Semi-automatic watershed merging method

Jakub Smołka[1][*], Maria Skublewska-Paszkowska[1][†]

[1]*Institute of Computer Science, Faculty of Electrical Engineering and Computer Science,
Lublin University of Technology,
ul. Nadbystrzycka 36b, 20-618 Lublin, Poland.*

Abstract – Watershed transformation frequently produces over-segmented images. The authors propose a solution to this problem. It utilizes hierarchical cluster analysis for grouping watersheds which are treated as objects characterized by a number of attributes. Initially the watershed merging method was meant only for gray-scale images, but later it was adapted for colour images. This paper presents further extension of the method that allows it to either automatically select the number of classes or to provide a hint as to which numbers in a specified range should be considered first. Segmentation quality assessment functions for colour images are presented. The results obtained using an extended watershed merging method are discussed. The examples of segmentations selected by the method, along with the graphs of assessment functions, are shown.

1 Introduction

Over-segmentation is the major concern when using watershed transformation. In [**1**], the authors propose a method for merging watersheds in over-segmented images. It consists in using hierarchical cluster analysis for grouping watersheds. The method was tested on the gray-scale CT and MRI scans. In [**2**] its use was extended to colour images. The hierarchical clustering methods are versatile because they can be considered modular. It is possible to change, for example, the clustering strategy without modifying any other component such as similarity coefficient. Such changes can be made quickly and easily if the method is properly implemented. Additionally, new clustering methods, similarity/dissimilarity measures and watershed attributes can be introduced. This way our approach can be easily extended or adapted for a

---

[*]jakub.smolka@pollub.pl

[†]maria.paszkowska@pollub.pl

particular task. This paper focuses on a new addition to our method that provides help with selecting the number of classes in the final segmentation.

2 Merging watersheds with hierarchical cluster analysis methods

The following section provides a summary of our method including the new addition. The approach consists in treating watersheds in an over-segmented image as objects characterized by a set of attributes. A list of available attributes can be extended, however, in our tests the following attributes are used: size, mean value, variance, standard deviation, value spread, minimal value and maximal value. In the case of colour images attributes are computed for each colour component separately (for example, three average values were computed: one for each of the RGB colour components).

Attribute values may differ by orders of magnitude. That is why optionally they are standardized in order to equalize their contribution to dissimilarity values. We used four different methods: (1) standardization that causes attributes to have a mean value of 0 and a standard deviation of 1, (2) linear scaling of the attribute values to the range of [0,1], the lowest value corresponds to 0 and the highest to 1, (3) linear scaling similar to (2) but only the highest value is guaranteed to be equal to 1 (the lowest is not necessarily equal to 0), (4) linear scaling that causes the normalized data to sum up to 1. The equations can be found in [**1, 3**]. It is important to note that attributes (for example average values) computed for different colour components are standardized together (i.e. treated as one attribute) in order to preserve colour information.

Watersheds' resemblances are measured with a similarity or dissimilarity (distance) coefficient. There are a number of coefficients that can be used. The authors implemented the following dissimilarity measures: Euclidean distance, average Euclidean distance, Canberra metric, Bray-Curtis coefficient and coefficient of shape difference. Additionally, two similarity measures were used – namely: the cosine coefficient and the correlation coefficient. Their values belong to the interval of $[-1; 1]$. However, in order to simplify the implementation of the algorithm and to increase compatibility with standardization methods [**3**], they are converted to dissimilarity measures that attain values from the range of $[0; 1]$. The equations for the coefficients can be found in [**1, 3**]. Using one of the mentioned measures, all possible (between all possible pairs of objects) similarity values are computed and saved in the similarity matrix.

The watersheds are grouped using one of the clustering methods. These include: complete linkage (CLINK), single linkage (SLINK), unweighted pair-group method using the arithmetic averages (UPGMA) and the Ward's minimum variance method [**1, 3, 4**]. Regardless of the selected method, the result is a similarity hierarchy. It can be represented as a tree in which a node represents merging of two clusters and has a similarity measure associated with it. Once the similarity tree is available, obtaining a given number of clusters is straightforward. It can be thought of as cutting the tree at

a given level (of similarity). The tree is divided into two parts. The top part containing the final clustering steps is discarded. The mergers in the bottom part are carried out.

The level at which the similarity hierarchy is cut determines the final number of clusters and hence the number of classes in the segmentation. There are different strategies as to how to select this level. They fall into two categories: general-purpose and specific-purpose. One of the general methods selects the largest gap (in terms of dissimilarity coefficient) between consecutive cluster mergers [3]. However, the authors propose a specific method described in more detail in the following section.

