# Minimum Barrier Salient Object Detection at 80 FPS

Jianming Zhang[1]     Stan Sclaroff[1]     Zhe Lin[2]     Xiaohui Shen[2]     Brian Price[2]     Radomír Měch[2]

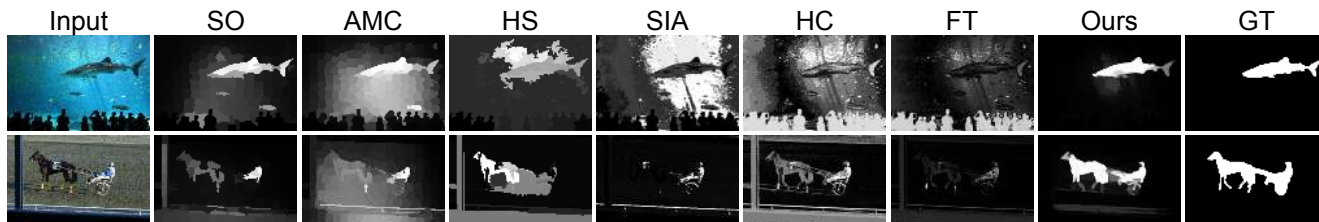[1]Boston University          [2]Adobe Research

Figure 1: Sample saliency maps of several state-of-the-art methods (SO [39], AMC [15], HS [34] and SIA [6]) and methods with fast speed (HC [5], FT [1] and ours). Our method runs at about 80 FPS using a single thread, and produces saliency maps of high quality. Previous methods with similar speed, like HC and FT, usually cannot handle complex images well.

## Abstract

*We propose a highly efficient, yet powerful, salient object detection method based on the Minimum Barrier Distance (MBD) Transform. The MBD transform is robust to pixel-value fluctuation, and thus can be effectively applied on raw pixels without region abstraction. We present an approximate MBD transform algorithm with 100X speedup over the exact algorithm. An error bound analysis is also provided. Powered by this fast MBD transform algorithm, the proposed salient object detection method runs at 80 FPS, and significantly outperforms previous methods with similar speed on four large benchmark datasets, and achieves comparable or better performance than state-of-the-art methods. Furthermore, a technique based on color whitening is proposed to extend our method to leverage the appearance-based backgroundness cue. This extended version further improves the performance, while still being one order of magnitude faster than all the other leading methods.*

## 1. Introduction

The goal of salient object detection is to compute a saliency map that highlights the salient objects and suppresses the background in a scene. Recently, this problem has received a lot of research interest owing to its usefulness in many computer vision applications, *e.g.* object detection, action recognition, and various image/video processing applications. Due to the emerging applications on mobile de-

vices and large scale datasets, a desirable salient object detection method should not only output high quality saliency maps, but should also be highly computationally efficient. In this paper, we address both the quality and speed requirements for salient object detection.

The *Image Boundary Connectivity Cue*, which assumes that background regions are usually connected to the image borders, is shown to be effective for salient object detection [39, 33, 36, 35]. To leverage this cue, previous methods, geodesic-distance-based [33, 39] or diffusion-based [15, 35], rely on a region abstraction step to extract superpixels. The superpixel representation helps remove irrelevant images details, and/or makes these models computationally feasible. However, this region abstraction step also becomes a speed bottleneck for this type of methods.

To boost the speed, we propose a method to exploit the image boundary connectivity cue without region abstraction. We use the *Minimum Barrier Distance* (MBD) [30, 8] to measure a pixel's connectivity to the image boundary. Compared with the widely used geodesic distance, the MBD is much more robust to pixel value fluctuation. In contrast, the geodesic distance transform often produces a rather fuzzy central area when applied on raw pixels, due to the small-weight-accumulation problem observed in [33].

Since the exact algorithm for the MBD transform is not very efficient, we present FastMBD, a fast raster-scanning algorithm for the MBD transform, which provides a good approximation of the MBD transform in milliseconds, being two orders of magnitude faster than the exact algorithm [8]. Due to the *non-smoothness* property [11] of MBD, er-

ror bound analysis of this kind of Dijkstra-like algorithm was previously regarded as difficult [8]. In this work, to the best of our knowledge, we present the first error bound analysis of a Dijkstra-like algorithm for the MBD transform.

The proposed salient object detection method runs at about 80 FPS using a single thread, and achieves comparable or better performance than the leading methods on four benchmark datasets. Compared with methods with similar speed, our method gives significantly better performance. Some sample saliency maps are shown in Fig. 1.

The main contributions of this paper are twofold:

1. We present `FastMBD`, a fast iterative MBD transform algorithm that is 100X faster than the exact algorithm, together with a theoretic error bound analysis.

2. We propose a fast salient object detection algorithm based on the MBD transform, which achieves state-of-the-art performance at a substantially reduced computational cost.

In addition, we provide an extension of the proposed method to leverage the appearance-based backgroundness cue [16, 23, 19]. This extension uses a simple and effective color space whitening technique, and it further improves the performance of our method, while still being at least one order of magnitude faster than all the other leading methods.

## 2. Related Work

Previous works in saliency detection indicate that saliency emerges from several closely related concepts such as rarity, uniqueness and local/global contrast [17, 14, 4, 2]. While saliency detection methods are often optimized for eye fixation prediction, salient object detection aims at uniformly highlighting the salient regions with well defined boundaries. Therefore, many salient object detection methods combine the contrast/uniqueness cue with other higher level priors [5, 24, 34, 26, 6], *e.g.* region uniformity, spatial compactness and centeredness.

The image boundary prior has been used for salient object detection, assuming that most image boundary regions are background. Salient regions can be inferred by their dissimilarity in appearance [19, 16, 23], or their connectivity [39, 33, 36] with respect to the image boundary regions.

Some saliency detection methods use a diffusion-based formulation to propagate the saliency values [21, 23, 35]. Other methods formulate the problem based on low rank matrix recovery theory [29] and Markov random walks [13].

Some recent methods [16, 20, 23, 38] have attained superior performance by using machine learning techniques. These methods require off-line training and complex feature extraction, *e.g.* region-level features via object proposal generation and Convolutional Neural Networks. These

methods are often much slower than the training-free methods. Thus, they are not directly comparable with training-free methods using simple features.

