Design of connected operators using the image foresting transform

Alexandre X. Falcão[a], Bruno S. da Cunha[a], and Roberto de A. Lotufo [b]

[a] Institute of Computing, University of Campinas, Campinas, SP, 13083-970, Brazil

[b] Faculty of Electrical Engineering, University of Campinas, Campinas, SP, 13081-970, Brazil

ABSTRACT

The Image Foresting Transform (IFT) reduces optimal image partition problems from seed pixels into a shortest-path forest problem in a graph, whose solution can be obtained in linear time. It has allowed a unified and efficient approach to edge tracking, region growing, watershed transforms, multiscale skeletonization, and Euclidean distance transform. In this paper, we extend the IFT to introduce two connected operators: cutting-off-domes and filling-up-basins. The former simplifies grayscale images by reducing the height of its domes, while the latter reduces the depth of its basins. By automatically or interactively specifying seed pixels in the image and computing a shortest-path forest, whose trees are rooted at these seeds, the IFT creates a simplified image where the brightness of each pixel is associated with the "length" of the corresponding shortest-path. A label assigned to each seed is propagated, resulting a labeled image that corresponds to the watershed partitioning from markers. The proposed operators may also be used to provide regional image filtering and labeling of connected components. We combine the cutting-off-domes and filling-up-basins to implement regional minima/maxima, h-domes/basins, opening/closing by reconstruction, leveling, area opening/closing, closing of holes, and removal of pikes. Their applications are illustrated with respect to medical image segmentation.

Keywords: morphological image reconstruction, leveling, watershed transform, interactive image filtering, image segmentation, openings and closings, graph algorithms, shortest-path forest.

1. INTRODUCTION

In image processing and analysis, many problems can be thought as an optimal image partition problem from seed pixels, where each seed defines an influence zone composed by its "closest" pixels. The image foresting transform (IFT) reduces such problems into a shortest-path forest problem in a graph, whose solution can be obtained in linear time.[1] The IFT allows an efficient and unified approach to edge tracking, region growing, watershed transforms, multiscale skeletonization, and Euclidean distance transform.[2–4] In this paper, we extend the IFT to introduce two connected operators: cutting-off-domes and filling-up-basins. The former simplifies grayscale images by reducing the height of its domes (bright regions), while the latter reduces the depth of its basins (dark regions).

Morphological grayscale reconstructions have been presented as image operators that take two input images, a mask and a marker, with the same domain where the mask is greater or equal to the marker (or vice-versa), and computes the "reconstruction" of the mask from the marker image.[5,6] Such a strategy allows to simplify grayscale images and to detect image features that can be used as markers in many segmentation and non-linear filtering tasks. Filters by reconstruction have been identified as belonging to a larger class called *connected operators* which interact with the image by means of its *flat zones* (the largest connected components where the image is constant).[7,8] Connected operators are reminded by the advantage of not creating false edges, a problem typically verified in linear image filtering. The proposed IFT-based connected operators have also the advantage of creating the watershed partitioning from markers[9] at same time the image is simplified.

The image is modeled as a weighted and oriented graph. Seed pixels are specified in the image, where each seed has assigned to it a label and a reference level (gray value). A shortest-path forest, whose trees are rooted at these seeds, is computed in this graph. The IFT creates a simplified image, where the brightness of each pixel is associated with the "length" of the corresponding shortest-path in the graph, and a labeled image which corresponds to the

Send correspondence to A.X.F., Av. Albert Einstein 1251, CEP 13083-970, Campinas, SP, Brasil, E-mail: afalcao@ic.unicamp.br, Phone: 55-19-37885881, Fax: 55-19-37885847.

Medical Imaging 2001: Image Processing, Milan Sonka, Kenneth M. Hanson, Editors,
Proceedings of SPIE Vol. 4322 (2001) © 2001 SPIE · 1605-7422/01/$15.00

watershed partitioning from markers, as thought the watershed transform were computed on the simplified image using the seed pixels as markers. That makes the watershed transform part of the connected operation and as far as we know this has never been proposed before.

Since the *IFT* is a recent subject, we start by presenting the principles underlying this technique. Subsequently, we present the cutting-off-domes and filling-up-basins operators, their respective linear-time algorithm, and show how their combination can lead to connected operations such as regional minima/maxima, h-domes/basins, opening/closing by reconstruction, leveling, area opening/closing, closing of holes, and removal of pikes.[5,6,10] The applications of the work are presented next with respect to medical image segmentation, and we finish the paper by stating our conclusions and directions for future research.

2. IMAGE FORESTING TRANSFORM

Consider an optimal image partition problem from seed pixels, where each seed should define a respective influence zone composed by its "closest" pixels. To solve this problem, the *IFT* maps the image into a graph, where the pixels are the nodes of the graph and the arcs are defined by an *adjacency relation* between distinct pixels. A *weight function* assigns a non-negative value to each arc representing its "length". The "distance" from a seed to a pixel is defined as the "length" of the shortest-path from the seed to the pixel in the graph. Such a "distance" is found by minimizing a *smooth path-function.* A smooth path-function is any non-negative function pf of the arc weights in the path that satisfies the following condition in the graph. Let $P \cdot \langle a \rangle$ denote the result of appending arc a to path P. For any nodes $u \neq w$ such that w is reachable from u, there must exist a shortest path P from u to w of the form $P = P' \cdot \langle a \rangle$, where $a = (v, w)$, such that:

(C1) $pf(P) \geq pf(P')$,

(C2) P' is a shortest path from u to v, and

(C3) for any shortest path P'' from u to v, $pf(P'' \cdot \langle a \rangle) = pf(P)$.

Note that, we may define directed graphs with asymmetric weight functions and path-functions. The choice of the adjacency relation, the weight function, and the path-function depends on the nature of the problem.

