Salient Object Detection and Segmentation

Ming-Ming Cheng, Niloy J. Mitra, Xiaolei Huang, *Member, IEEE,*
Philip H. S. Torr, *Senior Member, IEEE,* and Shi-Min Hu, *Member, IEEE,*

Abstract—Automatic estimation of salient object regions across images, without any prior assumption or knowledge of the contents of the corresponding scenes, enhances many computer vision and computer graphics applications. We introduce a regional contrast based salient object extraction algorithm, which simultaneously evaluates global contrast differences and spatial weighted coherence scores. The proposed algorithm is simple, efficient, naturally multi-scale, and produces full-resolution, high-quality saliency maps. These saliency maps are further used to initialize a novel iterative version of GrabCut for high quality salient object segmentation. We extensively evaluated our algorithm using traditional salient object detection datasets, as well as a more challenging Internet image dataset. Our experimental results demonstrate that our algorithm consistently outperforms existing salient object detection and segmentation methods, yielding higher precision and better recall rates. We also show that our algorithm can be used to efficiently extract salient object masks from Internet images, enabling effective sketch-based image retrieval (SBIR) via simple shape comparisons. Despite such noisy internet images, where the saliency regions are ambiguous, our saliency guided image retrieval achieves a superior retrieval rate compared with state-of-the-art SBIR methods, and additionally provides important target object region information.

Index Terms—Salient object detection, salient object segmentation, visual attention, saliency map, image retrieval.

1 Introduction

We, as humans, are experts at quickly and accurately identifying the most visually noticeable or familiar foreground object in the scene, known as salient objects, and adaptively focusing our attention on such perceived important regions. In contrast, computationally identifying such salient object regions [1], [2], that match the human annotators' behaviour when they have been asked to pick a salient object in an image, is very challenging. Being able to automatically and accurately estimate salient object regions, however, is highly desirable given the immediate ability to characterise the spatial support for feature extraction, isolate the object from potentially confusing background, and preferentially allocate computational resources for subsequent image processing. While essentially solves a segmentation problem, salient object detection models segment only the salient foreground object from the background, rather than partition an image into regions of coherent properties as in general segmentation algorithms [2]. Salient object detection models also differ from eye fixation prediction models which aims at predicting a few fixation points in an image rather than uniformly highlighting the entire salient object region [2]. The value of salient object detection methods lies in their wide applications in many fields: including object-of-interest image segmentation [3]–[5], object recognition [6]–[9], adaptive compression of images [10], content-aware image resizing [11]–[14], and image retrieval [15]–[19].

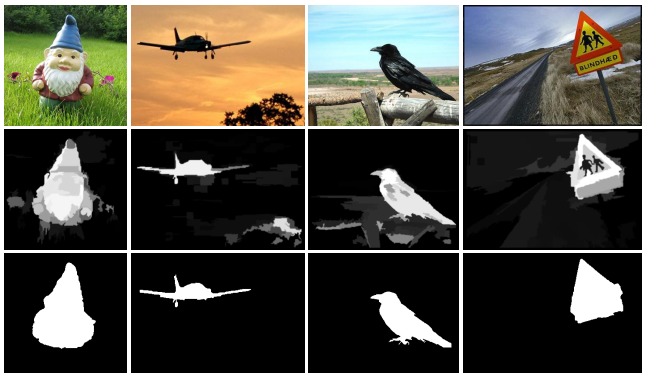


Fig. 1. Given input images (top), a global contrast analysis is used to compute high resolution saliency maps (middle), which can be used to produce masks (bottom) around regions of interest.

Although extraction of salient objects in a scene is related to accurate image segmentation and object retrieval, interestingly reliable saliency estimation is often feasible at the image-level without any actual scene understanding. This is feasible since often saliency, as widely believed, is bottom-up. Such a hypothesis is favored by an evolutionary incentive to parallel process large volumes of low-level image cues, without any computationally expensive global coupling. Thus saliency originates from visual uniqueness, unpredictability, rarity, or surprise, and is often attributed to variations in image attributes like color,

- *M.M. Cheng was with TNList Tsinghua university when doing the majority part of this work. He is currently with Oxford Brookes University. E-mail: cmm.thu@gmail.com*
- *N.J. Mitra is with UCL/KAUST.*
- *X. Huang is with Lehigh University.*
- *P.H.S. Torr is with Oxford Brookes University.*
- *S.M. Hu is with TNList, Tsinghua University.*

gradient, edges, and boundaries. Not surprisingly, visual saliency, being tightly related to our perception and processing of visual stimuli, is investigated across many disciplines including cognitive psychology [20], [21], neurobiology [22], [23], and computer vision [24], [25]. Based on our observed reaction times and estimated signal transmission times along biological pathways, human attention theories hypothesize that the human vision system processes only parts of an image in detail, while leaving others nearly unprocessed. Early work by Treisman and Gelade [26], Koch and Ullman [27], and subsequent attention theories proposed by Itti, Wolfe and others, suggest two stages of visual attention: (i) a fast, pre-attentive, bottom-up, data driven saliency extraction; and (ii) a slower, task dependent, top-down, goal driven saliency extraction.

We focus on bottom-up data driven salient object detection using image contrast (see Fig. 1)[1], with the supposition that a salient object exists in an image [1]. Motivated by the popular belief that human cortical cells may be *hard wired* to preferentially respond to high contrast stimulus in their receptive fields [42], we propose contrast analysis for extracting high-resolution, full-field saliency maps based on the following considerations:

- A global contrast based method, which separates a large-scale object from its surroundings, is desirable over local contrast based methods producing high saliency values at or near object boundaries. Global considerations enable assignment of comparable saliency values across similar image regions, and can uniformly highlight entire objects.
- Saliency of a region primarily depends on the contrast of the region with respect to its nearby regions, while contrasts to distant regions are less significant (see also [43]).
- In man-made photographs, object are often concentrated towards the inner regions of the images, away from image boundaries (see [40] and references therein).
- Saliency maps should be fast, accurate, have low memory footprints, and easy to generate to allow processing of large image collections, and facilitate efficient image classification and retrieval.

We propose a *histogram-based contrast method* (HC) to measure saliency. HC-maps assign pixel-wise saliency values based simply on color separation from all other image pixels to produce full resolution saliency maps. We use a histogram-based approach for efficient processing, while employing a smoothing procedure to control quantization artifacts.

As an improvement over HC-maps, we incorporate spatial relations to produce *region-based contrast* (RC) maps where we first segment the input image into regions, and then assign saliency values to them. The saliency value of a region is then calculated using a global contrast score, measured by the region's contrast and spatial distances to other regions in the image. Note that this approach better acknowledges the relation between image segmentation and saliency determination.

Segmenting regions of interest in still images is of great practical importance in many computer vision and computer graphics applications. Researchers have devoted significant efforts to minimize user interaction during this process. GrabCut [44], which iteratively optimizes the energy function and considers both texture and edge information, has successfully simplified the user interaction to simply dragging a rectangle around the desired object. We propose an improved iterative version of GrabCut and combine it with our saliency detection method to achieve superior performance compared to state-of-the-art unsupervised salient object extraction methods.

We build a database with 10,000 pixel-accurate human-labeled ground truth images (see also Sec. 6.1.1), which is an order of magnitude bigger than previous largest public available dataset of its kind [25]. We have extensively evaluated our methods on this dataset, and compared our methods with 15 state-of-the-art saliency methods as well as with manually created ground truth annotations[2]. The experiments show significant improvements over previous methods both in terms of precision and recall rates. Overall, compared with HC-maps, RC-maps produce better precision and recall rates, but at the cost of increased computational overheard. In our extensive empirical evaluations, we observe that the saliency cuts extracted using our saliency maps are, in most cases, comparable to the manually annotated ground truths. We also demonstrate applications of the extracted saliency maps to segmentation and sketch-based image retrieval.

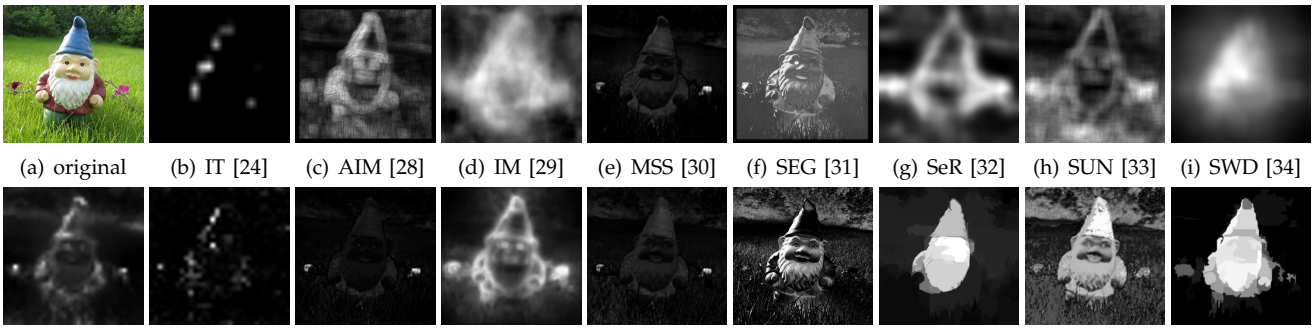
2 Related Work

Our work belongs to the active research field of visual attention modeling, for which a comprehensive discussion is beyond the scope of this paper. We refer readers to recent survey papers for a detailed discussion of 65 models [45], as well as quantitative analysis of different methods in the two major research directions: human fixation prediction [46], [47] and saliency object detection [2].

