The definitive version of this paper is available from Elsevier at: http://dx.doi.org/10.1016/j.patrec.2017.09.010

Pattern Recognition Letters
journal homepage: www.elsevier.com

Joint upsampling of random color distance maps for fast salient region detection

Maiko M. I. Lie[a,**], Gustavo B. Borba[b], Hugo Vieira Neto[a], Humberto R. Gamba[a]

[a] *Graduate Program in Electrical and Computer Engineering, Federal University of Technology – Paraná, Curitiba, Brazil*
[b] *Department of Electronics, Federal University of Technology Paran, Curitiba, Brazil*

ABSTRACT

The human visual system is capable of rapid response, even in the presence of massive quantities of visual information. This is possible because it restricts the operation of further processing stages to a small, potentially important, subset of the incoming information. This mechanism is called *visual attention* and is drawn by distinctive, *visually salient*, regions of the scene. Detection of visually salient regions is widely employed in vision-based applications, since a reduction in visual search space can lead to significant improvement in computational performance. Despite recent advances in salient region detection, most efforts have focused on improving accuracy, at the expense of increased execution time, significantly hindering their applicability. To address this, a fast and accurate salient region detection method is presented in this work, based on an efficient saliency estimate called *random color distance map*. This estimate is joint upsampled into an accurate saliency map, which is assessed and compared to saliency maps obtained by other four state-of-the-art methods on the MSRA1K, MSRA10K and SED2 datasets, showing that it is highly competitive in both accuracy and execution time.

1. Introduction

As a consequence of its limited processing capacity, the human visual system employs a mechanism that reduces the amount of incoming visual information that is effectively processed. This mechanism, *visual attention*, restricts the activity of further cognitive processes to a small, potentially important, subset of the observed scene, significantly decreasing their burden (Wolfe, 1994; Frintrop, 2006). The usefulness of such mechanism is not restricted to the human visual system – any vision-based system can benefit from this selective reduction of information. For this reason, it has been extensively explored in computational applications such as image compression (Ouerhani et al., 2001), content-based image retrieval (Marques et al., 2006), visual novelty detection (Vieira Neto, 2011) and object detection (Silva et al., 2014).

Empirical evidence indicates that visual attention is drawn to distinctive regions of the observed scene (Treisman and Gelade, 1980; Elazary and Itti, 2008). While significantly more complex processes are involved during extended and task-related

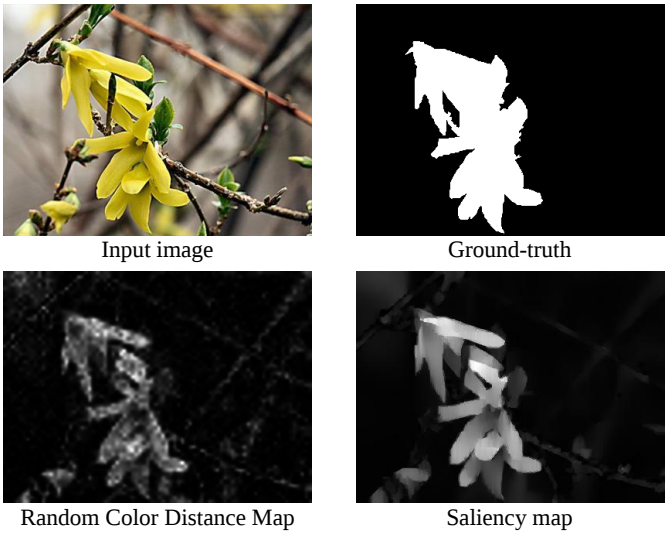

Fig. 1: Fast salient region detection. An efficient saliency estimate is computed as a sparse, downsampled, *random color distance map*. The result is joint upsampled into an accurate full-resolution saliency map, taking only a fraction of the time it would require to compute it using densely-sampled color distances on the full-resolution input.

**Corresponding author.
e-mail: minian.lie@gmail.com (Maiko M. I. Lie)

viewing (Theeuwes, 2010), stimulus-driven distinctiveness – *visual saliency* – is known to model the early stages of visual attention with high accuracy. Despite its short duration, the early stages of visual attention have a very significant impact on vision-based applications. Moreover, since it is stimulus-driven, instead of task-driven like later visual attention stages, it is also less subjective and more generally applicable, consequently, most research on computational visual attention has been dedicated to visual saliency detection.

Despite the high accuracy of modern saliency detection methods, their complexity makes many of them inadequate for real-time applications. Saliency detection has been reported to take less than 150 ms in the human visual system (Theeuwes, 2010) while, for instance, among the most accurate methods in the most extensive benchmark available to date (Borji et al., 2015), there are methods that take several seconds to process a single image with 400×300 pixels on a Xeon E5645 2.4 GHz CPU with 8 GB RAM. This work addresses this issue by presenting a salient region detection method based on the concept of a *random color distance map*, which is a bottom-up, unsupervised, and computationally efficient approach to saliency estimation. While this estimate itself is not adequate for salient region detection, it can be made so when combined with joint upsampling, resulting in computationally efficient, accurate, saliency maps (Figure 1).

The main contributions of this paper are: (i) showing that the color uniqueness of a pixel can be estimated with linear computation time, as the accumulated color distance to a small set of pixels randomly sampled from the scene; (ii) demonstrating that the proposed estimate can be joint-upsampled for efficient salient region detection, with competitive accuracy in comparison to state-of-the-art methods; (iii) an assessment of the data reduction parameters of the proposed model, showing that the amount of data necessary for salient region detection can be drastically reduced, substantially reducing execution time, without significant decrease in accuracy. The proposed method is assessed and compared to other four state-of-the-art saliency detection methods on the MSRA1K, MSRA10K and SED2 datasets in terms of precision, recall, F-measure and execution time. The results show that it is highly competitive with state-of-the-art methods, presenting one of the best trade-offs between accuracy and execution time.

This paper extends the results presented by Lie et al. (2016), providing more detailed descriptions and experiments, in addition to improvements to the method itself. In particular, execution time was significantly improved without noticeable impact on accuracy, by restricting the RGB to CIELAB colorspace conversion to the downsampled version of the input image instead of the full-resolution input itself. Furthermore, accuracy was also significantly improved, with practically no additional computational cost, by adopting background prior information, which is shown to be trivial to be incorporated in the method.

2. Related Work

Despite the variety of computational approaches for visual saliency estimation, their formulations are mostly based on the framework proposed by Koch and Ullman (1987), in which differences of low-level features are combined into a topological representation of relative conspicuity in the scene, a *saliency map*. This role of feature combination in selective attention is grounded mainly on the psychological experiments by Treisman and Gelade (1980), which show that, in human visual attention, low-level features are registered simultaneously at an early stage, and only later combined to identify individual objects. While empirical evidence indicates that a series of low-level features is involved in this process (Braun and Julesz, 1998), the high accuracy of most recent computational models suggest that color may be the most informative feature in natural images (Borji et al., 2015).