A few attempts have been made to speed up salient object detection. One approach is to leverage the pixel-level color contrast [1, 5], which only requires simple computation of some global color statistics like the color mean or color histogram. Speedup can also be obtained by down-sampling the input image before processing [33, 37], but this will significantly affect the quality of the saliency maps.

The MBD transform introduced in [30, 8] is shown to be advantageous in seeded image segmentation. An exact MBD transform algorithm and two approximations are presented in [30, 8]. However, the past papers [30, 8] did not propose a raster-scanning algorithm to make the MBD transform practical for fast salient object detection. Moreover, to our knowledge, we provide in this paper the first error bound analysis of the Dijkstra-like upper-bound approximation algorithms for the MBD transform.

## 3. Fast Approximate MBD Transform

In this section, we present `FastMBD`, a fast raster scanning algorithm for the MBD transform, together with a new theoretic error bound result, which we believe should be useful beyond the application of salient object detection, *e.g.* in image/video segmentation and object proposal [18].

### 3.1. Background: Distance Transform

The image distance transform aims at computing a distance map with respect to a set of background seed pixels. As a very powerful tool for geometric analysis of images, it has been a long-studied topic in computer vision [27].

Formally, we consider a 2-D single-channel digital image $\mathcal{I}$. A path $\pi = \langle \pi(0), \cdots, \pi(k) \rangle$ on image $\mathcal{I}$ is a sequence of pixels where consecutive pairs of pixels are adjacent. In this paper, we consider 4-adjacent paths. Given a path cost function $\mathcal{F}$ and a seed set $S$, the distance transform problem entails computing a distance map $\mathcal{D}$, such that for each pixel $t$

$$\mathcal{D}(t) = \min_{\pi \in \Pi_{S,t}} \mathcal{F}(\pi), \qquad (1)$$

where $\Pi_{S,t}$ is the set of all paths that connect a seed pixel in $S$ and $t$.

The definition of the path cost function $\mathcal{F}$ is application dependent. In [33, 39], the geodesic distance is used for salient object detection. Given a single-channel image $\mathcal{I}$, the geodesic path cost function $\Sigma_{\mathcal{I}}$ is defined as

$$\Sigma_{\mathcal{I}}(\pi) = \sum_{i=1}^{k} |\mathcal{I}(\pi(i-1)) - \mathcal{I}(\pi(i))|. \qquad (2)$$

where $\mathcal{I}(\cdot)$ denotes the pixel value. Recently, a new path

**Alg. 1** `FastMBD`
***

**input** : image $\mathcal{I} = (I, V)$, seed set $S$, number of passes $K$
**output** : MBD map $\mathcal{D}$
**auxiliaries** : $\mathcal{U}, \mathcal{L}$
***

set $\mathcal{D}(x)$ to 0 for $\forall x \in S$; otherwise, set $\mathcal{D}(x)$ to $\infty$.
set $\mathcal{L} \leftarrow \mathcal{I}$ and $\mathcal{U} \leftarrow \mathcal{I}$.
**for** $i = 1 : K$ **do**
 **if** $\mod (i, 2) = 1$ **then**
  RasterScan($\mathcal{D}, \mathcal{U}, \mathcal{L}; \mathcal{I}$).
 **else**
  InvRasterScan($\mathcal{D}, \mathcal{U}, \mathcal{L}; \mathcal{I}$).
***

**Alg. 2** `RasterScan`($\mathcal{D}, \mathcal{U}, \mathcal{L}; \mathcal{I}$)
***

**for** *each $x$, which is visited in a raster scan order* **do**
 **for** *each $y$ in the masked area for $x$* **do**
  compute $\beta_\mathcal{I}(\mathcal{P}_y(x))$ according to Eqn. 5.
  **if** $\beta_\mathcal{I}(\mathcal{P}_y(x)) < \mathcal{D}(x)$ **then**
   $\mathcal{D}(x) \leftarrow \beta_\mathcal{I}(\mathcal{P}_y(x))$.
   $\mathcal{U}(x) \leftarrow \max\{\mathcal{U}(y), \mathcal{I}(x)\}$.
   $\mathcal{L}(x) \leftarrow \min\{\mathcal{L}(y), \mathcal{I}(x)\}$.
***

cost function has been proposed in [30]:

$$\beta_\mathcal{I}(\pi) = \max_{i=0}^{k} \mathcal{I}(\pi(i)) - \min_{i=0}^{k} \mathcal{I}(\pi(i)). \quad (3)$$

The induced distance is called the *Minimum Barrier Distance*, and it is shown to be more robust to noise and blur than the geodesic distance for seeded image segmentation [30, 8]. However, the exact algorithm for the MBD transform takes time complexity of $O(mn \log n)$ [8], where $n$ is the number of pixels in the image and $m$ is the number of distinct pixel values the image contains. In practice, an optimized implementation for the exact MBD transform can take about half a second for a $300 \times 200$ image [8].

### 3.2. Fast MBD Transform by Raster Scan

Inspired by the fast geodesic distance transform using the raster scanning technique [10, 31], we propose `FastMBD`, an approximate iterative algorithm for the MBD transform. In practice, `FastMBD` usually outputs a satisfactory result in a few iterations (see Sec. 3.3), and thus it can be regarded as having linear complexity in the number of image pixels. Like all raster scan algorithms, it is also cache friendly, so it is highly efficient in practice.

Similar to the raster scan algorithm for the geodesic or Euclidean distance transform, during a pass, we need to visit each pixel $x$ in a raster scan or inverse raster scan order. Then each adjacent neighbor $y$ in the corresponding half of neighborhood of $x$ (see illustration in Fig. 2) will be used to iteratively minimize the path cost at $x$ by

$$\mathcal{D}(x) \leftarrow \min \begin{cases} \mathcal{D}(x) \\ \beta_\mathcal{I}(\mathcal{P}(y) \cdot \langle y, x \rangle) \end{cases}, \quad (4)$$
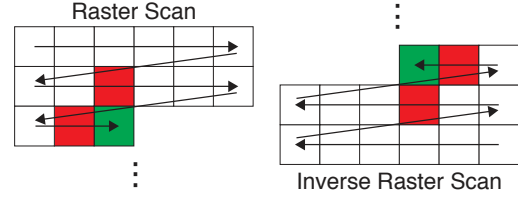


Figure 2: Illustration of the raster scan pass and the inverse raster scan pass. The green pixel is the currently visited pixel, and its masked neighbor area for 4-adjacency is shown in red.

where $\mathcal{P}(y)$ denotes the path currently assigned to the pixel $y$, $\langle y, x \rangle$ denotes the edge from $y$ to $x$, and $\mathcal{P}(y) \cdot \langle y, x \rangle$ is a path for $x$ that appends edge $\langle y, x \rangle$ to $\mathcal{P}(y)$.

