Evaluation of fuzzy clustering methods for segmentation of environmental images

Barbara Barišić
University of Split, University centre for professional studies, Livanjska 5, Croatia
Mirjana Bonković
University of Split, Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture, Ruđera Boškovića bb
Vladan Papić
Faculty of science, University of Split, Teslina 12, Croatia

Abstract: **The article compares three methods for segmentation of environmental images. Hue and saturation values of the image pixels were used as the input values for the clustering. The methods that have been examined are K-medoid, Fuzzy C-means and Gustafson-Kessel. Results of the fuzzy clustering methods were compared with the results obtained with method using the mean-shift algorithm.**

1. INTRODUCTION

Image segmentation is a low-level operation that is crucial for the success of higher-level image processing operations such as recognition, semantic representation and interpretation. It can be defined as a process of partitioning an image into disjoint and homogeneous regions.

Because of the increasing computer processing power and higher application demands, in the last decade the focus of the research has been shifted from gray image segmentation towards the color image segmentation [10]. Color uniformity is the most frequent criterion for partitioning color images [3]. For that purpose, the distribution of the pixel colors can be analyzed in the image plane [11][12] or in the color space. Zhang [2] devided methods into three categories: *pixel-based methods*, *region-based methods* and *boundary-based methods*. Pixel- based methods group the pixels with similar features, such as color or texture, without considering the spatial relationship among pixel groups. Examples of these methods include clustering using adaptive K-means, K-medoid, Gustafson-Kessel and Fuzzy C-means [14], among others.

Previous research [4] showed significant variations in sensitivity regarding detection of the small segments in an image. While some methods such as those using the Mean-shift algorithm [10] showed high sensitivity to small segments, performance of some others such as K-means [16] was quite disappointing. The sensitivity is very important for applications dealing with the detection of small objects in large scenes such as, for example, surveillance of the landscape and target detection from the distance. Segmentation method that produces only large segments makes successful object detection and recognition impossible no matter what higher level image operations are used at latter processing stages.

Choice of the color space used in a particular computer vision application is not a trivial task because different color spaces have different advantages and disadvantages [13]. Generally, it can be stated that traditional RGB color space is not convenient for this kind of applications due to the high correlation between color components. Although HSI (Hue, Saturation, Intensity) as well as HSV (Hue, Saturation, Value) color spaces has also some problems especially with the low saturation images, they are better choice for wide range of Computer Vision applications. After the clustering of the pixels in a color space, output is, generally, noisy segmentation with small regions scattered through the image. In order to obtain better segmentation results, spatial-based postprocessing should be performed.

In [2] the overview of different kinds of segmentation evaluation methods is presented. Depending on whether a human evaluator examines the segmented image visually or not, these evaluation methods can be divided into two major categories: *Subjective Evaluation* and *Objective Evaluation*. In the objective evaluation category, some methods examine the impact of a segmentation method on the larger system/application employing this method, while others study the segmentation method independently. Zhang [2] divides objective evaluation methods into *System-level Evaluation* and *Direct Evaluation*. The direct objective evaluation can be further divided into *Analytical Methods* and *Empirical Methods*, based on whether the method itself or the results that the method generated are being examined. Finally, the empirical methods are divided into *Unsupervised Methods* and *Supervised Methods*, based on whether the method requires a ground-truth reference image. On contrary to the supervised objective evaluation, which are objective methods that require access to a ground truth reference (manually-segmented reference image), unsupervised objective evaluation, does not require comparison with a manually-segmented reference image.

In this paper, we will try to investigate and evaluate three fuzzy clustering methods for the segmentation of environmental images taken from the long distance. Optimal number of clusters will be determined using the unsupervised objective evaluation, that is a proposed combined index for determining the optimal number of clusters. Region merging and other image processing in order to obtain larger regions after color clustering in HSV color space will not be presented because the focus is on the comparison of the methods and their applicability for the particular type of applications. This raw segmentation results as well as speed of each algorithm will be compared with a Mean shift algorithm [15] in order to find if these algorithms can improve performance of the applications targeting the segmentation of the enviromental images ie. natural images of the terrain taken from the airplane or helicopter. The rest of this paper is organized as follows. In section 2, algorithms used for the segmentation are presented. Also, a combined index for determining the optimal number of clusters is proposed. Section 3 gives the results of the algorithms and their comparison, which is followed by the discussion and conclusion in the section 4.

