

# Oversegmentation Reduction Via Multiresolution Image Representation

Maria Frucci, Giuliana Ramella, and Gabriella Sanniti di Baja

Institute of Cybernetics "E.Caianello", CNR, Pozzuoli (Naples), Italy  
{m.frucci, g.ramella, g.sannitidibaja}@cib.na.cnr.it

**Abstract.** We introduce a method to reduce oversegmentation in watershed partitioned images, that is based on the use of a multiresolution representation of the input image. The underlying idea is that the most significant components perceived in the highest resolution image will remain identifiable also at lower resolution. Thus, starting from the image at the highest resolution, we first obtain a multiresolution representation by building a resolution pyramid. Then, we identify the seeds for watershed segmentation on the lower resolution pyramid levels and suitably use them to identify the significant seeds in the highest resolution image. This is finally partitioned by watershed segmentation, providing a satisfactory result. Since different lower resolution levels can be used to identify the seeds, we obtain alternative segmentations of the highest resolution image, so that the user can select the preferred level of detail.

## 1 Introduction

Any image analysis task requires a segmentation step to distinguish the significant components of the image, i.e., the foreground, from the background.

A frequently adopted segmentation technique is based on the watershed transformation, [1,2]. Basically, watershed transformation originates a partition of a gray-level image into regions characterized by a common property, such as an almost homogeneous gray-level distribution. The partition regions are then assigned to either the foreground or the background, by taking into account the properties expected to characterize the two sets. If the user perceives as more significant the regions with locally higher (lower) intensity, hence the regions locally lighter (darker), the assignment criterion could be based on the maximal difference in gray-level among adjacent partition regions. This problem is still partially open, especially because its solution is strongly conditioned by the quality of the image partition.

Unfortunately, watershed segmentation is generally affected by excessive fragmentation into regions. This, besides requiring a suitable complex process to reduce the number of seeds from which the partition originates, may bias the successive assignment of the partition regions to the foreground and the background. We think that an effective way to reduce the number of seeds can be found by resorting to multiresolution representation. If a gray-level image is observed at different resolutions, only the most significant regions will be perceived at all resolutions, even if in a more coarse way at lower resolution. Regions that, at the highest resolution image, can be interpreted as noise or constitute fine details are generally not preserved when resolution

decreases. Thus, if the seeds for watershed segmentation of the highest resolution image are identified in a lower resolution level, the resulting partition is expected to be characterized by a reduced number of regions, corresponding to the most significant image parts. In this communication, we face this problem.

Starting from a gray-level image, we create a multiresolution representation by building a resolution pyramid. To this purpose, we modify the algorithm illustrated in [3,4]. Here, we use a different  $3 \times 3$  mask of weights to compute the gray-level of (parent) pixels at lower resolution, in order to obtain more faithful representations of the original input at all resolutions. Then, we identify the seeds for watershed segmentation at one of the lower resolution levels. These seeds are suitably projected onto the highest resolution level of the pyramid and are used to select among the seeds originally detected at that resolution, only those corresponding to the most significant regions. All other seeds originally found in the highest resolution image undergo a suitable removal process, aimed at merging the corresponding partition regions. The watershed segmentation of the highest resolution image is finally accomplished, by using only the seeds that survived the removal process. Different segmentations are suggested for the same image, depending on the pyramid level used to identify the seeds to be projected and, hence, on the desired detail of information to be preserved.

The paper is organized as follows. In Section 2, we briefly discuss the method to build the resolution pyramid. In Section 3, we illustrate the process that, starting from the seeds identified at a selected lower resolution level, allows us to identify among all seeds at the highest resolution, only those regarded as the most significant. In Section 4, we show the results of the watershed segmentation of the highest resolution image by using seeds computed at lower resolution levels. We also show the results obtained after we apply to the partitioned image a method to distinguish the foreground from the background. Finally, in Section 5 we give some concluding remarks.

## 2 The Resolution Pyramid

We consider images where the locally darker regions (i.e., those whose associated gray-level is locally lower) constitute the foreground. In our images, gray-levels are in the range  $[0, 255]$ . Let  $G_1$  be a  $2^n \times 2^n$  gray-level image. If the input image has a different number of rows/columns, a suitable number of rows/columns is added to build  $G_1$ . Pixels in the added rows/columns are assigned the maximum gray-level present in the original image, i.e., are seen as certainly belonging to the background. Through this paper,  $G_1$  is interpreted as a 3D landscape, where for every pixel in position  $(x,y)$ , its gray-level plays the role of the z-coordinate in the landscape. This interpretation is helpful to describe our process in a simple and intuitive way.