We focus on relevant literature targeting pre-attentive bottom-up saliency region detection, which are biologically motivated, or purely computational, or involve both aspects. Such methods utilize low-level processing to determine the contrast of image regions to their surroundings, and use feature attributes

1. A preliminary version of this work appeared at CVPR [41].

2. Results for 10,000 images and prototype software are available at the project webpage: http://cg.cs.tsinghua.edu.cn/people/~cmm/saliency2/.



(a) original (b) IT [24] (c) AIM [28] (d) IM [29] (e) MSS [30] (f) SEG [31] (g) SeR [32] (h) SUN [33] (i) SWD [34]

(j) GB [35] (k) SR [36] (l) AC [37] (m) CA [38] (n) FT [25] (o) LC [39] (p) CB [40] (q) HC (r) RC

Fig. 2. Saliency maps computed by different state-of-the-art methods (b-p), and with our proposed HC (q) and RC methods (r). Most results highlight edges, or are of low resolution. See also Fig. 9 (and our project webpage).

such as intensity, color, and edges [25]. We broadly classify the algorithms into local and global schemes. Note that the classification is not strict as some of the research efforts can be listed under both categories.

Local contrast based methods investigate the rarity of image regions with respect to (small) local neighborhoods. Based on the highly influential biologically inspired *early representation* model introduced by Koch and Ullman [27], Itti et al. [24] define image saliency using central-surrounded differences across multi-scale image features. Ma and Zhang [48] propose an alternate local contrast analysis for generating saliency maps, which is then extended using a fuzzy growth model. Harel et al. [49] propose a bottom-up visual saliency model to normalize the feature maps of Itti et al. to highlight conspicuous parts and permit combination with other importance maps. The model is simple, biologically plausible, and easy to parallelize. Liu et al. [1] find multi-scale contrast by linearly combining contrast in a Gaussian image pyramid. More recently, Goferman et al. [38] simultaneously model local low-level clues, global considerations, visual organization rules, and high-level features to highlight salient objects along with their contexts. Such methods using local contrast tend to produce higher saliency values near edges instead of uniformly highlighting salient objects (see Fig. 2). Note that Reinagel et al. [43] observe that humans tend to focus attention in image regions with high spatial contrast and local variance in pixel correlation.

Global contrast based methods evaluate saliency of an image region using its contrast with respect to the entire image. Zhai and Shah [39] define pixel-level saliency based on a pixel's contrast to all other pixels. However, for efficiency they use only luminance information, thus ignoring distinctiveness clues in other channels. Achanta et al. [25] propose a frequency tuned method that directly defines pixel saliency using a pixel's color difference from the average image color. The elegant approach, however, only considers first order average color, which can be insufficient to analyze complex variations common in natural images. In Figures 9 and 10, we show qualitative and quantitative weaknesses of such approaches. Furthermore, these methods ignore spatial relationships across image parts, which can be critical for reliable and coherent saliency detection (see Sec. 6).

Saliency maps are widely employed for unsupervised object segmentation: Ma and Zhang [48] find rectangular salient regions by fuzzy region growing on their saliency maps. Ko and Nam [4] select salient regions using a support vector machine trained on image segment features, and then cluster these regions to extract salient objects. Han et al. [3] model color, texture, and edge features in a Markov random field framework to grow salient object regions from seed values in the saliency maps. More recently, Achanta et al. [25] average saliency values within image segments produced by mean-shift segmentation, and then find salient objects by identifying image segments that have average saliency above a threshold that is set to be twice the mean saliency value of the entire image. We propose a different approach that extends GrabCut [44] method and automatically initialize it using our saliency detection methods. Experiments on our $10,000$ images dataset (see Sec. 6.1.1) demonstrate the significant advantages of our method compared to other state-of-the-art methods.

Subsequent to our preliminary results [41], Jiang et al. [40] propose a comparable method also making use of region level contrast to model image saliency. In the segmentation step, their method also expands and shrinks the initial trimap and iteratively applies graphcut and histogram appearance model. Since GrabCut is an iterative process of using graphcut and GMM appearance mode, the two segmentation methods share a strong similarity. Compared to the CB method [40], experimental results show that our RC salient object region detection and segmentation is more accurate (Fig. 10(a)(c)), $20\times$ faster (Fig. 7), and more robust to center-bias (Fig. 10(b)).

3 Histogram Based Contrast

Our biological vision system is highly sensitive to contrast in visual signal. Based on this observation,

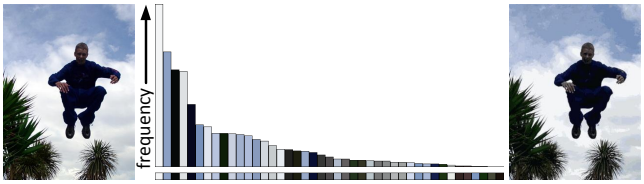


Fig. 3. Given an input image (left), we compute its color histogram (middle). Corresponding histogram bin colors are shown in the lower bar. The quantized image (right) uses only 43 histogram bin colors and still retains sufficient visual quality for saliency detection.

we propose a histogram-based contrast (HC) method to define saliency values for image pixels using color statistics of the input image. Specifically, the saliency of a pixel is defined using its color contrast to all other pixels in the image, i.e., the saliency value of a pixel I_k in image I is defined as,

$$S(I_k) = \sum_{\forall I_i \in I} D(I_k, I_i), \quad (1)$$

where $D(I_k, I_i)$ is the color distance metric between pixels I_k and I_i in the $L^*a^*b^*$space for perceptual accuracy. Equ. (1) can be expanded by pixel order to have the following form,

$$S(I_k) = D(I_k, I_1) + D(I_k, I_2) + \cdots + D(I_k, I_N), \quad (2)$$

where N is the number of pixels in image I. It is easy to see that pixels with the same color value have the same saliency value under this definition, since the measure is oblivious to spatial relations. Hence, rearranging Equ. (2) such that the terms with the same color value c_j are grouped together, we get saliency value for each color as,

$$S(I_k) = S(c_l) = \sum_{j=1}^{n} f_j D(c_l, c_j), \quad (3)$$

where c_l is the color value of pixel I_k, n is the number of distinct pixel colors, and f_j is the probability of pixel color c_j in image I. Note that in order to prevent salient region color statistics from being corrupted by similar colors from other regions, one can develop a similar scheme using varying window masks. However, given the strict efficiency requirement, we take the simple global approach.

3.1 Histogram based speed up

Naively evaluating the saliency value for each image pixel using Equ. (1) takes $O(N^2)$ time, which is computationally too expensive even for medium sized images. The equivalent representation in Equ. (3), however, takes $O(N) + O(n^2)$ time, implying that computational efficiency can be improved to $O(N)$ if $O(n^2) \leq O(N)$. Thus, the key to speed up is to reduce the number of pixel colors in the image. However, the true-color space contains 256^3 possible colors, which is typically larger than the number of image pixels.

Zhai and Shah [39] reduce the number of colors, n, by only using luminance. In this way, $n^2 = 256^2$ (typically $256^2 \ll N$). The method, however, ignores distinctiveness of color information. In this work, we use the full color space instead of luminance only. To reduce the number of colors needed to consider, we first quantize each color channel to have 12 different values, which reduces the number of colors to $12^3 = 1728$. Considering that color in a natural image typically covers only a small portion of the full color space, we further reduce the number of colors by ignoring less frequently occurring colors. By choosing more frequently occurring colors and ensuring these colors cover the colors of more than 95% of the image pixels, we typically are left with around $n = 85$ colors (see Sec. 6 for experimental details). The colors of the remaining pixels, which comprise fewer than 5% of the image pixels, are replaced by the closest colors in the histogram. A typical example of such quantization is shown in Fig. 3. Note that due to efficiency considerations we select the simple histogram based quantization instead of optimizing for an image specific color palette.