3 Selecting the number of classes in final segmentation and segmentation quality assessment

Our method is a specific-purpose solution for determining the level at which the similarity tree is cut. It utilizes three different functions that are designed specifically for segmentation quality assessment. The functions are described below.

In this section the following symbols are used: I – original image, I_a – image with classes' average values (colour components are denoted by a c superscript – for example, I_a^c is the component c of the average values' image), S – set containing all colour components, x – a pixel, n – pixel count, k – number of different colours in the original image, n_c – number of classes in the final segmentation, A_i – number of pixels in i^{th} class, $r(a)$ – number of classes with pixel count equal to a, e_i – colour error of ith class, R_i – i^{th} class, A_{max} – the area of the largest class expressed in pixels (it can be determined by finding the maximum of a histogram of the label image I_l), N,M – image width/height expressed in pixels.

The first assessment function is one introduced by Liu and Yang [5]. Its value is usually normalized by the image size and a constant factor of $1/1000$. It is given by:

$$F(I) = \frac{1}{1000 \cdot N \cdot M} \cdot \sqrt{n_c} \cdot \sum_{i=1}^{n_c} \frac{e_i^2}{\sqrt{A_i}} \tag{1}$$

The colour error e_i is given by $e_i = \sum_{x \in R_i} \sqrt{\sum_{c \in S}(I^c(x) - I_a^c(x))^2}$ [5, 6]. The function F consists of three factors. The first normalizes the function value. The second $\sqrt{n_c}$ penalizes over-segmentation. The third factor favours uniform regions and worsens the score of segmentations which have classes with a large colour error e_i.

M. Borsotti, P. Campadelli and R. Schettini in [7] criticized F (equation 1) and proposed an alternative function F' that according to them performs better. It is given by:

$$F'(I) = \frac{1}{10000(N \cdot M)} \cdot \sqrt{\sum_{a=1}^{A_{max}} [r(a)]^{1+1/a}} \cdot \sum_{i=1}^{n_c} \frac{e_i^2}{\sqrt{A_i}} \tag{2}$$

It replaces the $\sqrt{n_c}$ with $\sqrt{\sum_{a=1}^{A_{max}} [r(a)]^{1+1/a}}$ which puts more emphasis on penalizing segmentations that have many small regions. Additionally, M. Borsotti et.al. proposed another assessment function whose values, in their opinion, reflect the differences in

segmentation quality better that those of F'. The new function – Q is given by the following equation:

$$Q(I) = \frac{1}{10000(N \cdot M)} \cdot \sqrt{n_c} \cdot \sum_{i=1}^{n_c} \left[\frac{e_i^2}{1 + logA_i} + \left(\frac{r(A_i)}{A_i} \right)^2 \right] \tag{3}$$

The all aforementioned functions have built-in factors that penalize segmentations containing too many classes. This makes them well suited for determining the optimal number of classes and determining the level at which the similarity hierarchy should be cut.

In our solution look-up tables are generated for all possible cutting thresholds based on the information saved during clustering. The look-up tables allow generating final segmentation from the over-segmented image almost instantly. Using the tables, the watershed merging algorithm automatically searches the range, specified by the user, in which the desired number of classes lies. It does so by generating segmentations for all possible thresholds and computing values of selected quality assessment functions. As shown in section 4 the functions usually have multiple local minima. If the range containing the desired number of classes is narrow (and contains one local minimum) the algorithm can quickly find the optimal number of classes. If the range is wider, the algorithm can look for a global minimum or provide hints as to which tree division levels are significant (these are the levels corresponding to the local minima of the assessment function). Since generating segmentations (using look-up tables) is a fast process, they can be browsed in an interactive manner.

Additionally, quality assessment functions give the user the ability to evaluate the segmentation not only qualitatively but also quantitatively.