The most straightforward approach to estimate the saliency of an image location in terms of color is to compute its color distance to all other image locations. This is reasonable as long as the image is represented in a perceptually uniform colorspace, that is, a colorspace in which the Euclidean distance approximates perceptual difference (Reinhard et al., 2008). For an image I, Zhai and Shah (2006) formulated the saliency $S(x, y)$ of each pixel $I(x, y)$ in this approach as defined in Equation 1:

$$S(x, y) = \sum_{(x_i, y_i) \in P} \|I(x, y) - I(x_i, y_i)\|, \tag{1}$$

where P is the set of all pixel locations in I. In other words, the saliency of a pixel is defined as its accumulated color distance to all other pixels in the image. Although straightforward, this approach has $O(N^2)$ computational complexity for an image with N pixels, resulting in a poor solution for real-time applications. To reduce computational burden, Zhai and Shah (2006) proposed estimating the saliency of each color instead, which can improve performance if there are significantly more pixels than colors in the image, since simply assigning a precomputed color saliency to each pixel results in linear complexity. However, having more pixels than possible colors in an image is rarely the case – for instance, while a 1920×1080 true-color image has around 2 million pixels, it has over 16 million colors. Considering this, the approach is restricted to luminance information, losing the distinctiveness of color information but significantly improving computational performance, since the number of operations is quadratic with respect of the number of values. A strategy to improve the computational performance of this approach, without completely sacrificing color information, was presented by Cheng et al. (2015). Histogram-based color quantization is employed, followed by a selection of the most frequent remaining colors, in order to reduce their number to 85 – a very significant improvement over the luminance approach. In practice, the luminance approach is still faster, since it avoids the overhead of histogram quantization and conversion to a perceptually uniform colorspace, but the color quantization approach predicts salient regions much more accurately while achieving competitive speed (Borji et al., 2015).

An altogether different approach to estimate saliency efficiently, in terms of color distinctiveness, is to compare each pixel to a color summary of the image. This approach was adopted by Achanta et al. (2009), who defined the saliency

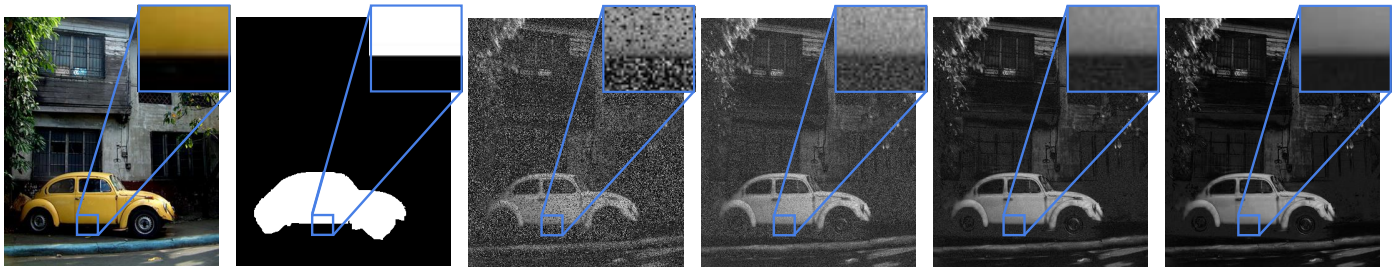

Fig. 2: Saliency estimation for different sizes N_R of the random set P_{rand}. Increasing N_R sharpens the output, but for values as small as $N_R = 1$, the salient region is already evident.

$S(x, y)$ of a pixel $I(x, y)$ as given in Equation 2:

$$S(x, y) = \|I(x, y) - I_G\|, \tag{2}$$

where I_G denotes the average color of the Gaussian filtered input I. This method is computationally efficient, since it has $O(N)$ complexity and computing the average color of I, as well as its Gaussian filtering, can also be performed very efficiently.

An intermediate approach, which does not require comparison to all pixels of the input, but also does not summarize their color in a single value, is the stochastic method by Vikram (2013), in which color distances are computed for randomy sampled pairs of pixels. Random pairs are resampled and compared until an empirically determined number of iterations is reached. This method is largely based on a previous and more general approach by Stentiford (2007), which does not sample random pairs of pixels but pairs of randomly shaped templates. Motivated by the random scattering of receptive fields in the human visual system, Vikram et al. (2012) devised a method which estimates the saliency of a pixel as the difference of its value to the mean value of the randomly generated windows that contain it. These windows have random positions and sizes, while their number was determined empirically as $0.02{\cdot}N$ for an image with N pixels.

3. Random Color Distance Map

The proposed method follows a stochastic approach, with a significant distinction with respect to the previously mentioned stochastic saliency detection methods – its purpose is not to be more biologically plausible, but to improve computational performance. Random sampling is used solely to obtain a representative color summary for the image, so that saliency can be estimated more efficiently than comparing each pixel to all others and more robustly than comparing only to the average color of the image. Considering this, for an image I, which is converted from the RGB colorspace to CIELAB, in order to leverage its perceptual uniformity, the saliency $S(x, y)$ of the pixel $I(x, y)$ is estimated as defined by Equation 3:

$$S(x, y) = \sum_{(x_r, y_r) \in P_{\text{rand}}} \|I(x, y) - I(x_r, y_r)\|, \tag{3}$$

where P_{rand} is a set of N_R random pixel locations in I. Equation 3 is essentially the same as Equation 1, but instead of a sum across the entire image, only a subset of pixels at positions (x_r, y_r), $\forall\, r \in [1\,..\,N_R]$ is adopted instead, where each coordinate is randomly sampled from a discrete uniform distribution in the interval $[1\,..\,L]$, L being the image width for x_r and height for y_r.

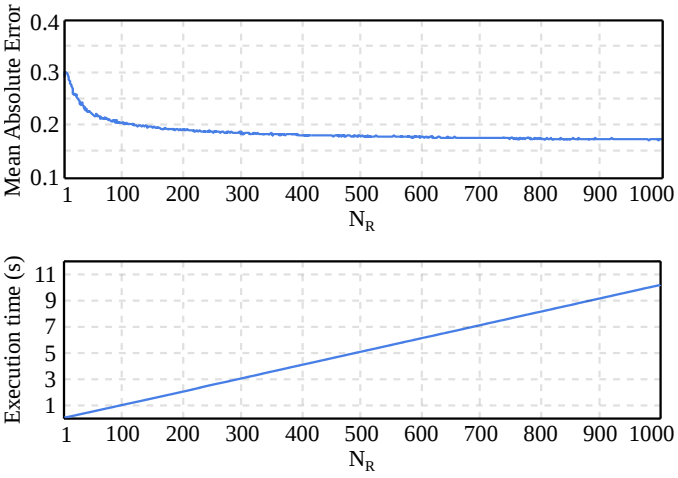

Fig. 3: Mean absolute error and execution time for random color distance map computation. Despite already being reasonably small, error decays exponentially as N_R is increased. As expected, execution time increases linearly. However, for $N_R = 100$, it already exceeds one second, showing that adopting large values for N_R is not an efficient approach to increase accuracy. The algorithm was computed for the example in Figure 2 (which has 345×400 pixels), on an Intel Core i7-860 2.80 GHz CPU with 4 GB RAM.

