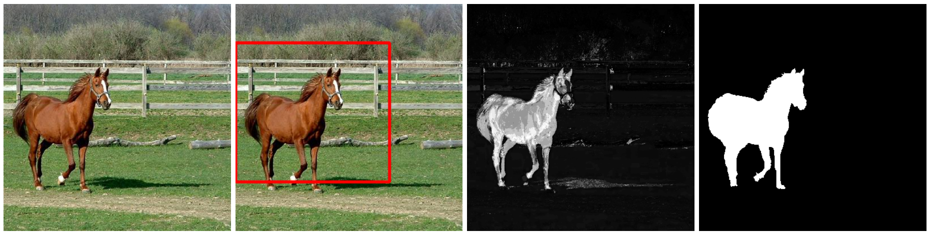
FASA: Fast, Accurate, and Size-Aware Salient Object Detection

Gökhan Yildirim, Sabine Süsstrunk

School of Computer and Communication Sciences
École Polytechnique Fédérale de Lausanne

Abstract. Fast and accurate salient-object detectors are important for various image processing and computer vision applications, such as adaptive compression and object segmentation. It is also desirable to have a detector that is aware of the position and the size of the salient objects. In this paper, we propose a salient-object detection method that is fast, accurate, and size-aware. For efficient computation, we quantize the image colors and estimate the spatial positions and sizes of the quantized colors. We then feed these values into a statistical model to obtain a probability of saliency. In order to estimate the final saliency, this probability is combined with a global color contrast measure. We test our method on two public datasets and show that our method significantly outperforms the fast state-of-the-art methods. In addition, it has comparable performance and is an order of magnitude faster than the accurate state-of-the-art methods. We exhibit the potential of our algorithm by processing a high-definition video in real time.

1 Introduction



(a) Original image (b) Position & size (c) Saliency map (d) Ground truth

Fig. 1. FASA processes (a) the 400 × 400 pixel image in 6 miliseconds and outputs (b) the parameters of rectangles that enclose the salient objects and (c) a saliency map, which is comparable to (d) the ground truth.

When we examine an image without specifying a task, our visual system attends mostly to the low-level distinctive or *salient* regions. Visual saliency can thus be defined as how much a certain image region visually stands out, compared to its surrounding area.

Low-level visual saliency deals with color and texture contrast, and their spatial constraints. The studies on these types of saliency-extraction techniques are, in general, inspired by the human visual system (HVS). A leading investigation in this field was done by Itti et al. [1], where, to generate a saliency map, center-to-surround differences in color, intensity, and orientations are combined at different scales with a non-maximum suppression.

One of the branches of visual saliency is salient-object detection. It is a popular and extensively studied topic in the computer vision community. It has the ability to mimic the human visual attention by *rapidly* finding important regions in an image. Therefore, similar to how the human brain operates [2], various applications, such as video compression [3], image retargeting [4], video retargeting [5, 6] and object detection [7], can allocate the processing resources according to saliency maps given by the detectors. As saliency detection is a pre-processing step, it should process the image in an efficient and accurate manner and provide as much information as possible for the successive step.

In this paper, we satisfy the efficiency, accuracy, and information criteria by introducing a **F**ast, **A**ccurate, and **S**ize-**A**ware (FASA) salient object detector. To achieve these goals, our method first performs a color quantization and forms a histogram in perceptually uniform CIEL*a*b* color space. It then uses the histogram to compute the spatial center and variance of the quantized colors via an efficient bilateral filtering approach. We show that the said variables are related to the position and the size of the salient object. Our method builds a probabilistic model of salient-object sizes and positions in an image. In order to compute the probability of saliency, the spatial center and variance of the colors are used in this model. The saliency probabilities are then multiplied with a global contrast value that is efficiently calculated using the quantized color differences. Finally, to obtain a full-resolution saliency map, the saliency values are linearly interpolated and are assigned to individual pixels using their quantization (histogram) bins.

On average, our method computes the saliency maps in 4.3 miliseconds (ms), when it uses the SED-100 [8] dataset and in 5.5 ms, when it uses the MSRA-1000 dataset [9]. On the same datasets, we show that our method is one order of magnitude faster than the accurate state-of-the-art methods and has a comparable performance in salient-object detection. Furthermore, it performs significantly better than other fast saliency-detection methods. In addition to fast computation and accurate saliency maps, our method also supplies the position and the size of the salient regions. We demonstrate the potential of our algorithm on a public high-definition (HD) video by detecting the saliency maps in *real time* (over 30 frames per second), as well as the position and the size of the salient object. In Figure 1, the outputs of our algorithm are illustrated with an example.

The outline of our paper is as follows: In Section 2, we summarize the recent research that is related to our method. In Section 3, we explain and discuss our efficient saliency detection method and its outputs. In Section 4, we present numerical and visual results of our method and compare it with other techniques on the MSRA-1000 [9] and SED-100 [8] datasets. In Section 5, we recapitulate the main points of our paper and explain possible future research directions.

2 Related Work

The main objective of a salient-object detector is to *rapidly* and *accurately* generate a pixel-precision map of visually distinctive objects and to provide additional information about them. We thus divide the previous studies on salient-object detection into two groups.

2.1 Fast Methods

One approach to accelerate saliency detection without introducing undesirable loss of accuracy is to quantize the intensity levels and/or compute the histogram of an image. In Zhai et al. [10], global color contrast is calculated using separate histograms for each channel. Whereas, in Cheng et al. [11], the color contrast is computed using a joint histogram. Both of the methods rely on global color contrast as the primary measure of visual saliency and they perform color quantization in sRGB color space. As sRGB is not perceptually uniform, histograms require a large number of quantization bins, otherwise they can suffer from large quantization errors. In order to avoid this, in our method, we perform the color quantization in CIEL*a*b* color space and, consequently, need fewer quantization bins. This provides faster, more accurate, and perceptually uniform color-contrast estimations.

