

# A Toolbox for Interactive Segmentation Based on Nested Partitions

F. Zanoguera, B. Marcotegui and F. Meyer  
*Centre de Morphologie Mathématique*  
*Ecole Nationale Supérieure des Mines de Paris*  
*77305 Fontainebleau Cedex, France*  
{zanoguera,marcotegui,meyer}@cmm.ensmp.fr

## Abstract

*This paper presents a toolbox for interactive image segmentation based on a series of nested partitions of increasing coarseness. The user can navigate among the different resolution levels and select regions with simple mouse clicks. The whole family of partitions can be created during a single morphological flooding of the image, leading to a very fast algorithm. The information is stored in a minimum spanning tree.*

## 1. Introduction

Automatic segmentation has proved to be a difficult issue in applications where generic types of images must be treated. In multimedia applications, where no *a priori* knowledge of the image exists, interactive segmentation presents an attractive solution. The goal here is to minimise the effort required from the user to obtain the desired output.

Most existing interactive segmentation tools suffer from limitations. Some lack flexibility in the interaction process [6] [7] [2] while others rely on expert knowledge from the user side [7] or make assumptions on the characteristics of the images to be segmented [3].

A system for interactive segmentation of generic images is presented. No expert knowledge from the user is required, and the interaction mechanisms are flexible and appear as natural.

We show in the first place how a multiscale segmentation can be defined as the set of balls of increasing radius for an ultrametric distance. We describe afterwards how this multiscale segmentation can be created during a single morphological flooding and stored in the form of a minimum spanning tree. Finally, we show how the interaction with the user is carried out.

## 2. Multiscale segmentation

Our system is based on the generation of a set of nested partitions  $P_1, P_2, \dots, P_k$  of the image. These partitions are

organised as a multiscale segmentation, where the lowest level  $P_1$  gives the finest resolution with  $k$  regions. The highest level  $P_k$  corresponds to a partition with the whole image as only one region. A partition at a higher level is obtained by merging regions of the partition below. For this reason each contour present in a given partition is also present in all partitions of lower level.

### 2.1. Neighbourhood graph and minimum spanning tree

**2.1.1. The watershed transform.** The watershed transform [1] constitutes the paradigm of morphological segmentation. The morphological (colour) gradient of the image to segment is first constructed. This image, considered as a topographic surface, is steadily flooded. Starting from the minima, lakes are formed which eventually meet with other lakes. When this happens, they are prevented from merging by erecting a zero-thickness dam between them. The water level increases until the whole image has been flooded. At the end of this process, a partition of the image is obtained, which contains one region for each catchment basin of the gradient image. In the following, the watershed transform will be referred to as uniform flooding, as the water level is kept constant during the flooding.

The process described produces a very strong over-segmentation of the image. The solution to this problem has traditionally been the use of markers. In the marker approach, the topographic surface is not flooded from sources placed at all the image minima, but from selected sources, called markers, placed at different points of the image. The segmentation displaces the over-segmentation problem into finding good markers.

**2.1.2. The neighbourhood graph.** The gradient image (interpreted as a topographic surface) can be represented by a neighbourhood graph, which contains a node for each catchment basin of the gradient image. Two nodes are linked by an edge if their corresponding catchment basins are neighbours. The edges are weighted with a dissimilarity measure between regions.

The flooding from markers may be carried out on the neighbourhood graph. Sources are placed on some nodes, and the edge weights represent the height of the borders that the water must cross. If the dissimilarity measure chosen to weigh the graph edges is the lowest pass point along the border separating the two catchment basins on the gradient image, then flooding the neighbourhood graph is equivalent to flooding the image. However, the graph representation is more general and can be used in any context where a fine partition and a dissimilarity measure between neighbouring regions are defined.

Flooding the neighbourhood graph is much faster than flooding the topographic surface, as the number of nodes on the graph is much smaller than the number of pixels on the image.

**2.1.3. The minimum spanning tree.** A close observation of the flooding process reveals that the water always follows the path of lowest sup-section (sup-section: highest value along the path). This is as well the path chosen by the minimum spanning tree of the neighbourhood graph [5]. Therefore, the minimum spanning tree of the neighbourhood graph contains all the necessary information for flooding purposes.

