

Algorithm for selecting optimal clustering parameters used for over-segmentation reduction

Abstract. The authors proposed a solution to the over-segmentation of color images processed by watershed segmentation algorithm. The solution utilizes hierarchical cluster analysis and treats watersheds as objects characterized by a number of attributes. This paper briefly discusses the solution (clustering methods, their parameters, selected watershed attributes) and presents an algorithm used for selecting optimal parameters for cluster analysis. Detailed results obtained for a set of test images are presented and discussed.

Streszczenie. Autorzy zaproponowali rozwiązanie problem nadsegmentacji obrazów barwnych. Zjawisko to występuje w wyniku zastosowania transformacji wododziałowej. Rozwiązanie wykorzystuje hierarchiczną analizę skupień i traktuje zlewiska jako obiekty opisane zestawami atrybutów. Artykuł, krótko omawia zaproponowaną metodę (metody analizy skupień, jej parametry, wybrane atrybuty zlewisk) a także przedstawia algorytm zastosowany do doboru optymalnych parametrów analizy skupień. Artykuł przedstawia i omawia również wyniki otrzymane dla obrazów testowych. (Algorytm doboru optymalnych parametrów analizy skupień zastosowanej do redukcji nadsegmentacji).

Keywords: over-segmentation reduction, watershed transformation, cluster analysis.

Słowa kluczowe: redukcja nadsegmentacji, transformacja wododziałowa, analiza skupień.

Introduction

Watershed transformation is an interesting segmentation method introduced by Beucher and Lantuejoul [1]. The algorithm simulates pouring water onto a landscape created on the basis of a digital image. Water flows with gravity and collects in catchment basins (watersheds) [2]. As input, the algorithm requires an image where higher pixel values indicate the presence of boundaries. Most images do not satisfy this requirement. A gradient filter is frequently used for converting the original image to an appropriate height function. The watershed transformation produces a region for each of the image's local minima and, since the gradient operator is very sensitive to noise [3], the resulting number of regions (watersheds) is significantly larger than the number of objects in the image. Consequently the image is usually strongly over-segmented. In spite of this problem, results obtained using watershed transformation are useful because all region boundaries are continuous (which is not the case for many segmentation algorithms) and over-segmentation can be removed by appropriate watershed merging.

Review of over-segmentation methods

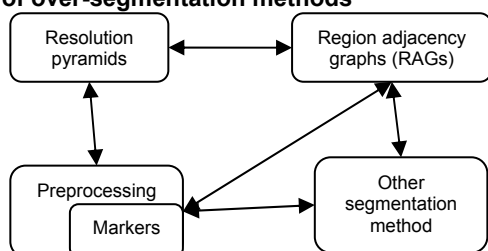


Fig. 1. Categories of over-segmentation reduction methods. Arrows connect methods that are frequently combined.

There are many over-segmentation reduction methods. The main categories are shown in figure 1. Preprocessing is the method most widely used. It aims at reducing the number of local minima in the gradient image. Preprocessing can be performed on the original image, on the gradient image or both. Solutions proposed in the literature for use with an original image consist of using morphological filters [4] or selective smoothing filters [5, 6]. These filters reduce noise and remove spurious details. Preprocessing the gradient image consists of thresholding it [5, 7] or of imposing markers on it [8]. The goal of

thresholding is to remove small gradient values whereas the goal of imposing markers is to eliminate local minima that are not selected by the markers. Frequently the local minima are filled up using morphological reconstruction. The main difficulty with this approach is marker selection.

Another very popular group of methods used in computer vision are graph algorithms [9]. Solutions using region adjacency graphs were developed for over-segmentation reduction. These methods usually consists in merging adjacent regions based on: (1) an arbitrarily selected similarity measure built upon the average value of watershed pixels and the watershed size [5], (2) dynamics of watershed contours [10], (3) a similarity measure determined using neural networks [11]. Additionally, region growing algorithms can be extended for use with region adjacency graphs and hence used for over-segmentation reduction [12, 13]. Unfortunately the last two approaches require initialization: neural networks require training while region growing algorithms require seeds for the region growing process. Another proposed solution is performing watershed transformation on graphs [14]. This allows for creating a hierarchy of segmentations with different degrees of detail: watershed transformation executed on a graph merges watersheds into larger regions, which in turn results in a new graph on which the watershed transformation can again be executed. It is an interesting approach. However, it does not allow one to select the number of classes in the final segmentation.

A more recent advance in over-segmentation reduction is resolution pyramids (multiscale segmentation). Solutions proposed in the literature take advantage of the fact that the gradient of a scaled down image contains a lower number of local minima. They use one of the following approaches: (1) project those minima onto the full resolution image [8, 15], (2) perform the watershed transformation on the smallest image and project the results onto the full resolution image [16, 17], (3) apply the watershed transformation to all levels of the image pyramid and then determine relations between watersheds on this and consecutive levels [18].

Finally other segmentation methods (apart from region growing) may be adapted for watershed merging. An example of such a method is optimal thresholding [19] and active contours (snakes) [20, 21]. In the latter approach selected watershed lines merely serve as contours that initialize a snake's evolution.