Another approach for fast saliency estimation is to analyze the frequency content of the images with salient objects. Achanta et al. [9] show that visual distinctiveness is related to almost all frequency bands. Thus, saliency can be detected by simply computing the color contrast between image pixels and the average color of the scene. This corresponds to a band-pass filter. This method is limited to the images, where average scene color is sufficiently different from the color of the salient object. It does not suppress multi-colored backgrounds very well. Our modeling of the spatial center and variances of colors overcomes this limitation.

2.2 Accurate Methods

Other studies have focused on the accuracy of the saliency map and ignored the computational efficiency. In order to estimate the visual saliency, Perazzi et al. [12] fuse the spatial variance of a color of a superpixel and its contrast with its local surroundings. The study in [11] is extended in Cheng et al. [13], where a Gaussian mixture model with a spatial constraint is introduced for a spatially-aware (non-global) color quantization. This step is followed by a further color clustering for calculating the spatial variance and contrast of the colors in a scene. To achieve an initial foreground and background segregation, Yang et al. [14] form a superpixel graph and combine the saliency maps with different image boundary queries. They find the final saliency map by querying the initial foreground estimation.

Similar to ours, some of the techniques use the spatial variance of a color [12, 13]. They assume that the spatial variance of a salient object is smaller than the background. Therefore, they inversely correlate the variance and saliency, and introduce a bias towards detecting smaller objects as more salient. In Section 3.2 and 3.5, we discuss this relationship and introduce a statistical model based on salient-object position and size to correctly address the problem.

Salient-object detection is a pre-processing step. In order to properly allocate the processing resources for higher-level tasks, such as object detection and recognition, the detectors should rapidly detect the salient regions. Therefore, we propose that the efficiency of salient object detectors is as important as their accuracy.

3 Our Method

Our saliency-detection method, FASA, combines a probability of saliency with a global contrast map. Figure 2 provides a scheme illustrating our method. For computational efficiency, our algorithm first quantizes an image to reduce the number of colors. Then, in order to estimate the position and the size of the salient object, the spatial center and variances of the quantized colors are calculated. These values are put in an object model to compute the probability of saliency. The same quantized colors are used to generate global contrast values as well. Finally, the saliency probabilities of the colors and the contrast values are fused into a single saliency map.

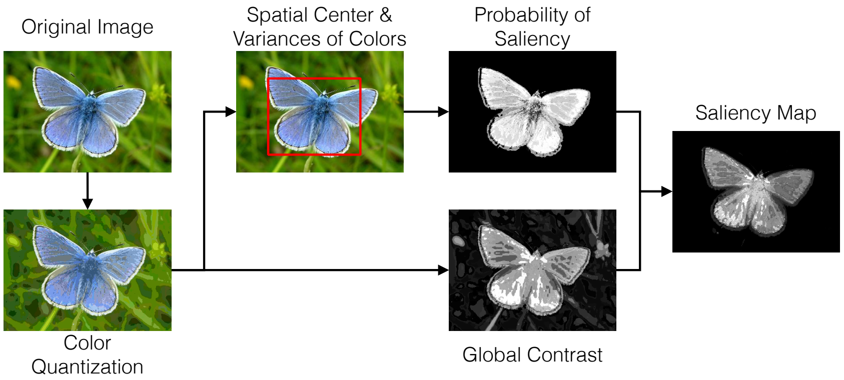


Fig. 2. Scheme of our method.

3.1 Spatial Center and Variances of a Color

One of the prominent components of visual saliency is the spatial variance of a color in a scene [12, 13]. In order to compute it, we first define a position and a color vector notation.

$$\mathbf{p}_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix}, \mathbf{C}_i = \begin{bmatrix} L^*(\mathbf{p}_i) \\ a^*(\mathbf{p}_i) \\ b^*(\mathbf{p}_i) \end{bmatrix} \tag{1}$$

Here, $\mathbf{p}_i$ is the position vector, which represents the coordinates (x_i, y_i) of the i^{th} pixel. $\mathbf{C}_i$ is the color vector of the pixel at position $\mathbf{p}_i$ in CIEL*a*b* color space.

The spatial center $\{m_x(\mathbf{p}_i), m_y(\mathbf{p}_i)\}$ and the horizontal and vertical variances $\{V_x(\mathbf{p}_i), V_y(\mathbf{p}_i)\}$ of a color can be calculated using the following equation:

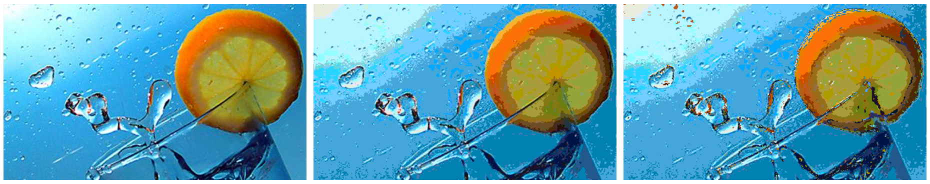
$$\begin{aligned} m_x(\mathbf{p}_i) &= \frac{\sum_{j=1}^{N} w^c\big(\mathbf{C}_i, \mathbf{C}_j\big) \cdot x_j}{\sum_{j=1}^{N} w^c\big(\mathbf{C}_i, \mathbf{C}_j\big)} \\ V_x(\mathbf{p}_i) &= \frac{\sum_{j=1}^{N} w^c\big(\mathbf{C}_i, \mathbf{C}_j\big) \cdot \big(x_j - m_x(\mathbf{p}_i)\big)^2}{\sum_{j=1}^{N} w^c\big(\mathbf{C}_i, \mathbf{C}_j\big)} \end{aligned} \tag{2}$$