In addition to the parameters that define a graph, a label may also be assigned to each seed pixel. Thus, the *IFT* grows a shortest-path tree, simultaneously, from each seed in the graph, computing the following properties to each pixel in an *annotating image*: the label of its closest seed, the "distance" from its closest seed, and the parent pixel that leads it back to its closest seed. At least one of these properties should be relevant for a given problem. For example, region growing and multiscale skeletonization output the label image that represents the influence zones of all seeds in the optimal image partition. In the latter, a difference image is computed from the label image and multiscale one-pixel-wide connected Euclidean skeletons are created by thresholding the difference image.[2] The parent pixel has been used for edge tracking and geodesic path computation. Distance transforms use the "distance" image, which represents the distance map in the correspondent metric. In the next section, we show how this framework can be used to design connected operators.

3. CONNECTED OPERATORS USING THE *IFT*

A *flat zone* of a grayscale image I is a maximal connected component wherein all pixels have the same gray value. A grayscale operator $\mathcal{O}$ is *connected* if and only if any pair of pixels belonging to a given flat zone in I also belongs to a same flat zone in $\mathcal{O}(I)$.[8] There are many ways of designing connected operators according to this definition. In this section, we use the concept of threshold decomposition of a grayscale image to introduce two types of connected operators based on the *IFT*: *cutting-off-domes* and *filling-up-basins.* The cutting-off-domes operator simplifies grayscale images by reducing the height of its domes (bright regions), while the filling-up-basins operator reduces the depth of its basins (dark regions).

3.1. Cutting-off-domes and filling-up-basins operators

Given an image $I(p)$ which assigns an integer value to each pixel p in the interval $[0, K]$, a stack of binary images $T_k(p)$, $k = 0, 1, \ldots, K$, can be created by assigning value 1 (foreground) to the pixels p where $I(p) \geq k$, $k \in [0, K]$, and value 0 (background) to the remaining pixels. The cutting-off-domes and filling-up-basins operators basically remove/add connected components from/to the stack of binary images. For sake of simplicity, we illustrate the operations using the 1-D profile representation of a grayscale image.

Figure 1a shows an image with a selected seed pixel s at a gray level k_s. The cutting-off-domes operation consists of selecting all foreground connected components from the respective stack of binary images that contain s at levels $k \leq k_s$. This "region growing" process is also done from top to bottom along the stack of binary images. Such components are represented by dashed lines in Figure 1a and the simplified image is represented by the profile of the shaded area in the same figure. Note that this operation has removed all foreground connected components that do not contain s and those that contain s at levels $k > k_s$.

Figure 1b illustrates the filling-up-basins operator applied to the same image with a selected seed pixel s at a gray level k_s. The filling-up-basins operation consists of removing all background connected components (i.e. bringing them up to the foreground) from the stack of binary images that do not contain s and those that contain s at levels $k \leq k_s$. This "region growing" process is done from bottom to top along the stack of binary images. Such components are represented by dashed lines in Figure 1b and the simplified image is represented by the profile of the shaded area in the same figure. Note that this is the same as only background connected components from the stack of binary images that contain s at levels $k > k_s$ remain in the background.

Now suppose that two seed pixels, s and s', are specified at levels k_s and $k_{s'}$, respectively, as shown in Figure 1c, and the cutting-off-domes operation is computed. The seeds define influence zones, represented by the shaded areas in Figure 1c, that meet each other at level k, where the foreground connected component contains both seeds. The simplified image is represented by the profile of the union of the shaded areas in the same figure. This image partition aims at maximizing the influence zone of each seed. Observe that s and s' have been selected in two domes of the image that meet each other at level k, $k < k_s, k_{s'}$. It is important to notice that if s' were selected at level $k_{s'} < k$, then the influence zone of s would invade and eliminate the influence zone of s', because the process is done from top to bottom along the stack of binary images. In other words, s' would be ineffective for cutting off domes. A similar example is shown in Figure 1d with the filling-up-basins operation. The simplified image is represented by the profile of the union of the shaded areas in the same figure, where each shaded area indicates the influence zone of one seed. The seeds have been selected in two basins of the image that meet each other at level k, $k > k_s, k_{s'}$. Now if s' were selected at level $k_{s'} > k$, then the influence zone of s would invade and eliminate the influence zone of s', because the process is done from bottom to top along the stack of binary images.

The essential difference between the presented connected operators and morphological image reconstruction[6] is that they bring together both image simplification and partition tasks. It is important to notice that the image partition that results from the filling-up-basins operation corresponds to the watershed transform from markers[9] (seeds s and s') as if it were applied to the simplified image. The same observation is valid for the cutting-off-domes operation, where the image partition corresponds to the dual of the watershed transform from markers. As far as we know, such a relationship between connected operators and the watershed transform has not been noticed before.

The implementation of the cutting-off-domes and filling-up-basins operators using the IFT requires the definition of the following parameters: an adjacency relation, a weight function, a path-function, and a set of seeds where each seed has assigned to it a label and a reference level. Adjacency relation, weight function, and path-function are fixed for each operator and the selection of seeds, their labels and reference levels, define the type of connected operation. In the context of the IFT, the reference level of the seeds is a parameter that has never been used in previous applications. Since there are interactive and automatic ways to select seeds, their labels and reference levels, we will postpone this discussion to the next section and concentrate on the definition of the other parameters now.

Given an image I, a set $\mathbf{S} = \{s_1, s_2, \ldots, s_m\}$ of m seed pixels in I with labels $l_{s_i} \neq 0$ and reference levels $k_{s_i} \in [0, K]$, $i = 1, 2, \ldots, m$, respectively, we think of I as a directed graph where the pixels in I are the nodes of the graph and each ordered pair (p, q) of 4-neighboring pixels (larger neighborhood sizes are also valid) in I defines an oriented arc that goes from p to q in the graph.