Let $\mathcal{P}_y(x)$ denote $\mathcal{P}(y) \cdot \langle y, x \rangle$. Note that

$$\beta_\mathcal{I}(\mathcal{P}_y(x)) = \max\{\mathcal{U}(y), \mathcal{I}(x)\} - \min\{\mathcal{L}(y), \mathcal{I}(x)\}, \quad (5)$$

where $\mathcal{U}(y)$ and $\mathcal{L}(y)$ are the highest and the lowest pixel values on $\mathcal{P}(y)$ respectively. Therefore, the new MBD cost $\beta_\mathcal{I}(\mathcal{P}_y(x))$ can be computed efficiently by using two auxiliary maps $\mathcal{U}$ and $\mathcal{L}$ that keep track of the highest and the lowest values on the current path for each pixel.

Given the image $\mathcal{I}$ and the seed set $S$, the initialization of the distance map $\mathcal{D}$ and the auxiliary map $\mathcal{U}$ and $\mathcal{L}$ is described in Alg. 1. Then the two subroutines, a raster scan pass and an inverse raster scan pass, are applied alternately to update $\mathcal{D}$ and the auxiliary maps, until the required number of passes is reached (see Alg. 1). The subroutine for a raster scan is described in Alg. 2. An inverse raster scan pass is basically the same as Alg. 2, except that it enumerates the pixels in reverse order and uses different neighborhood masks, as illustrated in Fig. 2.

Each iteration of Alg. 2 updates $\mathcal{U}$ and $\mathcal{L}$ accordingly when path assignment changes. Thus, at any state of Alg. 1, $\mathcal{D}(x)$ is the MBD path cost of some path that connects the seed set $S$ and $x$. It follows that $\mathcal{D}(x)$ is an upper bound of the exact MBD of $x$ at any step. Alg. 1 will converge, since each pixel value of $\mathcal{D}$ is non-negative and non-increasing during update. The converged solution will also be an upper bound of the exact MBD for each pixel.

### 3.3. Approximation Error Analysis

The update rule of `FastMBD` (Eqn. 4) shares the same nature with Dijkstra's Algorithm for solving the shortest path problem. However, it is shown in [30] that the MBD transform cannot be exactly solved by Dijkstra-like Algorithms due to the *non-smoothness* property [11] of the MBD. Therefore, the converged solution of `FastMBD` generally does not equal the exact MBD transform. To facilitate discussion, we first introduce the following concept.

**Definition 1.** For an image $\mathcal{I}$, the maximum local differ-

ence $\varepsilon_\mathcal{I}$ is the maximum absolute pixel value difference between a pair of pixels that share an edge or a corner on $\mathcal{I}$.

For a lower-bound approximation algorithm to the MBD transform [30], it has been proved that the corresponding errors are bounded by $2\varepsilon_\mathcal{I}$ when the seed set is singleton [30, 8] or connected [37]. We remind the readers that $2\varepsilon_\mathcal{I}$ is a very loose bound, because for natural images, $\varepsilon_\mathcal{I}$ is usually above $127/255$. Nevertheless, such bounds can provide insight into the asymptotic behavior of an algorithm when an image approaches its continuous version, *e.g.* an *idealized image* in the continuous domain $\mathbb{R}^2$ [9], or a simple up-sampled version using bilinear interpolation.

The error-bound analysis techniques presented in previous works [30, 8, 37] cannot be applied on a Dijkstra-like algorithm for the MBD transform. In what follows, we show a non-trivial sufficient condition when the converged solution of `FastMBD` is exact. We first introduce a slightly modified version of `FastMBD`, denoted as `FastMBD*`, which is the same as `FastMBD` except that the input image first undergoes a discretization step. In the discretization step, we use a rounding function $G(v; \varepsilon_\mathcal{I}) = \left\lfloor \frac{v}{\varepsilon_\mathcal{I}} \right\rfloor \varepsilon_\mathcal{I}$ to map each pixel value $v$ to the largest integer multiples of $\varepsilon_\mathcal{I}$ below $v$. Then the discretized image $\widetilde{\mathcal{I}}$ is passed to Alg. 1 to obtain a distance map for the original image $\mathcal{I}$.

**Lemma 1.** *Given an image $\mathcal{I}$ and a seed set $S$, let $d_{\beta_\mathcal{I}}(x)$ denote the MBD from $S$ to the pixel $x$, and $\mathcal{D}$ denote the converged solution of `FastMBD*`. Assuming 4-adjacency, if the seed set $S$ is connected[1], then for each pixel $x$,*

$$|\mathcal{D}(x) - d_{\beta_\mathcal{I}}(x)| < \varepsilon_\mathcal{I}. \tag{6}$$

The proof of Lemma 1 is provided as supplementary material. The above error bound applies to a connected seed set, which is more general than the assumption of a single seed set in previous works [30, 8].

**Corollary 2.** *Let $\mathcal{I}$ be an image with integer pixel values. Assuming 4-adjacency, if the seed set is connected and $\varepsilon_\mathcal{I} = 1$, the converged solution of `FastMBD` is exact.*

*Proof.* When $\varepsilon_\mathcal{I} = 1$, `FastMBD` will be the same as `FastMBD*` since $\widetilde{\mathcal{I}} = \mathcal{I}$. According to Lemma 1, $\mathcal{D}(x)$ will equal $d_{\beta_\mathcal{I}}(x)$ because $|\mathcal{D}(x) - d_{\beta_\mathcal{I}}(x)|$ must be an integer and it is less than 1. □

The condition of $\varepsilon_\mathcal{I} = 1$ can be achieved by upsampling an integer-valued image by bilinear interpolation. Note that the MBD is quite robust to upsampling and blur [8], due to its formulation in Eqn. 3. Thus, Corollary 2 can be regarded as a theoretic guarantee that `FastMBD` is exact in the limit.

