recall from 0 to 1. Several methods such as LMC [47] and BMS [52], have a significant larger minimum rate of recall than other methods, which means that they have a uniform saliency assignment in the detected salient regions. In contrast, the minimum recalls of other methods which are near 0% usually obtain continuous saliency maps [37]. On the right side, the far right ends of the curves all approaching $\sim$0.2 to 0.3 exhibit that, on average, the salient regions take about 20% to 30% of pixels in the images from all datasets.

As shown in Figure 6b, our approach outperforms other evaluated methods for all given recalls on MSRA-B which is the largest dataset that is used in our experiments. Also on its subset ASD, our approach achieves the highest AUC value from all 12 methods. On the datasets SOD and SED1, our method is among the top three methods, with only a small difference in AUC value to the top methods. The results on SED2, in which our method also achieves the highest AUC value of all methods, show that our approach is also able to cope with multiple objects per image. Figure 6f shows the overall evaluations on the dataset that combines all 5 datasets, which demonstrates the robustness of our approach.

C. Salient region segmentation by adaptive thresholding

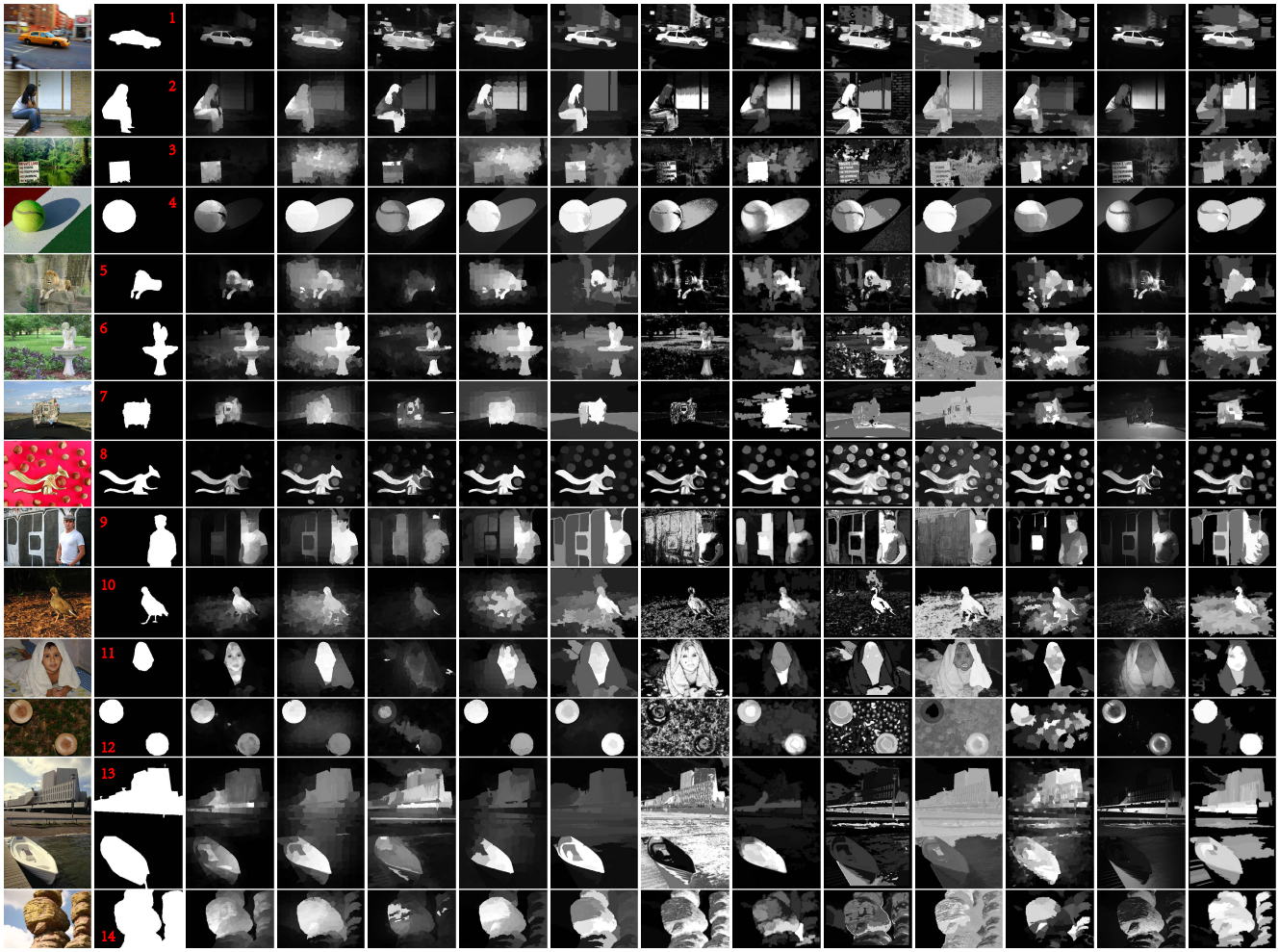
Instead of using a constant threshold value when binarizing different images, a simple, yet adaptive way to extract the foreground objects is to set the threshold dependent on the average saliency. In this part, we evaluate the performance of our proposed algorithm with the adaptive thresholding method introduced in [9], which defines the threshold as follows:

$$T_a = \frac{2}{W \cdot H} \sum_{x=1}^{W} \sum_{y=1}^{H} \text{saliency}(x, y) \tag{28}$$

where W and H are width and height of each tested saliency map, respectively. Furthermore, based on the obtained precision and recall value, the F-measure is computed to compare the performance of each method over the whole database, which is defined as

$$F_\beta = \frac{(1 + \beta^2) \times \mathit{precision} \times \mathit{recall}}{\beta^2 \times \mathit{precision} + \mathit{recall}}. \tag{29}$$

Similar to in [9] and [18], β^2 is set to 0.3 in our evaluation. Figure 8 compares the precision, recall and F-measure values



(a) original (b) GT (c) Ours (d) AMC (e) CHM (f) MR (g) HSD (h) SIA (i) BMS (j) STD (k) LMC (l) GS (m) SF (n) RC

Fig. 7. Visual comparison of other approaches to our proposed method. From left to right: original image, ground truth, our approach, AMC [26], CHM [50], MR [25], HSD [51], SIA [46], BMS [52], STD [53], LMC [47], GS [24], SF [18], and RC [17].

of all evaluated approaches. We observed that AMC [26], MR [25] and our approach are with similar performance on ASD, MSRA-B, SOD and SED1. As demonstrated in Figure 8e, our approach shows better performance than the others on SED2 which is a consistent two-objects dataset. Similar to Figure 6f, Figure 8f shows the overall evaluations of all methods on 5 datasets.

The evaluations of F-measure also show that our approach always achieves higher precision values than the others, that is, usually capable of avoiding annotating background regions. Taking several samples that are illustrated in Figure 7 for example, our saliency maps have relatively clean background. It is worth noting that, our approach gets lower recall values than some methods with the thresholds selected by Eq. (28). Distinct regions of an object may be emphasized with varying degrees in our salient map. As shown in the 4^{th} row of Figure 7, regions of the ball are assigned with different salient scores by our saliency computation. Unlike some methods that uniformly annotate the whole ball, high threshold values may cause incomplete segmentations to our saliency map.

D. Interactive segmentation using saliency maps

Compared to the unsupervised segmentation, the interactive manner is more useful since the extraction of foreground objects often greatly depends on the selective human vision. The supervised segmentation is usually initialized by one or several manually marked regions which can be also selected by saliency detection approaches. Cheng et al. [17] iteratively perform the *GrabCut* algorithm [57] on the original image and take a binarized saliency map instead of manually selected regions in each iteration. Federico et al. [18] and Xie et al. [47] use their saliency map to initialize the graph and employ the *min-cut* algorithm introduced in [58] as the post segmentation.