The premise of this approach is that a random set of pixels provides an adequate summary of the entire image in terms of color, which tends to be true as the set size N_R is increased. Our interest, however, is in the particular case when N_R is small with respect to the entire image, since in this case it can be sampled with minimal computational effort, providing an efficient color summary. The question is whether this summary is still representative when the set size is small. This seems to be the case, as illustrated by the example in Figure 2, which shows that the salient region is already evident even when N_R assumes very small values, despite presenting a noisy aspect.

This is illustrated more precisely in Figure 3 (top), which shows the mean absolute error of saliency estimation for different values of N_R, considering the input image and ground-truth depicted in Figure 2. Error decays exponentially as set size increases, despite being reasonably small even for small values

of N_R. However, increasing set size is not a computationally efficient solution. As Figure 3 (bottom) illustrates, even for medium sized images (e.g. 345×400), the set size required to successfully "denoise" the output image is computationally prohibitive for real-time applications.

The result of Equation 3 over an input image is called a *random color distance map*. While, for a small N_R, it does not result in an adequate saliency map for salient region detection, it can be made so when combined with image abstraction, since this lessens the impact of pixel level inaccuracies via region-level processing. As it turns out, this combination not only results in accurate saliency maps, but it is also computationally efficient, since there are several fast image abstraction methods in the literature (Gastal and Oliveira, 2011; He et al., 2013; Min et al., 2014) and, when combined with such methods, N_R can assume very small values without significant decrease in accuracy.

Moreover, due to its simplicity, extending the *random color distance map* is straightforward. For instance, *background prior*, the assumption that image boundaries belong to the background, can be incorporated in the model simply by restricting the sampling of P_{rand} to the image boundaries. In other words, for each pixel location (x_r, y_r), instead of randomly sampling the coordinates x_r and y_r from the interval $[1\,..\,L]$, they are randomly sampled from $[1\,..\,BL] \cup [(L - BL)\,..\,L]$, where L is the image width for x_r and height for y_r, while B is the *boundary ratio*, a parameter which defines the proportion of the image dimensions to adopt as boundary size for the prior. Adopting $B = 0.5$ disregards boundary prior, since it indicates that two opposing boundaries take half of the image each, setting the entire image as boundary. In the benchmark by Borji et al. (2015), it was shown that the six most accurate salient region detection methods adopted boundary prior, suggesting that it is a significant factor in state-of-the-art accuracy. In Section 5, it is shown that this is indeed the case, as the accuracy of the proposed method is significantly improved by adopting this prior – with practically no additional computational cost.

4. Joint Upsampling

Most of recent saliency detection methods are region-based, meaning that instead of performing pixelwise saliency estimation, they estimate the saliency of image patches. These patches are obtained using image abstraction methods, which decompose the image into perceptually homogeneous regions (Cheng et al., 2013). Salient object detection using this approach is usually more accurate and scales better (Cheng et al., 2015).

Based on the salient region detectors considered in the extensive survey by Borji et al. (2014), the most common image abstraction methods are the graph-based segmentation (EGBS) by Felzenszwalb and Huttenlocher (2004), SLIC superpixels (Achanta et al., 2012), and Mean Shift (Comaniciu and Meer, 2002). However, these methods are too computationally expensive to incorporate into a fast saliency detector. Instead, an edge-preserving smoothing filter was adopted here for image abstraction – the Fast Global Smoother (FGS) by Min et al. (2014). While there are several edge-preserving filters in the literature (Tomasi and Manduchi, 1998; Gastal and Oliveira, 2011; He et al., 2013), FGS was chosen because it presents linear complexity, short execution time and ease of parameterization. Table 1 shows the execution time of the image abstraction methods mentioned. The source code from the original authors was used in the experiments whose results are reported, except in the case of Mean Shift, which was assessed using the EDISON (Christoudias et al., 2002) implementation, since it is commonly used in saliency detectors.

Table 1: Execution time of image abstraction algorithms. FGS has the shortest execution time, less than half of the time taken by the second fastest algorithm (SLIC). The algorithms were executed using their default parameters, using an input image with 400×300 pixels, on an Intel Core i7-860 2.80 GHz CPU with 4 GB RAM.

Method	Mean shift	EGBS	SLIC	FGS
Execution time (s)	0.90	0.13	0.11	0.04

An edge-preserving smoothing filter removes image details without blurring edges. This kind of filter blurs the image like an usual Gaussian low-pass filter, but has its effect "weighted down" near edges. This is accomplished by filtering in both *space* and *range*, meaning that the output of the filter depends not only on the *geometric closeness* of the pixels in its support, but also on their *photometric similarity* (Tomasi and Manduchi, 1998). FGS (Min et al., 2014) formulates this as an optimization framework, which is approximated as a sequence of 1D subsystems, one for each row/column, that minimizes the energy function presented in Equation 4:

$$J(u) = \sum_n \left((u_n - f_n)^2 + \lambda \sum_{i \in \mathcal{N}(n)} w_{n,i}(g)(u_n - u_i)^2 \right), \quad (4)$$

where f, g and u are rows/columns of the *input*, *guide* and *output* images, respectively. The function $J(u)$ is computed along $n \in [1\,..\,L]$, where L corresponds to the width of the image if the input is a row or height if it is a column. $\mathcal{N}$ is a set that contains the two neighbors of n, λ is the *smoothness parameter*, and $w_{n,i}(g)$ is a function that determines the similarity between pixels n and i in the image g, and is defined in Equation 5:

$$w_{n,i}(g) = exp\left(\frac{-\|g_n - g_i\|}{\sigma_c} \right), \quad (5)$$

where σ_c is the *range parameter*. In this work, FGS was computed adopting 3 iterations and $\sigma_c = 0.03$, as suggested by Min et al. (2014), and $\lambda = 10^2$, which was determined empirically.

The formulation described by Equations 4 and 5 considers two sources of input data, f and g – spatial and range data, respectively. The same image can be used for both inputs, as in ordinary edge-preserving smoothing, but not necessarily. For instance, in image colorization, f can be a color image with sparse scribbles, whereas g is a grayscale image (Kopf et al., 2007). While the smoothness component of the filter spreads the colors from f in space, the range component restricts them inside the edges of g, resulting in uniform region colorization.

Upsampling a sparse solution using a full-resolution input as guide image is called *joint upsampling*. This approach was popularized by the *joint bilateral upsampling* (Kopf et al., 2007),

which is based on the bilateral filter (Tomasi and Manduchi, 1998) and has shown to improve efficiency in tasks such as tone mapping, colorization and depth from stereo. When a solution is too costly to be computed in the full-resolution input itself, joint upsampling can be very advantageous. As shown in Section 3, this is the case when computing a dense random color distance map. Considering this, a sparse random distance map ($N_R << N$) is computed instead and then joint upsampled into a full-resolution saliency map. Moreover, since the solution is upsampled, its execution time can be further improved by computing it in a downsampled copy of the input. While downsizing degrades region contours and texture information, the former is corrected by joint upsampling with the full-resolution input as guide image, while the latter actually improves accuracy, since the purpose of salient region detection is detecting homogeneous regions, not the details inside them (Borji et al., 2014). As it turns out, accuracy does not decrease significantly even within a wide range of downsampling scales, allowing a very significant reduction of the amount of processed data.