Aside from the worst-case error bounds, in practice, mean errors and the convergence rates are of more importance. Therefore, we test `FastMBD` on the PASCAL-S

---

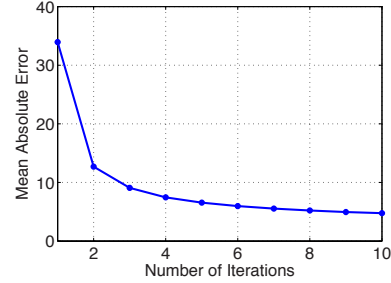[1] A set $S$ is connected if any pair of seeds are connected by a path in $S$.



Figure 3: Mean absolute distance approximation error against the number of iterations $\mathcal{K}$ in the presented fast algorithm for the MBD transform. The pixel values of the test images range between 0 to 255. The mean error drops below $10/255$ after three scan passes.

dataset [20] and set all of the image boundary pixels as seeds. We convert the input images to gray-scale, and compute the mean absolute approximation error of Alg. 1 w.r.t. the exact MBD transform. The result is shown in Fig. 3. The average mean error drops below $10/255$ after three scan passes (two forward and one backward passes), and each pass costs only about 2ms for a $320 \times 240$ image. The proposed `FastMBD` using three passes is about 100X faster than the exact algorithm proposed in [8], and over 30X faster than the fastest approximation algorithm proposed in [8]. In the application of salient object detection, there is no noticeable difference between `FastMBD` and the exact MBD transform in performance.

## 4. Minimum Barrier Salient Object Detection

In this section, we describe an implementation of a system for salient object detection that is based on `FastMBD`. Then an extension of our method is provided to further leverage the appearance-based backgroundness cue. Lastly, several efficient post-processing operations are introduced to finalize the salient map computation.

### 4.1. MBD Transform for Salient Object Detection

Similar to [33], to capture the image boundary connectivity cue, we set the pixels along the image boundary as the seeds, and compute the MBD transform for each color channel using `FastMBD`. Then the MBD maps for all color channels are pixel-wise added together to form a combined MBD map $\mathcal{B}$, whose pixel value are further scaled so that the maximum value is 1. We use three passes in `FastMBD`, as we find empirically that increasing the number of passes does not improve performance.

An example is given in Fig. 4 to illustrate why the geodesic distance is less favorable than the MBD in our case. We show the combined MBD map $\mathcal{B}$ (middle top) and the combined geodesic distance map (middle bottom). The computation of the geodesic distance map is the same as the
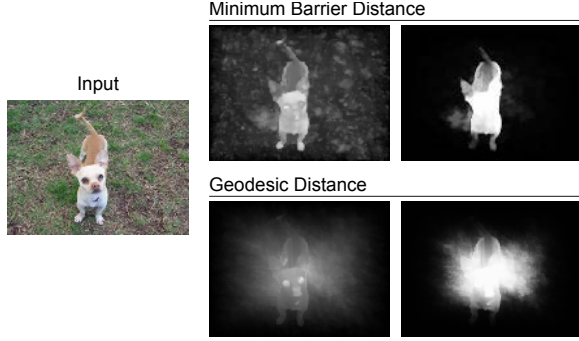
Figure 4: A test image is shown on the left. In the middle, the distance maps using the Minimum Barrier Distance (top) and geodesic distance (bottom) are displayed. The corresponding resultant final saliency maps are shown in the last column. The geodesic distance map has a fuzzy central area due to its sensitivity to the pixel value fluctuation.

MBD map, except that Eqn. 2 is used for the distance transform. Furthermore, as in [33], an adaptive edge weight clipping method is applied on the geodesic distance map to alleviate the *small-weight-accumulation* problem [33]. However, the geodesic distance map still has a rather fuzzy central area, due to the fact that we compute the distance transform on raw pixels instead of superpixels as in [33]. The MBD map does not suffer from this problem. As a result, the final saliency map (right top) using the MBD suppresses the central background area more effectively than using the geodesic distance (right bottom).

## 4.2. Combination with Backgroundness Cue

We provide an extension of the proposed method by integrating the appearance-based backgroundness cue [16], which assumes that background regions are likely to possess similar appearance to the image boundary regions. This appearance-based cue is more robust when the salient regions touch the image boundary, and it is complementary to the geometric cue captured by the MBD map $\mathcal{B}$. Instead of using various regional appearance features as in previous works [16, 23], we present a more efficient way to leverage this backgroundness cue using color space whitening.

We compute an *Image Boundary Contrast (IBC) Map* $\mathcal{U}$ to highlight regions with a high contrast in appearance against the image boundary regions. To do this, we consider four image boundary regions: 1) upper, 2) lower, 3) left and 4) right. Each region is $r$ pixels wide. For such a boundary region $k \in \{1, 2, 3, 4\}$, we calculate the mean color $\bar{\mathbf{x}}_k = [\bar{x}_1, \bar{x}_2, \bar{x}_3]$ and the color covariance matrix $\mathbf{Q}_k = [q_{ij}]_{3 \times 3}$ using the pixels inside this region. Then the corresponding intermediate IBC map $\mathbf{U}_k = [u_k^{ij}]_{W \times H}$ is computed based on the Mahalanobis distance from the mean color:

$$u_k^{ij} = \sqrt{\left(\mathbf{x}_k^{ij} - \bar{\mathbf{x}}\right) \mathbf{Q}^{-1} \left(\mathbf{x}_k^{ij} - \bar{\mathbf{x}}\right)^T}. \tag{7}$$

$\mathbf{U}_k$ is then normalized by $u_k^{ij} \leftarrow \frac{u_k^{ij}}{\max_{ij} u_k^{ij}}$, so that its pixel values lie in $[0, 1]$. The above formulation is equivalent to measuring the color difference in a whitened color space [28]. In a whitened color space, the Euclidean distance from the sample mean can better represent the distinctiveness of a pixel, because the coordinates of the whitened space are de-correlated and normalized.