3.2 Color space smoothing

Although we can efficiently compute color contrast by building a compact color histogram using color quantization and choosing more frequent colors, the quantization itself may introduce artifacts. Some similar colors may be quantized to different values. In order to reduce noisy saliency results caused by such randomness, we use a smoothing procedure to refine the saliency value for each color. We replace the saliency value of each color by the weighted average of the saliency values of similar colors (measured by $L^*a^*b^*$distance). This is actually a smoothing process in the color feature space. We choose $m = n/4$ nearest colors to refine the saliency value of color c by,

$$S'(c) = \frac{1}{(m-1)T} \sum_{i=1}^{m} (T - D(c, c_i)) S(c_i) \quad (4)$$

where $T = \sum_{i=1}^{m} D(c, c_i)$ is the sum of distances between color c and its m nearest neighbors c_i, and the normalization factor comes from $\sum_{i=1}^{m}(T - D(c, c_i)) = (m-1)T$. Note that we use a linearly-varying smoothing weight $(T - D(c, c_i))$ to assign larger weights to colors closer to c in the color feature space. In our experiments, we found that such linearly-varying weights are better than Gaussian weights, which fall off too sharply. Fig. 4 shows the typical effect of color space smoothing with the corresponding histograms sorted by decreasing saliency values. Note that similar histogram bins are closer to each other after such smoothing, indicating that similar colors have higher likelihood of being assigned similar saliency values, thus reducing quantization artifacts (see Fig. 10).

3.3 Implementation details

To quantize the color space into 12^3 different colors, we uniformly divide each color channel into

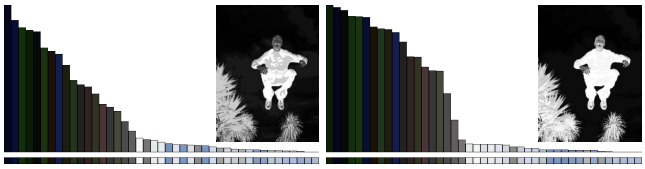


Fig. 4. Saliency of each color, normalized to the range $[0, 1]$, before (left) and after (right) color space smoothing. Corresponding saliency maps are shown in the respective insets.

12 different levels. While the quantization of colors is performed in the RGB color space, we measure color differences in the $L^*a^*b^*$color space given its perceptual accuracy. We do not, however, perform quantization directly in the $L^*a^*b^*$color space since not all colors in the range $L^* \in [0, 100]$, and $a^*, b^* \in [-127, 127]$ necessarily correspond to real colors. Experimentally we observed worse quantization artifacts using direct $L^*a^*b^*$color space quantization. Best results were obtained by quantization in the RGB space while measuring distance in the $L^*a^*b^*$color space, as opposed to performing both quantization and distance calculation in a single color space, either RGB or $L^*a^*b^*$.

4 Region Based Contrast

Humans pay more attention to image regions with high contrast to their surroundings [50]. Besides contrast, spatial relationships are important in human attention. High contrast to ones surrounding regions is usually stronger evidence for saliency of a region than comparable contrast to far-away regions. Since directly introducing spatial relationships when computing pixel-level contrast is computationally expensive, we introduce a contrast analysis method, *region contrast* (RC), so as to integrate spatial relationships into region-level contrast computation. In RC, we first segment the input image into regions, then compute color contrast at the region level, and finally define saliency for each region as the weighted sum of the region's contrasts to all other regions in the image. The weights are set according to the spatial distances with farther regions being assigned smaller weights.

4.1 Region contrast by histogram comparison

We first segment the input image into regions using a graph-based image segmentation method [51]. Then we build the color histogram for each region as in Sec. 3. For a region r_k, we compute its saliency value by measuring its color contrast to all other regions in the image,

$$S(r_k) = \sum_{r_k \neq r_i} w(r_i) D_r(r_k, r_i), \quad (5)$$

where $w(r_i)$ is the weight of region r_i and $D_r(\cdot, \cdot)$ is the color distance metric between the two regions. We weight the distances by the number of pixels in r_i as $w(r_i)$ to emphasize color contrast to bigger regions. The color distance between two regions r_1 and r_2 is,

$$D_r(r_1, r_2) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} f(c_{1,i}) f(c_{2,j}) D(c_{1,i}, c_{2,j}) \quad (6)$$

where $f(c_{k,i})$ is the probability of the i-th color $c_{k,i}$ among all n_k colors in the k-th region r_k, $k = \{1, 2\}$. Note that we use the probability of a color in the probability density function (i.e., normalized color histogram) of the region as the weight for this color to further emphasize the color differences between dominant colors.

Storing and calculating the regular matrix format histogram for each region is inefficient since each region typically contains a small number of colors in the color histogram of the whole image. Instead, we use a sparse histogram representation for efficient storage and computation.

4.2 Spatially weighted region contrast

We further incorporate spatial information by introducing a spatial weighting term in Equ. (5) to increase the effects of closer regions and decrease the effects of farther regions. Specifically, for any region r_k, the spatially weighted region contrast based saliency is:

$$S(r_k) = w_s(r_k) \sum_{r_k \neq r_i} e^{\frac{D_s(r_k, r_i)}{-\sigma_s^2}} w(r_i) D_r(r_k, r_i) \quad (7)$$

where $D_s(r_k, r_i)$ is the spatial distance between regions r_k and r_i, σ_s controls the strength of spatial distance weighting, $w(r_i)$ is the weight of region r_i defined by the number of pixels in r_i, and $w_s(r_k)$ is a spatial prior weighting term similar to center bias (CB [40]). We use $w_s(r_k) = \exp(-9d_k^2)$, where d_k is the average distance between pixels in region r_k and the center of the image, with pixel coordinates normalized to $[0, 1]$. Thus, $w_s(r_k)$ gives a high value if region r_k is close to the center of the image and it gives a low value if the region is a border region away from the center. For σ_s, larger values of σ_s reduce the effect of spatial weighting so that contrast to farther regions would contribute more to the saliency of the current region. The spatial distance between two regions is defined as the Euclidean distance between their centroids. In our implementation, we use $\sigma_s^2 = 0.4$ with pixel coordinates normalized to the range $[0, 1]$.

4.3 Further improvement of RC saliency maps

We further refine our RC saliency maps in two steps. First, we use the spatial prior to explicitly estimate the non-salient (background) region. Second, we apply the color space smoothing as described in Sec. 3.2.

We observe that regions with long borders overlapping with image borders are typically non-salient background regions, which we call border regions. We incorporate them as another spatial prior ($w_s(\cdot)$ in

(a) (b) (c) (d) (e) (f) (g)

Fig. 5. Region based contrast computation: (a) input image, (b) image regions generated by Felzenszwalb and Huttenlocher's segmentation method [51], (c) region contrast without distance weighting and spatial prior (Equ. (5)), (d) region contrast with distance weighting, (e) region contrast further considering spatial prior (Equ. (7)), (f) region contrast after improvement by border region estimation and color space smoothing, (g) using our saliencyCut (Sec. 5), we get a high quality cut that is comparable to human labeled ground truth.

Equ. (7)) to detect non-salient regions. In our implementation, we normalize the number of pixels located in the 15 pixel-wide image-border area by the region size, and consider regions with this value higher than a threshold to be border regions. In practice, this hard constraint improves both the saliency maps as well as the convergence speed of SaliencyCut (Sec. 5) by improving the initial condition. Our border region estimation aims at high precision, rather than high recall. A strict fixed threshold, which on average corresponds to 2% miss alarm rate in our dataset, is chosen to detect border regions.

In order to uniformly highlight the entire saliency region of the image, we get the average saliency of each color in the color histogram and adopt the color space smoothing (Sec. 3.2) to improve our RC saliency map. After smoothing, some border region pixels may get non-zero saliency values. We reset the saliency of border region to zero and re-estimate the saliency of each region as the average saliency value of its corresponding pixels. Since initial RC maps are typically more uniformly highlighted compared to HC saliency maps without color space smoothing, we typically choose smaller number of nearest colors ($m = n/10$ in this part). Fig. 5(f) demonstrates such an example. The jumping man region is more uniformly highlighted compared to Fig. 5(e).

5 SaliencyCut: Automatic Salient Region Extraction

In a highly influential work, GrabCut [44] made critical changes to the graphcut formulation to allow processing of noisy initialization. This enabled users to roughly annotate (e.g., using a rectangle) a region of interest, and then use GrabCut to extract a precise image mask. Using our estimated saliency masks, we remove even the need for user annotated rectangular regions. In this section, we introduce *SaliencyCut*, which uses the computed saliency map to assist in automatic salient object segmentation. This immediately enables automatic analysis of large internet image repositories. Specifically, we make two enhancements to GrabCut [44]: "iterative refine" and "adaptive fitting", which together handle considerably more noisy initializations. Thanks to the robustness of the new approach, we are able to automatically initialize the segmentation according to the detected saliency map.

