A SEGMENTATION PYRAMID FOR THE INTERACTIVE SEGMENTATION OF 3-D IMAGES AND VIDEO SEQUENCES

F. ZANOGUERA, B. MARCOTEGUI and F. MEYER
Centre de Morphologie Mathématique - Ecole des Mines de Paris
35, rue Saint-Honoré, 77305 Fontainebleau Cedex, France
email: zanoguera@cmm.ensmp.fr

Abstract. This paper presents a technique for the generation of a segmentation pyramid for a video sequence, designed for use in an interactive segmentation context. The pyramid is represented as a minimum spanning tree, which allows an efficient access to the information and avoids unnecessary recalculations. Several ways to introduce user interaction are also proposed.

Key words: segmentation pyramid, minimum spanning tree, interactive segmentation, 3-D image segmentation, video sequence segmentation, region hierarchy

1. Introduction

Image segmentation is the first step in many computer vision systems, and it has been used for a long time in industrial and medical applications. Recently, along with the growth of the Internet and the availability of powerful computers to the general public, new standards like MPEG-4 and MPEG-7 are appearing. They address the so-called content-based applications, where images and video-sequences are treated not at the pixel level, but at the object level. These applications rely on a successful segmentation of the objects present in the scene.

In multimedia applications, the treated images can be very diverse and automatic segmentation becomes problematic as no *a priori* knowledge on the contents of the images is available. In this context, interactive segmentation offers an attractive solution by automating the task of finding homogeneous regions while leaving object definition to the user. Segmentation pyramids are very well suited to interactivity, as they offer a high degree of flexibility. The user can navigate between the different resolution levels, choosing the regions that form the object of interest.

Several types of bottom-up segmentation pyramids have been proposed in the literature [13], [12], [14], [5], [1], [2], [15]. In these approaches, regions are progressively merged into larger regions until a single region is obtained or a stop condition is satisfied. The hierarchy is then represented in the form of a graph. Two main issues arise when examining the existing literature:
1. The merging criterion is in most cases based on a *local* dissimilarity measure, which does not take into account global measures such as size of the regions. This produces good results at the first merging steps, but at coarser levels of the hierarchy semantic aspects start to play a role and local merging criteria

are no longer valid.
2. Although some of the techniques proposed can directly be applied to the segmentation of 3D images, no proposals have been made for the generation of a pyramid which is *coherently preserved* throughout a complete video sequence shot. In most cases, moving video sequences are treated in a tracking-based manner, where an initial mask is interactively defined for the first frame of the sequence, to be tracked in the remaining frames [7], [3]. Whenever a hierarchy is used, it is for the analysis of individual images and it is not preserved from one frame to the next. It would however be very desirable to produce a hierarchy which is coherently preserved throughout the video sequence shot. In this way, when the user interacts to re-partition a certain region or to merge it with a neighbour, this interaction affects all the images of the sequence and the process of video object generation is much faster.

The first issue has been addressed in [17], [6], in the context of interactive segmentation of still images. In [17], a segmentation pyramid is produced by establishing a hierarchy between the regions of the watershed. The merging order defining the hierarchy is based on the volume extinction values [16], a global measure which is well adapted to the characteristics of human perception, due to the trade-off between size and contrast it makes. As the complete hierarchy can be obtained in a single watershed of the image, it results in very fast calculation. The hierarchy is represented in the form of a Minimum Spanning Tree, which allows for very fast manipulation.

This paper deals with the issue of video sequence processing by proposing an extension of the techniques described in [17] to the segmentation of video sequences and 3-D images. We consider video sequences as a particular case of 3-D images with two spatial dimensions and one temporal dimension. Video sequences being too long to process them as a single 3D block, a time recursive approach is proposed.

A segmentation pyramid for the whole sequence (or 3-D image) is made available to the user, who can navigate between the different resolution levels to create or modify the object partition. The actions carried out by the user affect the whole sequence.