We follow the method introduced in [17] in this subsection. Instead of using a manually labeled rectangle, the GrabCut algorithm is initialized by a four-valued mask which is produced by pre-segmented saliency maps. The saliency maps are first segmented with that fixed threshold which achieves 85% recall on average with regard to the examined approach (cf. Figure 6). For the foreground of each segmented map (S_f), the erosion and dilation operations are performed for obtaining two resized regions based on S_f. The foregrounds produced by erosion operation (E_f) and the dilation operation (D_f) as

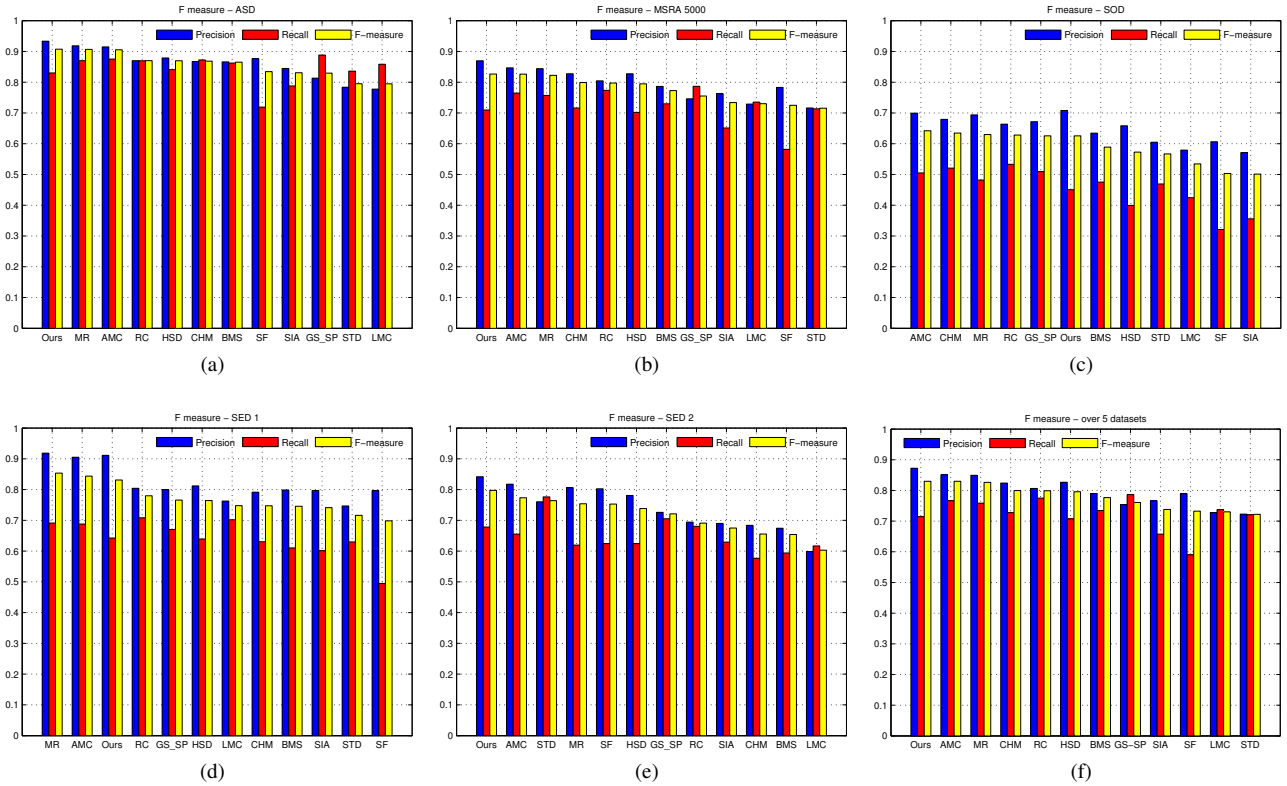


Fig. 8. The precision, recall and F-measure of all evaluated methods using adaptive thresholding on 5 saliency datasets: (a) ASD, (b) MSRA-B, (c) SOD. (d) SED 1, (e) SED2, and (f) Overall evaluation on 5 datasets. All methods are sorted by descending F-measure.