The methods briefly discussed above are often designed with a particular segmentation task in mind (e. g. road lane segmentation in [4]) or are designed to be universal [5, 8]. Specific methods cannot be easily adapted for other uses while universal methods may not deliver the best possible segmentation results.

Over-segmentation reduction using hierarchical cluster analysis

Because of the reasoning outlined in the previous section, the authors proposed a method for merging watersheds in over-segmented images [22]. It was tested on gray-scale CT and MRI scans. In [23] its use was extended to color images. Clustering methods are popular in image processing [24,25] however they had rarely been used for over-segmentation reduction. Our method uses hierarchical cluster analysis, which was chosen for its versatility. The hierarchical clustering methods can be considered modular. It is possible to change clustering strategy, similarity measure or watershed properties used for merging without modifying any other component. If the method is properly implemented, such changes can be made quickly and easily. If needed, new clustering methods, similarity/dissimilarity measures and watershed attributes can be introduced. Any quantitative measure of watershed properties can be used as the watershed's attribute. This means that the proposed approach can be easily extended or adapted to a particular task. For example, attributes from [26] may be used. New similarity measures and attributes, similar to those in [27], would be needed for clustering watersheds in HSV color space because of the hue component which is expressed as an angular value. However, as long as angular values are not used (RGB, Lab or S and V components from the HSV space), the presented approach can be used without change.

Our approach consists in treating watersheds as objects which are characterized by a set of attributes. Attribute values may be standardized in order to equalize their contribution to similarity values. Watersheds' resemblances are measured with a similarity or dissimilarity coefficient. There are a number of coefficients that can be used. Once all possible similarities are obtained, the watersheds are grouped using one of the clustering methods. The result is a similarity hierarchy which can be represented as a tree. Each tree node represents the merging of two clusters and has a similarity measure associated with it. With the similarity tree available, obtaining the required 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: top and bottom. The top part, which contains the final clustering steps, is left out while the bottom part determines how the watersheds are merged.

The remainder of this paper focuses on an algorithm that was used for finding clustering parameters that produce good segmentations. By good segmentation we mean one that receives high grades from an assessment function. It is important to note that while assessment functions are frequently designed to give values that correspond to an assessment of a human observer [28, 29], the segmentation goal may not always be to produce a visually pleasing segmentation. Using the results shown below, a simplified version of the presented algorithm may be used for automatic, semi-optimal, over-segmentation reduction.

Hierarchical clustering parameters

This section contains only basic information about cluster analysis methods used in the presented algorithm's

test. More detailed information may be found, for example, in [30,31]

Four clustering methods are used in the test (because of modular implementation the list can easily be extended). The methods are: single linkage (SLINK), complete linkage (CLINK), unweighted pair-group method using arithmetic averages (UPGMA) and Ward's minimum variance method. The SLINK, CLINK and UPGMA methods differ in the way they update the similarity (or dissimilarity) matrix after the clusters are merged [30]. With the SLINK method, two clusters are considered as similar as their two most similar components. The CLINK method considers two clusters as similar as their two most dissimilar components. UPGMA averages the similarity/dissimilarity measures of all possible pairs of components in the two clusters. Ward's minimum variance method differs significantly from the previous three methods. It does not use similarity/dissimilarity coefficients. Ward's method looks for a merger that will cause a minimal increase in the total within-cluster error sum of squares [31].

As stated before our approach consists in treating watersheds as objects described by attributes. The following attributes are used in this paper: (1) watershed size (number of pixels), (2) watershed mean value (listed as avg. in table 1), (3) variance (listed as var.), (4) standard deviation (denoted s.d.), (5) value spread (denoted spr.), (6) minimal value (denoted min.), (7) maximal value (denoted max.). These attributes reflect (to some extent) values, texture and degree of over-segmentation.

In the case of color images each attribute produces as many values as there are color components. For example, three averages are computed for each watershed in a RGB image.

Different attributes have values from different ranges. Standardization may be used to prevent one attribute from having too much weight in watershed similarity. It is an optional step. The following methods are used in this paper [30]: (1) a method that causes the standardized attribute to have a mean value of 0 and a standard deviation of 1 (denoted sigma in table 1), (2) a method that linearly scales the data and guarantees the highest value to be 1 (listed as L1), (3) a method that linearly scales the data, and guarantees the minimum to be 0 and the maximum to be 1 (listed as L01), (4) a method that causes the normalized data to sum up to 1 (denoted LS1). Additionally, attributes of a particular type (such as averages in an RGB image) are standardized together in order to prevent color information from being distorted.