Given the computed intermediate IBC maps $\{\mathbf{U}_k : k = 1, 2, 3, 4\}$ for the four image boundary regions, the final IBC map $\mathcal{U} = [u^{ij}]$ is computed by

$$u^{ij} = \left(\sum_{k=1}^{4} u_k^{ij}\right) - \max_k u_k^{ij}. \tag{8}$$

Compared with simply summing up all the intermediate IBC maps, the above formulation is more robust when one of the image boundary regions is mostly occupied by the foreground objects. Finally, we scale the values of $\mathcal{U}$ so that the maximum value is 1.

To integrate the IBC map $\mathcal{U}$ into our system, we pixel-wise add the MBD map $\mathcal{B}$ and the IBC map $\mathcal{U}$ together to form an enhanced map $\mathcal{B}^+ = \mathcal{B} + \mathcal{U}$. We find that although using $\mathcal{U}$ alone gives substantially worse performance than using $\mathcal{B}$, a simple linear combination of them consistently improves the overall performance.

## 4.3. Post-processing

We describe a series of efficient post-processing operations to enhance the quality of the final saliency map $\mathcal{S}$, given either $\mathcal{S} = \mathcal{B}$ or $\mathcal{S} = \mathcal{B}^+$. These operations do not add much computational burden, but can effectively enhance the performance for salient object segmentation.

Firstly, to smooth $\mathcal{S}$ while keeping the details of significant boundaries, we apply a morphological smoothing step on $\mathcal{S}$, which is composed of a reconstruction-by-dilation operation followed by a reconstruction-by-erosion [32]. The marker map for reconstruction by dilation (erosion) is obtained by eroding (dilating) the source image with a kernel of width $\delta$. To make the smoothing level scale with the size of the salient regions, $\delta$ is adaptively determined by

$$\delta = \alpha \sqrt{s}, \tag{9}$$

where $\alpha$ is a predefined constant, and $s$ is the mean pixel value on the map $\mathcal{B}$.

Secondly, similar to many previous methods [29, 12], to account for the center bias that is observed in many salient object detection datasets [3], we pixel-wise multiply $\mathcal{S}$ with a parameter-free centeredness map $\mathbf{C} = [c^{ij}]_{W \times H}$, which is defined as

$$c^{ij} = 1 - \frac{\sqrt{\left(i - \frac{H}{2}\right)^2 + \left(j - \frac{W}{2}\right)^2}}{\sqrt{\left(\frac{H}{2}\right)^2 + \left(\frac{W}{2}\right)^2}}. \tag{10}$$

Lastly, we scale the values of $\mathcal{S}$ so that its maximum value is 1, and we apply a contrast enhancement operation on $\mathcal{S}$, which increases the contrast between foreground and background regions using a sigmoid function:

$$f(x) = \frac{1}{1 + e^{-b(x-0.5)}}, \qquad (11)$$

where $b$ is a predefined parameter to control the level of contrast.

## 5. Experiments

**Implementation.** In our implementation, input images are first resized so that the maximum dimension is 300 pixels. We set $\alpha = 50$ in Eqn. 9, assuming the color values are in $[0, 1]$. We set $b = 10$ in Eqn. 11. For our extended version, the width $r$ of the border regions is set to 30. These parameters are fixed in the following experiments and we have found that, in practice, the performance of our algorithm is not sensitive to these parameter settings. An executable program of this implementation is available on our project website[2].

**Datasets.** To evaluate the proposed method, we use four large benchmark datasets: MSRA10K [22, 1, 5] (10000 images), DUTOmron [35] (5168 images), ECSSD [34] (1000 images) and PASCAL-S [20] (850 images). Among these, the PASCAL-S and DUTOmron datasets are the most challenging, and the PASCAL-S dataset is designed to avoid the dataset design bias. Note that MSRA10K is an extension of the ASD [1] and MSRA5K [22] datasets, which are widely used in the previous literature.

**Compared Methods.** We denote our method and the extended version as MB and MB+ respectively. MB only uses the MBD map $\mathcal{B}$, and MB+ uses $\mathcal{B}^+$ which integrates the appearance-based backgroundness cue. We compare our method with several recently published methods: SO [39], AMC [15], SIA [6], HSal [34], GS [33][3] and RC [5]. We also include several methods with an emphasis on the speed performance: HC [7] and FT [1].

To demonstrate the advantages of MBD over the geodesic distance, we also evaluate a baseline method, denoted as GD. GD is the same as MB but uses the geodesic distance (Eqn. 2) to compute the combined distance map $\mathcal{B}$ with the same post-processing applied. Adaptive edge weight clipping [33] is applied to alleviate the small-weight-accumulation problem. Parameters in the post-processing function are tuned to favor GD.

### 5.1. Speed Performance

The speed performance of the compared methods are reported in Fig. 5. FT, HC, SIA, RC and our methods are
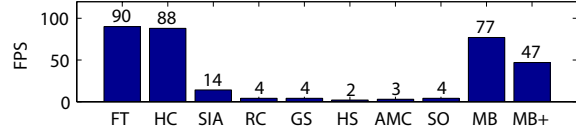
Figure 5: Speed performance. Our methods MB and MB+ run at 77 and 47 FPS respectively. While FT and HC are a bit faster, their accuracy is much lower (see Fig. 6 and 7).

implemented in C, and the rest use C and Matlab. The evaluation is conducted on a machine with 3.2GHz×2 CPU and 12GB RAM. We do not count I/O time, and do not allow processing multiple images in parallel. The test image size is the same as used in our methods (300 pixels in largest dimension) for all evaluated methods. Our method MB runs at about 80 FPS, which is comparable with the speed of FT and HC. Our extended version MB+ runs at 47 FPS, which is one order of magnitude faster than the state-of-the-art methods such as GS, HS, AMC, and SO.