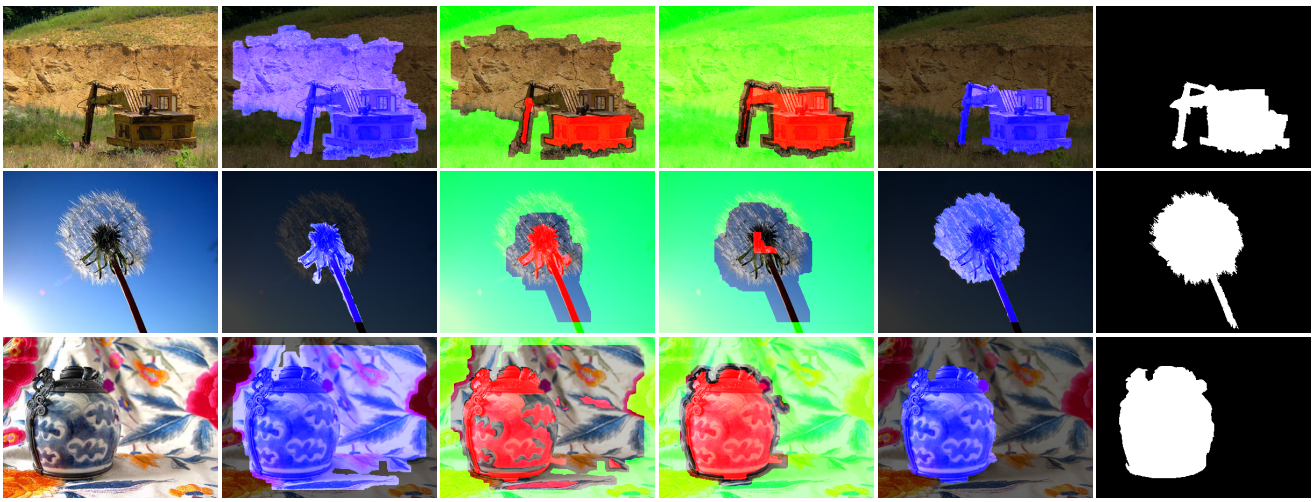
5.1 Algorithm initialization

Instead of manually selecting a rectangular region to initialize the process, as in classical GrabCut, we automatically initialize using a segmentation obtained by binarizing the saliency map using a fixed threshold. Similar to GrabCut, we use incomplete trimaps for the initialization. Regions with saliency value below a certain threshold are labeled as background regions. Other regions correspond to the unknown part of the trimap. Note that we do not initialize any hard foreground labeling. These unknown regions are initially used to train foreground color models thus helps the algorithm to identify the foreground pixels.

Since the initial background regions are retained while other regions may be changed during the GrabCut optimization, we give preference to confident background labels in the trimaps. Thus we initialize the GrabCut algorithm using threshold given high recall of potential foreground region and let the iterative optimization process to increase its precision. In our experiments, the threshold is chosen empirically to be the threshold that gives 95% recall rate in our fixed thresholding experiments (see Sec. 6.2). When initialized using RC saliency maps, this threshold is 70 with saliency values normalized to $[0, 255]$.

5.2 Segmentation by iterative fitting

Once initialized, we iteratively run GrabCut [44] to improve the saliency cut result (maximum of 4 iterations in our experiments). After each iteration, we use dilation and erosion operations on the current segmentation result to get a new trimap for the next GrabCut iteration. As shown in Fig. 6(c, d), the region outside the dilated region is set to background, the region inside the eroded region is set to foreground, and the remaining areas are set to unknown in the



(a) (b) (c) (d) (e) (f)

Fig. 6. Demonstration of SaliencyCut: (a) original image, (b) initial segmentation by fixed thresholding the saliency map, (c) trimap after first iteration, (d) trimap after second iteration, (e) final segmentation, and (f) manually labeled ground truth. In the segmented images (e), blue is foreground, gray is background, while in the trimaps (b–d), foreground is red, background is green, and unknown regions are left unchanged.

trimap. GrabCut, which by itself is an iterative process using Gaussian mixture models and graph-cut [52], helps to refine salient object regions at each step.

Different from one-pass GrabCut or the even simpler graph cut based saliency segmentation [53], the new scheme in SaliencyCut *iteratively refines* the initial salient regions. Such an iterative design is important to handle noisy initializations supplied by the saliency detection algorithm rather than human annotations. In case of incorrect initialization as shown in flower example in Fig. 6 (b), the initial background region incorrectly contains foreground object(s). Although we can still get a segmentation result containing many parts of the flower using GrabCut, the remaining flowers in the initial background region would never be correctly extracted using GrabCut since the background gets a hard labeling. One may consider relaxing the hard constrain of GrabCut to solve this problem. However, experimental results show this would make the method not stable, often producing results containing all foreground or all background.

We iteratively refine the initial segmentation and adaptively change the initial condition to fit with newly segmented salient region. The *adaptive fitting* is based on an important observation: regions closer to an initial salient object region are more likely to be part of that salient object than far-away regions. Thus, our new initialization enables GrabCut to include nearby salient regions, and exclude non-salient regions according to color feature dissimilarity. After each GrabCut iteration, SaliencyCut incorporates the constraints given by the newly obtained trimap, and train a better appearance model according to previous results.

Fig. 6 shows three examples. In the flower example (second row), SaliencyCut successfully expanded the initial salient regions (obtained directly from the saliency map) and converged to an accurate segmentation result. In the excavator and teapot examples, unwanted regions are correctly excluded during GrabCut iterations. The intermediate steps show how SaliencyCut successfully extracted the object regions of interest in these challenging examples. A comprehensive quantitative evaluation of different saliency segmentation methods is presented in Sec. 6.3.

6 Experimental Comparisons

In this work, we extensively evaluated our saliency detection method on three different types of benchmark datasets, and compared it against 15 alternate methods — SR [36], IT [24], IM [29], SUN [33], AC [37], SeR [32], AIM [28], GB [35], LC [39], CA [38], FT [25], SWD [34], SEG [31], MSS [30], LP [54] and CB [40], respectively. Following [25], we selected these methods according to: number of citations (IT, SR, SUN, AIM and FT), recency (SeR, MSS, SEG, IM, CA

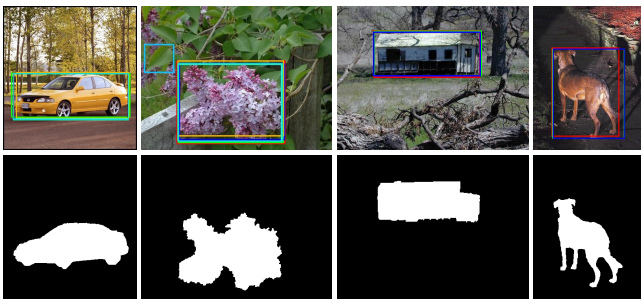


Fig. 8. Ground truth examples: (first row) original images with ground truth rectangles from [1], (second row) our ground truth, which have more precisely marked important regions at pixel level accuracy.

Method	IT [24]	AIM [28]	IM [29]	MSS [30]	SEG [31]	SeR [32]	SUN [33]	SWD [34]	CB [40]
Time (s)	0.246	4.288	0.991	0.106	4.921	1.019	1.116	0.100	5.568
Code Type	Matlab	Matlab	Matlab	C++	Matlab	Matlab	Matlab	Matlab	Matlab & C
Method	GB [35]	SR [36]	FT [25]	AC [37]	CA [38]	LC [39]	HC	RC	
Time (s)	1.614	0.064	0.102	0.109	53.1	0.018	0.019	0.254	
Code Type	Matlab	Matlab	C++	Matlab	Matlab	C++	C++	C++	

Fig. 7. Average time taken to compute a saliency map for images in the THUS10000 database (most have resolution 400×300). We use parallel computing environment for all Matlab functions for efficient computation.

and SWD), variety (IT is biologically-motivated, LC is purely computational, GB and LP are hybrid, SR works in the frequency domain, AC and FT output full resolution saliency maps), and being related to our approach (LC and CB).

The effectiveness of a saliency detection method depends on the applications. We evaluated our method on several core computer vision and graphics applications, including: salient region segmentation by fixed thresholding, object of interest image segmentation, and sketch based image retrieval.

Fig. 7 compares the average time taken by each method on a Dual Core 2.6 GHz machine with 2GB RAM. Our algorithms, HC and RC, are implemented in C++. For the other methods namely IT, AIM, IM, MSS, SEG, SeR, SUN, GB, SR, AC, CA, FT and CB, we used the authors' implementations, while for LC, we implemented the algorithm in C++ since we failed to obtain the authors' implementation. For typical natural images, our HC method runs in $O(N)$, which is sufficient for real-time applications. In contrast, our RC variant is slower as it requires image segmentation [51], but produces superior quality saliency maps.

In order to comprehensively evaluate the accuracy of our methods for salient object segmentation, we performed two experiments using different objective comparison measures. In the first experiment, to segment salient objects and calculate precision and recall curves, we binarized the saliency map using every possible fixed threshold (similar to [25]). In the second experiment, we segment salient objects by our SaliencyCut approach (Sec. 5).

6.1 Benchmark datasets for saliency detection

6.1.1 Images with unambiguous salient object

Similar to existing salient object region detection methods [1], [25], [30], [40], we first evaluate our methods on images with unambiguous salient object. The largest dataset of this kind is provided by Liu et al. [1]. This dataset contains 20,000+ images (mostly at 400×300 resolution), with bounding box labeling by 3-9 users. Since objects can still be recognized at low resolution, the dataset has limited scale and location variations of salient objects, i.e., implicitly the images have scale and location priors (Flickr like).