On the minimum spanning tree, every pair of nodes is linked by a unique path, which corresponds to the one of lowest sup-section among all the existing paths between the two nodes on the neighbourhood graph. Because on the minimum spanning tree every pair of nodes is linked by a unique path, eliminating  $n - 1$  edges of the tree yields  $n$  connected components, and therefore a segmentation of the image into  $n$  regions. For different values of  $n$  we obtain partitions containing a different number of regions, and therefore a multiscale segmentation.

The minimum spanning tree can be created from the neighbourhood graph by using any of the algorithms found in the literature. However, a more direct algorithm may be derived for our application by taking advantage of the fact that among all possible paths, the minimum spanning tree always chooses the one with lowest sup-section. This is also the path chosen by the water during the flooding. The minimum spanning tree can therefore be created simultaneously to the flooding process.

Initially, a node of the minimum spanning tree is created for each minimum of the gradient image. The gradient image is then flooded as described. Every time that two lakes meet for the first time, an edge is added between their corresponding nodes of the tree, weighted with the current water level. At the end of the process, the minimum spanning tree has been created.

## 2.2. Flooding ultrametrics

An ultrametric distance is a metric that satisfies the following conditions:

1.  $d(x, x) = 0 \quad \forall x$
2.  $d(x, y) = d(y, x) \quad \forall x, y$
3.  $d(x, y) \leq \text{Max}(d(x, z), d(z, y)) \quad \forall x, y, z$

The water level for which each pair of catchment basins meet for the first time defines an ultrametric distance.

The first two conditions are obviously satisfied by this distance. The third condition is also satisfied, as the right hand part of the inequality represents the water level for which the catchment basins  $x$ ,  $y$  and  $z$  meet for the first time while the left hand part represents the water level for which only two of these lakes meet for the first time.

This ultrametric distance may alternatively be defined on the minimum spanning tree between any pair of nodes as the sup-section of the unique path linking the two nodes.

The ultrametric distance has a very interesting property:

**Proposition 1** *Two open (resp. closed) balls  $B(x, r)$  and  $B(y, r)$  of same radius and different centre are either disjoint or identical.*

**Interpretation:** A ball of centre  $C$  and radius  $R$  is the set of points that are at a distance smaller or equal than  $R$  from the centre  $C$ . On the minimum spanning tree, the distance between two nodes has been defined as the sup-section of the path linking the two nodes. Cutting the edges of the tree with value higher than  $R$  and taking the connected component that contains  $C$  yields the ball of centre  $C$  and radius  $R$ . If another node of the same subtree is taken as centre, the same subtree is obtained. If an element of another subtree is taken as centre, the subtree obtained is disjoint with the previous one.

The set of maximal balls with radius smaller than a certain value  $R$  produce a multiscale segmentation for increasing values of  $R$  [4]. These balls can be obtained from the minimum spanning tree by cutting the edges with value higher than  $R$ .

However, the partitions obtained using the uniform flooding ultrametrics are not perceptually satisfactory. Other ultrametrics must be defined in order to obtain meaningful results.

## 2.3. New ultrametrics defined through synchronous flooding

In the synchronous flooding [4], the topographic surface is not flooded at a constant water level, but in such a way that all the lakes keep constant a certain measure (depth,

surface, volume) at any given time. When a catchment basin is about to overflow into a neighbouring catchment basin, the first is already full and stops growing until its neighbour has reached the same water level. At this moment, the full catchment basin is assigned to the same class as its neighbour, i.e. it is absorbed by it. As the flooding continues, successive absorptions will take place, until the whole image has been flooded and all the lakes belong to the same class.

New ultrametries can be defined through synchronous flooding, by associating to each pair of catchment basins the measure (the same one chosen for flooding the lakes: depth, surface or volume) of the lake that was full when they were joined into the same class for the first time.

The edge weights of the minimum spanning tree may be modified so that the distances calculated from it correspond to the new ultrametric distances. This can again be done simultaneously to the synchronous flooding. Initially, a node of the tree is created for each catchment basin of the gradient image. Every time that a lake is about to overflow into another lake, an edge is added between their corresponding nodes on the tree, weighted with the measure of the full lake. At the end of the process, the tree has been created, its edge weights representing the chosen ultrametries.