### 5.2. Evaluation Using PR Curve

Similar to [1, 7, 15], we use Precision-Recall (PR) Curve to evaluate the overall performance of a method regarding its trade-off between the precision and recall rates. For a saliency map, we generate a set of binary images by thresholding at values in the range of $[0, 1]$ with a sample step 0.05, and compute the precision and recall rates for each binary image. On a dataset, an average PR curve is computed by averaging the precision and recall rates for different images at each threshold value.

In the top row of Fig. 6, we show the PR curves for our methods MB and MB+, the baseline GD and the methods with similar speed, FT and HC. MB outperforms GD, HC and FT with a considerable margin across all datasets. The extended version MB+ further improves the performance of MB. On the most challenging dataset PASCAL-S, using the backgroundness cue only slightly increases the performance of MB+ over MB. Note that in many images in PASCAL-S, the background is complex and the color contrast between the foreground and background is low.

In the bottom row of Fig. 6, we show the PR curves of our methods and the state-of-the-art methods. MB gives a better precision rate than SIA, RC and GS over a wide range of recall rate across all datasets. Note that GS is based on the same image boundary connectivity cue, but it uses the geodesic distance transform on superpixels. The superior performance of MB compared with GS further validates the advantage of using the MBD over the geodesic distance. Compared with HS, AMC and SO, MB achieves similar performance under the PR metric, while being over 25X faster. Our extended version MB+ consistently achieves state-of-the-art performance, and is over 10X faster than the other leading methods, such as HS, AMC and SO.
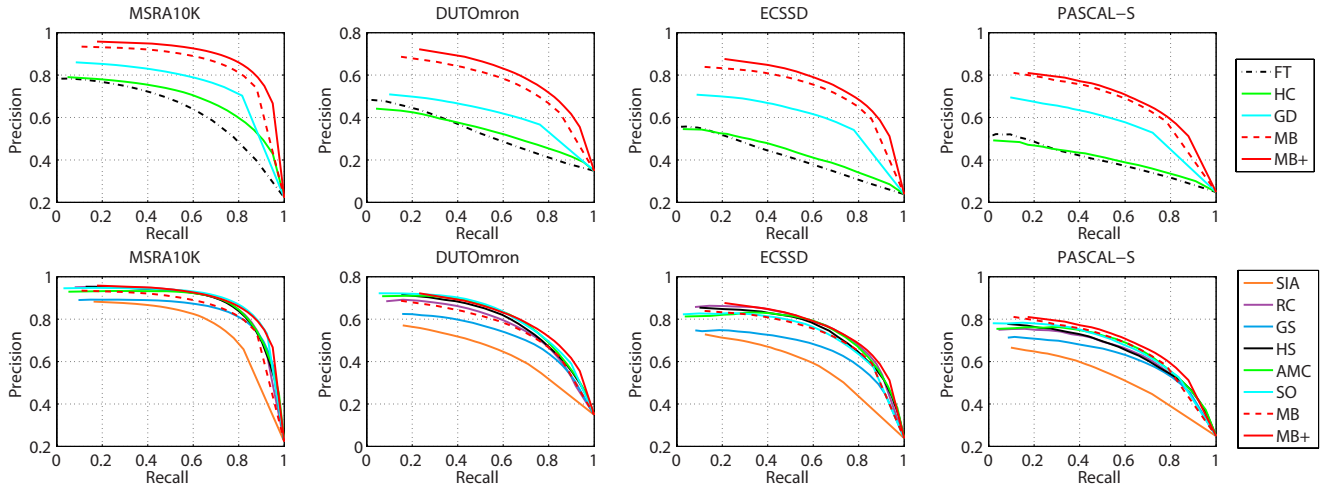
Figure 6: Precision-recall curves of the compared methods. Our methods MB and MB+ significantly outperform methods that offer similar speed across all datasets (top row), and achieve state-of-the-art performance (bottom row). The PR curves of the baseline GD using the geodesic distance are significantly worse than its MBD counterpart MB, validating the advantage of the MBD over the geodesic distance in our application.

## 5.3. Evaluation Using Weighted-$F_\beta$

To rank models, previous works use metrics like Area Under the Curve (AUC) [16, 23], Average Precision (AP) [23] and the $F_\beta$-measure [1, 7, 15]. However, as noted in [25], these metrics may not reliably evaluate the quality of a saliency map, due to the curve interpolation flaw, improper assumptions about the independence between pixels, and equal importance assignment to all errors. Therefore, we adopt the weighted-$F_\beta$ metric proposed in [25], which suffers less from the aforementioned problems. We use the code and the default setting provided by the authors of [25]. For more information about the weighted-$F_\beta$ metric, we refer the readers to [25]. We also provide the AUC and $F_\beta$ scores in our supplementary material.

The weighted-$F_\beta$ scores are shown in Fig. 7. MB achieves significantly better scores than the methods with similar speed (FT and HC), and it compares favorably with SIA, RC, GS, HS and AMC across all the datasets under the weighted-$F_\beta$ metric. MB gives similar scores as the leading method SO on the MSRA10K and DUTOmron datasets, and it attains the best scores on the ECSSD and PASCAL datasets. Our extended version MB+ further improves the scores of MB, and attains the top weighted-$F_\beta$ scores on all the datasets. The scores of the baseline method GD are substantially worse than those of MB, which is again consistent with our observation about the disadvantage of applying the geodesic distance on raw pixels.

To control the effect of post-processing on the ranking, we apply all possible combinations of the proposed post-processing steps for the other methods. The results are included in our supplementary material due to limited space.
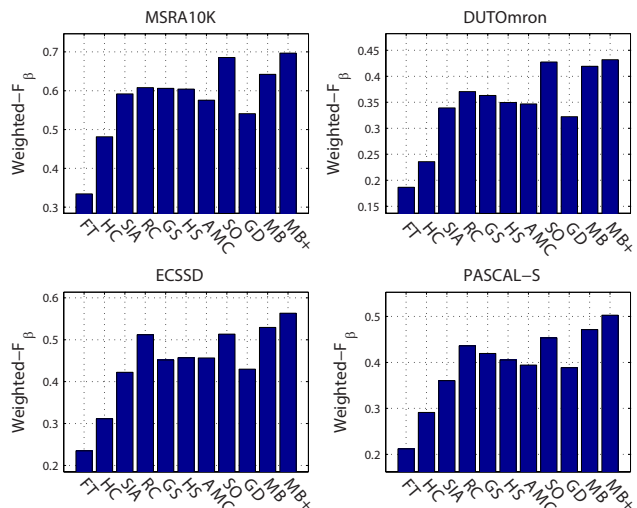


Figure 7: Weighted-$F_\beta$ scores of compared methods. Our methods MB and MB+ consistently attain comparable or better scores than the competitors.