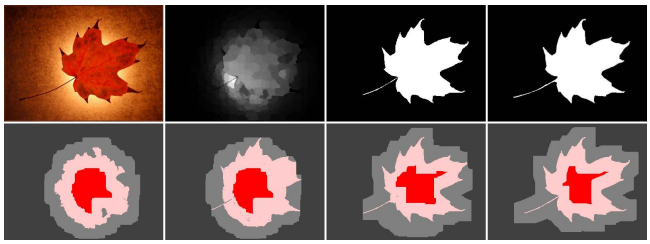


Fig. 9. The images in the first row, from left to right, refer to the original image, the saliency map, the segmentation result and the ground truth, respectively. The images in the second row, from left to right, refer to the initial four-valued mask and three segmentation results after each iteration. Each image in the second row is divided into 4 regions which are labeled with different colors: red (foreground), pink (probably foreground), gray (probably background) and dark gray (background).

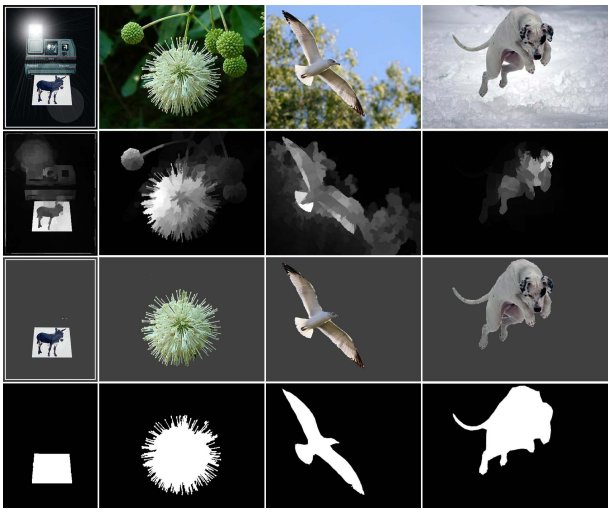


Fig. 10. From top to bottom: the original images, our saliency maps, segmentation results of the GrabCut algorithm initialized with the pre-segmented saliency maps, and the corresponding ground truths.

well as S_f divide the mask into 4 regions. Each pixel of the four-valued mask belongs to one of the regions E_f, $S_f - E_f$, $D_f - S_f$, and outside of D_f, hence is labeled as foreground, probably foreground, probably background, and background, respectively. We iterate the GrabCut algorithm several times (3 in our experiments). After each iteration, the intermediate result of segmentation is used to update S_f and further to obtain new E_f and D_f for the next time. Figure 9 shows an example of the initial four-valued mask and the segmentation results after each iteration.

Several segmentation results are shown in Figure 10. It is evident that our saliency maps work well for interacting with the GrabCut algorithm. Taking the images in the 1st column for example, without prior knowledge, the segmentation methods are more likely to extract the entire printer as the foreground since the original image has a very clean background. By giving a labeled foreground offered by our saliency map, the GrabCut algorithm can successfully find the manually labeled salient object. Conversely, the post segmentation significantly improves the result of our saliency detection method even when the saliency maps are not as good as expected. e.g., sev-