Below a complete outline of our method is provided: (1) The clustering parameters are specified. The parameters include: set of attributes, optional standardization method, dissimilarity measure and clustering strategy. Additionally, the range containing the desired number of classes and segmentation assessment function needs to be specified. (2) For each watershed attribute values are computed. In the case of colour images, each attribute produces as many values as there are colour components. They are calculated separately for each colour component. (3) If required, the attributes are standardized. All values produced by a particular attribute are standardized together in order to prevent colour information from being distorted. (4) Selected dissimilarity measure is used to construct the dissimilarity matrix. (5) The two most similar clusters are found (clusters with the minimal dissimilarity measure). Initially, each cluster consists of a single watershed. (6) The two clusters are merged. (7) The dissimilarity matrix is updated, according to the selected clustering strategy, in order to reflect the most recent merger. (8) A new node is added to the similarity tree. The node represents the most recent merger. (9) If there is more than one cluster left, the algorithm returns to step 5. (10) The fact that all the watersheds are in one cluster means that the complete similarity hierarchy is available. (11) The algorithm starts generating segmentations based on the similarity hierarchy. The selected level at which the tree is divided corresponds to the smallest number of classes in the specified range. (12) The
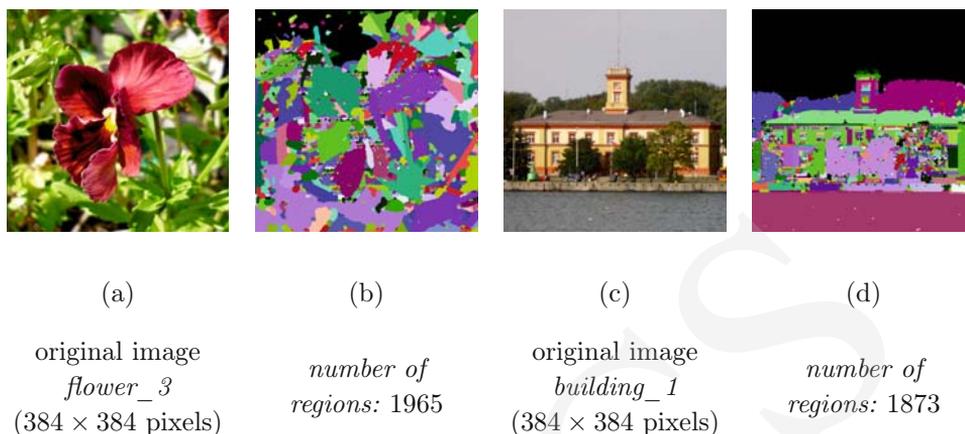
(a)	(b)	(c)	(d)
original image *flower_3* (384 × 384 pixels)	*number of regions:* 1965	original image *building_1* (384 × 384 pixels)	*number of regions:* 1873

FIG. 1. The test images and their over-segmented versions. The following parameters were used for preparing the over-segmented image – *flower_3*: gradient function *sum-of-squares-gradient*, threshold *5%*, merging level *10%*, *building_1*: gradient function *pca-gradient*, threshold *0%*, merging level *9%*.

similarity tree is divided into two parts. The top part of the tree is left out. All cluster mergers that are recorded in the bottom part of the tree are carried out on the corresponding watersheds – that is, the watersheds are grouped into final clusters. (13) The algorithm iterates through all clusters. Pixels that constitute watersheds belonging to a particular cluster are given a label unique to that cluster. This way image segmentation is obtained. (14) The value of the selected segmentation assessment function is computed. (15) Unless all segmentations in the specified range have been generated, the algorithm proceeds to generate a segmentation containing larger (by 1) number of classes. It selects the appropriate tree division level and returns to step 12. (16) The segmentation with the best assessment value is selected to produce the final segmentation. Alternatively, segmentations corresponding to the local minima of the assessment are presented to the human observer who then selects the best segmentation.

4   Results

Figures 1a and 1c show two test images: *flower_3* and *builting_1* (they are available at http://pluton.pol.lublin.pl/~jsmolka/2012_annales.

The test images were preprocessed with the use of a curvature anisotropic diffusion filter [**8, 9**] (there were used the following parameters: *conductance* $= 0.3$, $\Delta T = 0.12$ and 5 iterations). Filtering reduced noise is presented in the images.

The edges were extracted using two different gradient filters. In the case of the *flower_3* image a filter that finds gradient magnitude as the square-root of the sum of
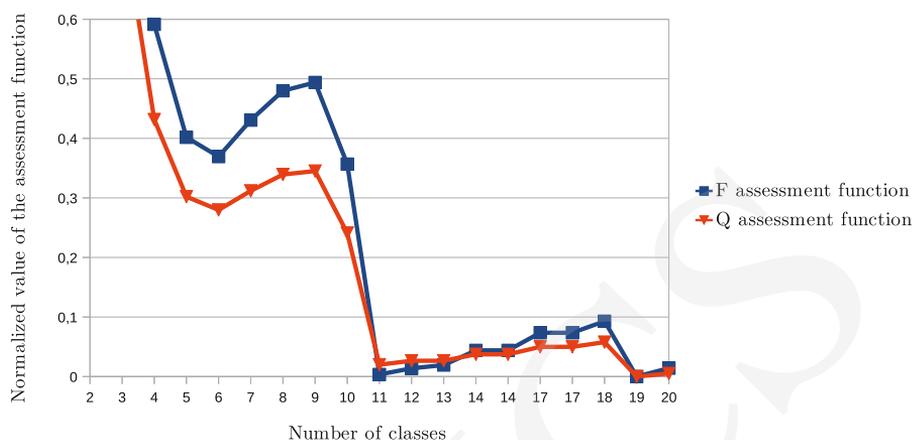
FIG. 2. Graph showing the dependence of the functions F and Q on the number of classes for the *flower_3* image. Function values were scaled to the interval of $[0; 1]$. The chart shows only the $[0; 0, 6]$ range in order to make local minima more visible.