We find that the proposed post-processing routine substantially improves the scores of FT and HC, but it does not significantly improve or even degrade the scores of the other compared models. Controlling for this factor does not lower the rankings of MB and MB+ on all the datasets.

Some sample saliency maps are shown in Fig. 8. Our methods MB and MB+ often give saliency maps with better visual quality than the other methods. The baseline GD tends to produce a rather fuzzy central area on the saliency map due to the small-weight-accumulation problem. More results are included in our supplementary material.
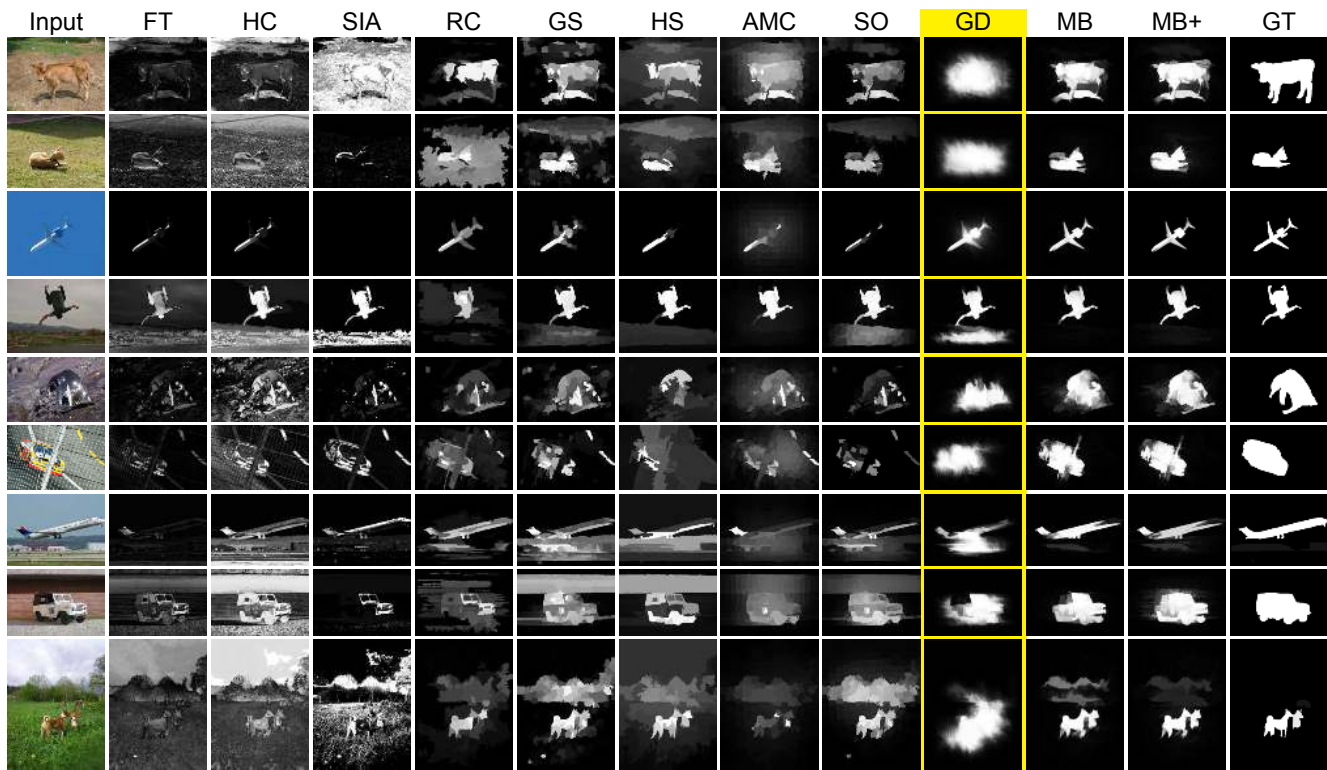
Figure 8: Sample saliency maps of the compared methods. The baseline using the geodesic distance (GD) often produces a rather fuzzy central area, while our methods based on MBD (MB and MB+) do not suffer from this problem.

## 5.4. Limitations

A key limitation of the image boundary connectivity cue is that it cannot handle salient objects that touch the image boundary. In Fig. 9, we show two typical examples of this case. Our method MB fails to highlight the salient regions that are connected to the image boundary, because it basically only depends on the image boundary connectivity cue. Our extended version MB+, which further leverages the appearance-based backgroundness prior, can help alleviate this issue if the foreground region has a high color contrast against the image boundary regions (see the top right image in Fig. 9). However, when such backgroundness prior does not hold, *e.g.* in the second test image in Fig. 9, MB+ cannot fully highlight the salient region, either.

## 6. Conclusion

In this paper, we presented `FastMBD`, a raster scanning algorithm to approximate the Minimum Barrier Distance (MBD) transform, which achieves state-of-the-art accuracy while being about 100X faster than the exact algorithm. A theoretical error bound result was shown to provide insight into the good performance of such Dijkstra-like algorithms. Based on `FastMBD`, we proposed a fast salient object detection method that runs at about 80 FPS. An extended ver-
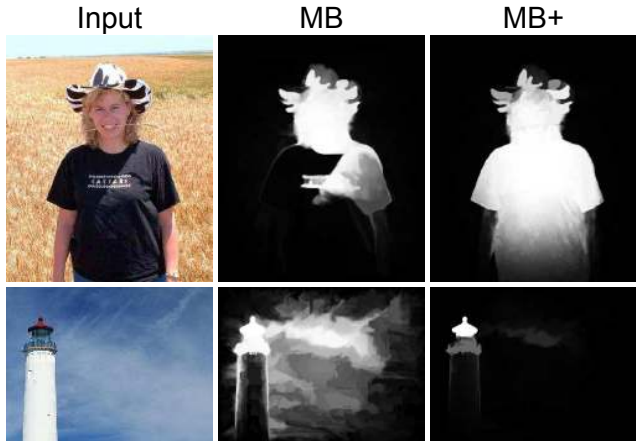


Figure 9: Some failure cases where the salient objects touch the image boundary.

sion of our method was also provided to further improve the performance. Evaluation was conducted on four benchmark datasets. Our method achieves state-of-the-art performance at a substantially smaller computational cost, and significantly outperforms the methods that offer similar speed.

