

Unsupervised Multi-Label Image and Texture Segmentation based on Optimal Feature Representation

Mehmet Cem CATALBAS

Abstract: In this paper, an adaptive and robust unsupervised texture segmentation algorithm is proposed. One of the novelties of this proposed algorithm is to determine optimal sub-image size by pattern analysis and another one is optimizing segmentation process by providing the most successful representation of pattern on images. The realization of the algorithm consists of various stages. Firstly, the parameters of regional minima and maxima for images are obtained to extract real pattern information about images. In the next stage, these outputs are combined to each other to calculate the centroid of distribution of patterns. The input image and several texture analysis features are analyzed in the light of this pattern information. Then, the dimension reduction process is performed by the PCA with Horn's parallel analysis method before the data clustering is applied to the dataset. This method allows optimal representation of data after dimension reduction process. The output image is improved by subjecting it to various morphological operations which are called an adaptive morphological mask and the performance of the proposed algorithm is compared with other texture segmentation algorithms.

Keywords: Feature Extraction; Fuzzy C-Means; Image Texture Segmentation; Morphological Operations; Object Analysis; Pattern Analysis; Texture Analysis

1 INTRODUCTION

Image segmentation is one of the most popular topics in the area of image processing. The general purpose of image segmentation provides much more understandable and clear outputs according to raw input images [1]. The image segmentation can be defined as two main headings such as color [2] and contour-based [3] image segmentation. There are some advantages and disadvantages of each approach according to the area of usage. The color-based image segmentation algorithms are easy to realize and they are more robust to change the parameter of the algorithm in contrast to contour based algorithms. Generally, they do not provide successful results at high rates when compared to contour-based algorithms [4]. On the other hand, contour-based segmentation algorithms are more complex for implementation, and the computation of these approaches takes time. In this paper, an unsupervised segmentation algorithm is performed and this method has been specially developed for texture segmentation. The texture information of the image is very important and it is one of the areas that are frequently worked in the image processing field [5, 6]. The definition of texture can be defined as a region that differs locally within an image and this difference should be meaningful and contain a unique pattern. The image texture segmentation methods should provide several criteria together, such as reliable, robust to noises and adaptable to different types of the pattern on images and these methods require at least preliminary information from users and therefore the unsupervised segmentation methods are preferred [7]. The proposed method was created in the light of that information. One of the sample outputs of the proposed method is shown in Fig. 1. This hyperspectral image is received from the satellite and the image contains road and field information [8]. There are two different textures on the related image and signal-to-noise ratio (SNR) is 15 dB for the hyperspectral image. Although the textures in the hyperspectral image are difficult to separate even from the eye, the proposed method successfully separates different textures. As shown in Fig. 1, the proposed method gives much more successful

and robust results and the boundaries of the roads and the fields are easily observable and the fuzzy c-means method, which is very popular among color-based segmentation methods, has failed because of high noise ratio at the image.

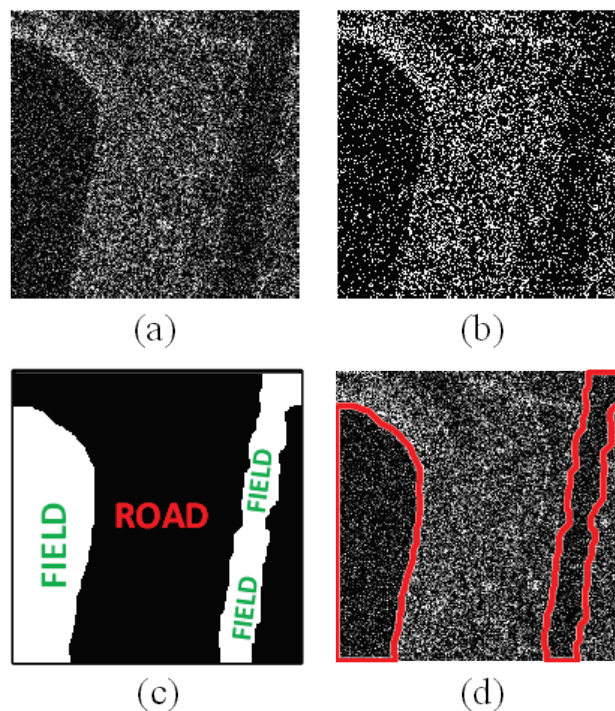


Figure 1 (a) Input image (b) Fuzzy c-mean (c) Proposed Method (d) Boundaries of the proposed method

This means that the sole usage of color-based segmentation methods for segmentation in hyperspectral images is inadequate; such is used in the first example. The application area of texture segmentation is quite varied, such as medical, hyperspectral and tissue images. In particular, the success of the texture segmentation method in the medical field is very important for accurate diagnosis or disease analysis and this is usually accomplished by determining the boundaries of different tissues. But, the success of this kind of texture segmentation algorithm is

highly related to the correct selection of initial parameters by users. Some of these important parameters are the observation screen size, which of the features to be extracted is important for segmentation, and the optimal representation of the observation space. An algorithm is proposed that automatically determines all the parameters that need to be defined by the user and does not lag behind other segmentation algorithms used in the literature. Additionally, it has been observed that the proposed algorithm gives successful results even for different types of images.