Since all paths start from a seed pixel, we define a weight function and a path-function as follows. Let P be a path from a seed pixel s to a pixel q such that $P = P' \cdot \langle (p, q) \rangle$ is the result of appending arc (p, q) to path P' in the

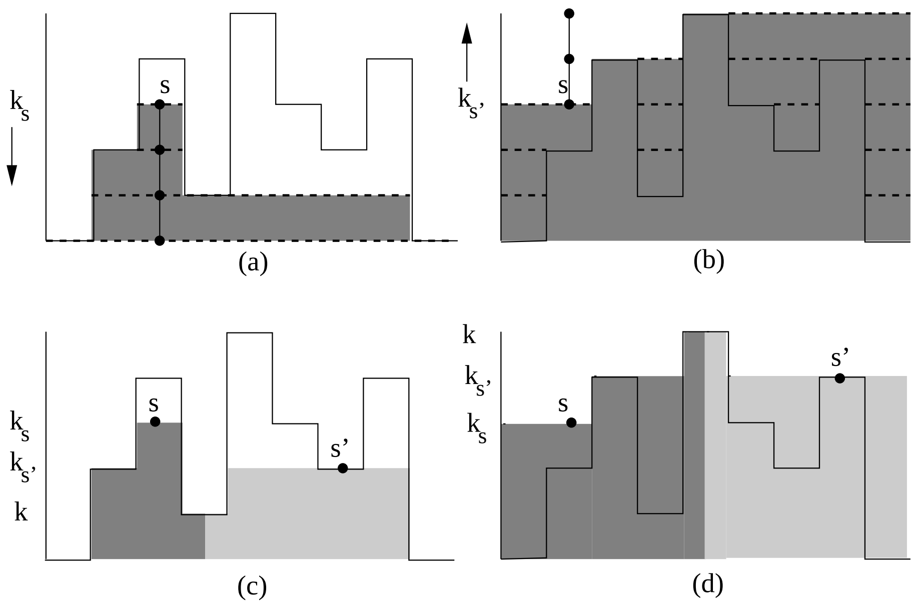


Figure 1. (a) Cutting-off-domes operation from a selected seed s at level k_s. (b) Filling-up-basins operation from a selected seed s at level k_s. (c) Cutting-off-domes operation from two selected seeds s and s' at levels k_s and $k_{s'}$, respectively. (d) Filling-up-basins operation from two selected seeds s and s' at levels k_s and $k_{s'}$, respectively. In all figures, the shaded areas show the influence zones of the seeds and the profile of the union of the shaded areas represent the simplified image.

graph. Then, the weight function $w(p,q)$ and the path-function $pf(P)$ for filling up basins are defined as:

$$w(p,q) = I(q), \tag{1}$$

$$pf(P) = \begin{cases} \max\{pf(P'), w(p,q)\} & \text{if } P' \neq \emptyset, \\ \max\{k_s, w(p,q)\} & \text{otherwise.} \end{cases} \tag{2}$$

Similarly, the weight function $w(p,q)$ and the path-function $pf(P)$ for cutting off domes are defined as:

$$w(p,q) = K - I(q), \tag{3}$$

$$pf(P) = \begin{cases} \max\{pf(P'), w(p,q)\} & \text{if } P' \neq \emptyset, \\ \max\{(K - k_s), w(p,q)\} & \text{otherwise.} \end{cases} \tag{4}$$

In both operations, the IFT computes a shortest-path forest where trees are rooted at the seed pixels defined in $\mathbf{S}$ and creates an annotating image with two properties: the propagated labels and the computed path-function values. The image of path-function values corresponds to the simplified image in the filling-up-basins operation and its complement in the cutting-off-domes operation. The label image corresponds to the watershed transform from markers in the filling-up-basins operation and its dual in the cutting-off-domes operation. Both operations can be implemented by the algorithm below.

IFT Algorithm for cutting off domes and filling up basins

Input: A grayscale image I, a set $\mathbf{S} = \{s_1, s_2, \ldots, s_m\}$ of m seed pixels in I with labels $l_{s_i} \neq 0$ and reference levels $k_{s_i} \in [0, K]$, $i = 1, 2, \ldots, m$, respectively;
Output: An annotating image where $l(p)$ and $pf(p)$ are arrays of propagated labels and path-function values, respectively;
Auxiliary Data Structures: A priority queue Q. An array $s(p)$ that indicates three possible values for the status of a pixel p in Q: *initial* - p was never inserted in Q; *inserted* - p has been inserted in Q; and *removed* - p has been removed from Q. A variable *tmp* for path-function computation.

begin

1. set $pf(p)$ to ∞, $l(p)$ to 0, and $s(p)$ to *initial* for all pixels $p \in I$;
2. *for* each seed s_i in $\mathbf{S}$ *do*
   a. set $pf(s_i)$ to $K - k_{s_i}$ for cutting off domes or set $pf(s_i)$ to k_{s_i} for filling up basins;
   b. set $l(s_i)$ to l_{s_i};
   c. insert s_i in Q and set $s(s_i)$ to *inserted*;
3. *while* $Q \neq \emptyset$ *do*
   a. remove a pixel p from Q such that $pf(p) = \min_{p' \in Q}\{pf(p')\}$ and set $s(p)$ to *removed*;
   b. *for* each pixel q, such that q is 4-neighbor of p and $s(q) \neq$ *removed do*
      (i) set *tmp* to $\max\{pf(p), w(p,q)\}$;
      (ii) *if tmp* $< pf(q)$ *then*
         a. set $pf(q)$ to *tmp* and $l(q)$ to $l(p)$;
         b. *if* $s(q) =$ *initial then* insert q in Q and set $s(q)$ to *inserted*;
         endif;
      endfor;
   endwhile;

end

The algorithm above executes in linear time if Q is implemented using the bucket-sort method.[11,12] Generically, the position of a pixel $q \in Q$ should be updated in step 3b(ii)b, but as we have demonstrated for the watershed transform in,[3] such an operation is not required in this particular graph model.