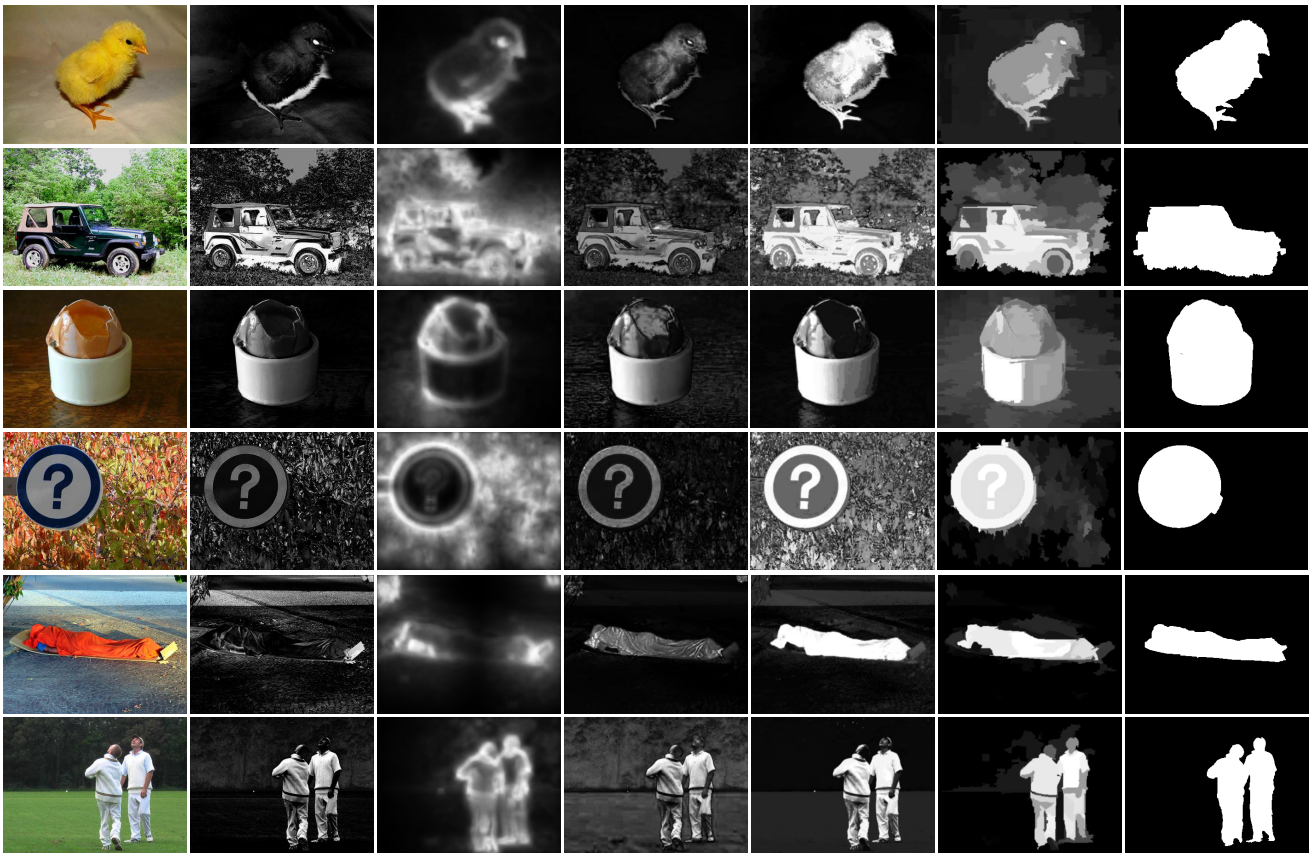
Although an invaluable recourse to evaluate saliency detection algorithms, the database with the marked bounding boxes, however, is often too coarse for fine grained evaluation as observed by Wang and Li [55], and Achanta et al. [25]. In order to do more extensive and accurate evaluation, we randomly selected 10,000 images with consistent bounding box labeling in database provided by Liu et al. [1] and the consistent measure is the same as choosing image dataset B in their paper. As shown in Fig. 8, we accurately marked pixels in salient object regions. We call this dataset THUS10000 because it contains 10,000 images with *pixel-level* saliency labeling (publicly available on our project page). Our dataset is 10 times bigger than what was previously the largest public available database of its kind [25] with pixel-level salient region marking. In our experiments, we find that saliency detection methods using pixel level contrast (FT, HC, LC, MSS) don't scale well on this lager benchmark (see Fig. 10(a)), suggesting the importance of region level analysis.

6.1.2 Non-selected internet images

While state-of-the-art methods consistently produce excellent results when evaluated using the traditional benchmark dataset [25] (see Fig. 10(c)), ordinary users often report less satisfactory experiences when using their own images. This encourage us to think about two questions: 'How would these methods deal with random internet images?' and 'When can we trust the results of these methods?'. To better explore these issues, we evaluated salient object segmentation methods on a dataset with non-selected internet images [56]. This benchmark dataset, namely THUR15000 [56], contains about 3000 images downloaded from Flickr for each of the 5 keywords: "butterfly", "coffee mug", "dog jump", "giraffe", and "plane". Salient regions in THUR15000 images are marked at pixel accuracy. Note that not every image in the THUR15000 dataset contains a salient region label, as some images do not have any salient object region. Besides saliency detection, this dataset can also be used to evaluate the performance of sketch based image retrieval (SBIR).

6.1.3 Human fixation dataset

While our algorithm targets salient region detection, it is also interesting to evaluate its performance on human fixation prediction benchmarks. We use the most widely adopted human fixation benchmark [54] for such evaluation.



(a) original (b) LC (c) CA (d) FT (e) HC-maps (f) RC-maps (g) RCC

Fig. 9. Visual comparison of saliency maps. (a) original images, saliency maps produced using (b) Zhai and Shah [39], (c) Goferman et al. [38], (d) Achanta et al. [25], (e) our HC and (f) RC methods, and (g) RC-based saliency cut results. Our methods generate uniformly highlighted salient regions (see our project webpage for all results on the full benchmark dataset).

6.2 Segmentation by fixed thresholding

The simplest way to get a binary segmentation of salient objects is to threshold the saliency map with a threshold $T_f \in [0, 255]$. To reliably compare how well various saliency detection methods highlight salient regions in images, we vary the threshold T_f from 0 to 255. Fig. 10(a) shows the resulting precision vs. recall curves. Typical qualitative comparison of saliency maps obtained by the various methods are presented in Fig. 2 and Fig. 9.

Unlike most other methods, both the CB method and our RC method use the center location prior of the man-made photographs. However, for a fair comparison, Fig. 10(b) shows comparisons while disabling such a location prior. Specifically, RC1 shows the effect disabling the center location weighting (Equ. (7)) of RC method, while RC2 shows the effect of further disabling border region estimation (Sec. 4.3). Other methods in Fig. 10(b) also improve when we use the same segmentation, as used in RC, to average saliency values within each segment and re-normalize to $[0, 255]$ by uniform scaling. Note that many of these methods aim to predict human eye movements rather than perform salient object segmentation, as is our focus.

The precision and recall curves clearly show that our RC method outperforms the other methods. We observe a significant loss in precision Fig. 10(b) for the CB method (which has best performance in the benchmark paper [2]) indicating that the method heavily relies on location prior. The extremities of the precision vs. recall curve are interesting: At maximum recall where $T_f = 0$, all pixels are retained as positives, i.e., considered to be foreground, so all the methods have the same precision and recall values; precision 0.22 and recall 1.0 at this point indicate that, on average, there are 22% image pixels belonging to the ground truth salient regions. At the other end, the minimum recall values of our RC method are higher than those of the other methods, because the saliency maps computed by our RC method are smoother and contain more pixels with the saliency value 255. Our HC method also has better precision and recall compared to methods with similar computational efficiency (SR, FT, and LC). After comparison of a large number of saliency detection models, Borji et al. [2] proposed a combined model and show that integration of the few best models (with the initial version of our method as one of them) outperforms all