2. General description

To produce a segmentation pyramid for a 3-D image or a video sequence, we use a 3-D approach based on flat-zone detection followed by a watershed, both using 3-D connectivity. The 3-D approach has the advantage that the same algorithms can be applied to both 3-D images and video sequences. In the sequel we will focus on video sequences, although the technique is also valid for 3-D image segmentation.

As a video sequence can be very long and the computer memory is limited, the sequence must be divided into blocks of images for processing. In order to be able to preserve the region hierarchy between consecutive blocks, our algorithm divides the video sequence into *overlapping blocks* (one image overlap).

An intra-block algorithm produces the segmentation pyramid for each block.

In order to ensure a coherence of the regions and their hierarchy between blocks, a single global hierarchy is recursively produced, using a procedure that we will call *pyramid projection*.

The segmentation pyramid for the video sequence is represented by a minimum spanning tree (MST). This tree is progressively updated at each projection step using the information contained in each block of the sequence.

Section 3 explains the generation of the segmentation pyramid for an individual block. Section 4 describes the pyramid projection between two consecutive blocks. Finally, Section 5 proposes some mechanisms to interact with the segmentation pyramid.

3. Intra-block segmentation

This section describes how the segmentation pyramid is produced for a block of images and how it can be represented as a tree. The computation is carried out in two steps. Firstly, a fine partition of the block is produced and represented by a neighbourhood graph. Then, from the neighbourhood graph a minimum spanning tree representing the segmentation pyramid is derived.

3.1. Creation of a fine partition through flat-zone detection and gradient flooding

In our previous work with still images in [17], a fine partition is created by flooding a morphological gradient from all its minima. This produces a partition with as many regions as there are minima on the gradient image. This technique can be directly extended to still 3-D images by using a 3-D gradient and flooding using 3-D connectivity. However, for moving video sequences, using a 3-D gradient will produce erroneous spatial contours around the moving areas, as these areas will show high gradient values due to the image-to-image changes. Using 2-D connectivity to calculate the gradient is not a good approach either, as it will produce leaks in the propagation: when two regions with low 2-D gradient values become temporally connected due to motion, temporal propagation will produce the label of the first region to be propagated inside the second region from one image to the next.

Creating the fine partition by flat-zone detection on the initial image rather than detecting the catchment basins of a gradient image restricts the possibility of temporal leaks as the propagation is made only across regions having the same colour. However, the contours of the flat zones are in general very irregular. To take advantage of the better contours produced by the watershed without increasing the risk of leaks, a hybrid approach is used. Large 3-D flat zones are detected and taken as markers for a 3-D watershed propagation inside a gradient image produced using only 2-D connectivity. Small flat zones are discarded, considering that they belong to a transition area, where the gradient will very likely have a high value (and thus no leaks will take place). To take advantage of colour information, three separate 3-D watersheds for the three colour components are calculated, taking flat zones or pseudo-flat zones [11] as markers. The intersection of the three partitions is then calculated. In a

last step, very small regions are removed and a last watershed is computed to find the final contours. The partition intersection further minimises the risk of leaks (spatial and temporal), as there will only be a leak if it exists in the three partitions.

As an important pre-processing step, filtering by levelings [10] is applied to the three normalised colour components. This allows the enlargement of the flat zones without modifying the regions' contours.

3.2. The neighbourhood graph

Any partition can be represented by a neighbourhood graph. Each region of the partition is represented by a node of the graph. Two nodes are connected if the regions they represent are neighbours on the image. The graph edges can be weighted to express a local dissimilarity measure between regions. Possible dissimilarity measures that can be used to weight the graph edges are the lowest pass point along the border separating the two regions on the gradient image, the average gradient value along the border or a distance measure based on the colour components inside the region.

The neighbourhood graph provides a simplified representation of the image, and it can be flooded using a watershed algorithm where the edge weights represent the height of the borders the water must cross [9].