A very interesting variation in the cutting-off-domes (filling-up-basins) operator is its regional application to remove from the foreground (background) only the connected components above (below) the reference level of their seed (see Figure 2). This allows regional filtering and it is particularly useful to label (extract) connected components as we will see in many examples in this paper. Its implementation is very simple, since all we need is the following change in the algorithm above. In item 1, $pf(p)$ should be initialized to $I(p)$ for filling up basins, and to $(K - I(p))$ for cutting off domes, instead of to ∞ for all pixels p in I.

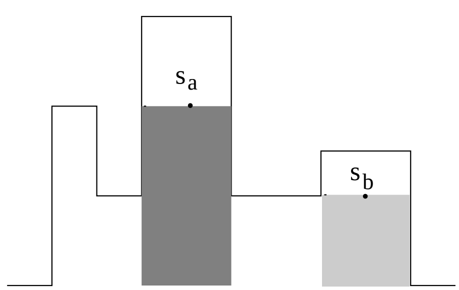


Figure 2. The cutting-off-domes is applied regionally to remove all connected components above the reference level of each seed.

3.2. Selection of seeds, labels, and reference levels

The selection of seeds, their labels and reference levels define the type of connected operation. Their interactive specification constitutes a new paradigm in non-linear image filtering which should be further investigated. In this section, we will concentrate on automatic seed selection to show how the cutting-off-domes and filling-up-basins operators can be combined to implement the following connected operations: regional maxima, regional minima, h-domes, h-basins, leveling, area opening, area closing, alternate sequential filters by reconstruction, opening by reconstruction, closing by reconstruction, closing of holes and removal of pikes.[5,6,10]

For sake of simplicity, we define $\mathcal{D}(I, \mathbf{S})$ and $\mathcal{B}(I, \mathbf{S})$ as the cutting off domes and the filling up basins, respectively, computed on image I using a set $\mathbf{S}$ of seed pixels s with labels l_s and reference levels k_s. Recall that their algorithm outputs an annotating image, where $l(p)$ and $pf(p)$ are arrays of propagated labels and path-function values, respectively. Therefore, we say that the operator $\mathcal{D}(I, \mathbf{S})$ outputs a simplified image D_s, where $D_s(p) = K - pf(p)$, and a label image D_l, where $D_l(p) = l(p)$, for each $p \in I$. The operator $\mathcal{B}(I, \mathbf{S})$ outputs a simplified image B_s, where $B_s(p) = pf(p)$, and a label image B_l, where $B_l(p) = l(p)$, for each $p \in I$.

- Regional Maxima $\mathcal{R}_M(I)$: Computes the flat zones of I whose gray values are greater than the values of their neighbors and outputs a binary image R_M, where $R_M(p) = 1$ if p belongs to a regional maximum in I and $R_M(p) = 0$ otherwise.

  1. Compute an image X such that $X(p) = I(p) - 1$ for each $p \in I$;
  2. Compute an image Y such that $Y(p) = 1$, if $X(p) \geq X(q)$ for all q 8-neighbor of p in X, and $Y(p) = 0$ otherwise;
  3. Put in $\mathbf{S}$ all pixels $s \in Y$ such that $Y(s) = 1$, set l_s to 0, since the labels are meaningless in this operation, and k_s to $X(s)$;
  4. Compute $\mathcal{D}(I, \mathbf{S})$ using these settings;
  5. Compute the image R_M such that $R_M(p) = I(p) - D_s(p)$ for each $p \in I, D_s$;

- Regional Minima $\mathcal{R}_m(I)$: Computes the flat zones of I whose gray values are less than the values of their neighbors and outputs a binary image R_m, where $R_m(p) = 1$ if p belongs to a regional minimum in I and $R_m(p) = 0$ otherwise.

  1. Compute an image X such that $X(p) = I(p) + 1$ for each $p \in I$;
  2. Compute an image Y such that $Y(p) = 1$, if $X(p) \leq X(q)$ for all q 8-neighbor of p in X, and $Y(p) = 0$ otherwise;
  3. Put in $\mathbf{S}$ all pixels $s \in Y$ such that $Y(s) = 1$, set l_s to 0 and k_s to $X(s)$;
  4. Compute $\mathcal{B}(I, \mathbf{S})$ using these settings;
  5. Compute the image R_m such that $R_m(p) = B_s(p) - I(p)$ for each $p \in B_s, I$;

- The h-domes $\mathcal{H}_d(I)$: Computes an image H_d which contains the domes of I whose height is greater than $h \in [0, K]$.

  1. Compute an image X such that $X(p) = I(p) - h$ for each $p \in I$;
  2. Compute $\mathcal{R}_M(X)$ and label each foreground connected component in R_M with a distinct value;
  3. Put in $\mathbf{S}$ all pixels $s \in R_M$ such that $R_M(s) = 1$, set l_s to its respective label and k_s to $X(s)$;
  4. Compute $\mathcal{D}(I, \mathbf{S})$ using these settings;
  5. Compute the image H_d such that $H_d(p) = I(p) - D_s(p)$ for each $p \in I, D_s$;

  Note that, the propagated labels in D_l represent the influence zones of each regional maximum of R_M. The only difference between this operator and an opening by reconstruction is that the image X is created by applying a morphological opening to I. A similar observation is valid for anti-extensive alternate sequential filters by reconstruction.

- The h-basins $\mathcal{H}_b(I)$: Computes an image H_b which contains the basins of I whose depth is greater than $h \in [0, K]$.