2. METHODS

2.1 K – medoid algorithm

K – medoid algorithm belongs to the hard partitioning methods, like K – means [1]. The objective function of the algorithm is to partition the data set X into c clusters. From a $N \times n$ dimensional data set algorithm allocates each data point to one of c clusters to minimize the within – cluster sum of squares:

$$\sum_{i=1}^{c} \sum_{k \in A_i} \left\| x_k - v_i \right\|_2 \quad (1)$$

where A_i is a set of data points in the i – th cluster and v_i is the mean for that points over cluster i. Equation (1) denotes a distance norm. The cluster centers are the nearest objects to the mean of data in one cluster. The function generates random cluster centers, so the number of clusters must be previously initialized. [1] The K – medoid algorithm is presented by steps in the table 1.

Table 1. The K – medoid algorithm

Given: data set X Choose: the number of clusters $1 < c < N$ Initialize: random cluster centers chosen from the data set X
Repeat for l=1,2,…
Compute the distances: $D_{ik}^2 = (x_k - v_i)^T (x_k - v_i),$ $1 \le i \le c, 1 \le k \le N$
Select: points for a cluster with the minimal distances, they belong to that cluster
Calculation of the fake cluster centers:
$v_i^{(l)*} = \dfrac{\sum_{j=1}^{N_i} x_i}{N_i}$
Choose the nearest data point to be the cluster center: $D_{ik}^{2*} = (x_k - v_i^*)^T (x_k - v_i^*),$ $x_i^* = \arg\min_i (D_{ik}^{2*}); v_i^{(l)} = x_i^*$ $until: \prod_{k=1}^{n} \max \left| v^{(l)} - v^{(l-1)} \right| \ne 0$ Ending: Calculate the partition matrix

2.2 Fuzzy C – means algorithm

Fuzzy C – means algorithm (FCM) [6] is based on minimization of an objective function called *C – means functional*, defined by Dunn [7]:

$$J(X;U,V) = \sum_{i=1}^{c} \sum_{k=1}^{N} (\mu_{ik})^m \left\| x_k - v_i \right\|_A^2 \quad (2)$$

$$\text{where } V = [v_1, v_2, \ldots, v_c], v_i \in \Re^n \quad (3)$$

is a vector of cluster centers, which have to be determined, and μ_{ik} represents fuzzy partitions which can attain real values in [0,1].

$$D_{ikA}^2 = \left\| x_k - v_i \right\|_A^2 = (x_k - v_i)^T A (x_k - v_i) \quad (4)$$

is a squared inner – product distance norm.

The minimization of the C – means functional (2) represents a non linear optimization problem. The simple iteration through the first - order conditions for stationary points of (2), is the fuzzy C –means algorithm. [1]

By adjoining the constraint to J by means of Lagrange multipliers, the stationary points of the objective function is found:

$$\bar{J}(X;U,V,\lambda) = \sum_{i=1}^{c} \sum_{k=1}^{N} (\mu_{ik})^m D_{ikA}^2 + \sum_{k=1}^{N} \lambda_k \left(\sum_{i=1}^{c} \mu_{ik} - 1 \right) \quad (5)$$

and by setting the gradients of $(\bar{J})$ with respect to U, V and λ to zero. If $D_{ikA}^2 > 0, \forall i,k$ and $m > 1$, then $(U,V) \in M_{fc} \times \Re^{n \times c}$ may minimize (2) only if

$$\mu_{ik} = \frac{1}{\sum_{j=1}^{c} (D_{ikA} / D_{jkA})^{2/(m-1)}} \quad (6)$$

$$1 \le i \le c, 1 \le k \le N$$

and

$$v_i = \frac{\sum_{k=1}^{N} \mu_{ik}^m x_k}{\sum_{k=1}^{N} \mu_{i,k}^m}, 1 \le i \le c \quad (7)$$

Equation (7) gives v_i as the weighted mean of the data items that belong to a cluster. The weights are the membership degrees. The value for the weighting parameter *m* is very important: if it approaches 1, the partition becomes hard, but if it approaches to infinity, the partition becomes maximally fuzzy.