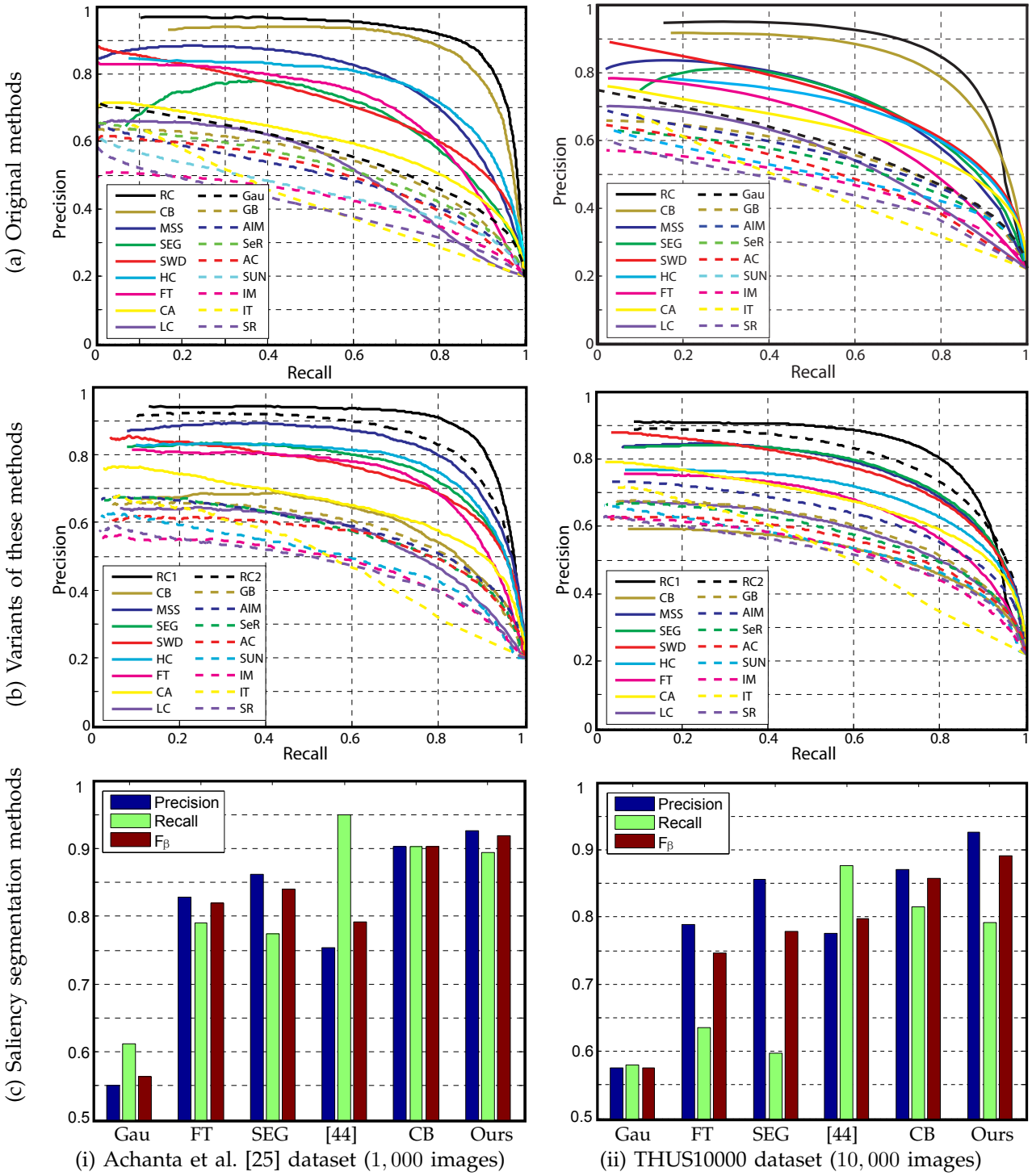


Fig. 10. Statistical comparison results of (a) different saliency region detection methods, (b) their variants, and (c) object of interest region segmentation methods, using largest public available dataset (i) and (ii) our THUS10000 dataset (to be made public available). (Please refer to our project webpage for details.)

models. We believe the the combined model of [2] will naturlly be benefit from performance improvement of our method.

As also observed in the survey papers [2], [45]–[47], center-bias naturally exists in human captured photos. Judd et al. [54] further found that a simple Gaussian blob performs better than many saliency detection methods when evaluated in famous eye fixation dataset. We experimentally find that such simple Gaussian blob, represented by 'Gau' in Fig. 10(a)(c), also performs better than many existing models for saliency region detection task. However, in the absence of explicit information, we prefer not to use such a strong prior that can potentially produce biased results, e.g., in automated imaging systems. When disabling the center bias term, our method still produces

Method	MSS [30]	CA [38]	LP [54]	Ours
ROC Area	0.683	0.844	0.849	0.830

Fig. 11. Comparison on human fixation dataset [54].

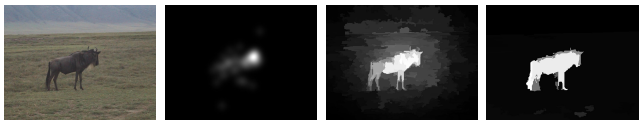


Fig. 12. From left to right, we show source image, ground truth eye fixation map by human observer, our RC result with the term encouraging similar appearance region receive similar saliency (Sec. 4.3) disabled, and result by our full RC method.

better results than other alternatives Fig. 10(b).

In the context of human fixation prediction, the CA [38] and LP [54] methods report the best performance. Although it avoids the heavy learning for combining multi-saliency models and object detectors, the CA [38] method still needs about 1 min to calculate a saliency map even for small images. Fig. 11 and Fig. 7 shows that our method, although initially designed for saliency region detection, has only slightly lower performance to state-of-the-art methods for predicting human fixation points, while being 200+ times more efficient. Readers can refer to [38], [47], [54] for more comparisons. Notice that the good performance of our RC method for predicting eye fixation points shown in Fig. 11 is achieved by disabling the term encouraging similar appearance region receive similar saliency value, thus improves human fixation point prediction as demonstrated in Fig. 12. Although disabling the process explained in Sec. 4.3 improves eye fixation prediction performance, we argue that uniformly highlighting the entire object region is better in many applications, including content aware image resizing [14], non-photorealist rendering [41], adaptive image compression [10], and image mosaic [38]. Thus, although their own method [38] achieves best performance on eye fixation dataset [54], Margolin et al. [57] still choose to integrate our RC saliency maps to achieve better effects for various of image manipulation applications.

6.3 Object of interest image segmentation

To objectively evaluate our new saliency cut method using our RC-map as initialization, we compare our results with results obtained by other state-of-the-art methods for object of interest segmentation, i.e., FT [25], SEG [31], GrabCut [44] (initialized using 5 pixel wide image boundary), and CB [40] (best parameters are selected for these methods). Average precision, recall, and F-Measure are compared against the entire ground-truth database [25], with the F-Measure defined as:

$$F_\beta = \frac{(1+\beta^2) Precision \times Recall}{\beta^2 \times Precision + Recall}. \quad (8)$$

We use $\beta^2 = 0.3$ as in Achanta et al. [25] to weigh precision more than recall. As can be seen from the comparison (see Fig. 10(c)), saliency cut using our RC saliency maps significantly outperforms other methods. As discussed by Liu et al. [1], recall rate is not as important as precision for attention detection. For example, a 100% recall rate can be achieved by simply selecting the whole image. Our approach reduced 57.2%, 50.9%, 46.5%, and 23.7% overall error rates on F-measure, compared with FT [25], SEG [31], GrabCut [44], and CB [40], respectively when evaluated using large accurate dataset (THUS10000). Besides producing higher F-Measure and robustness to location prior, SaliencyCut (demo software available on project webpage) is about 60 times faster (see Fig. 13) compared to CB [40].,

Although producing quite promising results for simple images as evaluated in Fig. 10, evaluation results for non-selected internet images Fig. 14 shows that there is still a need to develop more robust methods. For both datasets, our saliency cut's performance is the best. We believe that such high performance in predicting the entire salient object region can benefit object recognition [7], classification [6], and auto-cropping [57].

Method	FT [25]	SEG [31]	CB [40]	Ours
Time (s)	0.247	7.48	36.5	0.621
Code	Matlab	Matlab & C	Matlab & C	C++

Fig. 13. Comparison of average time taken for different saliency segmentation methods. Segmentation results for THUS10000 benchmark dataset using different methods are shared in our project page.

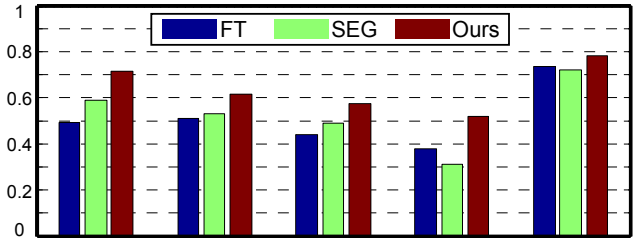


Fig. 14. Comparison of average F_β for different saliency segmentation methods: FT [25], SEG [31], and ours, on THUR15000 dataset [56].

6.4 Sketch based image retrieval

Outline sketches are typically easier and faster for users to generate than a complete color description of the desired image. Sketch based image retrieval (SBIR) techniques become vital for users to leverage the increasing volumes of available image database. A large majority of potential users fail to precisely express fine details in their drawings. Thus most SBIR

Fig. 15. Sketch based image comparison: first row shows images download from Flickr using keyword 'giraffe', second row shows our retrieval results obtained by comparing user input sketch with SaliencyCut result using shape context measure [58]; third row shows corresponding sketch based retrieval results using SHoG [59]. (see our project page for more examples.)