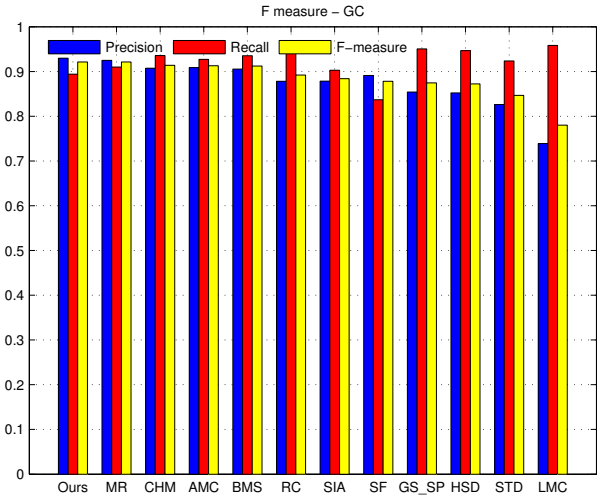


Fig. 11. The precision, recall and F-measure of all evaluated methods using the GrabCut algorithm as the post segmentation on ASD.

eral background regions are highlighted in the saliency maps in the 2nd and the 3rd columns of Figure 10, as well as some parts of the salient object behave to be much less salient in the last column. Figure 11 compares the precision, recall and F-measure of all evaluated approaches that use the GrabCut algorithm as the post segmentation on ASD.

E. Discussion

The evaluations on these benchmarks show that none of the methods outperforms all other ones on all datasets, while Figure 6f and Figure 8f demonstrate that our approach is in general more robust than the other methods. We believe that the multi-cue integration helps to achieve this performance. Taking HSD [51] for example, it computes the region-level saliency using a multi-scale, local-contrast-based approach that is similar to our local saliency model. the results of HSD [51] in the 7th and the 9th rows of Figure 7 show that background regions that stand out from their surroundings could be easily annotated as salient objects when solely considering the local contrast principle. Other single-cue approaches such as MR [25] and AMC [26] follow the hypothesis of the boundary prior [24] and formulate the saliency detection problem as a graph-based similarity propagation process. However, background regions that are near the center of an image usually have lower rankings (e.g., in MR [25]) or longer absorbed time (e.g., in AMC [26]) than the similar-looking regions that are near the image boundaries. Those center-located background regions may present competitive saliency when the salient object is near the image boundaries or has low contrast to them. Such examples are illustrated in their results in the 3rd, the 5th and the 10th rows of Figure 7. Another interesting method is CHM [50], which groups the superpixels into multi-scale cliques and scores the clique saliency according to the length of its boundaries. However, a large superpixel clique may contain both the salient and background regions. the result of CHM [50] in the 1st row of Figure 7 shows that, several background regions are also highlighted as they have similar appearances to the salient object.

The described failure cases show that, saliency detection models using single principle may produce unsatisfactory results in some cases. However, integrating the local, global and boundary priors with an effective $\mathcal{W}_2$ measure enables our approach to obtain more precise saliency maps than the results of the methods that are exemplarily analyzed above.

With the parameters chosen in our evaluation, on average, our algorithm takes about 0.85s for processing an image around the resolution of 300×400. Timings were tested on an Intel Core i7-4770 at 3.4 GHz with 4 GB RAM using double precision computations. The computational complexity of our algorithm relies essentially on the number of used scales, e.g., the processing time on level 1 to 3 scale is 0.07s, 0.18s and 0.60s, respectively. Most time is consumed on the distance computation and the clustering, which respectively occupies 40% and 43% of the processing time for an image. As the codes haven't been optimized, we are confident to make this system close to real-time performance by standard optimizations and parallelization (e.g., in the distance computation).

V. Conclusion

In this paper, we have presented a new computational salient object detection method based on multi-size superpixels. Each superpixel is represented by a multivariate normal distribution in CIE-Lab color space and their perceptual similarity is measured by the Wasserstein metric on the Euclidean norm. This distance measure is coherently used in the different parts of saliency computation. The overall saliency is composed of a global and a local component that capture different aspects of saliency: the global saliency assigns higher saliency values to compact image regions, while the local saliency emphasizes segments that visually stand out of their local environment. Experimental results demonstrate that our method achieves better performance than 11 recently published saliency detectors in overall comparisons on 5 widely used datasets.