  1. Compute an image X such that $X(p) = I(p) + h$ for each $p \in I$;
  2. Compute $\mathcal{R}_m(X)$ and label each foreground connected component in R_m with a distinct value;
  3. Put in $\mathbf{S}$ all pixels $s \in R_m$ such that $R_m(s) = 1$, set l_s to its respective label and k_s to $X(s)$;
  4. Compute $\mathcal{B}(I, \mathbf{S})$ using these settings;
  5. Compute the image H_b such that $H_b(p) = B_s(p) - I(p)$ for each $p \in B_s, I$;

  Note that, the propagated labels in B_l represent the influence zones of each regional minimum of R_m. The only difference between this operator and a closing by reconstruction is that the image X is created by applying a morphological closing to I. A similar observation is valid for extensive alternate sequential filters by reconstruction.

- Leveling $\mathcal{L}(I, J)$: Computes a leveling image L between two input images I and J with the same domain. The image L is such that $I(p) \leq L(p) \leq J(p)$ in regions where $I(p) \leq J(p)$, and $J(p) \leq L(p) \leq I(p)$ in regions where $J(p) \leq I(p)$.

  1. Compute an image X such that $X(p) = \min\{I(p), J \oplus e(p)\}$ for each $p \in I, J$, where $J \oplus e(p)$ is the image J dilated by a structuring element e;
  2. Compute $\mathcal{R}_M(X)$;
  3. Put in $\mathbf{S}$ all pixels $s \in R_M$ such that $R_M(s) = 1$, set l_s to 0 and k_s to $X(s)$;
  4. Compute $\mathcal{D}(I, \mathbf{S})$ using these settings;
  5. Compute an image Y such that $Y(p) = \max\{J \ominus e(p), D_s(p))\}$ for each $p \in I, D_s$, where $J \ominus e(p)$ is the image J eroded by a structuring element e;
  6. Compute $\mathcal{R}_m(Y)$;
  7. Put in $\mathbf{S}$ all pixels $s \in R_m$ such that $R_m(s) = 1$, set l_s to 0 and k_s to $Y(s)$;
  8. Compute $\mathcal{B}(I, \mathbf{S})$ using these settings;
  9. Compute the image L such that $L(p) = B_s(p)$ for each $p \in B_s$.

- Area opening $\mathcal{A}_O(I)$: Removes all foreground connected components from the stack of binary images of I whose area (number of pixels) is less than a threshold and outputs a simplified image A_O. Such an operation removes domes in I whose area is less than the threshold.

  1. Compute $\mathcal{R}_M(I)$ and label each regional maximum in R_M with a distinct value;
  2. For each regional maximum in R_M, use the regional cutting-off-domes operator to compute the highest reference level where the extracted foreground connected component has area greater than or equal to the desired threshold;
  3. Put in $\mathbf{S}$ all pixels $s \in R_M$ such that $R_M(s) = 1$, set l_s and k_s to their respective label and reference level;
  4. Compute $\mathcal{D}(I, \mathbf{S})$ using these settings;
  5. Compute the image A_O such that $A_O(p) = D_s(p)$ for each $p \in D_s$;

- Area closing $\mathcal{A}_C(I)$: Removes all background connected components from the stack of binary images of I whose area (number of pixels) is less than a threshold and outputs a simplified image A_C. Such an operation removes basins in I whose area is less than the threshold.

  1. Compute $\mathcal{R}_m(I)$ and label each regional minimum in R_m with a distinct value;
  2. For each regional minimum in R_m, use the regional filling-up-basins operator to compute the lowest reference level where the extracted background connected component has area greater than or equal to the desired threshold;

3. Put in $\mathbf{S}$ all pixels $s \in R_m$ such that $R_M(s) = 1$, set l_s and k_s to their respective label and reference level;
4. Compute $\mathcal{B}(I, \mathbf{S})$ using these settings;
5. Compute the image A_C such that $A_C(p) = B_s(p)$ for each $p \in B_s$;

- Closing of holes $\mathcal{C}_h(I)$: Let I be an image with dark regions (basins) surrounded by bright regions. This operator reduces the contrast between the dark regions and their surroundings by increasing the gray values of the dark regions. The resulting simplified image is called C_h.

  1. Put in $\mathbf{S}$ all pixels s belonging to the frame of I and set l_s and k_s to 0;
  2. Compute $\mathcal{B}(I, \mathbf{S})$ using these settings;
  3. Compute the image C_h such that $C_h(p) = B_s(p)$ for each $p \in B_s$;

- Removal of pikes $\mathcal{R}_p(I)$: Let I be an image with bright regions (domes) surrounded by dark regions. This operator reduces the contrast between the bright regions and their surroundings by decreasing the gray values of the bright regions. The resulting simplified image is called R_p.

  1. Put in $\mathbf{S}$ all pixels s belonging to the frame of I and set l_s and k_s to K;
  2. Compute $\mathcal{D}(I, \mathbf{S})$ using these settings;
  3. Compute the image R_p such that $R_p(p) = D_s(p)$ for each $p \in D_s$;

The operators presented above have many applications in medical image processing and analysis.[13] In the next section, we illustrate some operators and the power of their combination for medical image segmentation.

4. APPLICATIONS TO MEDICAL IMAGE SEGMENTATION

We have shown that the cutting-off-domes and filling-up-basins operators provide a powerful mechanism to design non-linear image filters, and that the propagated seed labels represent the watershed partitioning from markers. In this section, we present three examples where these connected operators together with other IFT-based operators address medical image segmentation.

Figure 3a presents an image of blood cells where they should be separated from each other automatically. The strategy is to first separate the cells from the background. Such a task requires an area opening of the image in Figure 3a, resulting a simplified image (Figure 3b) which is subsequently thresholded and the cell shapes are smoothed by a morphological opening (Figure 3c). To identify each cell, the Euclidean distance transform is computed inside the cells on the image of Figure 3c using the IFT-based algorithm reported in[2] (Figure 3d). The regional maxima of the distance map are used for cell identification. They are computed and dilated by a structuring element to create the seeds (dark spots) shown in Figure 3e. A distinct label is also assigned to each dilated regional maximum, their reference levels are set to 0, and the regional cutting-off-domes operator is computed on the image of Figure 3c. This operation assigns a different label to each cell, which is used to separate them as shown by black contours in Figure 3f.