TPR (%)	Butterfly		Coffee mug		Dog jump		Giraffe		Plane		**Average**	
	T50	T100	T50	T100	T50	T100	T50	T100	T50	T100	**T50**	**T100**
Flickr	28	28	58	51	56	55	30	25	44	48	**41.4**	**43.2**
Ours	58	52	88	93	86	90	72	66	88	90	**78.2**	**78.4**
SHoG [59]	36	40	82	78	74	73	18	18	90	91	**60.0**	**60.0**

Fig. 16. True positive ratios (TPR) among top 50 and 100 retrieval results. Results for SHOG are supplied by original authors. An image is considered as true positive if it contains a target object specified by the keywords.

systems, which employ global descriptors, are unsatisfactory as they are unreliable under affine variations. To overcome such drawbacks, Eitz et al. [59], [60] use local descriptors to achieve state-of-the-art retrieval performance. The success of their methods is mainly attributed to translation invariance of local descriptors while using large local feature size (in the order of $20-25\%$ of the image's diagonal) to still retain large scale characteristics. However, for such large window sizes, there is simply not much space left for translating the sketch, thus limiting the translation invariance. SBIR still suffers from relatively low accuracy thus restricting its commercial potential.

Matching object shapes with clean background, however, is a relatively mature field. Even for the very challenging MPEG-7 dataset, state-of-the-art methods can achieve 91.61% retrieval rates [61]. Classical shape methods such as Shape Contexts (SC) [58] and Chamfer Matching [62] are mostly successful when dealing with limited background clutter. Selecting clean object outlines without influence from irrelevant image edges has great potential to improve current SBIR systems. Based on the observation that good results cannot be achieved without selection of segments, Bai et al. [63] use a shape band model to coarsely select candidate of edge segments while using Shape Context distance to decide the optimal matching. However, the shape band model requires user sketch for further detection thus does not allow preprocessing. It needs a few minutes to process a single image making it unsuited for real-time image retrieval applications.

Our SaliencyCut algorithm provides another possibility for automatically finding the outlines of object of interest on large scale image datasets. After such preprocessing, it becomes possible to make use of proven shape matching algorithms. We simply rank the images by SC [58] distance between their salient region outlines and user input sketches and compare with a state-of-the-art SBIR method using SHoG [59].

Experiments indicate that although our SaliencyCut method may produce less optimal results for noisy internet images, the shape matching method is very efficient in selecting those well segmented results. A quantitative evaluation in our THUR15000 dataset is shown in Fig. 16. One can see that our retrieval method is more effective than SHoG in terms of selecting user-desired candidates. Sample qualitative results are shown in Fig. 15. Compared with SHoG, our method gives results that are more relevant to user input sketches. Moreover, our method produces the precise boundary of the desired object, which makes it possible to reuse these segmented image components in many applications, e.g., image composition [15], [16], semantic colonization [17], and information extraction [18]. Note that the extracted salient region features are complementary to other features like color, texture, and local edges.

Such SBIR methods also demonstrate an important technique for robustly integrating saliency detection in real-world applications. Although saliency detection methods cannot grantee robust performance on individual images, their efficiency and simplicity makes it possible to automatically process a large number of images, which can be subsequently filtered for reliability and accuracy [15]–[18].

7 Conclusions and Future Work

We presented global contrast based saliency computation methods, namely Histogram based Contrast (HC) and spatial information-enhanced Region based Contrast (RC). While the HC method is fast and generates

results with fine details, the RC method generates spatially coherent high quality saliency maps at the cost of reduced computational efficiency. We evaluated our methods on the largest publicly available dataset and compared our scheme with many other state-of-the-art methods consistently demonstrating that the proposed schemes is superior both in terms of precision and recall, while still being simple and efficient.

In the future, we plan to investigate efficient algorithms that incorporate spatial relationships in saliency computation while preserving fine details in the resulting saliency maps. Also, it is desirable to develop saliency detection algorithms to handle cluttered and textured background, which could introduce artifacts to our global histogram based approach. Finally, it may be beneficial to incorporate high level factors like human faces, and symmetry into saliency maps. We believe the proposed saliency maps can be further used for efficient object detection [64], reliable image classification, robust image scene analysis [65], leading to improved image retrieval [66], [67].

ACKNOWLEDGMENTS

We would like to thank all the anonymous reviewers for their many useful comments and feedback. This research was supported by the 973 Program (2011CB302205), the 863 Program (2009AA01Z327), the Key Project of S&T (2011ZX01042-001-002), and NSFC (U0735001). Ming-Ming Cheng was funded by Google PhD fellowship, IBM PhD fellowship, and New PhD Researcher Award (Ministry of Edu., CN).

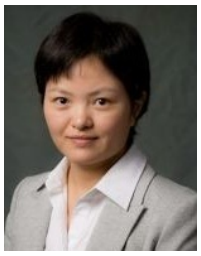
REFERENCES

[1] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, T. X., and S. H.Y., "Learning to detect a salient object," *IEEE TPAMI*, vol. 33, no. 2, pp. 353 –367, 2011.
[2] A. Borji, D. N. Sihite, and L. Itti, "Salient object detection: A benchmark," in *ECCV*, 2012.
[3] J. Han, K. Ngan, M. Li, and H. Zhang, "Unsupervised extraction of visual attention objects in color images," *IEEE TCSV*, vol. 16, no. 1, pp. 141–145, 2006.
[4] B. Ko and J. Nam, "Object-of-interest image segmentation based on human attention and semantic region clustering," *J. Opt. Soc. Am.*, vol. 23, no. 10, pp. 2462–2470, 2006.
[5] M. Donoser, M. Urschler, M. Hirzer, and H. Bischof, "Saliency driven total variation segmentation," in *IEEE ICCV*, 2009, pp. 817–824.
[6] U. Rutishauser, D. Walther, C. Koch, and P. Perona, "Is bottom-up attention useful for object recognition?" in *IEEE CVPR*, 2004, pp. 37–44.
[7] F. Fraundorfer and H. Bischof, "Detecting distinguished regions by saliency," *Image Analysis*, pp. 665–673, 2003.
[8] J. Carreira and C. Sminchisescu, "CPMC: Automatic object segmentation using constrained parametric min-cuts," vol. 34, no. 7, pp. 1312–1328, 2012.
[9] B. Alexe, T. Deselaers, and V. Ferrari, "Measuring the objectness of image windows," vol. 34, no. 11, 2012.
[10] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: an overview," *IEEE Trans. Consumer Elec.*, vol. 46, no. 4, pp. 1103–1127, 2002.
[11] S. Avidan and A. Shamir, "Seam carving for content-aware image resizing," *ACM Trans. Graph.*, vol. 26, 2007.
[12] Y.-S. Wang, C.-L. Tai, O. Sorkine, and T.-Y. Lee, "Optimized scale-and-stretch for image resizing," *ACM TOG*, vol. 27, no. 5, pp. 118:1–8, 2008.
[13] M. Rubinstein, A. Shamir, and S. Avidan, "Multi-operator media retargeting," *ACM Trans. Graph.*, vol. 28, no. 3, pp. 23:1–11, 2009.
[14] G.-X. Zhang, M.-M. Cheng, S.-M. Hu, and R. R. Martin, "A shape-preserving approach to image resizing," *Comput. Graph. Forum*, vol. 28, no. 7, pp. 1897–1906, 2009.
[15] T. Chen, M.-M. Cheng, P. Tan, A. Shamir, and S.-M. Hu, "Sketch2photo: Internet image montage," *ACM TOG*, vol. 28, no. 5, pp. 124:1–10, 2009.
[16] H. Huang, L. Zhang, and H.-C. Zhang, "Arcimboldo-like collage using internet images," *ACM TOG*, vol. 30, pp. 155:1–155:8, 2011.
[17] Y. S. Chia, S. Zhuo, R. K. Gupta, Y.-W. Tai, S.-Y. Cho, P. Tan, and S. Lin, "Semantic colorization with internet images," *ACM TOG*, vol. 30, no. 6, pp. 156:1–156:8, 2011.
[18] H. Liu, L. Zhang, and H. Huang, "Web-image driven best views of 3d shapes," *The Visual Computer*, pp. 1–9, 2012.
[19] J. He, J. Feng, X. Liu, T. Cheng, T.-H. Lin, H. Chung, and S.-F. Chang, "Mobile product search with bag of hash bits and boundary reranking," 2012, pp. 3005–3012.
[20] H. Teuber, "Physiological psychology," *Annual Review of Psychology*, vol. 6, no. 1, pp. 267–296, 1955.
[21] J. M. Wolfe and T. S. Horowitz, "What attributes guide the deployment of visual attention and how do they do it?" *Nature Reviews Neuroscience*, pp. 5:1–7, 2004.
[22] R. Desimone and J. Duncan, "Neural mechanisms of selective visual attention," *Annual review of neuroscience*, vol. 18, no. 1, pp. 193–222, 1995.
[23] S. K. Mannan, C. Kennard, and M. Husain, "The role of visual salience in directing eye movements in visual object agnosia," *Current biology*, vol. 19, no. 6, pp. 247–248, 2009.
[24] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE TPAMI*, vol. 20, no. 11, pp. 1254–1259, 1998.
[25] R. Achanta, S. Hemami, F. Estrada, and S. Süsstrunk, "Frequency-tuned salient region detection," in *IEEE CVPR*, 2009, pp. 1597–1604.
[26] A. M. Triesman and G. Gelade, "A feature-integration theory of attention," *Cognitive Psychology*, vol. 12, pp. 97–136, 1980.
[27] C. Koch and S. Ullman, "Shifts in selective visual attention: towards the underlying neural circuitry," *Human Neurbiology*, vol. 4, pp. 219–227, 1985.
[28] N. Bruce and J. Tsotsos, "Saliency, attention, and visual search: An information theoretic approach," *Journal of Vision*, vol. 9, no. 3, pp. 5:1–24, 2009.
[29] N. Murray, M. Vanrell, X. Otazu, and C. A. Parraga, "Saliency estimation using a non-parametric low-level vision model," in *IEEE CVPR*, 2011, pp. 433–440.
[30] R. Achanta and S. Süsstrunk, "Saliency detection using maximum symmetric surround," in *IEEE ICIP*, 2010, pp. 2653–2656.
[31] E. Rahtu, J. Kannala, M. Salo, and J. Heikkila, "Segmenting salient objects from images and videos," in *ECCV*, 2010, pp. 366–379.
[32] H. Seo and P. Milanfar, "Static and space-time visual saliency detection by self-resemblance," *Journal of vision*, vol. 9, no. 12, pp. 15:1–27, 2009.
[33] L. Zhang, M. Tong, T. Marks, H. Shan, and G. Cottrell, "SUN: A bayesian framework for saliency using natural statistics," *Journal of Vision*, vol. 8, no. 7, pp. 32:1–20, 2008.
[34] L. Duan, C. Wu, J. Miao, L. Qing, and Y. Fu, "Visual saliency detection by spatially weighted dissimilarity," in *IEEE CVPR*, 2011, pp. 473–480.
[35] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," in *NIPS*, 2007, pp. 545–552.
[36] X. Hou and L. Zhang, "Saliency detection: A spectral residual approach," in *IEEE CVPR*, 2007, pp. 1–8.
[37] R. Achanta, F. Estrada, P. Wils, and S. Süsstrunk, "Salient region detection and segmentation," in *IEEE ICVS*, 2008, pp. 66–75.
[38] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-aware saliency detection," *IEEE TPAMI*, vol. 34, no. 10, pp. 1915–1926, 2012.