In future research, we plan to integrate non-color descriptors such as gradient histograms into the local saliency part of our approach. It will improve the performance in the scene when the object isn't discriminative in color. Followed by our superpixel-level saliency map, a further pixel-level refinement will be also taken into account for obtaining a more detailed representation of objects. We are also interested in introducing the current approach into some high-level vision tasks such as object detection and recognition.

Acknowledgement

We express our gratitude to our colleagues of the Intelligent Vision Systems Group, University of Bonn: Germán Martín García and Shanshan Zhang for valuable discussions and some ideas in programming.

References

[1] H. E. Pashler, "The Psychology of Attention." MIT Press, Cambridge, Massachusetts, Oct. 1997.
[2] S. Frintrop, E. Rome, and H. I. Christensen, "Computational visual attention systems and their cognitive foundations, A survey," *ACM Trans. Appl. Percept.*, vol. 7, no. 1, pp. 1–39, Jan. 2010.

[3] J. Zhu, J. Wu, Y. Wei, E. Chang, and Z. Tu, "Unsupervised object class discovery via saliency-guided multiple class learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 3218–3225.

[4] B. Alexe, T. Deselaers, and V. Ferrari, "Measuring the objectness of image windows." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2189–2202, Nov. 2012.

[5] H. Fu, Z. Chi, and D. Feng, "Attention-driven image interpretation with application to image retrieval," *Pattern Recognit.*, vol. 39, no. 9, pp. 1604–1621, Sep. 2006.

[6] E. Nowak, F. Jurie, and B. Triggs, "Sampling strategies for bag-of-features image classification," in *Proc. European Conf. Comput. Vis.*, May 2006, pp. 490–503.

[7] L. Itti, "Automatic foveation for video compression using a neurobiological model of visual attention," *IEEE Trans. Image Process.*, vol. 13, no. 10, pp. 1304–1318, Oct. 2004.

[8] C. Guo and L. Zhang, "A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression," *IEEE Trans. Image Process.*, vol. 19, no. 1, pp. 185 –198, Jan. 2010.

[9] R. Achanta, S. Hemami, F. Estrada, and S. Süsstrunk, "Frequency-tuned salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2009, pp. 1597–1604.

[10] L. Wang, J. Xue, N. Zheng, and G. Hua, "Automatic salient object extraction with contextual cue," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 105–112.

[11] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, Nov. 1998.

[12] D. A. Klein and S. Frintrop, "Center-surround divergence of feature statistics for salient object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2214–2219.

[13] J. Feng, Y. Wei, L. Tao, C. Zhang, and J. Sun, "Salient object detection by composition," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 1028–1035.

[14] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H.-Y. Shum, "Learning to detect a salient object." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 353–367, Feb. 2011.

[15] E. Rahtu, J. Kannala, M. Salo, and J. Heikkilä, "Segmenting salient objects from images and videos," in *Proc. European Conf. Comput. Vis.*, Sep. 2010, pp. 366–379.

[16] R. Achanta and S. Süsstrunk, "Saliency detection using maximum symmetric surround," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 2653–2656.

[17] M. Cheng, N. J. Mitra, X. Huang, P. H. S. Torr, and S. Hu, "Salient Object Detection and Segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2014 (in press).

[18] F. Perazzi, P. Krahenbuhl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 733–740.

[19] X. Hou and L. Zhang, "Saliency Detection: A Spectral Residual Approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2007, pp. 1–8.

[20] C. Guo, Q. Ma, and L. Zhang, "Spatio-temporal Saliency detection using phase spectrum of quaternion fourier transform," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2008, pp. 1–8.

[21] Y. Zhai and M. Shah, "Visual attention detection in video sequences using spatiotemporal cues," in *Proc. ACM Int. Conf. Multimedia*, Oct. 2006, pp. 815–824.