CLINK, SLINK and UPGMA clustering methods require a similarity/dissimilarity coefficient for generating a similarity/dissimilarity matrix. Coefficients may be easily exchanged in order to take advantage of their properties. Seven coefficients are used in this paper [30]: (1) Euclidean distance (in n-dimensional space), (2) average Euclidean distance coefficient (it is able to compensate for missing values), (3) Canberra metric coefficient (equalizes the contribution of each attribute), (4) Bray-Curtis coefficient (similar to the Canberra metric, however, it allows one attribute to be dominant), (5) coefficient of shape difference (ignores additive translation), (6) cosine coefficient (ignores scaling), (7) correlation coefficient (insensitive to attribute additive translation)

Segmentation quality assessment

The algorithm presented in this paper relies on segmentation assessment functions. The following functions are used: (1) mean value of the difference image that is the difference of the original image and its segmentation (segmentation in which each class is represented by its average), it is similar to average absolute

deviation, (2) mean square error (MSE) [32,33] (segmentations in which classes contain many outliers will have larger MSE), (3) standard deviation of the difference

image (measures segmentation uniformity), (4) mutual information (measures how much information about

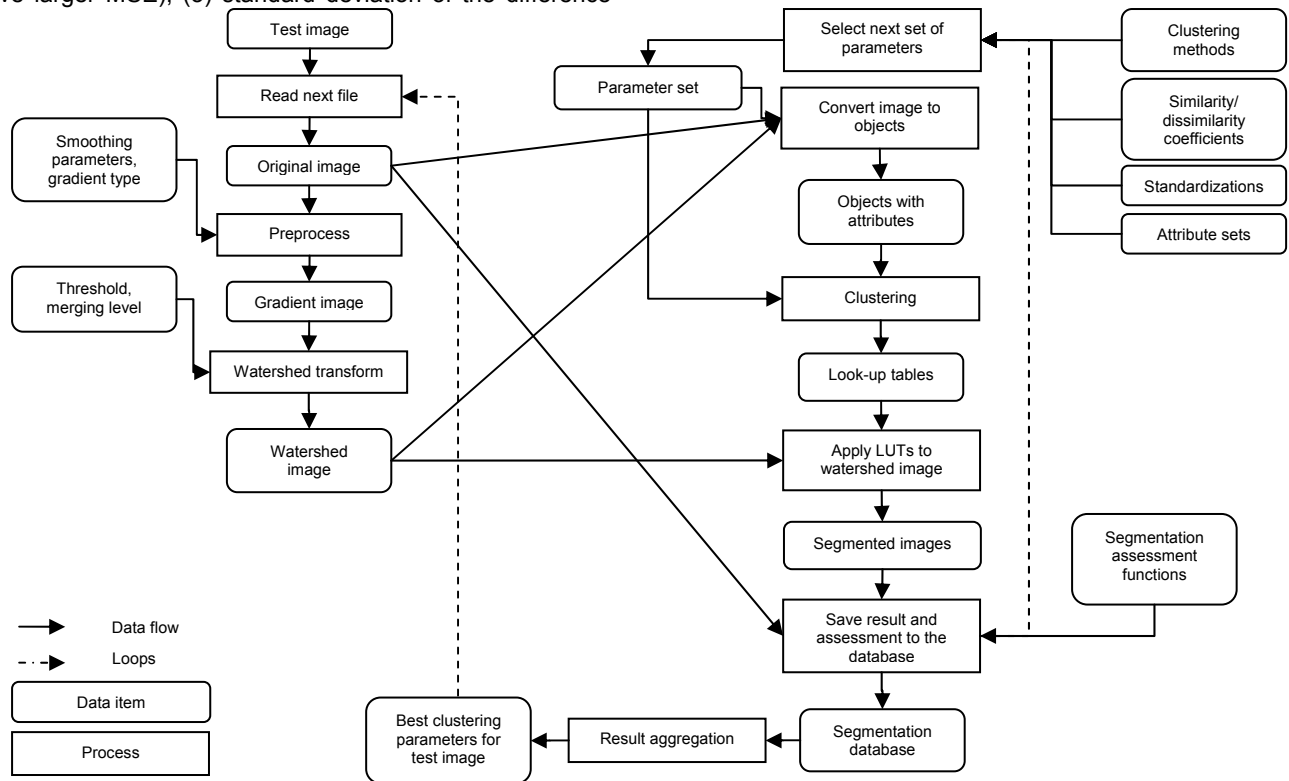


Fig. 2. Algorithm for selecting optimal clustering parameters.

the original image is contained in the segmentation) [34], (5) redundancy (a normalized variant of mutual information), (6) function F introduced by Liu and Yang [28], (7) function $F'(I)$ alternative of F proposed by M. Borsotti, P. Campadelli and R. Schettini [29], (8) function Q also proposed by Borsotti et al. [29].

Functions 6-8 are designed specifically with color image segmentation in mind. They consist of factors that penalize over-segmented images, nonhomogeneous regions and normalize the function's value. Functions 4 and 5 require histograms of the original and the segmented images. In our implementation each combination of red, green and blue color components were treated as one symbol. The histograms had separate bins for the following RGB colors: $\{0;0;1\}$, $\{0;0;2\}$, . . . , $\{255;255;255\}$. Joint histograms (required for mutual information) were optimized to reduce memory requirements (memory was reserved only for pairs of colors that exist in the images).

Gradient magnitude in color images

Determining the equivalent of gradient magnitude for a color image is problematic. Two different approaches are used in this paper. The first, a heuristic approach, finds gradient magnitude as the square-root of the sum of the squared partial derivatives of the individual image components [2]. This type of gradient will be called sum-of-squares-gradient in the remainder of this paper.

The second approach is based on principal component analysis. It was introduced by S. Di Zenzo and later extended by A. Cummani in [35]. This type of gradient will be called pca-gradient in the remainder of this paper.