Figure 4a shows a CT slice of a knee where the central bone is extracted semi-automatically. The user draws a black line outside the bone and a white line inside the bone. The pixels under these lines are selected as seeds with two correspondent labels, black and white, and with reference levels equal to their gray values. A filling-up-basins operation computed on this image using these settings results the simplified image shown Figure 4b. This alone constitutes a new interactive non-linear image filtering paradigm where the contrast of some objects can be preserved in detriment of the others. On the other hand, it is well known that the watershed transform from markers provides better segmentation when the transform is applied to the gradient image. We have also shown that the filling-up-basins operation outputs the watershed partitioning from markers. Figure 4c illustrates the segmentation result when the filling-up-basins operation is computed on the morphological gradient of the image in Figure 4a using the black and white seeds. Note that the watershed transform misses half of the border of the bone. At same time, a family of watershed transforms can be defined by changing the reference levels of theses seeds. For example, the gradient image is dilated and the reference levels of the seeds are set to their gray values in the dilated image. Figure 4d illustrates the same operation computed on the gradient image using the black and white seeds, but now

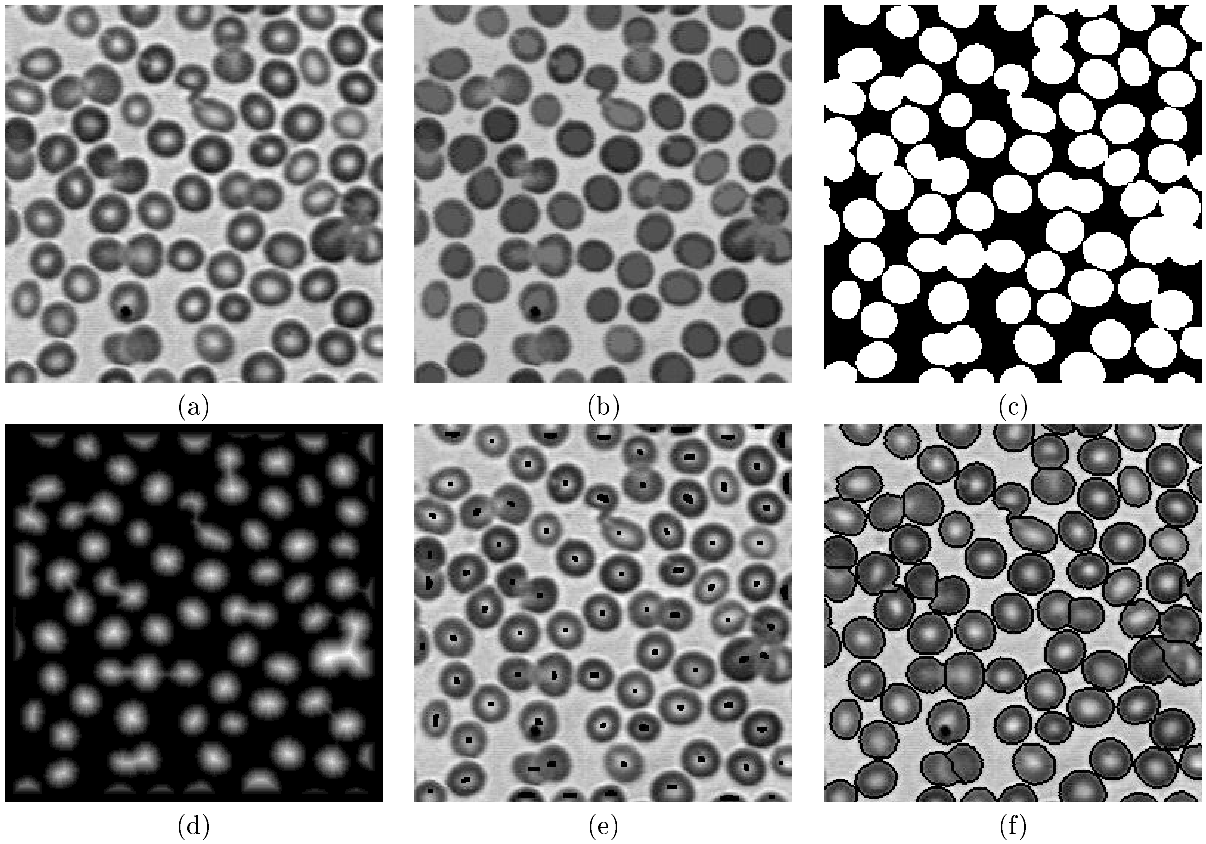


Figure 3. (a) An image of blood cells where they should be separated from each other automatically. (b) Image (a) is simplified by an area opening. (c) Image (b) is thresholded and the cell shapes are smoothed by a morphological opening. (d) The Euclidean distance transform is computed on (c). (e) Black spots indicate the regional maxima of (d) dilated by a structuring element. (f) The regional cutting-off-domes computed on (c) where a different label is assigned to each cell in order to separate them as shown by black contours.

with the new reference levels. Note that, the segmentation has improved but it still misses part of the border of the bone. Fortunately, the problem can be efficiently solved with our regional connected operators as follows.

Figure 5a presents the closing of holes computed on the image of Figure 4a as described in the previous section. The user draws a black line in the interior of the bone as shown in Figure 5a and the pixels under this line are selected as seeds with the same label. When the user sets their reference levels one level above their actual gray values in Figure 5a, the regional filling-up-basins computed on this image extracts the interior of the bone (Figure 5b). Alternatively when the user sets their reference levels one level below their gray values in Figure 5a, the regional cutting-off-domes computed on this image extracts the whole bone (Figure 5c). Moreover, Figure 5d illustrates that the white border of the bone can also be isolated by subtracting the image in Figure 5c from the image in Figure 5b and computing a morphological closing on the difference image.