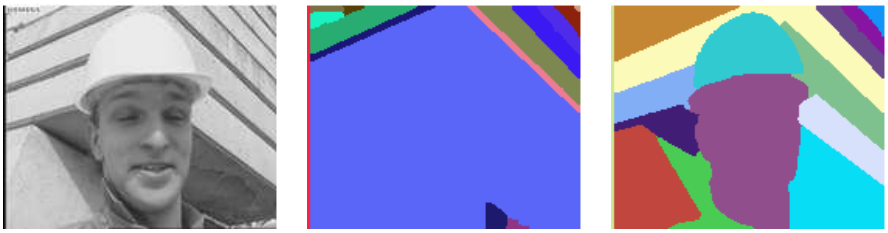
3.3. Minimum spanning tree and segmentation pyramids

In the watershed transformation, the lakes corresponding to two catchment basins A and B always meet through the path of lowest sup-section (sup-section: highest value along the path) between A and B. If we consider all possible pairs of catchment basins, we obtain a set of paths of minimal sup-section. This set of paths forms the minimum spanning tree of the neighbourhood graph [8]. Hence, the minimum spanning tree of the neighbourhood graph contains all the necessary information for flooding purposes. It can be created during the flooding of the neighbourhood graph by adding the edges to the MST in the order they are flooded on the neighbourhood graph, as long as they do not introduce a loop. When an edge introduces a loop on the MST, it is not added. At the end of the flooding, the MST has recorded the paths followed by the water to merge the different lakes. This algorithm corresponds to Bohuslav's algorithm for the calculation of the Minimum Spanning Tree.

On the MST, there is a unique path between any pair of nodes. Suppressing an edge of this tree produces a forest with two trees, which corresponds to a partition of the image into two regions. In the same way, suppressing $n-1$ edges of the tree partitions the image into n regions. Removing a variable number of edges of the tree produces a segmentation pyramid. It is then a question of knowing which edges to remove to obtain meaningful segmentations. The edge weights representing a dissimilarity measure between neighbouring regions, it seems reasonable to suppress them by decreasing weights. In this way, the most different regions will be separated first in the hierarchy.

The selection of the edge weights is a crucial issue to obtain meaningful segmentations. The approach used here consists of weighting the edges of the

neighbourhood graph with a local dissimilarity measure based on a colour distance. The MST of the graph is then computed and its edges are simultaneously re-weighted based on the volume extinction value [16], which can be seen as a global dissimilarity measure. When an image is flooded by placing sources on all its minima, every time that two lakes meet an absorption takes place. The lake with smaller volume is considered to be absorbed by the lake with larger volume. When a lake is absorbed by a larger lake, its volume at the moment of the absorption is called the volume extinction value of the corresponding catchment basin. At the end of the flooding each catchment basin except one has been assigned an extinction value. The same procedure can be used during the graph flooding, where the volume of a region can be approximated as its surface multiplied by the current flooding level. The volume extinction values rate the regions in a way which is close to human perception, as they take into account both the size and contrast of the regions. Figure 1 shows a comparison between the 15 best regions found using a dynamics criterion [4] and the volume extinction values.



(a) Original image (b) Dynamics (c) Volume extinction values

Fig. 1. Best 15 regions as found using the dynamics and the volume extinction values.

Both the MST and the extinction values can be calculated by storing some extra information during the graph flooding.

Figure 2 shows three different resolution levels of the segmentation pyramid for four consecutive images belonging to the same block (for a block of 5 images).

4. Pyramid projection

As treating a whole video sequence into one single block is not possible nor convenient, a projection step becomes necessary to put past and present information into correspondence.

For this purpose, the sequence is divided into overlapping blocks (one image overlap), and two continuity conditions are imposed on the common image between two blocks:

– The fine partition must be exactly the same on the common image for the two blocks.