Algorithm selecting optimal clustering parameters

The implementation of the algorithm is built around The Insight Toolkit Library and the filters available within this

library. The algorithm flow is illustrated in figure 2. It proceeds as follows. (1) One of the test files is read. (2) The file is preprocessed using ITK curvature anisotropic diffusion filter [35, 2] (selective blurring used for noise reduction) and one of the mentioned gradient filters is applied. (3) The watershed transform is applied to the gradient image. The ITK implementation performs basic over-segmentation reduction. Before transformation it thresholds the gradient image and removes small values. After transformation it merges shallow catchment basins with their neighbors if their depth does not exceed a value called the merging level [2]. Both threshold level and merging level are expressed as a percentage of the maximum value found in the gradient image. (4) The algorithm selects the next set of cluster analysis parameters (clustering method, similarity/dissimilarity coefficient, standardization method) and an attribute set that describes watersheds. (5) The watersheds are converted to an object described by attribute vectors (6) and the clustering is performed. (7) Using similarity hierarchy look-up tables are created. Each of the look-up tables allows one to generate a segmentation with a given number of classes (1, 2, 3...). (8) The LUTs are used to relabel watersheds. This results in a set of segmented images. (9) Each segmentation is assessed using all of the quality assessment functions, and the results are saved to the database. (10) the algorithm goes back to step 4. (11) Once all the possible segmentations for a given test image are completed, the algorithm aggregates the data from the segmentation. It finds the best segmentations according to each of the assessment functions. Then, for a particular number of classes, the algorithm finds the best (maximal or minimal - depending on the function) values of all the assessment functions that were calculated. It selects good segmentations. A segmentation is considered good when at

least one of the quality measures has a value differing by no more than 10% from the best segmentation (for this measure). (12) aggregated results are saved, and the algorithm proceeds to the next test file and goes back to step (1).

Algorithm test

The algorithm was tested on 9 images. The *building_1*, *flower_3*, *seagull_3*, *wall_3*, *painting_2* images are freely available at <http://cs.pollub.pl/english-segmentation-test-images/>, The Lena image is available at <http://www.cosy.sbg.ac.at/~pmeerw/Watermarking/lena.html> and the horses (197017), airplane (37073), and elephants (296059) images are a part of The Berkeley Segmentation Dataset [37] which is available at <http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/>). The test is an extended version of results presented in [38]. The results include new test images and lead to more comprehensive conclusions. The parameters used for selective blurring were conductance = 0.3, $\Delta T = 0.12$ and 5 iterations. The gradient filter used for a particular image is given in the figure description. For some images the resulting gradient image was thresholded to reduce oversegmentation. After the watershed transformation, shallow catchment basins were merged with neighbors if their depth did not exceed a value called merging level [2] (also given in the figure description). Both threshold and merging levels were selected manually. Special care was taken to prevent these preprocessing steps from removing significant features. Figures: 3a - 11a depict the over-segmented test images that were input to the clustering algorithm.

Four clustering algorithms, all standardization equations and all similarity/dissimilarity coefficients enumerated in previous sections were included in the test. No coefficients were used with Ward's method as it has a built-in measure. Watershed attributes were grouped in the following sets: (1) average, (2) average and standard deviation, (3) average, standard deviation and size, (4) average and variance, (5) average, variance and size, (6) minimal value, (7) maximal value, (8) average and value spread, (9) average and minimal value, (10) average and maximal value, (11) average and both minimal and maximal value, (12) average, minimal value, maximal value and the value spread. All valid combinations of the parameters were used in the test. For each parameter set segmentations containing 2 to 20 classes were created. This resulted in a total of 225720 segmentations that were generated and assessed.

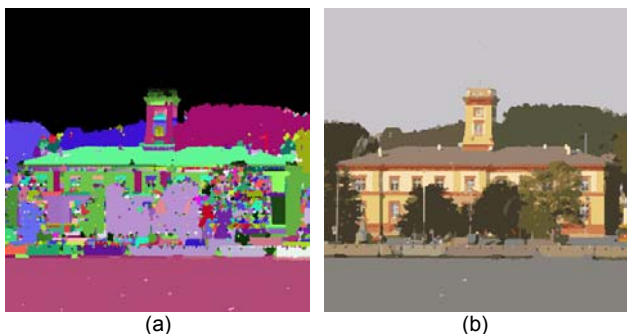


Fig. 3. Oversegmentation and oversegmentation reduction results for the *building_1* test image (384x384 pixels), (a) gradient: *pca*, threshold 0%, level 9%, catchment basins: 1873, (b) CLINK, Euclidean distance, no standardization, attributes: average, variance, 19 classes

Selected over-segmentation results are shown in figures 3b-11b. The best performing parameters are listed in table 1. All presented parameter sets produced 19 good segmentations according to the definition given in the

previous section. In order to differentiate between them the parameters were ordered with respect to the Q function value. If, for a given test image, the list is longer than 10 items, it is truncated.

The number of classes in segmentation results depicted in figures 3b-11b is also selected on the basis of the Q function (segmentations containing 2 to 20 classes were generated for each parameter set).