Similar calculations can be done for y dimension. Here, N is the total number of pixels in an image, and $w^c\big(\mathbf{C}_i, \mathbf{C}_j\big)$ are the color weights and are calculated using a Gaussian function.

$$w^c\big(\mathbf{C}_i, \mathbf{C}_j\big) = e^{-\frac{||\mathbf{C}_i - \mathbf{C}_j||^2}{2\sigma_c^2}} \tag{3}$$

Here, σ_c is a parameter to adjust the effect of the color difference. If we look at (2), we can notice that w^c in both of the equations depends on the spatial coordinates. These calculations correspond to a bilateral filter with a color kernel, namely $w^c\big(\mathbf{C}_i, \mathbf{C}_j\big)$. For computational efficiency, the spatial kernel (or support) is chosen to be the whole image, which turns our algorithm into a global saliency-detection method.

The computational complexity of (2) is $O(N^2)$. Here, for efficient bilateral filtering, we follow the approach proposed by Yang et al. [15], in which they quantize the intensity levels of a grayscale image. In this paper, the colors $\mathbf{C}_i$ of an image are quantized (*i.e.*, a color histogram is created) into a set of colors $\{\mathbf{Q}_k\}_{k=1}^{K}$, where K is the number of colors after the quantization. In practice, we can minimize K by assigning certain quantized colors that have very few pixels to the perceptually closest quantized color with a non-zero number of pixels. A similar color quantization in sRGB color space is performed in Cheng et al. [11]. However, we quantize the image in perceptually uniform CIEL*a*b* color space and thus need fewer quantization bins. An example of the color quantization is given in Figure 3.



(a) Original Image (b) 175 quantized colors (c) 50 quantized colors

Fig. 3. The L*a*b* histogram (8 bins in each channel, $8^3 = 512$ bins in total) of (a) the original image contains (b) 175 quantized colors with non-zero histogram bins and (c) 50 quantized colors that can cover 95% of the image pixels.

In our paper, $\mathbf{C}_i \rightarrow \mathbf{Q}_k$ indicates that the color of the pixel at $\mathbf{p}_i$ falls to the k^{th} color histogram bin after the quantization. If we quickly calculate the color histogram of the image and precompute $w^c\big(\mathbf{Q}_k, \mathbf{Q}_j\big)$, we can efficiently estimate the spatial center and variances of the quantized colors as follows:

$$\begin{aligned} m'_{xk} &= \frac{\sum_{j=1}^{K} w^c\big(\mathbf{Q}_k, \mathbf{Q}_j\big) \cdot \sum_{\forall x_i | \mathbf{C}_i \rightarrow \mathbf{Q}_j} x_i}{\sum_{j=1}^{K} h_j \cdot w^c\big(\mathbf{Q}_k, \mathbf{Q}_j\big)} \\ V'_{xk} &= \frac{\sum_{j=1}^{K} w^c\big(\mathbf{Q}_k, \mathbf{Q}_j\big) \cdot \sum_{\forall x_i | \mathbf{C}_i \rightarrow \mathbf{Q}_j} \big(x_i - m'_{xk}\big)^2}{\sum_{j=1}^{K} h_j \cdot w^c\big(\mathbf{Q}_k, \mathbf{Q}_j\big)} \end{aligned} \tag{4}$$

Similar calculations can be performed for y dimension. Here, $\{m'_{xk}, m'_{yk}\}$ is the spatial center and $\{V'_{xk}, V'_{yk}\}$ are the spatial variances of the k^{th} quantized color. $h_k = \big|\forall x_i | \mathbf{C}_i \rightarrow \mathbf{Q}_k\big|$ is the number of pixels in the k^{th} color histogram bin. The spatial center and variances at each pixel in (2) can be estimated as follows:

$$\begin{aligned} m_x(\mathbf{p}_i) &\approx m'_{xk} \quad \forall \mathbf{p}_i | \mathbf{C}_i \rightarrow \mathbf{Q}_k \\ V_x(\mathbf{p}_i) &\approx V'_{xk} \quad \forall \mathbf{p}_i | \mathbf{C}_i \rightarrow \mathbf{Q}_k \end{aligned} \tag{5}$$

Similar calculations can be performed for y dimension. We reduce the complexity of the bilateral filtering in (2) to $O(K^2)$ via the color quantization in (4). In addition, as explained in Section 3.2, $\{m'_{xk}, m'_{yk}\}$ and $\{V'_{xk}, V'_{yk}\}$ provide valuable position and size cues about the salient object.

3.2 The Center and the Size of a Salient Object

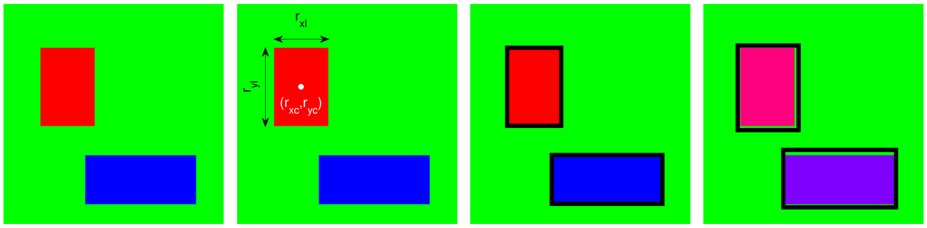
The spatial center $\{m'_{xk}, m'_{yk}\}$ shows the color-weighted center of mass of k^{th} quantized color of the image. The spatial variances $\{V'_{xk}, V'_{yk}\}$ depict how spatially distributed the same quantized color is within the image. In addition, it also gives us an idea about the "size" of that color. In order to show this relationship, in Figure 4(a), we illustrate a test image of size 256 × 256 pixels that includes a red and a blue rectangle.