The FCM algorithm is a simple iteration through (6) and (7). It computes with the standard Euclidean distance norm, which induces hyperspherical clusters. Hence it can only detect clusters with the same shape and orientation, because the common choice of norm inducing matrix is: **A=I** or it can be chosen as a $n \times n$ diagonal matrix that accounts for different variances in the directions of the coordinate axes of *X*:

$$A_D = \begin{bmatrix} (1/\sigma_1)^2 & 0 & \dots & 0 \\ 0 & (1/\sigma_2)^2 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & (1/\sigma_n)^2 \end{bmatrix} \quad (8)$$

or **A** can be defined as the inverse of the $n \times n$ covariance matrix: $A = F^{-1}$, with

$$F = \frac{1}{N}\sum_{k=1}^{N}(x_k - \bar{x})(x_k - \bar{x})^T \quad (9)$$

$\bar{x}$ denotes the sample mean of the data.

There are three input parameters needed to run this function: number of clusters, fuzziness weighting exponent and the maximum termination tolerance. If not given, the last two parameters have their default value. [1] The Fuzzy C - means algorithm is presented by steps in the table 2.

Table 2. The Fuzzy C – means algorithm

Given: data set X Choose: the number of clusters $1 < c < N$, the weighting exponent $m > 1$, the termination tolerance $\in > 0$, and the norm inducing matrix A. Initialize the partition matrix randomly, such that $U^{(0)} \in M_{fc}$
Repeat for l=1,2,…
Compute the cluster means: $v_i^{(l)} = \frac{\sum_{k=1}^{N}(\mu_{ik}^{(l-1)})^m x_k}{\sum_{k=1}^{N}(\mu_{i,k}^{(l-1)})^m}, 1 \le i \le c$
Compute the distances: $D_{ikA}^2 = (x_k - v_i)^T A(x_k - v_i), 1 \le i \le c, 1 \le k \le N$
Update the partition matrix: $\mu_{i,k}^{(l)} = \frac{1}{\sum_{j=1}^{c}(D_{ikA}/D_{jkA})^{2/(m-1)}}$
Until: $\left\| U^{(l)} - U^{(l-1)} \right\| < \in$

2.3 The Gustafson - Kessel algorithm

In order to detect clusters of different geometrical shapes in one data set, Gustafson and Kessel extended the standard fuzzy C – means algorithm by employing an adaptive distance norm [5]. Each cluster has its own – norm inducing matrix A_i, which yields the following inner – product norm:

$$D_{ikA}^2 = (x_k - v_i)^T A(x_k - v_i) \quad (10)$$
$$1 \le i \le c, 1 \le k \le N$$

Table 3. The Gustafson – Kessel algorithm

Given: data set X Choose: the number of clusters $1 < c < N$, the weighting exponent $m > 1$, the termination tolerance $\in > 0$, and the norm inducing matrix A. Initialize the partition matrix randomly, such that $U^{(0)} \in M_{fc}$
Repeat for l=1,2,…
Calculate the cluster centers: $v_i^{(l)} = \frac{\sum_{k=1}^{N}(\mu_{ik}^{(l-1)})^m x_k}{\sum_{k=1}^{N}(\mu_{i,k}^{(l-1)})^m}, 1 \le i \le c$
Compute the cluster covariance matrices: $F_i^{(l)} = \frac{\sum_{k=1}^{N}(\mu_{ik}^{(l-1)})^m (x_k - v_i^{(l)})(x_k - v_i^{(l)})^T}{\sum_{k=1}^{N}(\mu_{ik}^{(l-1)})^m}, 1 \le i \le c$
Add a scaled identity matrix: $F_i := (1-\gamma)F_i + \gamma(F_0)^{1/n} I$
Extract eigenvalues λ_{ij} and eigenvectors ϕ_{ij}, find $\lambda_{i,\max} = \max_j \lambda_{ij}$ and set: $\lambda_{i,\max} = \lambda_{ij} / \beta$ for which $\lambda_{i,\max} / \lambda_{ij} \ge \beta$
Reconstruct F_i by: $F_i = [\phi_{i,1} .. \phi_{i,n}] diag(\lambda_{i,1} .. \lambda_{i,n}) [\phi_{i,1} .. \phi_{i,n}]^{-1}$
Compute the distances: $D_{ikA_i}^2 = (x_k - v_i^{(l)})^T [(\rho_i \det(F_i))^{1/n} F_i^{-1}](x_k - v_i^{(l)})$
Update the partition matrix: $\mu_{ik}^{(l)} = \frac{1}{\sum_{j=1}^{c}(D_{ikA_i}(x_k, v_i)/D_{jk}(x_k, v_j))^{2/(m-1)}}$ $1 \le i \le c, 1 \le k \le N$ *Until:* $\left\| U^{(l)} - U^{(l-1)} \right\| < \in$