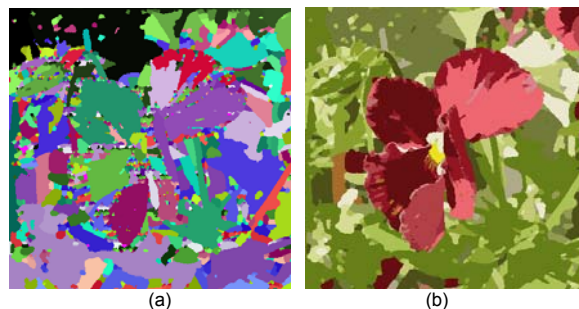


Fig. 4. Oversegmentation and oversegmentation reduction results for the *flower_3* test image (384x384 pixels), (a) gradient: sum-of-squares, threshold 5%, level 10%, catchment basins: 1965, (b) Ward's minimum variance, no standardization, attributes: average, standard deviation, 19 classes

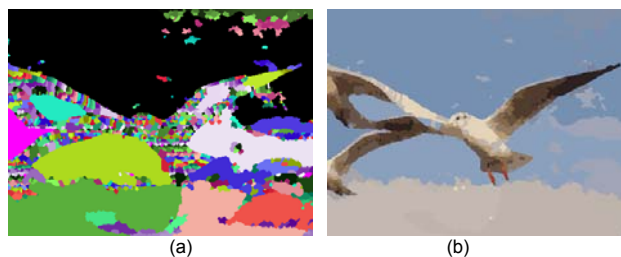


Fig. 5. Oversegmentation and oversegmentation reduction results for the *seagull_3* test image (384x288 pixels), (a) gradient: *pca*, threshold 0%, level 1%, catchment basins: 1512, (b) CLINK, Euclidean distance, L1 standardization, attributes: average, minimum, maximum, 18 classes

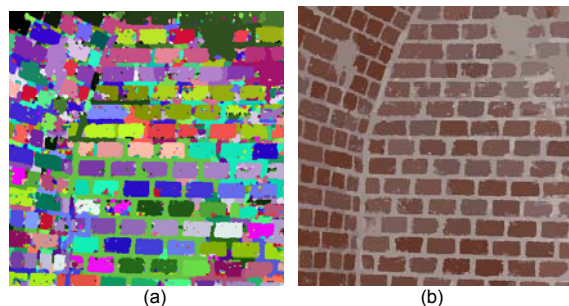


Fig. 6. Oversegmentation and oversegmentation reduction results for the *wall_3* test image (384x384 pixels), (a) gradient: *pca*, threshold 0%, level 8%, catchment basins: 2555, (b) Ward's minimum variance, no standardization, attributes: average, variance, 18 classes

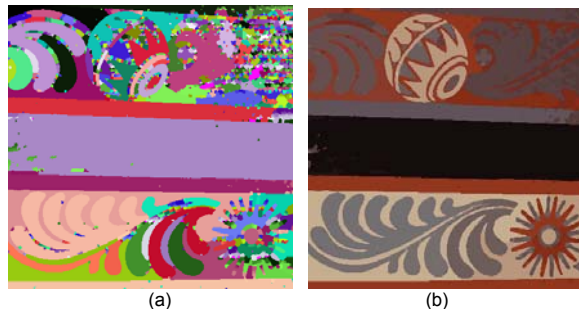


Fig. 7. Oversegmentation and oversegmentation reduction results for the *painting_2* test image (384x288 pixels), (a) gradient: sum-of-squares, threshold 5%, level 10%, catchment basins: 2144, (b) CLINK, Euclidean distance, no standardization, attributes: average, minimum, maximum, spread, 20 classes

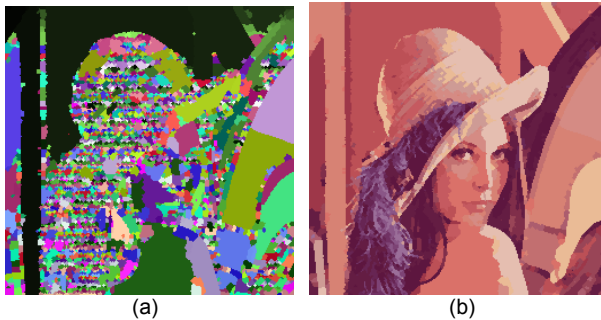


Fig. 8. Oversegmentation and oversegmentation reduction results for the *lena* test image (256x256 pixels), (a) gradient: sum-of-squares, threshold 1%, level 10%, catchment basins: 3164, (b) Ward's minimum variance, no standardization, attributes: average, variance, 20 classes

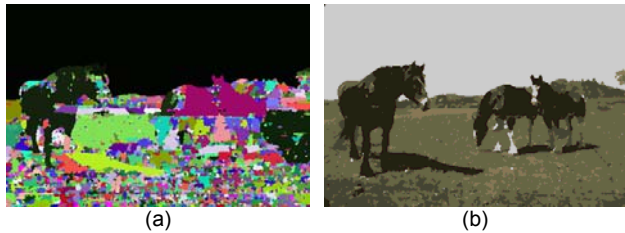


Fig. 9. Oversegmentation and oversegmentation reduction results for the *horses* (197017 from the *The Berkeley Segmentation Dataset*) test image (384x256 pixels), (a) gradient: pca, threshold 0%, level 6%, catchment basins: 3303, (b) Ward's minimum variance, sigma standardization, attributes: average, minimum, 5 classes