In this image, we have three dominant colors, i.e. $k \in \{red, green, blue\}$. As there is sufficient global color contrast between these colors, we can assume $w^c\big(\mathbf{Q}_k, \mathbf{Q}_j\big) \approx 0$ for $k \neq j$ and we know that $w^c\big(\mathbf{Q}_k, \mathbf{Q}_k\big) = 1$. By using this, we can rewrite (4) and estimate the center of the objects as follows:

$$m'_{x,rect} \approx \frac{1}{h_{rect}} \sum_{\forall x_i | \mathbf{C}_i \rightarrow \mathbf{Q}_{rect}} x_i \quad = r_{xc} \tag{6}$$

Here r_{xc} is the x coordinate of the center of the red rectangle. As rectangles are symmetrical in both horizontal and vertical dimensions, we can easily compute the center of the red rectangle (r_{xc}, r_{yc}) using (6). The size of an object can be calculated as follows:

$$V'_{x,rect} \approx \frac{1}{h_{rect}} \sum_{\forall x_i | \mathbf{C}_i \rightarrow \mathbf{Q}_{rect}} (x_i - r_{xc})^2 \approx \frac{r_{yl} \cdot \int_{-r_{xl}/2}^{r_{xl}/2} x^2 \cdot dx}{r_{yl} \cdot r_{xl}} = \frac{r_{xl}^2}{12} \tag{7}$$



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

Fig. 4. (a) A test image with two salient rectangles with (b) the center and size parameters of the red rectangle. (c) The estimated position and sizes are shown with black bounding rectangle. (d) The accuracy of the center and the size estimation degrades, when the color of the objects are similar.

Here, r_{xl} and r_{yl} are the width and the height of the red rectangle, respectively. Similar equations can be derived for the y dimension. As we can see from (6) and (7), given sufficient color contrast, we are able to estimate the center and the size of both rectangles and ellipses, which is illustrated with black boundaries in Figure 4(c).

Conventionally, a bounding rectangle is used to represent a detected object. However, in some cases, it could be useful to represent the objects using a bounding ellipse instead. The central position of a bounding ellipse can be computed using (6). To estimate the dimensions of an ellipse, we slightly modify (7):

$$V'_{x,ellipse} \approx \frac{\pi/4 \cdot e_{yl} \cdot \int_{-e_{xl}/2}^{e_{xl}/2} x^2 \sqrt{1 - \frac{x^2}{(e_{xl}/2)^2}} \cdot dx}{\pi^2/16 \cdot e_{yl} \cdot e_{xl}} = \frac{e_{xl}^2}{16} \qquad (8)$$

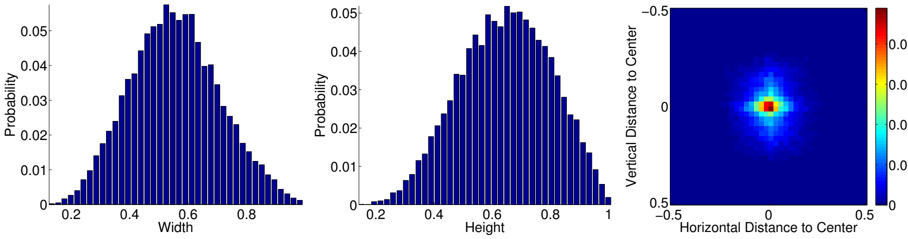
Here, e_{xl} and e_{yl} are the width and the height of an ellipse, respectively. The equation for estimating the height is similar to (8).

Natural images often contain non-rectangular objects and the color of the objects might interfere with each other as shown in Figure 4(d). However, the spatial center and variances still give us an idea about the position and the size of an object (or background), so that we can better calculate the saliency value. Moreover, this additional information is beneficial for object detection applications, as demonstrated in Section 4.4.

3.3 Computing Probability of Saliency

The salient objects tend to be smaller then their surrounding background. As they do not calculate the position and the size of an object, Perazzi et al. [12] and Cheng et al. [13] favor small spatial variations by using an inverting function to map the spatial variance to visual saliency. This creates a bias towards smaller objects.

In our method, we estimate the position and the size of the salient object, thus we can statistically model a mapping from these variables to a saliency probability. To generate our model, we use the MSRA-A dataset [16] that includes over 20'000 images with salient objects and their enclosing rectangles marked by three persons. The MSRA-1000 dataset is derived from the MSRA-A dataset. Therefore, to generate unbiased statistics, we exclude the images of the MSRA-1000 from the MSRA-A dataset. In Figure 5, we illustrate the probability distributions in terms of the width and the height of the salient objects, as well as their distance to the image center.



(a) (b) (c)

Fig. 5. Distributions of object (a) width (b) height, and (c) distance to image center in the MSRA-A dataset [16] based on the ground truth rectangles. All values are normalized by using the image dimensions.

We can see in Figure 5 that all probability distributions resemble a Gaussian distribution. Therefore, we model their joint distribution with a multivariate Gaussian function given as follows:

$$\begin{aligned} P(\mathbf{p}_i) &= \frac{1}{(2\pi)^2\sqrt{|\boldsymbol{\Sigma}|}} \exp\Big(-\frac{(\mathbf{g}_i-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{g}_i-\boldsymbol{\mu})}{2}\Big) \\ \mathbf{g}_i &= \left[\begin{matrix} \frac{\sqrt{12\cdot V_x(\mathbf{p}_i)}}{n_w} & \frac{\sqrt{12\cdot V_y(\mathbf{p}_i)}}{n_h} & \frac{m_x(\mathbf{p}_i)-n_w/2}{n_w} & \frac{m_y(\mathbf{p}_i)-n_h/2}{n_h} \end{matrix}\right]^T \end{aligned} \tag{9}$$

Here, $P\big(\mathbf{p}_i\big)$ is the probability of saliency of an input image with dimensions n_w and n_h. Note that the factor 12 in $\mathbf{g}_i$ comes from (7).