Execution time can also be improved by performing the RGB to CIELAB colorspace conversion in the downsampled version, rather than in the full-resolution input image, since it is only needed for perceptually uniform color distance computation. As consequence, the joint upsampling is guided by the edges of the input image in the RGB colorspace. While edges in the CIELAB colorspace might be more perceptually meaningful, this conversion significantly increases execution time without perceptible increase in accuracy. The joint upsampled random color distance map is also subject to gamma-correction ($\gamma = 3$) to suppress occasional noise in the background.

5. Experiments

The proposed method was assessed and compared to four state-of-the-art saliency detection methods: Frequency-tuned (FT) (Achanta et al., 2009), Spectral Residual (SR) (Hou and Zhang, 2007), Random Center Surround (RCS) (Vikram et al., 2012) and Absorbing Markov Chain (AMC) (Jiang et al., 2013). The criteria for these choices were number of citations (FT and SR have both more than 1,000 citations each, according to Google Scholar), similarity to the proposed approach (RCS is also based on random color distances) and performance (AMC is the fastest among the most accurate methods in the benchmark by Borji et al. (2015)). The experiments were performed on an Intel Core i7-860 2.80 GHz CPU with 4 GB RAM.

The proposed method was implemented in MATLAB, except for the Fast Global Smoother, for which the MEX interface and C++ source code of the original authors (Min et al., 2014) was used. The implementation languages of the compared methods are the following: AMC (MATLAB, C++), FT (C++), RCS (MATLAB), SR (MATLAB). The assessment was based on the source code made publicly available by the authors of each method.

5.1. Datasets

The experiments were performed on the publicly available MSRA1K (Achanta et al., 2009), MSRA10K (Cheng et al., 2015) and SED2 (Alpert et al., 2012) datasets, which provide accurate object-contour ground-truths indicating regions consistently labeled as salient by human subjects (Cheng et al., 2015). The two former were sampled from the MSRA Salient Object Database (Liu et al., 2007) and have an average image size of 400×300 pixels. The MSRA1K dataset contains 1,000 images, while MSRA10K contains 10,000 – despite the existence of some images common to both, one is not a complete subset of the other. Following the approach by Cheng et al. (2015), the experiments are performed on these two datasets to assess scalability. The SED2 dataset is comprised of 100 images with average size of approximately 300×225 pixels, each image containing two objects. Following the approach by Borji et al. (2014), the experiments are also performed on this dataset to assess accuracy when there is more than a single object in the scene.

5.2. Metrics

The assessment was based on accuracy and execution time. Accuracy is measured in terms of *precision*, *recall*, and *F-measure*, which are standard metrics in salient region detection assessment (Borji et al., 2014). *Precision* and *recall* are defined in Equation 6:

$$Precision = \frac{TP}{TP + FP}, \quad Recall = \frac{TP}{TP + FN}, \tag{6}$$

where TP (true positives) are salient pixels correctly detected as such, FN (false positives) are salient pixels detected as background and FP (false positives) are background pixels detected as salient. Since saliency maps are usually given in shades of gray, and these metrics are for binary values, the maps are thresholded for each value in the $[0 .. 255]$ interval. The accuracy of a method on an image is summarized as the precision-recall curve for all thresholds in this interval, while the accuracy for an entire dataset is summarized as the average precision-recall curve for all images.

Besides the precision-recall curve, accuracy can also be summarized by the F-measure, which is the weighted harmonic mean of precision and recall, as defined in Equation 7:

$$F_\beta = (1 + \beta^2)\frac{Precision \times Recall}{(\beta^2 \times Precision) + Recall}, \tag{7}$$

where β is used to emphasize the effect of precision or recall. Since many authors consider precision more important than recall for saliency detection, it is common to adopt $\beta^2 = 0.3$ (Achanta et al., 2009; Li et al., 2013; Cheng et al., 2015). While the precision-recall curve is computed for all thresholds in [0..255], F-measure is computed for a single adaptive threshold – *twice the average saliency of the image* – following the widely adopted assessment approach by Achanta et al. (2009).

5.3. Parameter assessment

There are three parameters in the computation of the random color distance map: *set size* $N_R \in [1 .. N]$, *downsize scale* $D \in (0, 1]$ and *boundary ratio* $B \in (0, 0.5]$. Each parameter was assessed by varying its value while the remaining parameters were fixed to default values. For D and B, default values

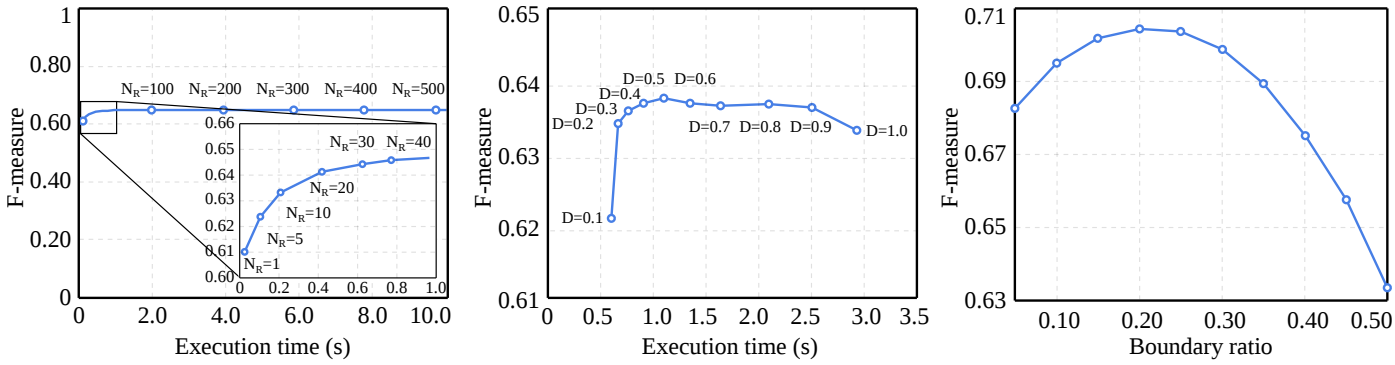

Fig. 4: Parameter assessment. All images from the MSRA1K, MSRA10K and SED2 datasets were considered. **Left:** Set size N_R ($D = 1.0$, $B = 0.5$). Small values offer the best trade-off. For $N_R > 10$, execution time increases significantly with only marginal accuracy improvement. **Center:** Downsize scale D ($N_R = 10$, $B = 0.5$). The best trade-off occurs for $D = 0.2$. Larger values incur significant computational cost for almost no accuracy improvement. **Right:** Boundary ratio B ($N_R = 10$, $D = 1.0$). Any valid boundary ratio value improves accuracy. The best trade-off occurs around $B = 0.2$.

are straightforward – downsizing and boundary prior can simply be disabled ($D = 1.0$, $B = 0.5$). For N_R there is no obvious default value, so it was determined by analysing accuracy vs. execution time for several values. As shown in Figure 4 (left), accuracy improvement due to increasing N_R saturates around an F-measure of 0.64, which is achieved with $N_R \approx 20$. Adopting $N_R > 10$ is not cost-effective, since execution time increases significantly with only marginal accuracy improvement. Considering this, $N_R = 10$ is adopted as default size for P_{rand}.