Matrices A_i are used as optimization variables in the c – means functional, thus allowing each cluster to adapt the distance norm to the local topological structure of the data. Let **A** denote a *c* – tuple of the norm - inducing matrices:

A= $(A_1, A_2, \ldots, A_c)$. The objective functional of the GK algorithm is defined by:

$$J(X;U,V,A) = \sum_{i=1}^{c}\sum_{k=1}^{N}(\mu_{ik})^m D_{ikA_i}^2 \qquad (11)$$

The objective function (11) cannot be directly minimized with respect to A_i, since it is linear in A_i. J can be made small by making A_i less positive definite, but A_i needs to be constrained in some way. Allowing the matrix A_i to vary with its determinant fixed corresponds to optimizing the cluster's shapes while its volume remains constant:

$$\|A_i\| = \rho_i, \rho > 0 \qquad (12)$$

where ρ_i is fixed for each cluster. Using the Lagrange multiplier method, the following expression is obtained:

$$A_i = [\rho_i \det(F_i)]^{1/n} F_i^{-1} \qquad (13)$$

where F_i is the *fuzzy covariance matrix* of the i – th cluster defined by:

$$F_i = \frac{\sum_{k=1}^{N}(\mu_{ik})^m (x_k - v_i)(x_k - v_i)^T}{\sum_{k=1}^{N}(\mu_{ik})^m} \qquad (14)$$

The algorithm detects the elongated clusters [1].
The numerically robust GK algorithm described by Babuška et al [9] is used for the clustering of test images. The Gustafson - Kessel algorithm is presented by steps in the table 3.

2.4 The optimal number of clusters

Haralick and Shapiro [8] proposed four criteria for a good segmentation:

(i) Regions should be uniform and homogeneous with respect to some characteristic
(ii) Adjacent regions should have significant differences with respect to the characteristic on which they are uniform
(iii) Region interiors should be simple and without holes
(iv) Boundaries should be simple, not ragged, and be spatially accurate

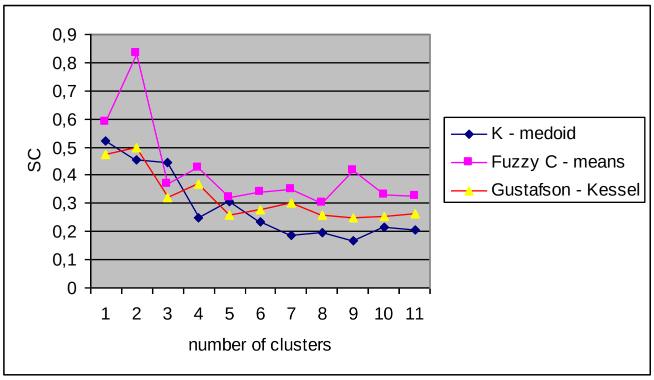
The characteristics of objects in the image are examined by the first two criteria, and the last two are based on how likely each region is regarded as a single object. According to [2], the criterion (iv) is usually not appropriate for natural images.