The mean vector and the covariance matrix of the joint Gaussian model that is illustrated in Figure 5 are given as follows:

$$\boldsymbol{\mu} = \begin{bmatrix} 0.5555 \\ 0.6449 \\ 0.0002 \\ 0.0063 \end{bmatrix}, \quad \boldsymbol{\Sigma} = \begin{bmatrix} 0.0231 & -0.0010 & 0.0001 & -0.0002 \\ -0.0010 & 0.0246 & -0.0000 & 0.0000 \\ 0.0001 & -0.0000 & 0.0115 & 0.0003 \\ -0.0002 & 0.0000 & 0.0003 & 0.0080 \end{bmatrix} \tag{10}$$

If we analyze $\boldsymbol{\mu}$ in (10), we can see that the average height is larger than the average width. This could be due to a tendency of the photographers to take landscape photographs over portraits, in order to emphasize salient objects. In addition, the average position is very close to the image center, thus validating the well-known center-bias phenomenon [17].

3.4 Global Contrast

High color-contrast is widely used as a measure of saliency [1, 9–14]. Once we have the quantized colors and the color differences $w^c(\mathbf{Q}_k, \mathbf{Q}_j)$, we can easily compute the global contrast of each quantized color as follows:

$$R(\mathbf{p}_i) = \sum_{j=1}^{K} h_j \cdot ||\mathbf{Q}_k - \mathbf{Q}_j||_2, \quad \forall \mathbf{p}_i | \mathbf{C}_i \rightarrow \mathbf{Q}_k \tag{11}$$

Here, h_j is the number of pixels in the j^{th} histogram bin and $\mathbf{Q}_j$ is the quantized color that corresponds to that bin. State-of-the-art methods, such as FT [9], LC [10], and HC [11], rely only on global color contrast. In order to generate a final saliency map, our method combines global color contrast with the probability of saliency.

3.5 Computing the Final Saliency Map

In order to combine the probability of saliency and the global contrast into a single saliency map, we use the following approach:

$$S(\mathbf{p}_i) = \frac{\sum_{j=1}^{K} w^c(\mathbf{Q}_k, \mathbf{Q}_j) \cdot P(\mathbf{p}_i) \cdot R(\mathbf{p}_i)}{\sum_{j=1}^{K} w^c(\mathbf{Q}_k, \mathbf{Q}_j)}, \quad \forall \mathbf{p}_i | \mathbf{C}_i \rightarrow \mathbf{Q}_k \tag{12}$$

Here, $S(\mathbf{p}_i)$ is the saliency value of the pixel at $\mathbf{p}_i$. All of the computations for $P(\mathbf{p}_i)$, $R(\mathbf{p}_i)$, and $S(\mathbf{p}_i)$ can be done using quantized colors. Therefore, our implementation performs the calculations by using K colors and assigns the corresponding saliency values to individual pixels based on their color quantization bins. The final computational complexity of our method is $O(N)+O(K^2)$, where $O(N)$ comes from the histogram computation and $O(K^2)$ comes from the bilateral filtering and other quantization related computations. A color weighting is used in (12) for smoother saliency values. After computing the final saliency map, we normalize the map between 0 and 1.

4 Experiments

In this section, to show the most recent state-of-the-art performance on the MSRA-1000 [9] and the SED-100 [8] datasets, we compare our **FASA** method to fast saliency detection methods, such as FT [9], LC [10], HC [11], as well as accurate methods, such as GC [13], SF [12], RC [11], and GMR [14]. FASA performs efficient and image-dependent quantization in CIEL*a*b* color space and benefits from a statistical object model. Therefore, it is one order of magnitude faster than accurate methods while significantly outperforming other fast techniques.

4.1 Experimental Setup

Given an input image, we quantize its colors in CIEL*a*b* color space. We use 8 bins in each channel. In order to minimize the quantization errors, instead of using the full range of CIEL*a*b* (for example $L^* \in [0, 100]$), we divide the range, defined by the minimum and the maximum value of each channel, into 8 bins. This gives us a histogram of size $8^3 = 512$. We further reduce the number of colors to a set of colors that represents 95% of the image pixels. We reassign the excluded non-zero color histogram bins to the perceptually closest bins. There are, on average, 55 and 60 colors per image in the MSRA-1000 and the SED-100 datasets, respectively. We use $\sigma_c = 16$ to calculate the color weights $w^c(\mathbf{Q}_k, \mathbf{Q}_j)$. For all methods, we run the corresponding author's implementations using an Intel Core i7 2.3GHz with 16GB RAM.

4.2 Performance Comparisons

The color quantization step, prior to the bilateral filtering, greatly reduces the computational complexity of our method while still retaining the saliency accuracy. Our algorithm estimates the visual saliency of an image in the MSRA-1000 and the SED-100 datasets in, on average, 5.5 and 4.3 ms, respectively. The comparison of execution times is given in Table 1. Note that the most time consuming step (superpixel segmentation) in GMR is implemented in C++ and it processes the MSRA-1000 images in approximately 200 ms, on average.

Table 1. Average computation time (in miliseconds) for the MSRA-1000 (12×10^4 pixels per image) and the SED-100 (8.7×10^4 pixels per image) datasets.