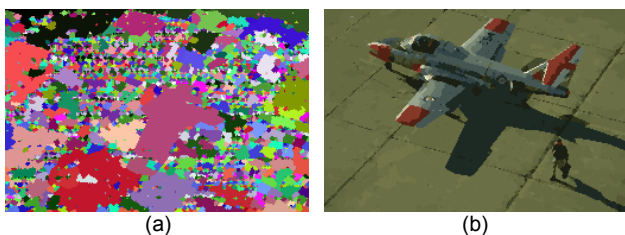


Fig. 10. Oversegmentation and oversegmentation reduction results for the *airplane* (37073 from the *The Berkeley Segmentation Dataset*) test image (256x171 pixels), (a) gradient: pca, threshold 1%, level 1%, catchment basins: 2985, (b) Ward's minimum variance, no standardization, attributes: average, variance, 20 classes



Fig. 11. Oversegmentation and oversegmentation reduction results for the *elephants* (296059 from the *The Berkeley Segmentation Dataset*) test image (384x256 pixels), (a) gradient: pca, threshold 0%, level 6%, catchment basins: 3107, (b) Ward's minimum variance, L1 standardization, attributes: average, 11 classes

Table 1. Clustering parameters that gave 19 good segmentations. Parameters are ordered according to ascending Q function values.

Q value	Method	Coeff.	Std.	Attr. set.
Building_1				
427	CLINK	Euclidean	none	avg. var.
427	CLINK	avg. Eucl.	none	avg. var.
442	CLINK	Euclidean	sigma	avg.
442	CLINK	Euclidean	L1	avg.
442	CLINK	Euclidean	L01	avg.
442	CLINK	Euclidean	LS1	avg.
442	CLINK	Euclidean	none	avg.

442	CLINK	avg. Eucl.	sigma	avg.
442	CLINK	avg. Eucl.	L1	avg.
442	CLINK	avg. Eucl.	L01	avg.
Flower_3				
2279	WARD	none	none	avg. s.d.
2441	WARD	none	none	avg. var.
3518	CLINK	cos.	sigma	avg.
3980	CLINK	Euclidean	none	avg. s.d.
3980	CLINK	avg. Eucl.	none	avg. s.d.
6436	CLINK	Canberra	sigma	min.
6540	CLINK	Bray-Curtis	sigma	min.
7266	CLINK	Bray-Curtis	sigma	avg. min. max. spr.
Seagull_3				
329	CLINK	Euclidean	L1	avg. min. max.
329	CLINK	avg. Eucl.	L1	avg. min. max.
339	CLINK	Euclidean	sigma	avg.
339	CLINK	Euclidean	L1	avg.
339	CLINK	Euclidean	L01	avg.
339	CLINK	Euclidean	LS1	avg.
339	CLINK	Euclidean	none	avg.
339	CLINK	avg. Eucl.	sigma	avg.
339	CLINK	avg. Eucl.	L1	avg.
339	CLINK	avg. Eucl.	L01	avg.
Wall_1				
309	WARD	none	none	avg. var.
317	WARD	none	L1	avg.
357	CLINK	Bray-Curtis	none	avg. var.
384	CLINK	Bray-Curtis	L1	avg.
384	CLINK	Bray-Curtis	LS1	avg.
384	CLINK	Bray-Curtis	none	avg.
414	CLINK	Euclidean	LS1	avg.
414	CLINK	avg. Eucl.	LS1	avg.
427	CLINK	Bray-Curtis	none	avg. s.d.
Painting_2				
241	CLINK	Euclidean	none	avg. min. max. spr.
241	CLINK	avg. Eucl.	none	avg. min. max. spr.
244	CLINK	Euclidean	L01	avg. min. max.
244	CLINK	avg. Eucl.	L01	avg. min. max.
250	CLINK	s.d.	none	avg. var.
261	CLINK	Euclidean	none	avg. min. max.
261	CLINK	avg. Eucl.	none	avg. min. max.
262	CLINK	s.d.	none	avg. s.d.
277	CLINK	Euclidean	LS1	avg. max
lena				
204	WARD	none	none	avg. var.
243	WARD	none	L1	avg.
252	CLINK	Euclidean	L1	avg. min. max.
252	CLINK	avg. Eucl.	L1	avg. min. max.
262	CLINK	Euclidean	L01	avg. min. max.
262	CLINK	avg. Eucl.	L01	avg. min. max.
266	CLINK	Euclidean	sigma	avg. min. max.
266	CLINK	avg. Eucl.	sigma	avg. min. max.
275	CLINK	Euclidean	none	avg. min. max.