The ultrametries chosen will in general have a strong repercussion on the quality of the results obtained. Figure 1 shows the original image and the 15 largest balls for three different ultrametries. The depth ultrametries (Figure 1-b) considers as largest balls the most contrasted regions, regardless of their size. The surface (Figure 1-c) rates large regions as important, regardless of their contrast. Finally, the volume (Figure 1-d) balances well size and contrast of the regions and is the distance we have chosen to create our multiscale segmentation.

The synchronous flooding is however difficult to program. An equivalent algorithm is proposed in the following section, based on the uniform flooding.

## 2.4. Proposed algorithm

The following algorithm is based on the fact that for both synchronous and uniform flooding processes, the relevant events happen at the same meeting points and for the same measures of the smaller lakes. For the uniform flooding, the relevant event is the meeting of two lakes having the same water level. When this happens both lakes are full. For the synchronous flooding, the relevant event is that only one of the lakes is full, the one with smaller measure.

The proposed algorithm is very similar to the one described in Section 2.1.3 for the creation of the minimum spanning tree during uniform flooding. Initially, every

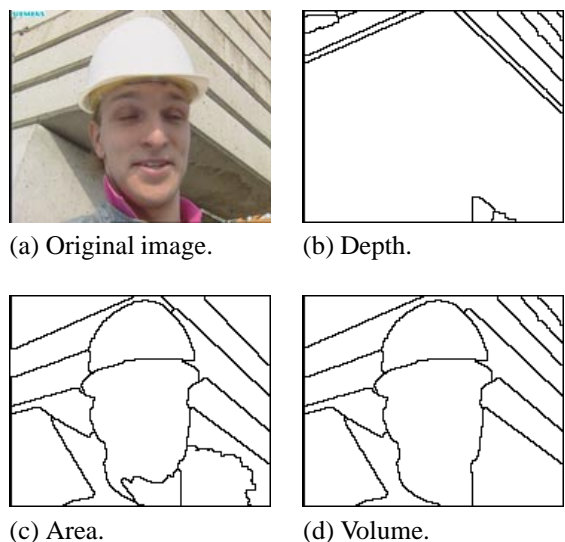


Figure 1: Comparison between the depth, area and volume ultrametries.

catchment basin belongs to a different class. Every time that two lakes that belong to different classes meet during the flooding process: (a) an edge is added between the nodes corresponding to their catchment basins weighted with the measure of the smaller lake (depth, surface, volume) and (b) both classes are joined together.

Figure 2 illustrates this algorithm. For each catchment basin there is a node on the minimum spanning tree. As the water level increases, the first pair of lakes to meet are 2 and 3. At this moment the volumes of the two lakes are compared. Lake 3 has a volume of 6 while the volume of lake 2 is only 4. Lake 3 and lake 2 are grouped together into the same class and an edge is introduced on the tree between nodes 2 and 3, of value 4 (smallest volume). The volume of the class is now updated to the sum of both, i.e. 10. The next pair of lakes to meet are 3 (which belongs to the same class than lake 2) and 4. Since lakes 3 and 4 belong to different classes, an edge will be introduced between them, weighted with the smallest class volume. Lake 3 has a class volume of 18 while the volume of lake 4 is 7. therefore, the new edge is weighted with the class volume of lake 4, i.e. 7. The last meeting is between lake 2 and lake 1. Again, class volumes are compared and the last edge is added to the tree.

## 3. Interaction with the user

The “good” segmentation does not exist in general, as it depends on both the image and the application. Often se-

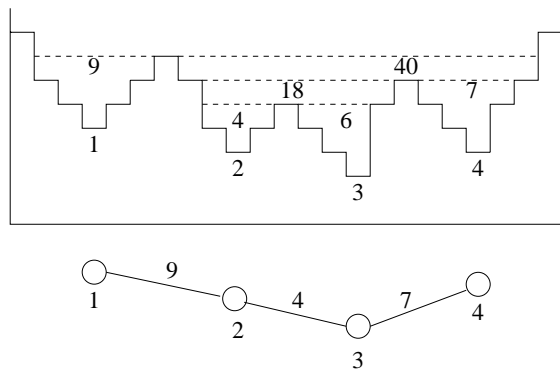


Figure 2: Flooding process with creation of a minimum spanning tree.

mantics plays an important role and this information must be introduced in the segmentation process. The injection of information by the user must be as easy and intuitive as possible. The nested segmentations produced so far provide a solid and natural support for human interaction, as will be shown in the next section.