The approach in determining the optimal number of clusters is in defining the validity function which evaluates a complete partition. An upper bound for the number of clusters must be estimated $(c_{\max})$, and the algorithms have to be run with each $c \in \{2,3,\ldots,c_{\max}\}$ [1]. For each partition the validity function provides a value such that the result of the analysis can be compared indirectly. A number of validity functions have been proposed by [1], and those are: Partition Coefficient (PC), Classification Entropy (CE), Partition Index (SC), Separation Index (S), Xie and Beni's Index (XB), Dunn's Index (DI) and Alternative Dunn Index (ADI). None of these indexes is perfect by oneself. The approach that is presented in this paper is based only on two indexes, an SC and S functions, which joint together evaluated the optimal number of clusters.
Partition index SC is the ratio of the sum of compactness and separation of the clusters. It is a sum of individual cluster validity measures [1]:

$$SC(c) = \sum_{i=1}^{c}\frac{\sum_{j=1}^{N}(\mu_{ij})^m \|x_j - v_i\|^2}{N_i \sum_{k=1}^{c}\|v_k - v_i\|^2} \qquad (15)$$

A lower value of SC indicates a better partition.
Separation index S, on the contrary, uses a minimum - distance separation for partition validity [1]:

$$S(c) = \frac{\sum_{i=1}^{c}\sum_{j=1}^{N}(\mu_{ij})^2 \|x_j - v_i\|^2}{N \min_{i,k}\|v_k - v_i\|^2} \qquad (16)$$



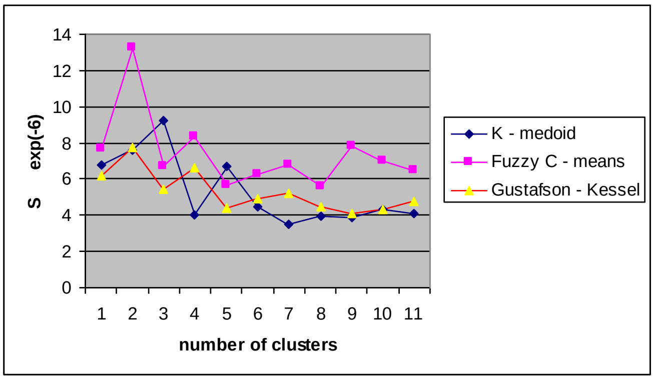


Figure 1. An example of finding optimal number of clusters for a test image: for K – medoid algorithm is c=10, for Fuzzy C – means c=8, and for Gustafson – Kessel c=10.

Because of the diversity of SC and S indexes, we propose index that is the combination of both, SC and S. For SC, the difference $|SC(c+1) - SC(c)|$ needed to be smaller then 0.05, to say that the optimal number of clusters was reached. As for S, $|S(c+1)-S(c)|$, it was sufficient that the difference was smaller than 0.1. The number of clusters of the smaller index value calculated for SC and S, was taken into account. The optimal number of clusters was then increased by one $(c+1)$.

3. RESULTS

Each algorithm, (K-medoid, Fuzzy C-means, Gustafson-Kessel), was executed on the base of 50 environmental images. The images of natural, mainly landscape images were taken from long distances. The purpose was to distinguish person from the environment, what would not be so hard if the images were taken from not such big distance. Each algorithm, K-medoid, Fuzzy C-means, Gustafson-Kessel, was performed on each of the 50 environmental images. Clustering was performed using the HSV color model: hue (H) and saturation (S) components were then used as the input data for the clustering process. The algorithms were performed for various numbers of clusters $c \in \{2,3,\ldots,12\}$ and for that each number of clusters the validity function was obtained. Using the information obtained from the SC and S matrix, the optimal number of clusters was determined. After determinating the optimal number of clusters, the algorithm was performed once again, but this time only for the estimated optimal number of clusters. On figure 1 the SC and S functions are shown, as an example of finding optimal number of clusters. Optimal number is estimated observing the function graph and detecting the point at which the function values are stabilized, (as explained in 2.4). Figure 1 shows SC and S for one environmental image: for K – medoid algorithm the optimal number of clusters is c=10, for Fuzzy C – means c=8, and for Gustafson – Kessel c=10.
Figure 2 shows the result of the segmentation for each method, for the same original image.
Table 4 shows total results of all 50 images for each algorithm: mean values (meanC) and standard deviations for optimal number of clusters, minimal (meanL) and maximal (meanH) clusters. The optimal number of clusters is almost the same for each algorithm, but from the standard deviation of minimal and maximal clusters Fuzzy C-means can be indicated as the best option for detection of small objects in an image.