275	CLINK	avg. Eucl.	none	avg. min. max.
Horses-197017				
732	WARD	none	sigma	avg. min.
816	CLINK	Euclidean	none	avg. var.
816	CLINK	avg. Eucl.	none	avg. var.
1174	UPGMA	Canberra	sigma	max.
Airplane-37073				
56	WARD	none	none	avg. var.
62	WARD	none	none	avg.
64	WARD	none	sigma	avg.
68	WARD	none	L1	avg.
70	WARD	none	L01	avg.
86	WARD	none	sigma	avg. max
90	WARD	none	none	avg. s.d.
Elephants-296059				
482	WARD	none	L1	avg.
517	WARD	none	sigma	avg.
522	WARD	none	LS1	avg.
580	WARD	none	L01	avg. min.
586	WARD	none	none	avg. var.
601	WARD	none	L01	avg. var.
602	WARD	none	L01	avg.
644	CLINK	Bray-Curtis	LS1	avg. min.

Conclusions

As expected, the same parameters don't give the best results in the case of all test images. However, conclusions on parameters that produce good segmentations may be drawn based on table 1.

As shown in table 1 in case of the Wall_1, Lena and Airplane-37073 images, the set of parameters that allowed for obtaining the best segmentation are: (1) Ward's minimum variance method, (2) no standardization and (3) an attribute set consisting of watershed's average and its variance. It is also worth noting that, in the case of Flower_3 and Elephants-296059 images, it is listed among the best sets, and the Q function values are not far from the minimal (the best) value for those images (2441 vs 2279 and 586 vs 482 respectively). These are small differences considering that the worst Q function values are close to 100000 and 25000 respectively. This means that, for both images, several parameter sets produce almost equally good segmentations.

In case of the Building_1 and Horses-197017 images the following set produced good results: (1) CLINK clustering method, (2) Euclidean distance, (3) no standardization, (4) an attribute set consisting of watershed's average and variance.

It is also worth noting that the (1) CLINK clustering method, combined with (2) Euclidean distance, average Euclidean distance or Bray-Curtis coefficient and an (3) attribute set containing more than one attribute produces good results. This is the case for eight test images.

Other clustering methods (SLINK and UPGMA) do not appear in table 1. While they don't generate good segmentations as defined by the assessment functions (general use), they are useful for other purposes. As shown in the authors' previous work, the SLINK method combined with watershed size is useful in the segmentation of noisy areas [38], and the UPGMA method is the optimal choice for merging watersheds in gray-scale CT scans [39].

The presented algorithm allowed for discovering clustering parameters that produce good segmentations (from over-segmented images). A simplified version of the algorithm may be used for automatic over-segmentation reduction (the algorithm needs only to use the best parameter sets, to assess the results and to select the best segmentation).

Authors: dr inż. Jakub Smółka, Politechnika Lubelska, Wydział Elektrotechniki i Informatyki, Instytut Informatyki, ul. Nadbystrzycka 36B, 20-618 Lublin, E-mail: jakub.smolka@pollub.pl, dr inż. Maria Skublewska-Paszowska, Politechnika Lubelska, Wydział Elektrotechniki i Informatyki, Instytut Informatyki, ul. Nadbystrzycka 36B, 20-618 Lublin, E-mail: maria.paszowska@pollub.pl, dr inż. Edyta Łukasik, Politechnika Lubelska, Wydział Elektrotechniki i Informatyki, Instytut Informatyki, ul. Nadbystrzycka 36B, 20-618 Lublin, E-mail: e.lukasik@pollub.pl

REFERENCES

- [1] Beucher S., Lantuejoul C., Use of watersheds in contour detection, *In Real-time Edge and Motion Detection/Estimation, International Workshop on Image Processing*, 1979.
- [2] Johnson H., Cormick M., Ibanez L., The ITK Software Guide. Book 2: Design and Functionality, *itk.org*, fourth edition, 2015.
- [3] Russ J., The Image Processing Handbook, *CRC Press*, fourth edition, 2002.
- [4] Beucher S., Yu X., Road recognition in complex traffic situations, *In 7th IFAC/IFORS Symposium on Transportation Systems: Theory and Application of Advanced Technology*, pages 413–418, 1994.
- [5] Haris K., Estradiadis S. N., Maglaveras N., Katsaggelos A. K., Hybrid image segmentation using watersheds and fast region merging, *IEEE Transactions on Image Processing*, 7(12):1684–1699, 1998.
- [6] Ciecholewski M. Automated coronal hole segmentation from Solar EUV Images using the watershed transform, *Journal of Visual Communications and Image Representation*, 33 (2015), 203-218
- [7] Hsieh F.-Y., Han C.-C., Wu N.-S., Chuang T., Fan K.-C., A novel approach to the detection of small objects with low contrast, *Signal Processing*, 86(1):71–83, 2006.
- [8] Frucci M., Ramella G., Sanniti di Baja G., Using resolution pyramids for watershed image segmentation, *Image and Vision Computing*, 25(6):1021–1031, 2007.
- [9] Węgliński T., Fabijańska A., On Cerebrospinal Fluid Segmentation from CT Brain Scans using Interactive Graph Cuts, *Informatyka, Automatyka, Pomiar w Gospodarce i Ochronie Środowiska*, pages 7-9, 2012(4b)
- [10] Meyer F., The dynamics of minima and contours, *In Mathematical Morphology and its Applications to Image and Signal Processing (ISMM'96)*, 1996.
- [11] Kuo W.-F., Lin C.-Y., Sun Y.-N., Brain MR images segmentation using statistical ratio: Mapping between watershed and competitive hopfield clustering network algorithms, *Computer Methods and Programs in Biomedicine*, 91(3):191–198, 2008.
- [12] Smółka J., Watershed based region growing algorithm, *Annales UMCS Informatica*, AI 3:169–178, 2005.
- [13] Tremeau A., Colantoni P., Regions adjacency graph applied to color image segmentation, *IEEE Trans. Image Processing*, 9(4):735–744, 2000.
- [14] Frucci M., Nappi M., Ricco D., Sanniti di Baja G., WIRE: Watershed based iris recognition, *Pattern Recognition*, 52 (2016), 148-159
- [15] Scheunders P., Sijbers J., Multiscale watershed segmentation of multivalued images, *In 16th International Conference on Pattern Recognition*, volume 3, 2002.
- [16] Jung C., Unsupervised multiscale segmentation of color images, *Pattern Recognition Letters*, 28(4):523–533, 2007.
- [17] Smółka J., Skublewska-Paszowska M., Wojdyga A., Improving performance of watershed clustering algorithm by using wavelet transform, *Polish Journal of Environmental Studies*, 18(3B):341–346, 2009.
- [18] Letteboer M., Olsen O., Dam E., Willems P., Vieregger M., Niessen W., Segmentation of tumors in magnetic resonance brain images using an interactive multiscale watershed algorithm. *Academic Radiology*, 11(10), 2004.
- [19] Smółka J., Multilevel near optimal thresholding applied to watershed grouping, *Annales UMCS Informatica*, AI 5:191–200, 2006.
- [20] Zhao C., Zhuang T., A hybrid boundary detection algorithm based on watershed and snake, *Pattern Recognition Letters*, 26(9):1256–1265, 2005.