Fig. 2. Three levels of the segmentation pyramid for four consecutive images belonging to the same block.

– The hierarchy of the regions that exist in the two blocks should be preserved. New regions will be accommodated into this hierarchy.

In order to fulfil the first condition, the marker image used to produce the fine partition is modified. Section 3.1 described how 3-D flat or pseudo-flat zones were detected and used as markers for constructing the watershed of a gradient image. The marker image is therefore a 3-D image containing the detected flat zones. Now the first frame of the block containing the flat zones is replaced by the last frame of the fine partition corresponding to the previous block. The flat zones of the remaining images of the block are then relabelled to establish a correspondence with the regions of the imposed partition. Flat zones that do not obtain a label from the partition of the previous block are considered as new regions. The resulting 3-D image is then taken as marker for the watershed in the current block.

To satisfy the second condition, a recursive approach is used to dynamically update the MST to represent the sequence up to the current block.

Consider that the block N is currently being processed, and that an updated tree ϕ_{N-1} is available representing the segmentation pyramid up to the block $N-1$. The new tree ϕ_N is obtained by incorporating into the existing tree ϕ_{N-1} all the regions newly appeared in block N. In order to accommodate the new regions into the hierarchy, their volume extinction values must be calculated.

The nodes corresponding to the new regions will then be plugged into ϕ_{N-1} with edges weighted with this measure.

For the calculation of the extinction values for the new regions, a complete neighbourhood graph G_N is computed for the current block. Also, a new graph ϕ_N is obtained from ϕ_{N-1} by adding a node to ϕ_{N-1} for each new region, but with no link. To determine the links between the new nodes, and between the new and the old nodes, a *partial flooding* of G_N is carried out. In the partial flooding, only the nodes of the graph corresponding to new regions are flooded. Flooding the existing nodes is not necessary, as they are already linked in ϕ_N. When two new regions are joined by the flooding, one of them extinguishes as explained in Section 3.3. An edge is then added to ϕ_N between these two nodes, weighted with the volume of the extinguished region. When a new node meets with an old node, the new node is considered as extinguished (without volume comparison) and an edge is added to ϕ_N, weighted with the volume of this region. At the end of this procedure, a tree ϕ_N has been obtained. The paths between regions existing in ϕ_{N-1} are preserved with the same weights. New regions have been plugged into the tree, weighted with their volume extinction values.

At each projection step the tree ϕ_N grows by adding the new regions into the hierarchy while retaining previous information.

Figure 3 shows the results of the pyramid projection for images 1, 40 and 100 of the *Mother and Daughter* sequence.

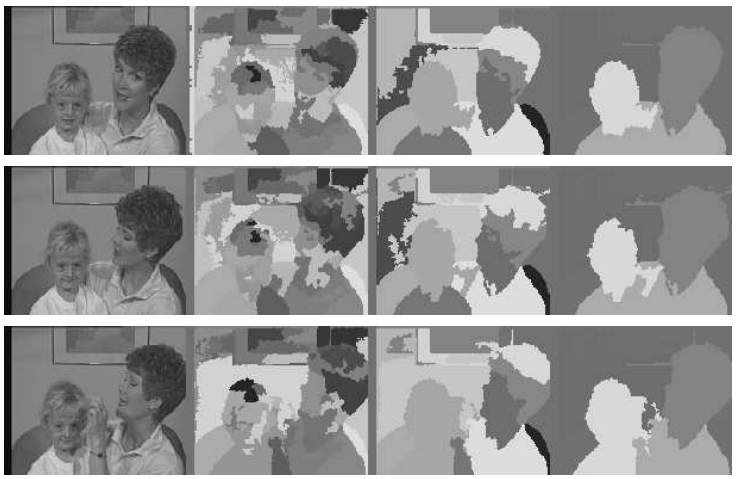


Fig. 3. Three different levels of the segmentation pyramid for images 1, 40 and 100 of the Mother and Daughter sequence. The segmentation pyramid has been projected block-to-block to span the whole sequence.