Factorization-Based algorithm is one of the important texture segmentation algorithms in the literature and comparison of Factorization-Based Texture Segmentation (FTSEG) and the proposed method is shown in Fig. 2 [9]. The estimated boundary of input image is nearly equal for each algorithm. Texture segmentation is performed fully automatically with proposed algorithm. Unlike the FTSEG algorithm, the user does not have to define an integration scale or predefined size of observation windows for this proposed algorithm. Thanks to these abilities, an adaptive and fully automatic texture segmentation algorithm is implemented according to the pattern in the input image. Proposed algorithm has the advantages of color-based image segmentation algorithms. Additionally, it has ability to solve high/mid-level segmentation problem thanks to adaptive combination of extracted features.

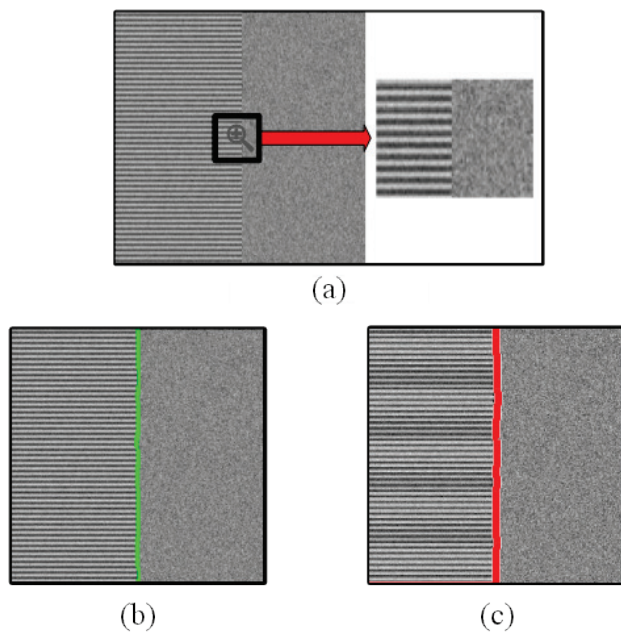


Figure 2 (a) Input image, (b) FTSEG (c) Proposed Method

As shown earlier, the direct usage of the clustering algorithm, such as k-means, k-medoids, fuzzy c-means etc. is not suitable for texture segmentation because the segmentation is performed only on the brightness values of the pixels. The users need more meaningful and divergent attributes before segmentation process. There is a need for a number of pre-operations that will allow you to analyze the patterns on the images. There are various features and transformations that provide suitable information for texture segmentation. Some of these features and transformations are the following: difference of Gaussian (*DoG*), local standard deviation of the image, the entropy

of image, range filter result of the image, Gabor transform, 2D Fourier transform, Wavelet transform, etc. The remaining sections of the paper are organised as follows. In section 2, the extracted features from images are briefly explained. In section 3, clustering algorithms which are commonly used in the area of image segmentation are given. In section 4, a brief information about proposed algorithm is given. The results about algorithm are given in section 5.

2 FEATURE EXTRACTION AND TEXTURE ANALYSIS

In this section, the methods which are used for feature extraction for texture segmentation in this study are briefly explained. Various parameters have been calculated to determine the pattern in this study, such as local standard deviation, entropy, range in images and DoG.