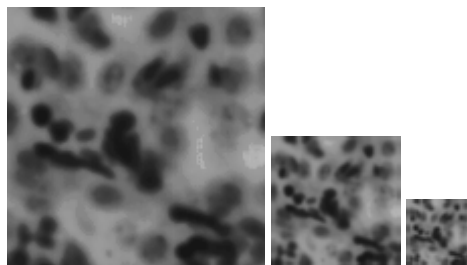
A multiresolution image representation is of interest in many contests, since it provides from coarse to fine representations of an input image, always preserving the most relevant features. In this framework, resolution pyramids are among the most common representation systems [5]. We here modify the discrete method [3,4] to build a resolution pyramid. In [3,4], we focused both on shift invariance and topology preservation. Here, we are still interested in shift invariance and aim at a more faithful computation of gray-levels for the parent pixels.

Pyramid construction is based on a recursive subdivision into quadrants of  $G_1$ . At each recursion step, resolution decreases by four and, in principle, the process terminates when the image including one single pixel is built. Actually, we do not compute resolution levels including less than  $32 \times 32$  pixels, as they would give too coarse representations of  $G_1$ . For the running example shown in this paper, the base of the pyramid, level 1, is the image  $G_1$  at full resolution ( $128 \times 128$ ), the next level of the pyramid, level 2, represents the image at a uniformly lower resolution ( $64 \times 64$ ), and the apex of the pyramid is the  $32 \times 32$  image, which constitutes level 3. We use a decimation process involving the use of a partition grid. When the grid is placed onto the current resolution image,  $G_k$ , the image is divided into blocks of  $2 \times 2$  children pixels, which correspond to parent pixels at the immediately lower resolution level  $G_{k+1}$ . Practically, we inspect in forward raster fashion only pixels belonging to even rows and columns of  $G_k$ , meaning that we use the bottom right child pixel in a block to find the coordinates of the parent pixel in  $G_{k+1}$ . Let us indicate with  $(i,j)$  the pixel in position  $(i,j)$ . For each inspected pixel  $(i,j)$  of  $G_k$ , the parent pixel in  $G_{k+1}$  will be  $(i/2,j/2)$ .

4	6	4
6	9	6
4	6	4

**Fig. 1.** The multiplicative mask of weights used to build the pyramid

To compute the gray-level of the parent pixel  $(i/2,j/2)$  in  $G_{k+1}$ , we average the gray-levels of  $(i,j)$  and of its eight neighbors in  $G_k$ . Since, the partition grid could be shifted on  $G_k$  and, hence, any pixel in the  $3 \times 3$  window centered on  $(i,j)$  could be the bottom right pixel of the block, we introduce a multiplicative mask of weights to evaluate the contribution given by the nine pixels in the  $3 \times 3$  window centered on  $(i,j)$ .



**Fig. 2.** The three levels of the pyramid computed for a  $128 \times 128$  input image

In this way, the gray-level of  $(i/2,j/2)$  will be computed almost independently of the position of the partition grid. To this aim, we consider the nine  $3 \times 3$  windows centered on  $(i,j)$  and on each of its eight neighbors. Pixel  $(i,j)$  is included in all nine windows, its horizontal/vertical neighbors are included in six windows and diagonal neighbors in four windows. The number of windows including a pixel constitutes the corresponding

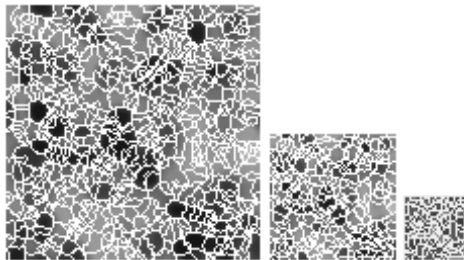
weight for the multiplicative mask. See Fig. 1. It can be noted that the weights in our mask are practically midway between the Gaussian and the uniform weights.