5. User interaction

This section describes some possibilities for user interaction with the segmentation pyramid. From the user point of view, the interaction simply consists of re-segmentations and mergings of regions, and marker drawing. The tree representation is not visible to the user.

Three types of interactions are proposed: automatic segmentation into a certain number of regions, local re-segmentations/mergings of regions and segmentation from markers.

5.1. Selection of the total number of regions

A starting point in the segmentation process could be for the user to ask for the image to be segmented into a number n of regions. This corresponds to a request for an automatic segmentation of the image into n regions. Such a request may easily be satisfied by suppressing the $n-1$ edges of the tree with highest weight. The interaction can be presented in a very intuitive way by means of a sliding bar which slides up and down to have more or less resolution.

5.2. Local interactions

The interaction type described in the previous section treats the image as a whole, finding the n best regions. However, the user may be interested in having some regions 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. This is done by locally suppressing/adding edges from/to the tree. Two operations allow the user to locally navigate up and down the pyramid.

In the *refine* operation, the user clicks on a certain area 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 $n-1$ (n being the parameter specified by the user) edges of highest weight are suppressed, but this time only the edges inside the selected region are considered.

In the *coarsen* operation, the user selects a region with a mouse click, and again a parameter n may be specified. The $n-1$ most similar neighbouring regions are merged to the selected one. 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 actions carried out by the user. As a starting step, an automatic segmentation into 10 regions is obtained. The user then clicks on the helmet and requests a re-segmentation of the region into two regions. The helmet is then separated from the background on all the images of the sequence where the helmet is present. The user then requests a re-segmentation of the shoulder region, in order to separate it as well from the background. In this way, finer or coarser segmentations can be obtained on certain areas by simple mouse-clicking.

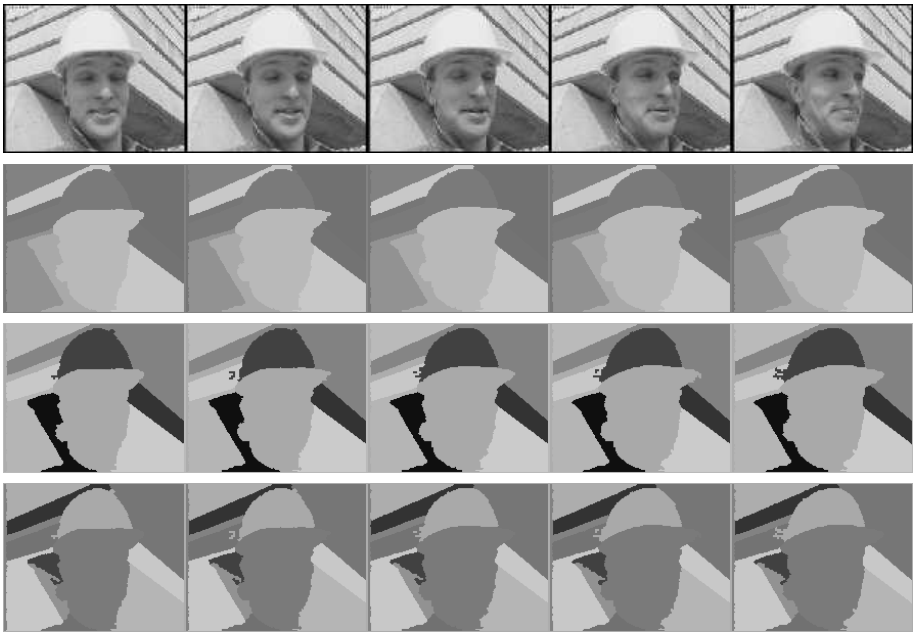


Fig. 4. Sequence of interactions, shown for one block of the video sequence. Top to bottom: original image, automatic segmentation into 10 regions, re-segmentation of the helmet area, re-segmentation of the shoulder.