The results of average processing time normalized according to the slowest method and the results of the calculated optimal number of clusters for 50 test images are presented in table 5. As it can be seen in case of one image, K-medoid is the fastest, Fuzzy C-means and Mean-Shift algorithm showed similar processing speed. Optimal number of clusters for 3 fuzzy methods was lower than number of clusters obtained with Mean Shift which must also be noticed because higher number of clusters implies higher processing time.

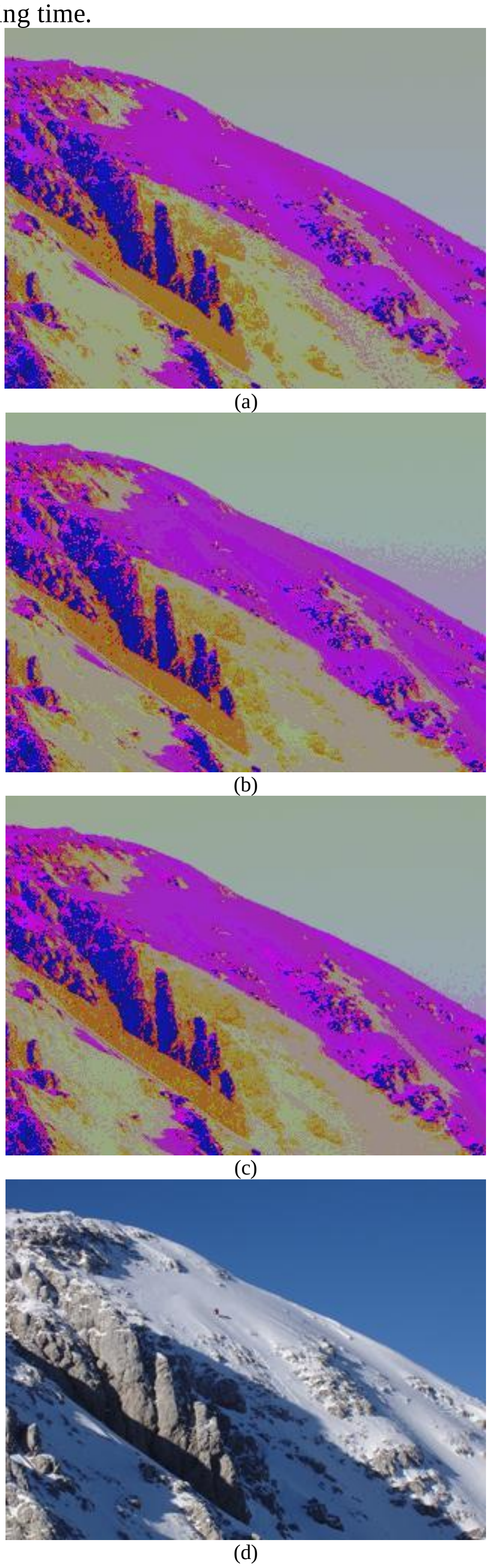


Figure 2. The segmentation results: (a) K-medoid algorithm, (b) Fuzzy C – means algorithm, (c) Gustafson – Kessel algorithm, (d) the original image

Table 4. The results for clustering and segmentation of 50 images

	K-medoid	Fuzzy C-means	Gustafson-Kessel
meanC(opt.num.)	6,84	6,66	6,74
SD(opt.num)	2,57	2,21	1,98
meanL(min.cl.)	4319,32	3881,60	4850,86
SD(min.cl.)	4310,59	2793,50	2958,50
meanH(max.cl.)	24783,30	21707,85	21089,51
SD(max.cl.)	7972,12	6742,50	7886,24

Table 5. Comparison results of average processing time and calculated optimal number of clusters for 50 test images.