### 3.1. Selection of the total number of regions

The most general and simple way of user interaction consists of the selection of the total number of regions in which the image must be segmented. The user indicates a total number  $n$  of regions. This corresponds to a request for an automatic segmentation of the image into  $n$  regions. The  $n$  largest balls of the ultrametrics must be presented to the user. To do this, the  $n - 1$  edges of the tree with highest weight are suppressed. This type of interaction is a starting point in the segmentation process that provides the user with a first proposal to work on. It can be presented in the form of a sliding bar that can be moved up and down to adapt the size of the regions to the specific application. This can be done in real user time because all the calculations are carried out on the minimum spanning tree rather than on the image.

Figure 3 shows three different levels of the multiscale segmentation.

### 3.2. Local actions on the image

The interaction type described in the previous section treats the image as a whole, finding the  $n$  largest balls from the point of view of the volume ultrametrics. However, the user may be interested in having some regions/objects segmented with more detail than others. In this case, the user must be offered the possibility to refine a certain area or to coarsen it by merging it with neighbouring regions.

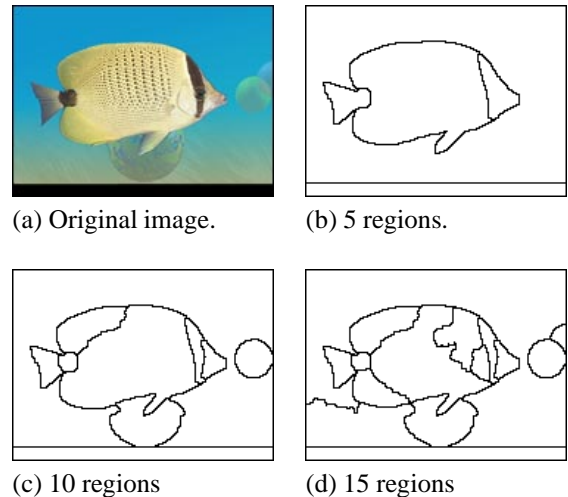


Figure 3: Original image and three different levels of the multiscale segmentation.

This is done by locally decreasing/increasing the radius of a selected ball. Two operations allow the user to locally navigate up and down the scale of segmentations.

In the *refine* operation, the user clicks on a certain area of the image with the mouse. At the same time, the number of regions in which the selected region must be subdivided may be specified. If it is not, a default value is used. The radius of the selected ball/region is decreased until it produces a re-segmentation into the requested number of regions. From the point of view of the implementation, the  $n - 1$  ( $n$  being the parameter specified by the user) edges of highest weight are searched for and eliminated, but this time only the edges inside the region selected by the user are considered.

In the *coarsen* operation, the user selects a region with a mouse click, and again a parameter  $n$  may be specified. The radius of the selected ball is increased by merging neighbouring regions. The  $n - 1$  most similar neighbouring regions are merged to the one selected. Among the previously eliminated edges, the  $n - 1$  of lowest weight that link a node belonging to the selected region with an external one are re-inserted.

Figure 4 shows an example of local operations. Starting from an automatic segmentation into 10 regions, the user has clicked on the large region inside the woman contour in Figure 4-b to request a re-segmentation of that region into 6 new regions. Figure 4-c shows the result. Next, the user has clicked on the left part of the background and requested a merging of the most similar 4 regions into one. The result is shown in Figure 4-d.