squared partial derivatives of the individual image components [**9**] was used (*sum-of-squares-gradient*). The gradient image of *building_1* was computed using *pca-gradient* – an approach that defines gradient magnitude as the difference between the two largest eigenvalues in principal component analysis of the partial derivatives of the colour components [**10, 11, 12**].

The gradient image of *flower_3* test image was thresholded in order to reduce over-segmentation. The gradient image of *building_1* was not modified. After thresholding, the watershed transformation was performed. Then shallow catchment basins were merged with their neighbours if their depth did not exceed a value called merging level [**9**]. This allowed us to further reduce the over-segmentation. Values of both the merging level and the threshold that were used are specified as a percentage of maximal brightness in the gradient image. While selecting these values, special care needs to be taken to prevent thresholding and merging from removing important features in images (that is, from merging catchment basins belonging to different objects). It is worth noting that the preliminary over-segmentation reduction usually significantly speeds up the clustering process. The results of watershed transformation and preliminary over-segmentation reduction are shown in Figs 1b and 1d.

Watersheds in the over-segmented images were grouped using cluster analysis. The parameters used for the test images are specified in the captions of Figs 3 and 5.

Once the clustering was finished, segmentations containing from 2 to 20 classes were generated. Then the values of the functions F, F' and Q for each segmentation were

6 classes   11 classes   19 classes

Fig. 3. Segmentations of *flower_3* test image chosen based on the F and Q function values. The following parameters were used during the over-segmentation reduction – clustering method: *Ward's minimum variance*, watershed attributes: *average and standard deviation*, standardization: *none*.

calculated. Figure 2 shows a graph of the F and Q functions obtained for segmentations of the *building_1* test image and Figure 4 shows the analogous graph for the *flower_3* image. The function values were scaled to the interval [0, 1] for presentation purposes. Scaling makes the comparison between the functions easier. The F' was omitted because it turned out that after scaling its shape did not visibly differ from the F function. This was the case for both test images. With the F, F' and Q functions smaller values correspond to better segmentations. As shown in Figures 2 and 4, these functions can have multiple local minima. Finding the global minimum can be lengthy, because it would require generating all possible segmentations (typically the number of all possible segmentations is in the magnitude of thousands). That is why the range containing the expected number of classes is required by our segmentation method. Fortunately, because the goal of the method is to provide images that are not over-segmented, this range is usually small.

For the *flower_3* test image assessment the functions F, F' and Q have three local minima in the considered range. They are attained for 7, 11 and 19 classes. For the *building_1* all the functions have minima corresponding to 6, 11 and 19 classes but, additionally, the Q function has a less significant minimum corresponding to 9 classes. Segmentations selected by all three quality assessment functions are shown in Figures 3 and 5.