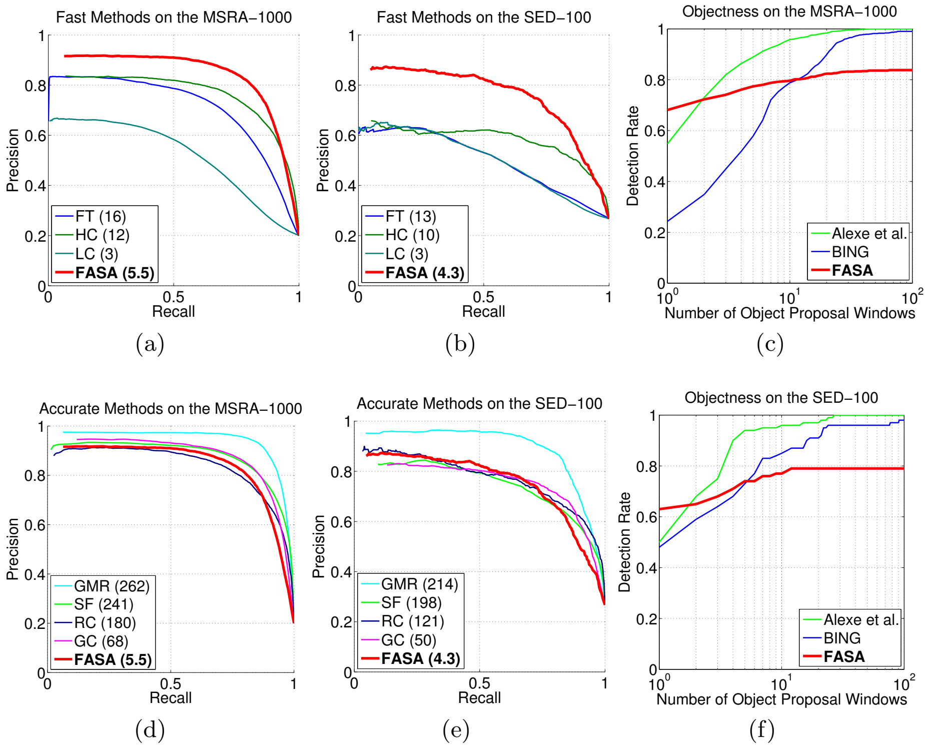
	Accurate				**Fast**			
	GMR [14]	SF [12]	RC [11]	GC [13]	FT [9]	HC [11]	LC [10]	**FASA**
MSRA-1000	262	241	180	68	16	12	3	**5.5**
SED-100	214	198	121	50	13	10	3	**4.3**
Code	Matlab/C++	C++	C++	C++	C++	C++	C++	C++

To reduce the computation time, the methods LC, HC, RC, and GC execute a color quantization step that is similar to ours. They quantize the colors in sRGB space by using either 255 bins (LC, independent histograms for each channel) or 12 bins (HC, RC, GC) per channel. Instead, we perform the quantization in the perceptually more uniform CIEL*a*b* color space and adjust the histogram using the minimum and the maximum values of L*a*b* channels of the processed image. Consequently, we need only 8 bins per channel and obtain a better and faster representation of the image.

In Figure 6, we compare the precision and recall curves of our method with the other methods on the MSRA-1000 and SED-100 datasets. As it can be seen from Figure 6(a) and 6(b), our algorithm significantly outperforms the fast methods, such as FT, LC, and HC, because they only take the color contrast (global and/or local) into account. The local saliency-detection methods, such as

SF and GC, also use the spatial variance as a part of their saliency computation. However, as stated before, they directly favor smaller objects by assuming an inverse correlation between the spatial variance and the object saliency. We use, instead, the position and the size information in a statistical model and, even though our method is global, we achieve an accuracy comparable to SF, GC, and RC and we are one order of magnitude faster.

As we state in Section 2.2, saliency detection is a pre-processing step for successive applications. Therefore, the speed of a salient-object detector can be as essential as its accuracy. In this paper, we focus on the efficiency and present a method that is 50 times faster than GMR, which is one of the most accurate state-of-the-art salient-object detectors.



(a) (b) (c)

(d) (e) (f)

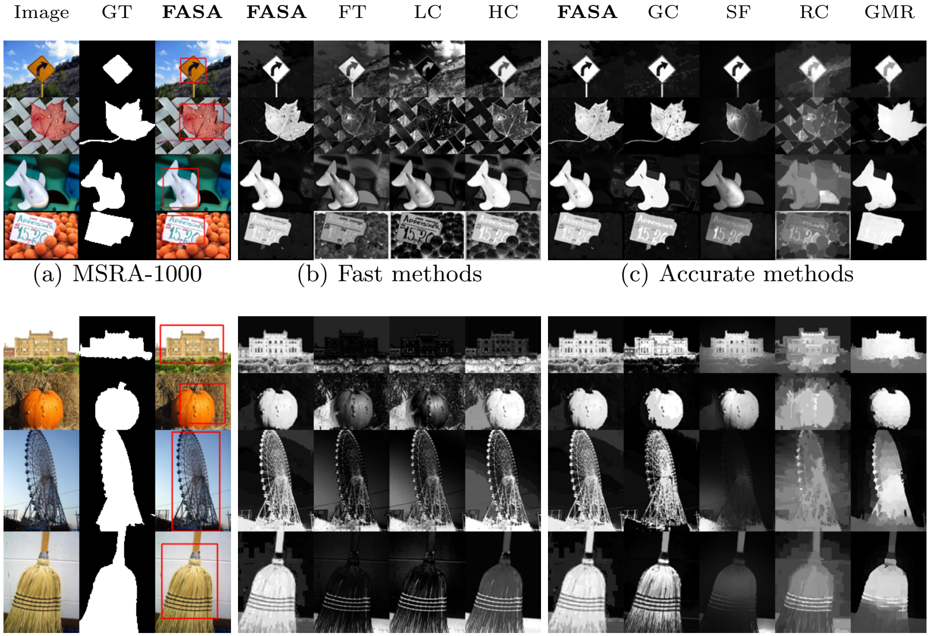
Fig. 6. Salient object detection performance of our method are compared to fast and accurate methods in (a, b, d, e). The numbers in parentheses indicate the execution times of the corresponding methods. The objectness detection rate of our method is compared to other methods in (c) and (e).