50 images	Mean Shift	K-medoid	Fuzzy C-means	Gustafson-Kessel
OPT.NUMB.CL.	9,38	6,84	6,66	6,74
TIME [sec]	0,51	0,41	0,53	1

4. DISCUSSION AND CONCLUSION

In this paper we have tried to research possible use of three fuzzy clustering methods for the segmentation of certain type of natural images dealing dominantly with the long distance images of non urban terrain. There have been developed many segmentation methods, but there is still no satisfactory performance measure, which makes it hard to compare them. We have proposed a combined index for determining the optimal number of clusters. There is no question that segmentation methods can be improved with further processing: segment merging, noise removal, etc. but here we have compared only raw results of the segmented images obtained after clustering of data in the chosen color space. Results of this "raw" segmentation using presented clustering methods showed that these methods can be competitive with other currently popular methods such as Mean Shift [15]. With the known number of clusters, processing speed can even faster although less clusters implies that there is a higher possibility that small segments will be merged with larger ones.

Main problem of presented fuzzy clustering methods is previously known and that is time needed to determine optimal number of clusters. However, obtained results suggest that K-medoid and Fuzzy C-means methods have potential to produce fast segmentation results. Regarding the sensitivity to detection of small segments, results suggest that Fuzzy C-means method edges the other two methods.

For our particular area of interest i.e. environmental images segmentation, there is a possibility of preprocessing expected optimal number of clusters (or interval of numbers) for particular terrain type which could significantly improve performance of these methods.

REFERENCES

[1] Balazs Balasko, Janos Abony and Balazs Feil: "Fuzzy Clustering and Data Analysis Toolbox".

[2] Hui Zhang, Jason E. Fritts, Sally A. Goldman: "Image segmentation evaluation: A survey of unsupervised methods", Computer Vision and Image Understanding 110 (2008) p.p. 260-280.

[3] Ludovic Macaire, Nicolas Vandenbroucke, Jack-Gerard Postaire: "Color image segmentation by analysis of subset connectedness and color homogeneity properties", Computer Vision and Image Understanding 102 (2006) 105-116.

[4] Hrvoje Turic, Vladan Papic, Hrvoje Dujmic: "A procedure for detection of humans from long distance images", Elmar 2008.

[5] D.E. Gustafson and W.C. Kessel. Fuzzy clustering with fuzzy covariance matrix. In Proceedings of the IEEE CDC, San Diego, p.p. 761-766. 1979.

[6] J. C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, 1981.

[7] J.C. Bezdek and J.C. Dunn: "Optimal fuzzy partitions: A heuristic for estimating the parameters in a mixture of normal dustrubutions", IEEE Transactions on Computers, p.p. 835-838, 1975.

[8] R. Haralick, L. Shapiro: "Survey: image segmentation techniques", Computer Vision, Graphics and Image Processing 29 (1985) p.p. 100-132.

[9] R. Babuška, P. J. van der Veen, and U. Kaymak. Improved covariance estimation for Gustafson-Kessel clustering. IEEE International Conference on Fuzzy Systems, p.p. 1081-1085, 2002.

[10] L. Lucchese, S.K. Mitra: "Color Image Segmentation: A State-of-the-Art Survey", Image Processing, Vision, and Pattern Recognition, Proc. of the Indian National Science Academy (INSA-A), New Delhi, India, Vol. 67 A, No. 2, Mar. 2001, pp. 207-221.

[11] A. Tremeau, N. Borel: "A region growing and merging algorithm to color segmentation", Pattern Recognition 30 (7) (1997) p.p. 1191–1203.

[12] Xu Jie, Shi Peng-fei: "Natural Color Image Segmentation", International Conference on Image Processing, ICIP 2003, p.p. 973-976, 2003.

[13] H.D.Cheng, X.H.Jiang, Y.Sun, Jing Li Wang: "Color Image Segmentation: Advances & Prospects", Pattern Recognition, 34(12), p.p. 2259-2281, 2001.

[14] P. Lambert, H. Grecu: "A Quick and Coarse Color Image Segmentation", International Conference on Image Processing, ICIP 2003, p.p. 965-968, 2003.

[15] D. Comaniciu, P. Meer. "Mean shift: A robust approach toward feature space analysis", IEEE Trans. Pattern Anal. Machine Intell, 24, 2002, p.p.603–619.

[16] J. Peña, J. Lozano, and P. Larrsñaga, "An Empirical Comparison of Four Initialization Methods for the k-Means Algorithm," Pattern Recognition Letters, vol. 20, p.p. 1027-1040, 1999.