[39] Y. Zhai and M. Shah, "Visual attention detection in video sequences using spatiotemporal cues," in *ACM Multimedia*, 2006, pp. 815–824.
[40] H. Jiang, J. Wang, Z. Yuan, T. Liu, N. Zheng, and S. Li, "Automatic salient object segmentation based on context and shape prior," in *BMVC*, 2011, pp. 1–12.
[41] M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu, "Global contrast based salient region detection," in *IEEE CVPR*, 2011, pp. 409–416.
[42] J. Reynolds and R. Desimone, "Interacting roles of attention and visual salience in v4," *Neuron*, vol. 37, no. 5, pp. 853–863, 2003.
[43] P. Reinagel, A. Zador *et al.*, "Natural scene statistics at the centre of gaze," *Network: Computation in Neural Systems*, vol. 10, no. 4, pp. 341–350, 1999.
[44] C. Rother, V. Kolmogorov, and A. Blake, ""GrabCut"– Interactive foreground extraction using iterated graph cuts," *ACM TOG*, vol. 23, no. 3, pp. 309–314, 2004.
[45] A. Borji and L. Itti, "State-of-the-art in visual attention modeling," *IEEE TPAMI*, 2012.
[46] A. Borji, D. Sihite, and L. Itti, "Quantitative analysis of human-model agreement in visual saliency modeling: A comparative study," *IEEE TIP*, 2012.
[47] T. Judd, F. Durand, and A. Torralba, "A benchmark of computational models of saliency to predict human fixations," MIT tech report, Tech. Rep., 2012.
[48] Y.-F. Ma and H.-J. Zhang, "Contrast-based image attention analysis by using fuzzy growing," in *ACM Multimedia*, 2003, pp. 374–381.
[49] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," in *NIPS*, 2006, pp. 545–552.
[50] W. Eihhauser and P. Konig, "Does luminance-constrast contribute to a saliency map for overt visual attention?" *European Journal of Neuroscience*, vol. 17, pp. 1089–1097, 2003.
[51] P. Felzenszwalb and D. Huttenlocher, "Efficient graph-based image segmentation," *IJCV*, vol. 59, no. 2, pp. 167–181, 2004.
[52] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE TPAMI*, vol. 23, no. 11, pp. 1222–1239, 2001.
[53] P. Mehrani and O. Veksler, "Saliency segmentation based on learning and graph cut refinement," in *BMVC*, 2010.
[54] T. Judd, K. Ehinger, F. Durand, and A. Torralba, "Learning to predict where humans look," in *IEEE ICCV*, 2009.
[55] Z. Wang and B. Li, "A two-stage approach to saliency detection in images," in *IEEE ICASSP*, 2008, pp. 965–968.
[56] M.-M. Cheng, N. J. Mitra, X. Huang, and S.-M. Hu, "Salientshape: Group saliency in image collections," GGC Group, Tsinghua University, Tech. Rep. TR-120624, June 2012.
[57] R. Margolin, L. Zelnik-Manor, and A. Tal, "Saliency for image manipulation," *The Visual Computer*, pp. 1–12, 2012.
[58] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE TPAMI*, vol. 24, no. 4, pp. 509–522, 2002.
[59] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa, "Sketch-based image retrieval: benchmark and bag-of-features descriptors," *IEEE TVCG*, 2011, preprint.
[60] ——, "An evaluation of descriptors for large-scale image retrieval from sketched feature lines," *Computers & Graphics*, vol. 34, no. 5, pp. 482–498, 2010.
[61] X. Bai, X. Yang, L. Latecki, W. Liu, and Z. Tu, "Learning context-sensitive shape similarity by graph transduction," *IEEE TPAMI*, pp. 861–874, 2010.
[62] A. Thayananthan, B. Stenger, P. Torr, and R. Cipolla, "Shape context and chamfer matching in cluttered scenes," in *IEEE CVPR*, vol. 1, 2003, pp. 127–133.
[63] X. Bai, Q. N. Li, L. J. Latecki, W. Y. Liu, and Z. W. Tu, "Shape band: A deformable object detection approach," in *IEEE CVPR*, 2009, pp. 1335–1342.
[64] A. Rosenfeld and D. Weinshall, "Extracting foreground masks towards object recognition," in *IEEE ICCV*, 2011, pp. 1–8.
[65] P. Isola, J. Xiao, A. Torralba, and A. Oliva, "What makes an image memorable?" in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 145–152.
[66] J. Jeon, V. Lavrenko, and R. Manmatha, "Automatic image annotation and retrieval using cross-media relevance models," in *ACM SIGIR*, 2003, pp. 119–126.
[67] Y. Gao, M. Wang, Z. Zha, Q. Tian, Q. Dai, and N. Zhang, "Less is more: Efficient 3d object retrieval with query view selection," *IEEE TMM*, vol. 11, no. 5, pp. 1007–1018, 2011.

Ming-Ming Cheng received his PhD degree from Tsinghua University in 2012. He is currently a research fellow in Oxford Brookes University, working with Prof. Philip Torr. His research interests includes computer graphics, computer vision, image processing, and image retrieval. He has received the Google PhD fellowship award, the IBM PhD fellowship award, and the "New PhD Researcher Award" from Chinese Ministry of Education.

Niloy J. Mitra received his PhD degree from Stanford University in 2006. He is currently a senior lecturer in the Computer Science Department at the University College London (UCL). His research interests span a range of topics including shape analysis, geometry processing, shape manipulation, image analysis, and recreational art. He is on the editorial boards of Computer & Graphics, and Visual Computer.



Xiaolei Huang received her PhD degrees from Rutgers University in 2006. She is currently an assistant professor in Lehigh University. Her research interests are in computer vision, computer graphics, and multimedia retrieval. Dr. Huang serves on the program committees of several international conferences on computer vision and computer graphics, and she reviews papers regularly for journals including the IEEE TPAMI, and the IEEE TIP. She is a member of the IEEE.

Philip H. S. Torr received the PhD degree from the Robotics Research Group of Oxford University under Professor David Murray. He worked for another three years at Oxford and is currently a visiting fellow at Oxford University, working closely with Professor Zisserman and the Visual Geometry Group. He left Oxford to work for six years as a research scientist for Microsoft Research, first in Redmond, in the Vision Technology Group, then in Cambridge, UK, founding the vision side of the Machine Learning and Perception Group. He is now a professor of computer vision and machine learning at Oxford Brookes University. He has won awards from several top vision conferences, including ICCV, CVPR, ECCV, NIPS and BMVC. He is a senior member of the IEEE, Royal Society Wolfson Research Merit Award holder, and program co-chair of ICCV 2013.

Shi-Min Hu received the PhD degree from Zhejiang University in 1996. He is currently a chair professor of computer science in the Dept. of Computer Science and Technology, Tsinghua University, Beijing. His research interests include digital geometry processing, video processing, rendering, computer animation, and computer-aided geometric design. He is on the editorial board of Computer Aided Design. He is a member of the IEEE.