Downsize scale has significant impact on execution time and should be kept as small as possible. As shown in Figure 4 (center), accuracy does not change significantly for a wide range of scales, remaining with F-measure just below 0.64 for $0.2 \leq D \leq 9.0$. The lowest accuracy occurs for $D = 0.1$, followed by $D = 1.0$, suggesting that – for joint upsampling – estimating saliency in the full-resolution input might actually hinder accuracy. The downsize scale is therefore set as $D = 0.2$ since larger values increase execution time without significantly improving accuracy.

Unlike the previous parameters, boundary ratio has no significant impact on execution time – it merely defines the area from which to sample pixels for color distance computation. As shown in Figure 4 (right), $B = 0.5$ (i.e. no boundary prior) results in the lowest accuracy, indicating that boundary prior always improves accuracy. The best performance is achieved with $B = 0.2$, which results in an accuracy increase of approximately 10% compared to ignoring boundary prior. Note that formulating boundary prior with random sampling ensures that the proposed method leverages the fact that boundary regions might correspond to background, but do not rely on this.

5.4. Quantitative analysis

Precision-recall curves for the compared methods on the MSRA1K dataset are presented in Figure 5 (left). The proposed method significantly outperforms all compared methods except AMC. FT and RCS present similar accuracy, while SR presents the lowest accuracy by a large margin. An assessment on the MSRA10K dataset, presented in Figure 5 (center), shows that almost all methods suffer a decrease in accuracy, the most severe by FT, which is expected, since it is known that pixel-level methods do not scale as well as region-based methods (Cheng et al., 2015). RCS suffers a more subtle decrease, probably due to its saliency maps always emphasizing the image center (see Figure 7, 5th column), which can increase performance in datasets with center-bias like MSRA1K and MSRA10K. In fact, it has been shown that a simple Gaussian blob at the center of the image can outperform many salient region detection methods on datasets with this characteristic (Cheng et al., 2015). Surprisingly, SR has improved accuracy on the larger dataset, however, the improvement is marginal and the method remains the most inaccurate among the assessed. Based on the precision-recall curve for the SED2 dataset, presented in Figure 5 (right), the number of objects in the scene does not seem to be an essential factor in the performance of any of the compared methods, since their relative performance remains similar to that of the previous datasets. Since SED2 presents simple scenes with two salient objects, dissimilarity to the average color of the image correlates reasonably well with saliency and center-bias is less severe. Consequently, FT outperforms RCS on this dataset. Despite still outperforming all other methods, of the three datasets considered, AMC presents its worst performance on SED2. This is very likely due to its difficulty in detecting small salient regions, as will be discussed later.

Table 2: Execution time (for an image with 400 × 300 pixels) and F-measure of the saliency detection methods assessed. The experiments were executed on an Intel Core i7-860 2.80 GHz CPU with 4 GB RAM.

Method	Execution time (s)	F-measure		
		MSRA1K	MSRA10K	SED2
AMC	0.1826	0.9059	0.8358	0.7375
Proposed	0.0638	0.7933	0.6956	0.7235
FT	0.0582	0.7070	0.5972	0.6246
RCS	0.7333	0.6607	0.6181	0.5709
SR	0.0090	0.4819	0.4900	0.4299

Considering execution time, shown in Figure 6 and described in more detail in Table 2, the proposed method presents one of the best trade-offs. Compared to FT, it achieves significantly superior accuracy with very similar execution time (it takes an additional 0.0056 seconds) while, compared to AMC, it achieves

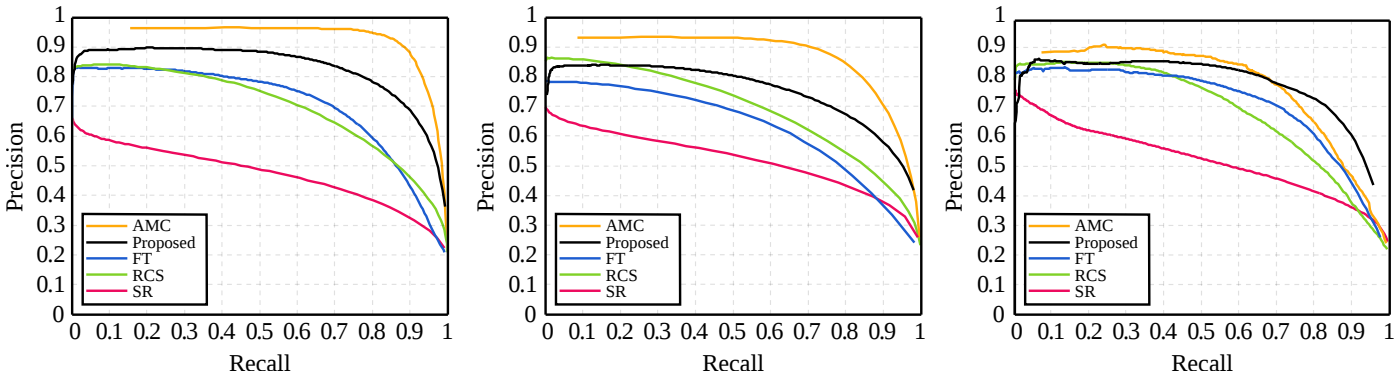

Fig. 5: Precision-recall curves of the compared methods on different datasets. **Left:** MSRA1K. **Center:** MSRA10K. **Right:** SED2. The proposed method is highly competitive on all three datasets, scaling well in terms of both dataset size and number of objects. From the MSRA1K to the larger MSRA10K dataset, there is a slight decrease in accuracy for all compared methods, except SR, for which there a slight increase. On the SED2 dataset, the proposed method has accuracy comparable to AMC and superior to all other methods.