The gray-levels computed by using the mask are then normalized to assume values in the range  $[0, 255]$ . Once the computation of  $G_{k+1}$  is done, the successive lower resolution level is built by the same procedure. The pyramid built for the running example is shown in Fig. 2.

### 3 Selection of the Significant Seeds

At each level  $k$ , the gradient image  $\nabla_k$  corresponding to  $G_k$  is interpreted as a 3D landscape. This interpretation is useful to illustrate in a simple manner the paradigm on which watershed segmentation is founded. High gray-levels correspond in the landscape to mountains, while low gray-levels correspond to valleys. If the bottom of each valley is pierced and the landscape is immersed in water, then valleys are filled in by water. Filling starts from the deepest valleys and then continues through less and less deep valleys. These begin to be filled as soon as the water level reaches their bottom. A dam (watershed) is built wherever water could spread from a basin into the close ones. When the whole landscape has been covered by water, the basins are interpreted as the parts into which the landscape is partitioned.

The regional minima found in  $\nabla_k$  are generally used as the seeds starting from which watershed transformation generates a partition of  $\nabla_k$  (and, hence, of  $G_k$ ) into regions characterized by some gray-level homogeneity. See Fig. 3, where the watershed lines found by using the regional minima in the three gradient images at the three pyramid levels are shown in white. There are respectively 709, 280, and 116 seeds (and, hence, basins), for pyramid levels 1, 2 and 3. We note that the image at level 1 is affected by excessive fragmentation, caused by the very large number of regional minima. Some (heavy) process is generally accomplished to select among the seeds found in  $\nabla_1$  only those that are significant to correctly partition  $G_1$ . See, e.g., [6].



**Fig. 3.** The watershed lines (white) found at the three pyramid levels starting from the relative regional minima, superimposed on the three gray-level images. The found basins are 709, at level 1, 280, at level 2, and 116, at level 3.

Since  $G_1$  is well represented even at the lowest resolution level of the pyramid (see Fig. 2) and, in turn, the seeds found in  $\nabla_3$  are considerably less than those found in  $\nabla_1$ , we will use the seeds found in  $\nabla_3$  to select among the seeds detected in  $\nabla_1$  the

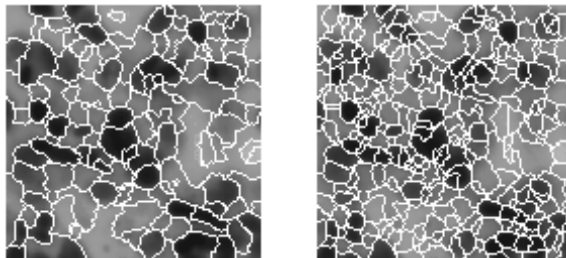
most significant ones and obtain, in this way, a less fragmented partition of  $G_1$ . To this aim, we project the seeds from level 3 to level 1. This is possible due to the fact that our pyramid construction method preserves the links parent-children. Thus, for each pixel at level 3 we can easily identify its descendants at level 1. Obviously, since any parent pixel at level 3 has four children at level 2 and each of these children has in turn four children at level 1, for each seed pixel found in  $\nabla_3$  we identify a  $6 \times 6$  block of descendants in  $\nabla_1$ . See Fig. 4 middle.



**Fig. 4.** Seeds found at level 1, left; descendants at level 1 of the seeds found at level 3, middle; descendants remaining after reduction (see text), right

Our idea is to regard as significant a seed originally detected in  $\nabla_1$ (Fig. 4 left), only provided that its associated partition region (Fig. 3, level 1) includes at least one descendant of the seeds found at level 3 (Fig. 4 middle). All other seeds detected in  $\nabla_1$  are regarded as non significant and, by means of a *flooding* process, the corresponding partition regions are merged.

Due to the large size of the sets of descendants originated from the seeds found at level 3, still too many seeds would be preserved at level 1. To reduce their number, we do the following process. Let  $M$  be the number of connected components of descendants,  $CCD_i$ , found at level 1. In the gradient image  $\nabla_1$ , we inspect the  $M$  sets  $C_i$  of pixels with homologous positions with respect to the pixels of the sets  $CCD_i$ . In each set  $C_i$ , we identify and preserve as seeds only the pixels, whose gray-level is minimal with respect to the gray-levels of the other pixels of  $C_i$ . All other descendants are removed (Fig. 4 right). Flooding is then applied at level 1 to merge all partition regions that do not include at least one descendant that survived the removal process.