Our method outputs additional position and size information about salient objects, which can be considered as an "objectness" measure. Therefore, we compare the object-detection capabilities of FASA to well-known objectness measuring methods, such as Alexe et al. [18] and BING [19]. The object-detection rate (probability of detecting an object) versus the number of object proposal windows for the MSRA-1000 and the SED-100 datasets are illustrated in Figure 6(c) and Figure 6(f), respectively. Our method is more accurate if we just consider the first proposal window. This is logical as our method focuses on (and is optimized for) estimating the salient objects and provides object center and size as additional information. This property can be helpful to provide single and accurate proposals for object detection and tracking.

4.3 Visual Comparisons

We visually compare the saliency maps generated by different methods in Figure 7. Due to the accurate quantization in CIEL*a*b* color space, our saliency maps are more uniform than the maps of LC and HC. Moreover, the background suppression of the probability of saliency is better than FT. Compared to the other fast methods, FASA generates visually better results.

Our maps are visually comparable to the maps of the accurate methods, such as RC, GC, SF, and even GMR. Considering that FASA is one order of magnitude faster than these methods, our method may be preferable for time-critical applications.



(a) MSRA-1000 (b) Fast methods (c) Accurate methods

(d) SED-100 (e) Fast methods (f) Accurate methods

Fig. 7. Four example images from the MSRA-1000 and the SED-100 datasets. (a,d) FASA estimates the position and the size of the objects (red rectangles). (b,e) Our saliency maps are better than the maps of the fast methods and (c,f) they are visually comparable to the maps of the accurate state-of-the-art methods (GT: Ground Truth).

4.4 Application to Videos

Our method is fast enough to use it as a real-time salient-object detector in videos. Furthermore, it provides the position and size information for the salient and non-salient parts of the image. This property can be used in applications such as object tracking in videos [20, 21].

In order to demonstrate the potential of FASA, we estimate, using different resolutions, the saliency of the publicly available video “Big Buck Bunny”[1]. We are able to process the HD version (1280 × 720) of the video with a speed of 30 frames per second (fps). Given that the frame-rate of the video is 24 fps, our method estimates the saliency map and the center and the size of the objects in real time. The fps values under different resolutions are given in Table 2. The fps value linearly changes with the number of pixels in a single video frame. In other words, the number of pixels our method can process in a second ($N\times$ fps) is largely independent from the resolution of the image. This shows that our method has a computational complexity of $O(N) + O(K^2) \approx O(N)$, where K is the number of colors after quantization and $K^2 \ll N$. Note that our method is optimized to perform saliency detection in images and individually processes each video frame.

Table 2. Average processing speed of FASA in frames per second (fps) for different resolutions of the video “Big Buck Bunny”.

		Resolution		
		1920 × 1080	1280 × 720	854 × 480
fps	Frames per second	13.7	**30.7**	66.5
N	Number of megapixels	2.07	0.92	0.41
$N\times$ fps	Number of megapixels per second	28.4	28.3	27.2

For visual results, in Figure 8, we illustrate 10 frames from the same video. We encircle the most salient regions (saliency > 0.75 after normalization between 0 and 1) using their estimated positions and sizes, and we display the corresponding saliency maps. Due to its global nature, our method is accurate in scenes with a single salient object or multiple salient objects with different colors. It has a limited performance when color interference or a complex scene is present, as shown in the last two frames in Figure 8.

5 Conclusion

In this paper, we introduce a salient-object detection method that quantizes colors in perceptually uniform CIEL*a*b* space and combines two components for a fast and accurate estimation of saliency. The first component deals with the spatial center and variances of the quantized colors via an efficient bilateral

[1] http://www.bigbuckbunny.org/index.php/download/

filtering. This component also produces a probability of saliency based on a statistical object model that is extracted from the MSRA-A dataset [16]. The center and variance values are shown to be related to the position and the size of the salient objects. The second component computes the global color contrast. The final saliency map is calculated by linearly interpolating the product of the two components. Our method performs significantly better than other fast state-of-the-art methods and it is an order of magnitude faster than the methods with similar precision-recall performances on the MSRA-1000 [9] and SED-100 [8] datasets. We also present the potential of our method by generating saliency maps and by computing the position and the size of salient objects in 30 fps on an HD video.

For computational efficiency, our algorithm globally detects the visual saliency. This can create limitations in scenes with multiple salient objects and multi-colored salient objects, because colors interfere with each other during the estimation of saliency. In order to generate more accurate saliency maps, we can enforce spatial constraints to our color histogram through bilateral filtering or to our contrast computation, without significantly sacrificing the execution time. Another future research direction is to calculate temporally-aware color histograms for video processing.