[22] V. Gopalakrishnan and D. Rajan, "Salient region detection by modeling distributions of color and orientation," *IEEE Trans. Multimedia*, vol. 11, no. 5, pp. 892–905, Aug. 2009.

[23] V. Gopalakrishnan, Y. Hu, and D. Rajan, "Random walks on graphs for salient object detection in images." *IEEE Trans. Image Process.*, vol. 19, no. 12, pp. 3232–3242, Dec. 2010.

[24] Y. Wei, F. Wen, W. Zhu, and J. Sun, "Geodesic Saliency Using Background Priors," in *Proc. European Conf. Comput. Vis.*, Oct. 2012, pp. 29–42.

[25] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency Detection via Graph-Based Manifold Ranking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 3166–3173.

[26] B. Jiang, L. Zhang, H. Lu, C. Yang, and M.-H. Yang, "Saliency Detection via Absorbing Markov Chain," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1665–1672.

[27] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li, "Salient Object Detection: A Discriminative Regional Feature Integration Approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 2083–2090.

[28] R. Margolin, A. Tal, and L. Zelnik-Manor, "What Makes a Patch Distinct?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 1139–1146.

[29] D. A. Klein and S. Frintrop, "Salient Pattern Detection Using W2 on Multivariate Normal Distributions," in *Proc. DAGM-OAGM Conf.*, Aug. 2012, pp. 246–255.

[30] S. Yantis, "Goal-directed and stimulus-driven determinants of attentional control," in *Attention and Performance*. Cambridge, MA: MIT Press, 2000, vol. 18, pp. 73–103.

[31] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2007, pp. 545–552.

[32] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-Aware Saliency Detection." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 10, pp. 1915–1926, Dec. 2011.

[33] H. J. Seo and P. Milanfar, "Nonparametric bottom-up saliency detection by self-resemblance," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. Workshops*, Jun. 2009, pp. 45–52.

[34] Y. Li, Y. Zhou, J. Yan, Z. Niu, and J. Yang, "Visual saliency based on conditional entropy," in *Proc. Asian Conf. Comput. Vis.*, Sep. 2010, pp. 246–257.

[35] X. Hou, J. Harel, and C. Koch, "Image Signature: Highlighting sparse salient regions." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 1, pp. 194–201, Jan. 2012.

[36] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *Int. J. Comput. Vis.*, vol. 59, no. 2, pp. 167–181, Sep. 2004.

[37] M. Park, M. Kumar, and A. C. Loui, "Saliency detection using region-based incremental center-surround distance," in *Proc. IEEE Int. Symp. Multimedia*, Dec. 2011, pp. 249–256.

[38] Z. Ren, Y. Hu, L.-T. Chia, and D. Rajan, "Improved saliency detection based on superpixel clustering and saliency propagation," in *Proc. ACM Int. Conf. Multimedia*, Oct. 2010, pp. 1099–1102.

[39] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.

[40] O. Michailovich, Y. Rathi, and A. Tannenbaum, "Image segmentation using active contours driven by the bhattacharyya gradient flow," *IEEE Trans. Image Process.*, vol. 16, no. 11, pp. 2787–2801, Nov. 2007.

[41] D. C. Dowson and B. V. Landau, "The Fréchet distance between multivariate normal distributions," *J. Multivar. Anal.*, vol. 12, no. 3, pp. 450–455, Sep. 1982.

[42] C. R. Givens and R. M. Shortt, "A class of wasserstein metrics for probability distributions," *Michigan Math. J.*, vol. 31, pp. 231–240, 1984.

[43] O. K. Smith, "Eigenvalues of a symmetric 3×3 matrix," *Commun. ACM*, vol. 4, p. 168, Apr. 1961.

[44] H. Li and K. N. Ngan, "A co-saliency model of image pairs." *IEEE Trans. Image Process.*, vol. 20, no. 12, pp. 3365–3375, Dec. 2011.