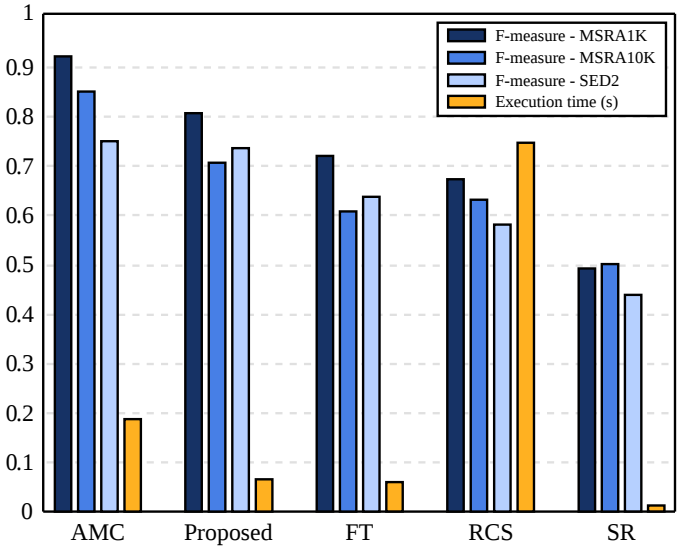

Fig. 6: F-measure and execution time of the compared methods, sorted by descending accuracy. The proposed method is highly competitive on the three datasets, being one of the most accurate while remaining one of the fastest. In terms of accuracy, only AMC is superior. In terms of execution time, SR and FT are faster, despite none of them being as accurate. While SR is too inaccurate for salient region detection, FT is significantly less accurate and only 0.0056 seconds faster on average for an image with 400×300 pixels.

inferior but competitive accuracy while performing three times faster. RCS is the slowest method, barely computing a single saliency map per second, while SR is very fast but too inaccurate for adequate salient region detection.

Despite being the most accurate among the assessed methods, AMC is also one of the slowest (only RCS is slower). It adopts a graph-based model with superpixels as nodes (Jiang et al., 2013), which are computed by the SLIC method (Achanta et al., 2012). As shown previously in Table 1, SLIC takes on average 0.11 seconds to compute a single 400×300 image. This accounts for more than half of the execution time of AMC, which corresponds to approximately twice the execution time of the entire proposed method. SR and FT are the fastest among the assessed methods. Both have simple and straightforward models: the former is a difference in the frequency domain, while the latter is based on color distances to the average color of the image. However, while SR relies on heavy downsampling (i.e. to 64 pixels in width or height) to achieve reasonable computational performance, but significantly compromising accuracy, FT operates on the full-resolution image. Since FT computes a single difference per pixel, it runs efficiently and in linear time, despite being slower than SR. The long execution time by RCS can be attributed to the large number of sub-windows involved in its computation: $0.02 \times N$ for an image with N pixels, which generates 2,400 windows for a single 400×300 image.

5.5. *Qualitative analysis*

Saliency maps computed using the compared methods are presented in Figure 7, ordered from left to right by decreasing accuracy. AMC outputs the most homogeneous saliency maps, since it assigns a single saliency value for each superpixel. For uncluttered images with high contrast salient regions, it outputs mostly binary saliency maps, closely resembling the ground-truth. The proposed method outputs similarly homogeneous regions, despite not suppressing details as well. This is mostly a consequence of the different image abstraction approaches adopted by each method. While SLIC superpixels output discrete labeled segments, the Fast Global Smoother is simply a low-pass filter sensitive to photometric similarity, which makes it susceptible to sharp details in salient regions. Considering that the effect of such details is mostly a slight decrease in homogeneity, adopting the Fast Global Smoother offers a good compromise, since it allows detecting salient regions three times faster. Another advantage of avoiding superpixel computation is that there is no need to specify the number or size of the superpixels. Adopting inadequate values for these parameters can have significant impact on detection accuracy, for instance, when the salient region is smaller than the superpixel size, as illustrated by the example in Figure 7 (second row). It is possible to mitigate that by computing superpixels on multiple scales (Tong et al., 2014), but this is not adequate for fast salient region detection, since it significantly increases execution time.

As mentioned previously, RCS overemphasizes the image center. In some cases, such as the saliency maps in the 4th,