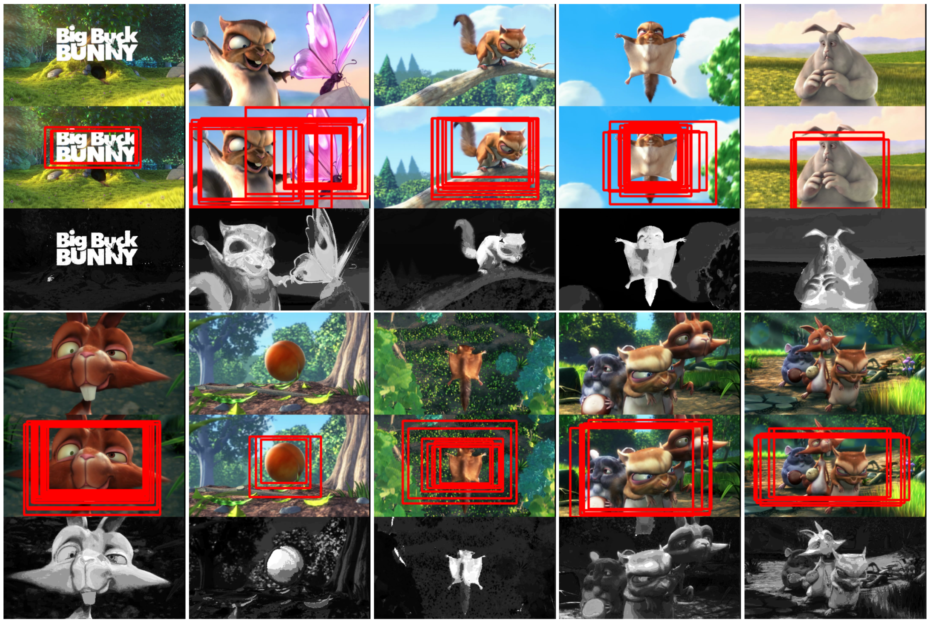


Fig. 8. 10 frames from the video "Big Buck Bunny". The first row shows the original frame, the second row position and the size of the most salient objects, and the third row the saliency map of the frame.

Acknowledgement. This work was supported by the Swiss National Science Foundation under grant number 200021-143406 / 1.

References

1. Itti, L., Koch, C., Niebur, E.: A model of saliency-based visual attention for rapid scene analysis. IEEE Transactions on PAMI **20** (1998) 1254–1259
2. Anderson, J.R.: Cognitive psychology and its implications. 5th edn. (2000)
3. Itti, L.: Automatic foveation for video compression using a neurobiological model of visual attention. IEEE Transactions on Image Processing **13** (2004) 1304 –1318
4. Achanta, R., Süsstrunk, S.: Saliency detection for content-aware image resizing. In: Proceedings of IEEE ICIP. (2009) 1001–1004
5. Xiang, Y., Kankanhalli, M.S.: Video retargeting for aesthetic enhancement. In: Proceedings of ACM Multimedia. (2010) 919–922
6. Wolf, L., Guttmann, M., Cohen-Or, D.: Non-homogeneous content-driven video-retargeting. In: Proceedings of IEEE ICCV. (2007) 1–6
7. Oliva, A., Torralba, A., Castelhano, M., Henderson, J.: Top-down control of visual attention in object detection. In: Proceedings of IEEE ICIP. Volume 1. (2003) 253–256
8. Alpert, S., Galun, M., Basri, R., Brandt, A.: Image segmentation by probabilistic bottom-up aggregation and cue integration. In: Proceedings of IEEE CVPR. (2007) 1–8
9. Achanta, R., Hemami, S., Estrada, F., Süsstrunk, S.: Frequency-tuned salient region detection. In: Proceedings of IEEE CVPR. (2009) 1597 – 1604
10. Zhai, Y., Shah, M.: Visual attention detection in video sequences using spatiotemporal cues. In: Proceedings of ACM Multimedia. (2006) 815–824
11. Cheng, M., Zhang, G., Mitra, N.J., Huang, X., Hu, S.: Global contrast based salient region detection. In: Proceedings of IEEE CVPR. (2011) 409–416
12. Perazzi, F., Krahenbuhl, P., Pritch, Y., Hornung, A.: Saliency filters: Contrast based filtering for salient region detection. In: Proceedings of IEEE CVPR. (2012) 733–740
13. Cheng, M.M., Warrell, J., Lin, W.Y., Zheng, S., Vineet, V., Crook, N.: Efficient salient region detection with soft image abstraction. In: IEEE ICCV. (2013) –
14. Yang, C., Zhang, L., Lu, H., Ruan, X., Yang, M.: Saliency detection via graph-based manifold ranking. In: Proceedings of IEEE CVPR. (2013) 3166–3173
15. Yang, Q., Tan, K., Ahuja, N.: Real-time O(1) bilateral filtering. In: Proceedings of IEEE CVPR. (2009) 557–564
16. Liu, T., Yuan, Z., Sun, J., Wang, J., Zheng, N., Tang, X., Shum, H.: Learning to detect a salient object. IEEE Transactions on PAMI **33** (2011) 353–367
17. Borji, A., Sihite, D.N., Itti, L.: Salient object detection: A benchmark. In: Proceedings of the ECCV. (2012) 414–429
18. Alexe, B., Deselaers, T., Ferrari, V.: Measuring the objectness of image windows. IEEE Transactions on PAMI **34** (2012) 2189–2202
19. Cheng, M.M., Zhang, Z., Lin, W.Y., Torr, P.H.S.: BING: Binarized normed gradients for objectness estimation at 300fps. In: Proceedings of IEEE CVPR. (2014)
20. Zia, K., Balch, T., Dellaert, F.: MCMC-based particle filtering for tracking a variable number of interacting targets. IEEE Transactions on PAMI **27** (2005) 1805–1819
21. Stalder, S., Grabner, H., Van Gool, L.: Cascaded confidence filtering for improved tracking-by-detection. In: Proceedings of ECCV. (2010) 369–382