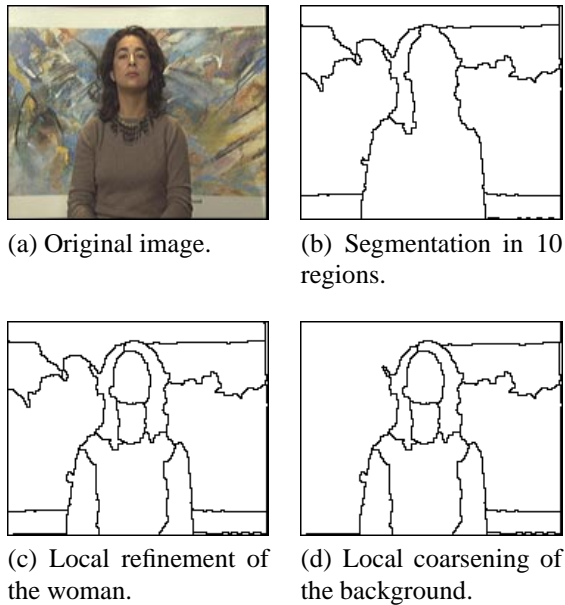


Figure 4: Example of local actions on the image.

### 3.3. Marker drawing

This type of interaction is an alternative to the selection of the total number of regions. The user draws a marker on every object of interest, and also a marker for the background. The algorithm will then again consider the edges for suppression in decreasing order, but validate a suppression only if each produced subtree contains at least one marker. Alternative algorithms can be found in [5]. Figure 5 shows the markers that have been drawn by the user to select the objects of interest and the results obtained. Further interaction of the types described in Section 3.2 is possible from this point.

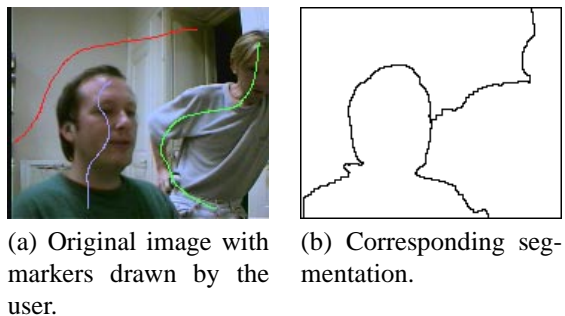


Figure 5: Example of marker drawing and the corresponding results.

## 4. Conclusion

A toolbox for the interactive segmentation of generic images has been presented. It is based on a multiscale segmentation calculated during a single flooding of the image, operation for which there exist fast implementations based on hierarchical queues. The results are stored in the form of a minimum spanning tree. Several tools have been developed, allowing an easy interactive navigation between the different levels of the scale of segmentations. Because the interaction with the user is based on simple operations on the minimum spanning tree, the computing time is fast enough for the user to perceive the results as immediate.

## References

- [1] S. Beucher and F. Meyer. “The morphological approach to segmentation: the watershed transformation”, *Mathematical morphology in image processing*, E. Dougherty editor, M. Dekker, 1993.
- [2] E. Chalom and V.M. Bove. “Segmentation of an image sequence using multi-dimensional image attributes”, *Proceedings of the IEEE International Conference on Image Processing, ICIP*, Lausanne, September 1996.
- [3] S. Delgado Olabarriaga, D. Koelma and A.W.M. Smeulders. “A Simple Application Framework for Interactive Segmentation Systems”, *ASCI*, 1997.
- [4] F. Meyer “Morphological multiscale and interactive segmentation”, *IEEE-EURASIP Workshop on Non-Linear Signal and Image Processing (NSIP'99)*, Antalya, Turkey, June 1999.
- [5] F. Meyer “Minimum spanning forests for morphological segmentation” *Mathematical Morphology and its Applications to Image Processing, ISMM'94*, J. Serra and P.Soleille Eds., Kluwer Academic Publishers, 1994.
- [6] N.E. O'Connor and S. Marlow. “Supervised semantic object segmentation and tracking via EM-based estimation of mixture density parameters”, *Noblesse Workshop on Non-Linear Model Based Image Analysis*, Glasgow, July 1998.
- [7] K.R. Subramanian, D.M. Lawrence and M.T. Mostafavi. “Interactive segmentation and analysis of fetal ultrasound images”, *8th EG Workshop on ViSC*, Boulogne sur Mer, April 1997.