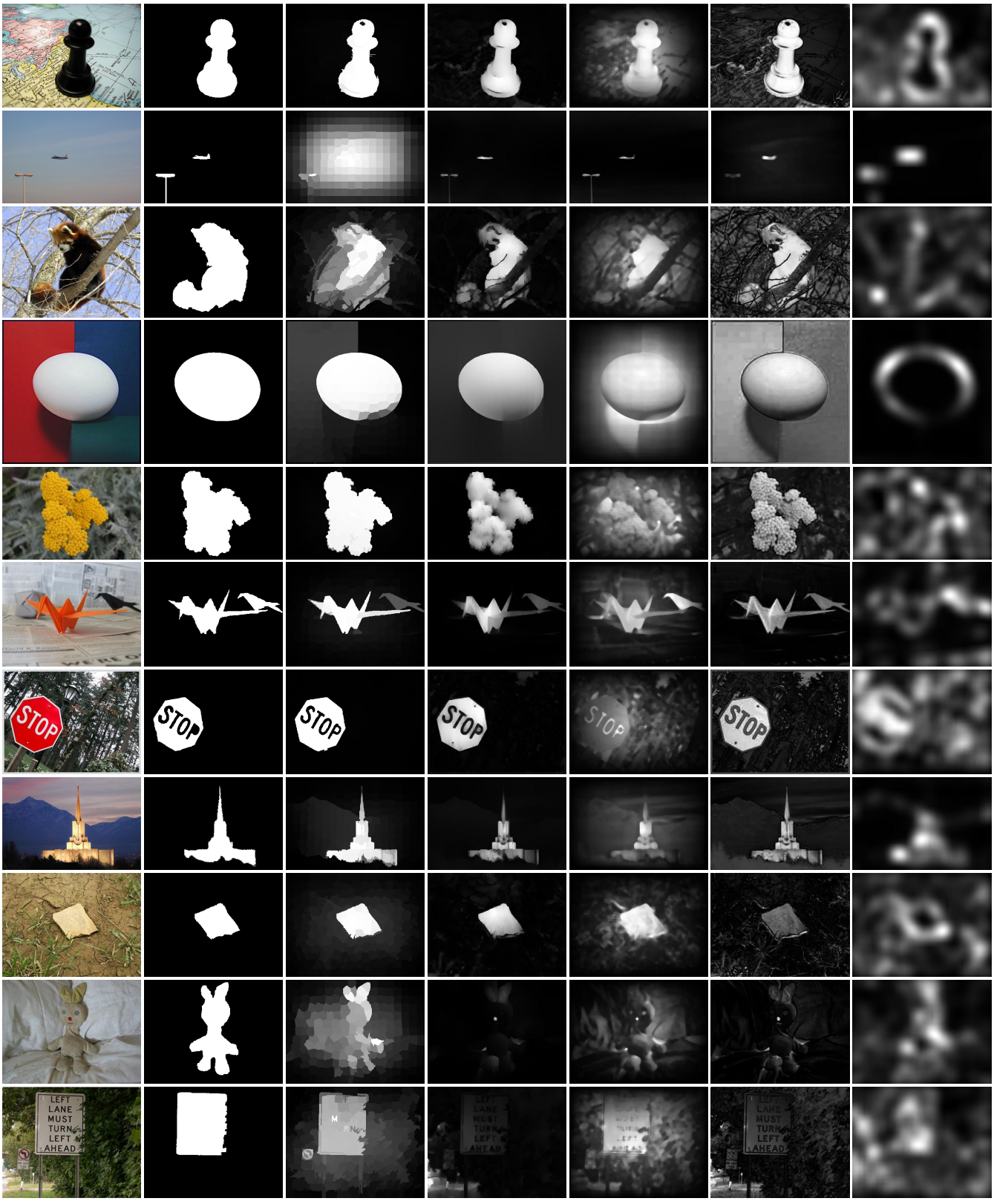

Fig. 7: Saliency maps computed using the compared methods, ordered from left to right by decreasing accuracy. AMC outputs the most homogeneous saliency maps, since it assigns a single value for each superpixel. However, it presents difficulties detecting small salient objects. The proposed method outputs similarly homogeneous regions, despite not suppressing details as well. RCS overemphasizes the center of the image, resulting in good accuracy on center-biased datasets but not necessarily by detecting salient regions. FT outputs sharp, accurate saliency maps, but is susceptible to small distractors in the background, due to its fine-grained approach. SR produces inaccurate, low-resolution, saliency maps, which might be useful for rough localization but are not adequate for salient region detection. The last two rows present failure cases of our method, which demonstrate that its limitations are also shared by some of the compared methods.

5th and 7th rows of Figure 7, it is possible to notice that the salient regions in the output are barely distinguishable from the background, but still manage to have significant overlap with the ground-truth simply because they emphasize the center of the image. FT adopts a fine-grained approach, capable of outputting sharp saliency maps with accurate boundaries. However, since it does not include any image abstraction method, it is susceptible to small distractors in the background. Additionally, in cases such as the 4th row of Figure 7, in which there are multiple regions with similar size and different colors, the method may fail since distance to the average color of the image will not be a good estimate of saliency. SR produces very inaccurate, low-resolution, saliency maps. In most cases, it emphasizes edges and might be susceptible to background texture, such as in the saliency maps in the 5th, 7th and 9th rows of Figure 7. Despite being too inaccurate for salient region detection, the elegant formulation and short execution time of this method make it attractive to applications that do not require accurate regions, such as fixation prediction (Borji et al., 2015), or as a building block for more elaborate methods (Silva et al., 2014).

5.6. Limitations

The proposed method is based on the premise that salient regions distinguish themselves from the background in terms of color – also known as the *color uniqueness hypothesis*. While, in practice, this is mostly true for natural images, there are cases in which this does not happen. The Figure 7 (tenth row) shows one such case, in which it is possible to notice that the nose of the doll is detected as salient due to its distinctive color, while the rest of the doll, which was expected to be the salient region, is not. Except for SR, all other methods also assume that the salient regions have distinctive color, consequently they share this limitation and also output incorrect saliency maps, some very similar to that of the proposed method.

Unlike approaches which explicitly model image boundaries as background (e.g. AMC) or which performance relies heavily on center-bias (e.g. RCS), the proposed method employs a "soft" boundary prior. This is accomplished by concentrating the random sampling for color distance computation on the image boundaries. This does not explicitly set the boundaries as background, which makes the method robust since it still allows salient regions inside them. However, this also makes the method susceptible to high-contrast boundary distractors, as shown in Figure 7 (last row). Notice that the same boundary distractor is detected as salient by FT, since its model relies on color contrast without distinguishing boundary regions.

6. Conclusions

This paper presented a bottom-up, unsupervised, computationally efficient method for salient region detection. It is based on a *random color distance map*, which estimates visual saliency as accumulated color distances to a color summary of the image, computed from a set of pixels randomly sampled from the scene. This map can be computed very efficiently if the set of random pixels is kept small, resulting in a highly descriptive, albeit noisy, saliency representation. By joint upsampling this noisy representation with the original input image, the proposed method computes an accurate, full-resolution, saliency map. The experimental results indicate that the method is highly competitive with the state-of-the-art in terms of accuracy and execution time on the MSRA1K, MSRA10K and SED2 datasets, achieving remarkable trade-off. In particular, using the adaptive threshold by Achanta et al. (2009), it achieves an F-measure of 0.7933, 0.6956 and 0.7235 on the MSRA1K, MSRA10K and SED2 datasets, respectively, with an average execution time of 0.0638 seconds per 400×300 image.

The parameter assessment showed that it is possible to compute accurate saliency maps with very few distance computations per pixel, and that this can be made in a heavily downsampled input image. It was shown that computing the saliency of each pixel as its color distance to 10 randomly sampled pixels, with the input image downsized to 20% of its original size, resulted in the best trade-off between accuracy and execution time. Computation with less downsampling did not significantly improve accuracy – in some cases it even decreased it. Moreover, an assessment of different boundary ratios indicated that boundary prior always improves accuracy. In the experiments, a boundary ratio of 20% resulted in an accuracy increase of approximately 10%.

Future work includes further investigation of extensions to the random color distance map. Boundary prior was trivially incorporated into the proposed method, significantly improving accuracy at practically no additional computational cost. Additional priors, such as spatial distribution (Liu et al., 2007) and objectness (Alexe et al., 2010), might be incorporated to further improve accuracy. The model might also be adapted for additional features besides color, providing a compact and efficient representation for applications requiring fast computation, which might be joint upsampled if high accuracy is required.

Acknowledgements

The authors acknowledge the Brazilian Coordination for the Improvement of Higher Education Personnel (CAPES) for the financial support of this work.

References

Achanta, R., Hemami, S., Estrada, F., Süsstrunk, S., 2009. Frequency-tuned Salient Region Detection, in: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, pp. 1597 – 1604. doi:10.1109/CVPR.2009.5206596.

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S., 2012. SLIC Superpixels Compared to State-of-the-Art Superpixel Methods. IEEE Transactions on Pattern Analysis and Machine Intelligence 34, 2274–2282. doi:10.1109/TPAMI.2012.120.

Alexe, B., Deselaers, T., Ferrari, V., 2010. What is an Object?, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 73–80. doi:10.1109/CVPR.2010.5540226.

Alpert, S., Galun, M., Brandt, A., Basri, R., 2012. Image Segmentation by Probabilistic Bottom-Up Aggregation and Cue Integration. IEEE Transactions on Pattern Analysis and Machine Intelligence 34, 315–327. doi:10.1109/TPAMI.2011.130.

Borji, A., Cheng, M.M., Jiang, H., Li, J., 2014. Salient Object Detection: A Survey. arXiv preprint arXiv:1411.5878 .

Borji, A., Cheng, M.M., Jiang, H., Li, J., 2015. Salient Object Detection: A Benchmark. IEEE Transactions on Image Processing 24, 5706–5722. doi:10.1109/TIP.2015.2487833.

Braun, J., Julesz, B., 1998. Withdrawing Attention at Little or No Cost: Detection and Discrimination Tasks. Perception & Psychophysics 60, 1–23. doi:10.3758/BF03211915.

Cheng, M.M., Mitra, N.J., Huang, X., Torr, P.H.S., Hu, S.M., 2015. Global Contrast Based Salient Region Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 37, 569–582. doi:10.1109/TPAMI.2014.2345401.