- [21]Dagher I., El Tom K., Waterballoons: A hybrid watershed balloon snake segmentation, *Image and Vision Computing*, 26(7):905–912, 2008.
- [22]Smołka J., Hierarchical cluster analysis methods applied to image segmentation by watershed merging, *Annales UMCS Informatica*, AI 6:73–84, 2007.
- [23]Smołka J., Skublewska-Paszkowska M., Watershed merging method for color images, *Annales UMCS Informatica*, AI 8(1):111–121, 2008.
- [24]Lazarek J., Metody analizy obrazu – analiza obrazu mammograficznego na podstawie cech wyznaczonych z tekstury, *Informatyka, Automatyka, Pomiary w Gospodarce i Ochronie Środowiska*, pages 10-13, 2013(4)
- [25]Musiał A., Szczepaniak P., Optical Character Recognition using Artificial Intelligence Technologies, *Informatyka, Automatyka, Pomiary w Gospodarce i Ochronie Środowiska*, pages 41-44, 2014; 4(2)
- [26]Fiderek P., Jaworski T., Wajman R., Kucharski J., Rozmyta klasteryzacja surowych trójwymiarowych danych tomograficznych dla potrzeb rozpoznawania przepływów dwufazowych, *Informatyka, Automatyka, Pomiary w Gospodarce i Ochronie Środowiska*, pages 12-15, 2015; 5(4)
- [27]Hanbury A. G., Serra J., Morphological operators on the unit circle, *IEEE Trans. Image Processing*, 10(12):1842–1850, 2001.
- [28]Liu J. and Yang Y.-H., Multiresolution color image segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(7):689–700, 1994.
- [29]Borsotti M., Campadelli P., Schettini R., Quantitative evaluation of color image segmentation results, *Pattern Recognition Letters*, 19:741–747, 1998.
- [30]Romesburg H., Cluster Analysis for Researchers, *Lulu Press*, 2004.
- [31]Everitt B., Landau S., Leese M., Cluster Analysis, *Arnold*, fourth edition, 2001.
- [32]Avcibas I., Image Quality Statistics and their Use in Steganalysis and Compression, *PhD thesis*, Uniwersytet Bogazici, 2001.
- [33]Węgliński T., Fabijańska A., Poprawa jakości obrazów tomograficznych o niskiej dawce promieniowania, *Informatyka, Automatyka, Pomiary w Gospodarce i Ochronie Środowiska*, pages 7-9, 2013(4)
- [34]Pluim J., Maintz J., Viergever M., Mutual information based registration of medical images: a survey, *IEEE Transactions on Medical Imaging*, 20, 2003.
- [35]Cumani A., Edge detection in multispectral images, *CVGIP: Graphical Models and Image Processing*, 53(1):40–51, 1991.
- [36]Whitaker R., Xue X., Variable-conductance, level-set curvature for image denoising, *In Proceedings. 2001 International Conference on Image Processing*, 3 (2001) 142–145.
- [37]Martin D., Fowlkes C., Tal D., Malik J., A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, *In Proc. 8th Int'l Conf. Computer Vision*, volume 2, pages 416–423, July 2001.
- [38]Smołka J., Adaptacyjny system wspomagający usuwanie nadsegmentacji w obrazach poddanych transformacji wododziałowej, *PhD thesis*, Politechnika Śląska 2010
- [39]Smołka J., Skublewska-Paszkowska M., Comparison of hierarchical cluster analysis methods applied to image segmentation by watershed merging, *In Computer Recognitions Systems 2, volume 45 of Advances in Soft Computing*, pages 84–91, 2007.