Finally, we present an example where connected operators overweight Gaussian filters in preprocessing for image segmentation using the live wire method.[12] Figure 6a shows the live-wire tracings computed on an MRI slice of a foot where the object of interest is the bone called talus. The user had to select seven points on the boundary to accomplish this segmentation task. Such a number is reduced to four when the image is simplified by a Gaussian filter (Figure 6b). On the other hand, the Gaussian filter creates false edges and blurs some relevant boundary details. A better image filtering is obtained by computing the leveling between the images in Figures 6a-b, as described in the previous section. Figure 6c illustrates the segmentation on the simplified image and also shows that the number of points selected on the boundary has reduced to three, further minimizing the user involvement.

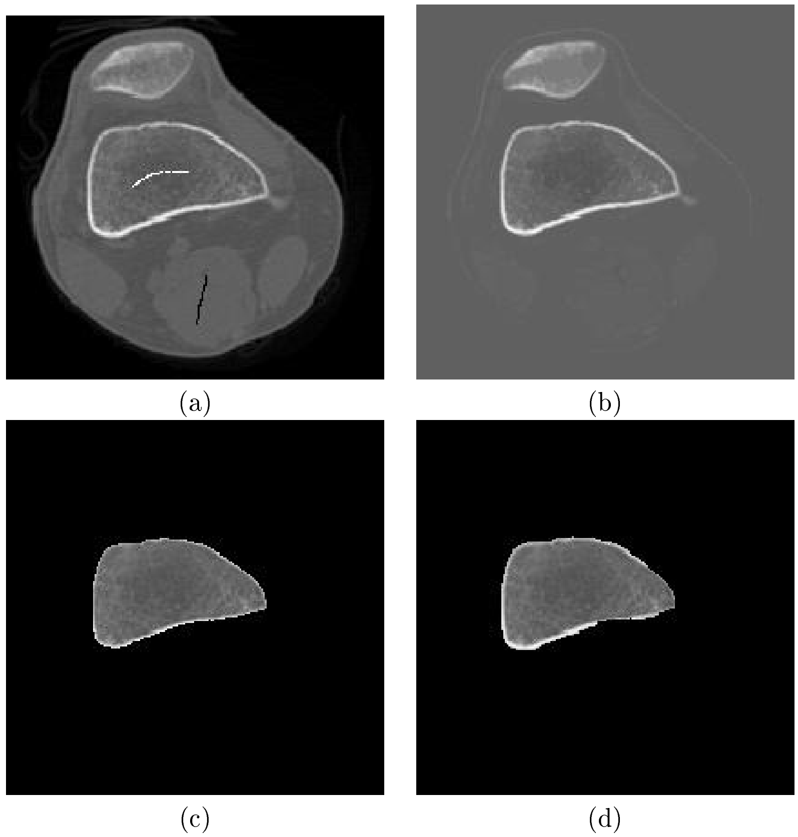


Figure 4. (a) A CT slice of a knee where the white and black lines indicate seeds inside and outside the object of interest (central bone), respectively. (b) The filling-up-basins computed on (a) using the pixels under the lines as seeds. It is also computed on the gradient of (a) and on the gradient of (a) dilated by a structuring element. The respective partitions (segmentation) are shown in (c) and (d).

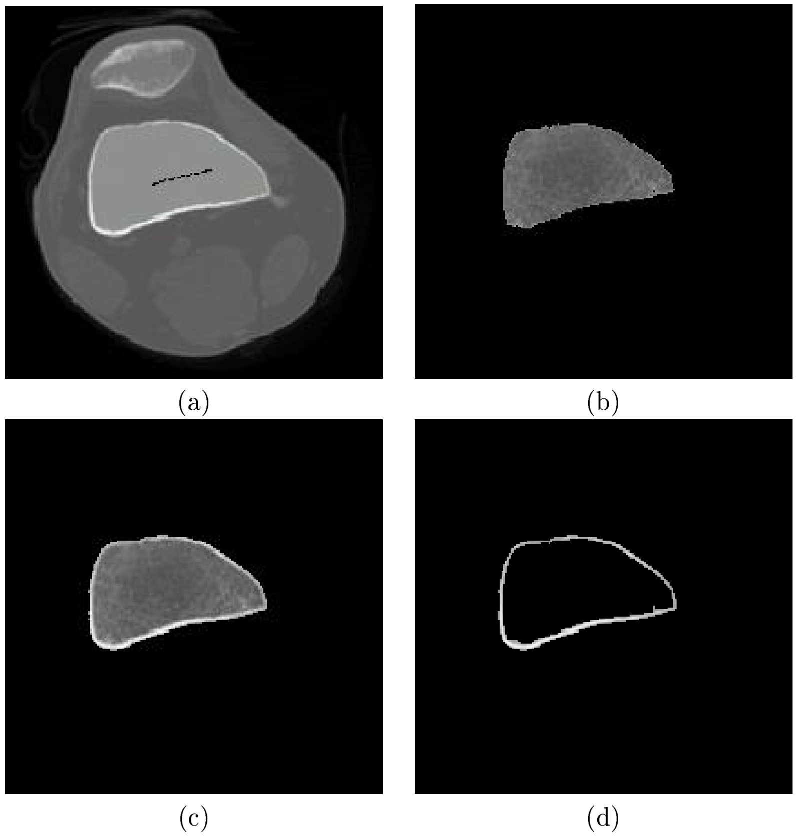
5. CONCLUSION AND DISCUSSION

We have presented two connected operators based on the IFT, named cutting-off-domes and filling-up-basins. They may be used regionally or globally to provide automatic and interactive non-linear image filtering, segmentation, and labeling of connected components. We have shown the design of several grayscale connected operators based on these two operations which also provide the watershed partitioning from markers. That makes the watershed transform part of the connected operation and as far as we know this has never been proposed before. The connected operators together with the previous image operators make the IFT a powerful and efficient tool to handle image processing problems.