**Fig. 5.** Partition of  $G_1$  at level 1 into 133 basins, by using the seeds found at level 3, left, and partition of  $G_1$  at level 1 into 284 basins, by using the seeds found at level 2, right

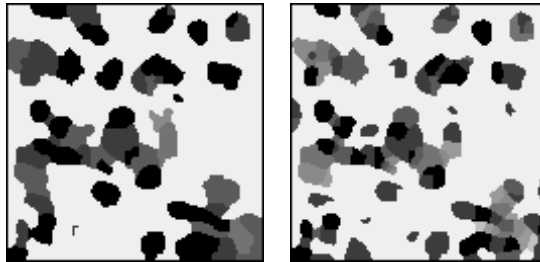
The watershed lines of the partition of  $G_1$ , obtained by using the seeds found at level 3 to identify the significant seeds at level 1, are superimposed in white on  $G_1$  in Fig. 5 left. By selecting a different lower resolution level, we can use the seeds found there to identify the significant seeds among those detected at level 1. For example, by using the seeds found at level 2 and by applying the same process described above, the watershed partition of  $G_1$  shown in Fig. 5 right is obtained.

By comparing the results shown in Figs. 5 and 3, we see that a considerable reduction of the fragmentation is obtained, as expected. To show that the obtained partitions are significant, we briefly illustrate in the following Section a process that allows us to assign to either the background or the foreground the partition regions.

## 4 Region Assignment to Foreground and Background

The model that we follow to assign the watershed partition regions to either the background or the foreground is inspired by visual perception. In our gray-level images, the foreground is perceived as characterized by locally lower intensity. We assume that the border separating the foreground from the background is placed wherever strong differences in gray-level occur. Assignment is done by means of a process requiring two steps. A more detailed description of this process can be found in [7].

The first step of the process globally assigns to the foreground and to the background the regions characterized by locally minimal and locally maximal average gray-levels (valleys and peaks and in the landscape representation).

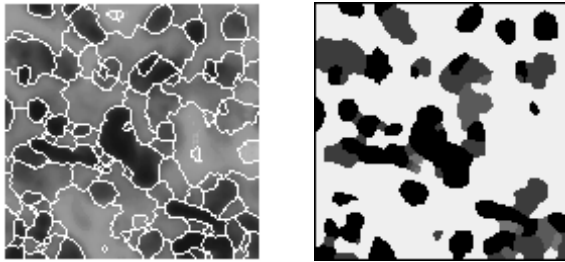


**Fig. 6.** Foreground components identified in correspondence of the two alternative partitions shown in Fig. 5. Gray-tones are related to the significance of the regions (see text). Darker regions are more significant.

The second step is guided by the maximal gray-level difference  $\Delta$  between all pairs of adjacent regions. It assigns to the foreground and to the background the partition regions placed along the slopes in the landscape. This step is iterated (with a new  $\Delta$  computed at each iteration) until all regions are assigned. Two cases are possible depending on the number  $N$  of adjacent regions with maximal  $\Delta$ . If  $N=1$ , the darker region in the pair of regions with difference  $\Delta$  is (locally) assigned to the foreground, while the lighter region is assigned to the background. In fact, in correspondence with these two adjacent regions with difference  $\Delta$ , we assume that a transition from background to foreground occurs. Based on the same assumption, we (globally) assign to

the foreground (background) also all the regions equally dark or darker (equally light or lighter) than the region, in the pair of adjacent regions with difference  $\Delta$ , assigned to the foreground (background). If  $N > 1$ , a conflictual assignment is possible if, for any of the pairs characterized by the maximal  $\Delta$ , say the  $i$ -th pair, the darker region, say  $DR_i$ , happens to be lighter of the lighter region, say  $LR_j$ , in another pair of regions characterized by the maximal  $\Delta$ , say the  $j$ -th pair. In fact,  $DR_i$  should be assigned to the foreground, by taking into account the average gray-levels in the  $i$ -th pair, but it should be assigned to the background, by taking into account the relation between the average gray-levels of  $LR_i$  and  $DR_j$ . If this is the case, a local process, still based on the maximal  $\Delta$ , is accomplished to assign the regions along the slope including  $DR_i$ . Once the conflictual cases have been treated and all the pairs with the maximal  $\Delta$  have been locally assigned, the same global process done for  $N=1$  is safely applied.