# References

[1] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk. Frequency-tuned salient region detection. In *CVPR*, 2009.

[2] A. Borji and L. Itti. Exploiting local and global patch rarities for saliency detection. In *CVPR*, 2012.

[3] A. Borji, D. N. Sihite, and L. Itti. Salient object detection: A benchmark. In *ECCV*. 2012.

[4] N. Bruce and J. Tsotsos. Saliency based on information maximization. In *NIPS*, 2005.

[5] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. S. Torr, and S.-M. Hu. Global contrast based salient region detection. *TPAMI*, 37(3):569–582, 2015.

[6] M.-M. Cheng, J. Warrell, W.-Y. Lin, S. Zheng, V. Vineet, and N. Crook. Efficient salient region detection with soft image abstraction. In *CVPR*, 2013.

[7] M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu. Global contrast based salient region detection. In *CVPR*, 2011.

[8] K. C. Ciesielski, R. Strand, F. Malmberg, and P. K. Saha. Efficient algorithm for finding the exact minimum barrier distance. *Computer Vision and Image Understanding*, 123:53–64, 2014.

[9] K. C. Ciesielski and J. K. Udupa. A framework for comparing different image segmentation methods and its use in studying equivalences between level set and fuzzy connectedness frameworks. *Computer Vision and Image Understanding*, 115(6):721–734, 2011.

[10] P.-E. Danielsson. Euclidean distance mapping. *Computer Graphics and image processing*, 14(3):227–248, 1980.

[11] A. X. Falcão, J. Stolfi, and R. de Alencar Lotufo. The image foresting transform: Theory, algorithms, and applications. *TPAMI*, 26(1):19–29, 2004.

[12] S. Goferman, L. Zelnik-Manor, and A. Tal. Context-aware saliency detection. *TPAMI*, 34(10):1915–1926, 2012.

[13] V. Gopalakrishnan, Y. Hu, and D. Rajan. Random walks on graphs to model saliency in images. In *CVPR*, 2009.

[14] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *TPAMI*, 20(11):1254–1259, 1998.

[15] B. Jiang, L. Zhang, H. Lu, C. Yang, and M.-H. Yang. Saliency detection via absorbing markov chain. In *ICCV*, 2013.

[16] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li. Salient object detection: A discriminative regional feature integration approach. In *CVPR*. IEEE, 2013.

[17] C. Koch and S. Ullman. Shifts in selective visual attention: towards the underlying neural circuitry. In *Matters of Intelligence*, pages 115–141. Springer, 1987.

[18] P. Krähenbühl and V. Koltun. Geodesic object proposals. In *ECCV*, 2014.

[19] X. Li, H. Lu, L. Zhang, X. Ruan, and M.-H. Yang. Saliency detection via dense and sparse reconstruction. In *ICCV*. IEEE, 2013.

[20] Y. Li, X. Hou, C. Koch, J. Rehg, and A. Yuille. The secrets of salient object segmentation. In *CVPR*, 2014.

[21] R. Liu, J. Cao, Z. Lin, and S. Shan. Adaptive partial differential equation learning for visual saliency detection. In *CVPR*, 2014.

[22] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H.-Y. Shum. Learning to detect a salient object. *TPAMI*, 33(2):353–367, 2011.

[23] S. Lu, V. Mahadevan, and N. Vasconcelos. Learning optimal seeds for diffusion-based salient object detection. In *CVPR*, 2014.

[24] R. Margolin, A. Tal, and L. Zelnik-Manor. What makes a patch distinct? In *CVPR*, 2013.

[25] R. Margolin, L. Zelnik-Manor, and A. Tal. How to evaluate foreground maps? 2014.

[26] F. Perazzi, P. Krahenbuhl, Y. Pritch, and A. Hornung. Saliency filters: Contrast based filtering for salient region detection. In *CVPR*, 2012.

[27] A. Rosenfeld and J. L. Pfaltz. Distance functions on digital pictures. *Pattern recognition*, 1(1):33–61, 1968.

[28] R. Rosenholtz. Search asymmetries? what search asymmetries? *Perception & Psychophysics*, 63(3):476–489, 2001.

[29] X. Shen and Y. Wu. A unified approach to salient object detection via low rank matrix recovery. In *CVPR*, 2012.

[30] R. Strand, K. C. Ciesielski, F. Malmberg, and P. K. Saha. The minimum barrier distance. *Computer Vision and Image Understanding*, 117(4):429–437, 2013.

[31] P. J. Toivanen. New geodosic distance transforms for grayscale images. *Pattern Recognition Letters*, 17(5):437–450, 1996.

[32] L. Vincent. Morphological grayscale reconstruction in image analysis: applications and efficient algorithms. *TIP*, 2(2):176–201, 1993.

[33] Y. Wei, F. Wen, W. Zhu, and J. Sun. Geodesic saliency using background priors. In *ECCV*. 2012.

[34] Q. Yan, L. Xu, J. Shi, and J. Jia. Hierarchical saliency detection. In *CVPR*, 2013.

[35] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang. Saliency detection via graph-based manifold ranking. In *CVPR*. IEEE, 2013.

[36] J. Zhang and S. Sclaroff. Saliency detection: a Boolean map approach. In *ICCV*, 2013.

[37] J. Zhang and S. Sclaroff. Exploiting surroundedness for saliency detection: a Boolean map approach. *Accepted at TPAMI*, 2015.

[38] R. Zhao, W. Ouyang, H. Li, and X. Wang. Saliency detection by multi-context deep learning. In *CVPR*, 2015.

[39] W. Zhu, S. Liang, Y. Wei, and J. Sun. Saliency optimization from robust background detection. In *CVPR*, 2014.