The graphs in Figures 2 and 4 show that segmentation quality assessment functions can provide clear hint as to what number of classes should be selected. They also show that increasing the number of classes does not always significantly increase the accuracy of the segmented image. This happens when, while travelling down through the similarity hierarchy, small classes are disjoined from larger clusters. The penalty
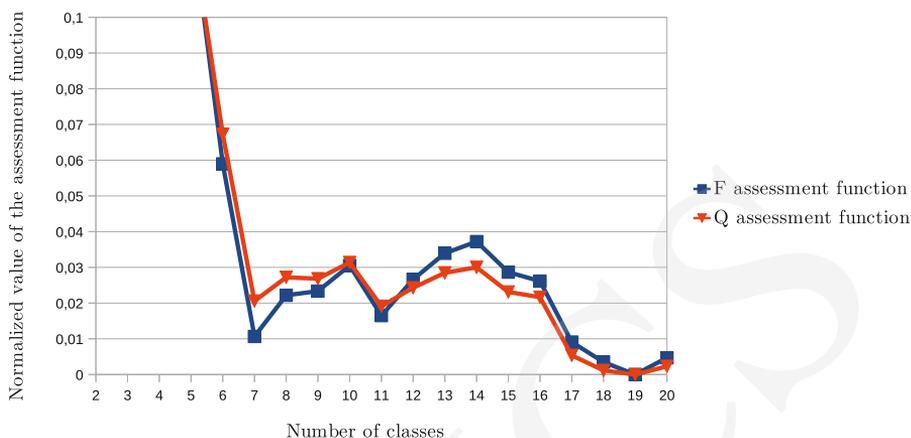
FIG. 4. Graph showing the dependence of the functions F and Q on the number of classes for the *building_1* image. The function values were scaled to the interval of $[0; 1]$. The chart shows only the $[0; 0,1]$ range in order to make local minima more visible.
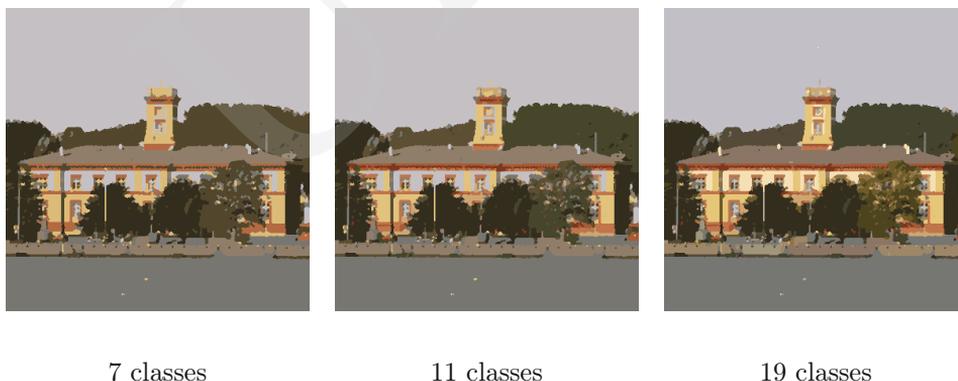

    7 classes   11 classes   19 classes

FIG. 5. Segmentations of *building_1* test image chosen based on the *F* and *Q* function values. The following parameters were used during the over-segmentation – clustering method: *CLINK*, watershed attributes: *average and variance*, standardization: *none*.

factor in the assessment function increases, but the colour error does not significantly decrease.

5   Conclusions

The results presented above show that the segmentation quality assessment functions can be used for selecting the optimal level at which to cut watershed similarity hierarchies generated by cluster analysis. The level determines the number of classes in the final segmentation. If the range of the desired number of classes is narrow, then the assessment functions can be used to select the optimal number of classes. If the range is wider, then the local minima provide a hint as to which segmentations should be assessed by a human observer: in other words they help to select the significant levels in the similarity hierarchy. The three assessment functions shown in section 3 provide coherent results. In most cases they attain local minima for the same number of classes. Like other elements of our watershed merging method, the quality assessment functions can easily be exchanged or replaced.

References

[1] Smołka J., Hierarchical cluster analysis methods applied to image segmentation by watershed merging, Annales UMCS Informatica AI 6 (2007): 73.

[2] Smołka J., Skublewska-Paszkowska M., Watershed merging method for color images, Annales UMCS Informatica AI 8 (1) (2008): 111.

[3] Romesburg H. C., Cluster Analysis for Researchers, Lulu Press (2004).

[4] Everitt B. S., Landau S., Leese M., Cluster Analysis, Arnold (2001).

[5] Liu J., Yang Y.-H., Multiresolution Color Image Segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 16 (7) (1994): 689.

[6] Zhang H., Fritts J. E., Goldman S. A., Image segmentation evaluation: A survey of unsupervised methods, Computer Vision and Image Understanding 110 (2) (2008): 260.

[7] Borsotti M., Campadelli P., Schettini R., Quantitative evaluation of color image segmentation results, Pattern Recognition Letters 19 (1998): 741.

[8] Whitaker R. S., Xue X., Variable-conductance, level-set curvature for image denoising, Proceedings of 3'rd International Conference on Image Processing (2001): 142.

[9] Ibanez L., Schroeder W., Ng L., Cates J., et al., The ITK Software Guide, Kitware Inc. (2005).

[10] Cumani A., Edge detection in multispectral images, CVGIP: Graphical Models and Image Processing 53 (1) (1991): 40.

[11] Zenzo S. D., A note on the gradient of a multi-image, Computer Vision, Graphics, and Image Processing 33 (1) (1986): 116.

[12] Scheunders P., Sijbers J., Multiscale watershed segmentation of multivalued images, 16th International Conference on Pattern Recognition 3 (855) (2002).