6. Segmentation from markers

Finally, another type of interaction is the classical marker drawing. The user is asked to roughly mark the objects of interest, including the background. The markers are imposed on the MST and the resulting segmentation is obtained for the whole sequence. An algorithm can be found in [8]. As a segmentation pyramid is available, further re-segmentations/mergings are possible if the regions obtained are not completely satisfactory.

7. Conclusions

We have presented a technique to create a segmentation pyramid corresponding to a video sequence. Instead of a single partition, a segmentation pyramid is available. This approach is very well suited to interactivity as it allows the user to build the desired video object by taking regions from different resolution levels and frames. The correspondence between regions is made *at all resolution levels* for all the images of the sequence. In this way, the actions carried out by the user automatically affect all the images of the video sequence.

References

1. J. Cichosz and F. Meyer. Morphological multiscale image segmentation. *Workshop on Image Analysis for Multimedia Interactive Services, WIAMIS'97*, 1997.
2. L. Garrido, P. Salembier and D. Garcia. Extensive operators in partition lattices for image sequence analysis. *IEEE Transactions on Image Processing*, April 1998.
3. C. Gomila and F. Meyer. Tracking of video objects for videophony applications. *Proc. Workshop on Image Analysis for Multimedia Interactive Services, WIAMIS'99*, Berlin, May-June 1999.
4. M. Grimaud. New measure of contrast: dynamics. *Proc. Image Algebra and Morphological Processing III, SPIE*, San Diego CA, 1992.
5. B. Marcotegui and F. Meyer. Bottom-up segmentation of image sequences for coding. *Annales des Télécommunications*, 52(7–8):397–407, 1997.
6. B. Marcotegui, P. Correia, F. Marqués, R. Mech, R. Rosa, M. Wollborn and F. Zanoguera. A video object generation tool allowing friendly user interaction. *Proc. IEEE International Conference on Image Processing, ICIP'99*, Kobe, Oct. 1999.
7. F. Marqués and J. Llach. Tracking of generic objects for video object generation. *Proc. IEEE International Conference on Image Processing, ICIP'98*, Chicago, 1998.
8. F. Meyer. Minimum spanning forests for morphological segmentation. *Mathematical Morphology and its Applications to Image Processing, ISMM'94*, pages 77–84, 1994. Kluwer Academic Publishers.
9. F. Meyer. Morphological segmentation on a neighborhood graph. *Acta Stereologica*, 16(3):175–182, 1997.
10. F. Meyer. From connected operators to levelings. *Mathematical Morphology and its Applications to Image Processing, ISMM'98*, pages 191–198, June 1998.
11. F. Meyer. The levelings. *Mathematical Morphology and its Applications to Image Processing, ISMM'98*, pages 199–206, June 1998.
12. A. Montanvert, P. Meer and A. Rosenfeld. Hierarchical image analysis using irregular tessellations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13:307–316, April 1991.
13. O. Morris, M. Lee and A. Constantinides. Graph theory for images analysis: an approach based on the shortest spanning tree. *Proc. IEE*, 133(2):146–152, April 1986.
14. P.F.M. Nacken. Image segmentation by connectivity preserving relinking in hierarchical graph structures. *Pattern Recognition*, 28(6):907–920, 1995.
15. X. Shen, M. Spann and P. Nacken. Segmentation of 2D and 3D images through a hierarchical clustering based on region modelling. *Pattern Recognition*, 31(9):1295–1309, 1998.
16. C. Vachier. Extraction de charactéristiques, segmentation d'image et morphologie mathématique. PhD thesis, Ecole Nationale Supérieure des Mines de Paris, 1995.
17. F. Zanoguera, B. Marcotegui and F. Meyer. A toolbox for interactive segmentation based on nested partitions. *Proc. IEEE International Conference on Image Processing*, Kobe, Oct. 1999.