A relevance parameter, taking into account the perceptual significance, is also set for the regions assigned to the foreground, which allows us to rank foreground components. The relevance parameter for regions detected during the first step assumes value 1 if the region (i.e., a valley in the landscape representation) has an average gray-level smaller than that characterizing all peaks in the landscape (i.e., all regions assigned to the background). It assumes value 2 otherwise, meaning that such a valley, though assigned to the foreground, has a perceptual significance smaller than that pertaining the other valleys. During the second step, the relevance parameter of a region assigned to the foreground is set to the number of foreground regions in the shortest path, linking that region to the most relevant part in the same foreground component. The result of this process applied to the watershed partitions shown in Fig. 5 is shown in Fig.6, where darker gray-tones correspond to more significant regions.



**Fig. 7.** Result of the segmentation process [6] applied to  $G_1$ . The final segmentation into 119 basins, left, and the result of the assignment process to identify the foreground, right.

If we apply to  $G_1$  the high performance, but computationally more expensive, segmentation algorithm [6], we obtain the result shown in Fig. 7. Also in this case, region assignment is done by using the algorithm [7]. We can now compare Figs. 6 and 7, by using Fig. 7 as a reference. We note that even starting from seeds found at low resolution (levels 2 and 3 of the pyramid), the results shown in Fig. 6 are comparable with those in Fig. 7. Obviously, more details are identified if the seeds are taken at level 2, as it is expected because of the higher resolution of level 2 with respect to level 3.

## 5 Conclusion

We have introduced a method to reduce the excessive fragmentation of gray-level images into regions, when watershed segmentation is used. Our method is based on the use of a multiresolution representation of the input image and on the detection of the most significant seeds for segmenting the highest resolution image, guided by the seeds found at lower resolution. The underlying idea is that the most significant components perceived in the highest resolution image will remain identifiable also at lower resolution. Thus, starting from the highest resolution image, we first build a resolution pyramid. Then, we identify the seeds for watershed segmentation on one of the lower resolution pyramid levels and suitably use them to identify the significant seeds in the highest resolution image. This image is finally partitioned by watershed segmentation, providing a satisfactory result. Since different lower resolution levels can be used to identify the significant seeds at the highest resolution, we obtain alternative segmentations of the highest resolution image, among which the user can select the best suited one for the specific task.

The performance of the method has been shown on a sample image only, but we have tested our procedure on a large set of biological images. The obtained results have been judged as satisfactory by the experts in the field.

## References

1. S.Beucher, C.Lantuejoul, "Use of watersheds in contour detection", *Proc. Int. Workshop on Image Processing, Real-Time Edge and Motion Detection/Estimation*, Rennes, France, 1979.
2. S.Beucher, F.Meyer, "The morphological approach of segmentation: the watershed transformation", in *Mathematical Morphology in Image Processing*, E.Dougherty (Ed.) M.Dekker, New York, 433-481, 1993.
3. G.Borgefors, G.Ramella, G.Sanniti di Baja, "Shape and topology preserving multi-valued image pyramids for multi-resolution skeletonization", *Pattern Recognition Letters*, 22, 741-751, 2001.
4. G.Ramella, G.Sanniti di Baja, "Grey level image components for multi-scale representation", in *Progress in Pattern Recognition, Image Analysis and Applications*, A.Sanfeliu, J.F.Martinez Trinidad, J.A.Carrasco Ochoa (Eds.), LNCS 3287, Springer, Berlin, 574 - 581, 2004.
5. A.Rosenfeld (Ed.), "Multiresolution Image Processing and Analysis", Springer, Berlin; 1984.
6. M.Frucci, "A novel merging method in watershed segmentation", *Proc. 4th Indian Conf. on Computer Vision, Graphics, and Image Processing*, Applied Publishing Private Ltd, Kolkata, India, 532-537, 2004.
7. M.Frucci, C.Arcelli, G.Sanniti di Baja, "Detecting and Ranking Foreground Regions in Gray-Level Images", in "Brain, Vision and Artificial Intelligence", M. De Gregorio, V. Di Maio, M. Frucci, C. Musio (Eds.), LNCS 3704, Springer, Berlin, 2005 (in press).