The interactive use of the presented connected operators introduces a new paradigm in image filtering, which can lead to improvements in image segmentation, enhancement and compression, and so should be further investigated. The extension of the work to the 3D case is straightforward and will be pursued next.

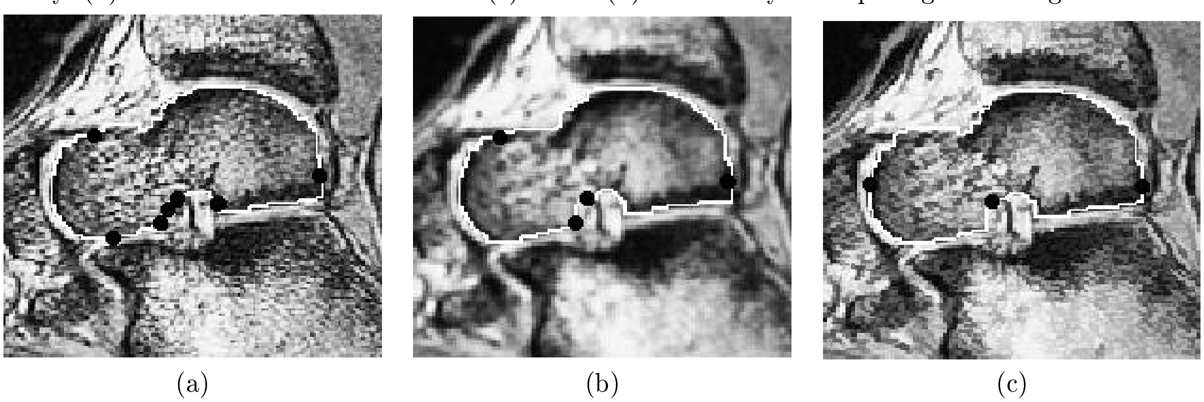
ACKNOWLEDGMENTS

Alexandre X. Falcão is grateful to Dr. Jayaram K. Udupa, Dr. Bruce E. Hirsch, and SDC information systems for the medical images used in this paper. His work is partially supported by grants from CNPq (Proc. 300698/98-4) and FAEP - University of Campinas. The work of Bruno S. da Cunha is financially supported by FAPESP (Proc. 99/10100-3).



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

Figure 5. (a) A closing of holes computed on the image of Figure 4a. Black seeds are drawn inside the bone and the regional filling-up-basins and cutting-off-domes extract the interior and the whole bone as shown in (b) and (c), respectively. (d) The white border results from (c) minus (b) followed by a morphological closing.



(a) (b) (c)

Figure 6. This figure illustrates the improvements on image segmentation using the live wire method[12] due to image pre-processing by smoothing filters. Leveling provides better results than Gaussian filters. (a) The live-wire tracings on an MRI slice of a foot, (b) on the image that results of a Gaussian filter computed on (a), and (c) on the simplified image which results of the leveling between (a) and (b).

REFERENCES

1. A. Falcão, R. A. Lotufo, G. Araujo, and J. Stolfi, "The image foresting transform," *IEEE on Pattern Analysis and Machine Intelligence* . submitted.

2. A. Falcão and B. da Cunha, "Multiscale shape representation by image foresting transform," in *Proceedings of SPIE on Medical Imaging'2001*, (San Diego, CA), Feb 2001. to appear.
3. R. Lotufo and A. Falcão, "The ordered queue and the optimality of the watershed approaches," in *5th International Symposium on Mathematical Morphology*, pp. 341–350, Kluwer Academic Publishers, (Palo Alto, USA), Jun 2000.
4. R. Lotufo, A. Falcão, and F. Zampirolli, "Fast Euclidean distance transform using a graph-search algorithm," in *XIII Brazilian Symposium on Computer Graphics and Image Processing*, pp. 269–275, (Gramado - RS, Brazil), Oct 2000.
5. L. Vincent, "Morphological area opening and closings for greyscale images," in *Shape in Picture'92 - NATO Workshop*, Springer Verlag, (Driebergen, The Netherlands), Sep 1992.
6. L. Vincent, "Morphological grayscale reconstruction in image analysis," *IEEE Transactions on Image Processing* **2**, pp. 176–201, Apr 1993.
7. P. Salembier, A. Oliveras, and L. Guarrido, "Antiextensive connected operators for image and sequence processing," *IEEE Transactions on Image Processing* **7**, pp. 555–570, Apr 1998.
8. P. Salembier and J. Serra, "Flat zones filtering, connected operators, and filters by reconstruction," *IEEE Transactions on Image Processing* **4**, pp. 1153–1160, Aug 1995.
9. S. Beucher and F. Meyer, "The morphological approach to segmentation: The watershed transformation," in *In Mathematical Morphology in Image Processing*, E. R. Dougherty, ed., ch. 12, pp. 433–481, Marcel Dekker, Inc., New York, NY, 1993.
10. F. Meyer, "The levelings," in *4th International Symposium on Mathematical Morphology*, H. Heijmans and J. Roerdink, eds., pp. 190–207, Kluwer Academic, 1998.
11. T. Cormen, C. Leiserson, and R. Rivest, *Introduction to Algorithms*, MIT Press, New York, NY, 1991.
12. A. Falcão, J. Udupa, and F. Miyazawa, "An ultra-fast user-steered image segmentation paradigm: Live-wire-on-the-fly," *IEEE Transactions on Medical Imaging* **19**, pp. 55–62, Jan 2000.
13. M. Sonka and J. Fitzpatrick, eds., *Handbook of Medical Imaging: Volume 2. Medical Image Processing and Analysis - Chapter 3*, SPIE Press, USA, 2000.