Cheng, M.M., Warrell, J., Lin, W.Y., Zheng, S., Vineet, V., Crook, N., 2013. Efficient Salient Region Detection with Soft Image Abstraction, in: Proceedings of the IEEE International Conference on Computer Vision, pp. 1529–1536. doi:10.1109/ICCV.2013.193.

Christoudias, C.M., Georgescu, B., Meer, P., 2002. Synergism in Low Level Vision, in: Proceedings of the International Conference on Pattern Recognition, pp. 150–155. doi:10.1109/ICPR.2002.1047421.

Comaniciu, D., Meer, P., 2002. Mean Shift: a Robust Approach Toward Feature Space Analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence 24, 603–619. doi:10.1109/34.1000236.

Elazary, L., Itti, L., 2008. Interesting Objects are Visually Salient. Journal of Vision 8, 3. doi:10.1167/8.3.3.

Felzenszwalb, P.F., Huttenlocher, D.P., 2004. Efficient Graph-Based Image Segmentation. International Journal of Computer Vision 59, 167–181. doi:10.1023/B:VISI.0000022288.19776.77.

Frintrop, S., 2006. VOCUS: A Visual Attention System for Object Detection and Goal-Directed Search (Lecture Notes in Computer Science / Lecture Notes in Artificial Intelligence). Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Gastal, E.S.L., Oliveira, M.M., 2011. Domain Transform for Edge-Aware Image and Video Processing. ACM Transactions on Graphics 30, 69:1–69:12. doi:10.1145/2010324.1964964.

He, K., Sun, J., Tang, X., 2013. Guided Image Filtering. IEEE Transactions on Pattern Analysis and Machine Intelligence 35, 1397–1409. doi:10.1109/TPAMI.2012.213.

Hou, X., Zhang, L., 2007. Saliency Detection: A Spectral Residual Approach, in: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, pp. 1–8. doi:10.1109/CVPR.2007.383267.

Jiang, B., Zhang, L., Lu, H., Yang, C., Yang, M.H., 2013. Saliency Detection via Absorbing Markov Chain, in: Proceedings of the IEEE International Conference on Computer Vision, pp. 1665–1672. doi:10.1109/ICCV.2013.209.

Koch, C., Ullman, S., 1987. Shifts in Selective Visual Attention: Towards the Underlying Neural Circuitry, in: Vaina, L.M. (Ed.), Matters of Intelligence: Conceptual Structures in Cognitive Neuroscience. Springer Netherlands, Dordrecht, pp. 115–141. doi:10.1007/978-94-009-3833-5_5.

Kopf, J., Cohen, M.F., Lischinski, D., Uyttendaele, M., 2007. Joint Bilateral Upsampling. ACM Transactions on Graphics 26. doi:10.1145/1276377.1276497.

Li, X., Lu, H., Zhang, L., Ruan, X., Yang, M.H., 2013. Saliency Detection via Dense and Sparse Reconstruction, in: Proceedings of the IEEE International Conference on Computer Vision, Washington, DC, USA. pp. 2976–2983. doi:10.1109/ICCV.2013.370.

Lie, M.M.I., Borba, G.B., Vieira Neto, H., Gamba, H.R., 2016. Fast Saliency Detection Using Sparse Random Color Samples and Joint Upsampling, in: Proceedings of the SIBGRAPI Conference on Graphics, Patterns and Images, São José dos Campos, SP, Brazil. pp. 217–224. doi:10.1109/SIBGRAPI.2016.038.

Liu, T., Sun, J., Zheng, N.N., Tang, X., Shum, H.Y., 2007. Learning to Detect A Salient Object, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8. doi:10.1109/CVPR.2007.383047.

Marques, O., Mayron, L.M., Borba, G.B., Gamba, H.R., 2006. Using Visual Attention to Extract Regions of Interest in the Context of Image Retrieval, in: Proceedings of the Annual Southeast Regional Conference, New York, NY, USA. pp. 638–643. doi:10.1145/1185448.1185588.

Min, D., Choi, S., Lu, J., Ham, B., Sohn, K., Do, M.N., 2014. Fast Global Image Smoothing Based on Weighted Least Squares. IEEE Transactions on Image Processing 23, 5638–5653. doi:10.1109/TIP.2014.2366600.

Ouerhani, N., Bracamonte, J., Hügli, H., Ansorge, M., Pellandini, F., 2001. Adaptive Color Image Compression Based on Visual Attention, in: Proceedings of the International Conference on Image Analysis and Processing, pp. 416–421. doi:10.1109/ICIAP.2001.957045.

Reinhard, E., Khan, E.A., Ahmet, Akyüz, O., Johnson, G.M., 2008. Color Imaging: Fundamentals and Applications. AK Peters, Wellesley, MA.

Silva, G., Schnitman, L., Oliveira, L., 2014. Constraining Image Object Search by Multi-scale Spectral Residue Analysis. Pattern Recognition Letters 39, 31 – 38. doi:10.1016/j.patrec.2013.08.025.

Stentiford, F., 2007. Attention-based Similarity. Pattern Recognition 40, 771 – 783. doi:10.1016/j.patcog.2006.05.014.

Theeuwes, J., 2010. Top-down and Bottom-up Control of Visual Selection. Acta Psychologica 135, 77 – 99. doi:10.1016/j.actpsy.2010.02.006.

Tomasi, C., Manduchi, R., 1998. Bilateral Filtering for Gray and Color Images, in: Proceedings of the International Conference on Computer Vision, Washington, DC, USA. pp. 839–846. doi:10.1109/ICCV.1998.710815.

Tong, N., Lu, H., Zhang, L., Ruan, X., 2014. Saliency Detection with Multi-Scale Superpixels. IEEE Signal Processing Letters 21, 1035–1039. doi:10.1109/LSP.2014.2323407.

Treisman, A.M., Gelade, G., 1980. A Feature-integration Theory of Attention. Cognitive Psychology 12, 97 – 136. doi:10.1016/0010-0285(80)90005-5.

Vieira Neto, H., 2011. On-line Visual Novelty Detection in Autonomous Mobile Robots, in: Yokota, S., Chugo, D. (Eds.), Introduction to Modern Robotics. iConcept Press, Annerley, Australia, pp. 241–265.

Vikram, T.N., 2013. Random center-surround approaches for modeling visual saliency. Ph.D. thesis. Bielefeld University.

Vikram, T.N., Tscherepanow, M., Wrede, B., 2012. A Saliency Map Based on Sampling an Image Into Random Rectangular Regions of Interest. Pattern Recognition 45, 3114 – 3124. doi:10.1016/j.patcog.2012.02.009.

Wolfe, J.M., 1994. Guided Search 2.0 A Revised Model of Visual Search. Psychonomic Bulletin & Review 1, 202–238. doi:10.3758/BF03200774.

Zhai, Y., Shah, M., 2006. Visual Attention Detection in Video Sequences Using Spatiotemporal Cues, in: Proceedings of the ACM International Conference on Multimedia, New York, NY, USA. pp. 815–824. doi:10.1145/1180639.1180824.