[45] B. J. Frey and D. Dueck, "Clustering by passing messages between data points." *Science*, vol. 315, no. 5814, pp. 972–976, Feb. 2007.

[46] M. Cheng, J. Warrell, W. Lin, S. Zheng, V. Vineet, and N. Crook, "Efficient Salient Region Detection with Soft Image Abstraction," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1529–1536.

[47] Y. Xie, H. Lu, and M.-H. Yang, "Bayesian saliency via low and mid level cues." *IEEE Trans. Image Process.*, vol. 22, no. 5, pp. 1689–1698, May 2013.

[48] T. Geweniger, D. Zühlke, B. Hammer, and T. Villmann, "Fuzzy variant of affinity propagation in comparison to median fuzzy c-means," in *Proc. Int. Workshop on Adv. Self-Organizing Maps*, Jun. 2009, pp. 72–79.

[49] A. Borji and L. Itti, "Exploiting local and global patch rarities for saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 478–485.

[50] X. Li, Y. Li, C. Shen, A. Dick, and A. V. D. Hengel, "Contextual Hypergraph Modeling for Salient Object Detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 3328–3335.

[51] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical Saliency Detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 1155–1162.

[52] J. Zhang and S. Sclaroff, "Saliency Detection: A Boolean Map Approach," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 153–160.

[53] C. Scharfenberger, A. Wong, K. Fergani, J. S. Zelek, and D. A. Clausi, "Statistical Textural Distinctiveness for Salient Region Detection in Natural Images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 979–986.

[54] V. Movahedi and J. H. Elder, "Design and perceptual validation of performance measures for salient object segmentation," in *Workshop on Percept. Organ. Compute. Vis.*, Jun. 2010, pp. 49–56.

[55] D. R. 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," in *Proc. IEEE Int. Conf. Comput. Vis.*, Jul. 2001, pp. 416–423.
[56] S. Alpert, M. Galun, A. Brandt, and R. Basri, "Image segmentation by probabilistic bottom-up aggregation and cue integration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 2, pp. 315–327, Feb. 2012.
[57] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309–314, Aug. 2004.
[58] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1124–1137, Sep. 2004.

Lei Zhu received his B.S. and M.S. degrees from the Department of Electronics and Information Engineering, Huazhong University of Science and Technology (HUST) in 2004 and 2007, respectively. He is currently a Ph.D. candidate of Pattern Recognition and Intelligent System in HUST. This work was finished when he stayed as a visiting researcher in the Intelligent Vision Systems Group, at the department of Computer Science, University of Bonn in 2012. His research interests include image processing and pattern recognition.

Dominik A. Klein holds a master degree in computer science (Dipl. Inform.) from the University of Bonn. There, since 2009 he is strengthening the Intelligent Vision Systems group as a researcher and Ph.D. candidate. His main areas of interest include robotics, pattern recognition, object tracking, and biologically inspired vision systems.

Simone Frintrop is a senior researcher at the Computer Science department at the University of Bonn and is currently heading the Cognitive Vision Group. She received a doctoral degree from the University of Bonn in 2005. Her research interests include computational visual attention, cognitive computer vision, and robot vision.

Zhiguo Cao received his PhD degree in pattern recognition and intelligence systems in 2001 from Huazhong University of Science and Technology, where he is currently a professor. His research interests are pattern recognition, image processing, and data fusion.

Armin B. Cremers is Professor of Computer Science at the University of Bonn. He holds a doctoral degree in mathematics from the University of Karlsruhe. His former faculty appointments were in Karlsruhe, Los Angeles and Dortmund. Since 2002 he is Director of the Bonn-Aachen International Center for Information Technology (B-IT) in Bonn. His research areas are software engineering, information systems, and artificial intelligence. Currently, Prof. Cremers also holds a Visiting Professorship at the School of Automation of HUST in Wuhan.