2.1 Local Range Filter

The parameters of a local range of the image are calculated according to Eq. (1). The S_{xy} is representing the set of coordinates in a sub-screen size n and the center coordinates are i, j . The output of local range filter [10] is defined as $R(i, j)$ and $g(s, t)$ is filtered sub-image. The sample output of local range filter is shown in Fig. 3.

$$R(i, j) = \max_{(s,t) \in S_{xy}} \{g(s, t)\} - \min_{(s,t) \in S_{xy}} \{g(s, t)\} \quad (1)$$

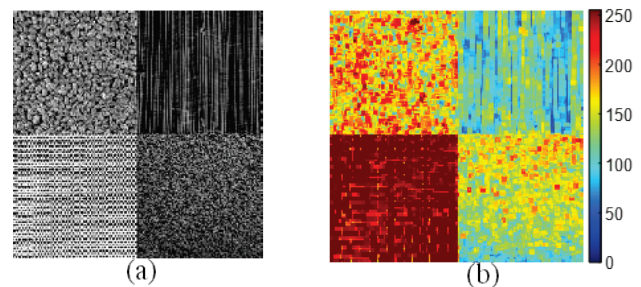


Figure 3(a) Input image (b) The output of local range filter

2.2 Local Standard Deviation Filter

The parameter of local standard deviation (LSD) which is another texture analysis parameter is calculated as follows: $I(i, j)$ is the value of the center pixel. i and j are represented by the coordinates of images. Local area or sub-images are defined by $(2n + 1)(2n + 1)$ for LSD, $(2n + 1) = s$ is an integer of selected screen size. The local mean of the images and local variances are calculated in Eq. (2) and Eq. (3) respectively [11].

$$m_x(i, j) = \frac{1}{(2n + 1)^2} \sum_{k=i-n}^{i+n} \sum_{l=j-n}^{j+n} I(k, l) \quad (2)$$

$$\sigma_x^2(i, j) = \frac{1}{(2n + 1)^2} \sum_{k=i-n}^{i+n} \sum_{l=j-n}^{j+n} [x(k, l) - m_x]^2 \quad (3)$$

The term of LSD which is $\sigma_x(i, j)$ equal to square root of local variances. This term gives information of distribution about the selected screen and is used for increasing the success of segmentation in this study. The

computation of LSD is quite easy compared to other edge based feature extraction algorithms. These parameters are used for obtaining edge information from observation screen of neighborhoods. The result of LSD matrix on the input image is illustrated in Fig. 4.

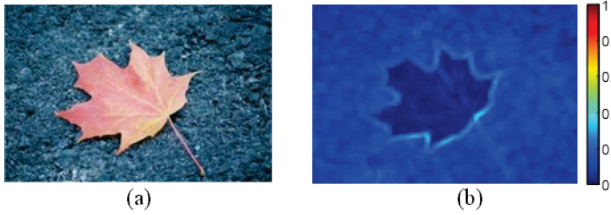


Figure 4 (a) Input image, (b) LSD result of input image

As shown in Fig. 4, edge information or fast pixel value changes at the image are representing red colour on the heat map. This feature gives information about the intensity of edge on input images.

2.3 Entropy Filter

In addition to these, another important feature for texture analysis is entropy information about the image. The calculation of image entropy is shown in Eq. (4) [12].

$$H(x) = \sum_i P(x_i) I(x_i) = -\sum_i P(x_i) \log_2 P(x_i) \quad (4)$$

2.4 Adaptive Difference of Gaussian

The Difference of Gaussians (*DoG*) is a feature extraction algorithm which is obtained via two blurred images obtained with different ratios [13]. These two images are subtracted with each other and info about intensity change is obtained as a result of this subtraction. This blurring process is performed with a Gaussian filter. The users have to define two different standard deviations (σ_1, σ_2) for filters. In literature, the ratio between these two standard deviations is selected as follows: $\sigma_1/\sigma_2 = 1.6$ for determination of optimal intensity change. However, there are some drawbacks to the constant selection of this parameter [13]. Because of this constant selection, some important features in the image may not be extracted. This parameter has to calculate adaptively according to the input image. The first, second Gaussian filters and filtered image results are shown in Eqs. (5), (6), (7) and (8) respectively.

$$G_{\sigma_1}(i, j) = \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp\left(-\frac{i^2 + j^2}{2\pi\sigma_1^2}\right) \quad (5)$$

$$G_{\sigma_2}(i, j) = \frac{1}{\sqrt{2\pi\sigma_2^2}} \exp\left(-\frac{i^2 + j^2}{2\pi\sigma_2^2}\right) \quad (6)$$

$$g_1(i, j) = G_{\sigma_1}(i, j) * I(i, j) \quad (7)$$

$$g_2(i, j) = G_{\sigma_2}(i, j) * I(i, j) \quad (8)$$

The *DoG* is obtained as shown in Eq. (9) and Eq. (10).

$$g_1(i, j) - g_2(i, j) = [G_{\sigma_1}(i, j) - G_{\sigma_2}(i, j)] * I(i, j) \quad (9)$$

$$DoG * I(i, j) = [G_{\sigma_1}(i, j) - G_{\sigma_2}(i, j)] * I(i, j) \quad (10)$$

The selection of the ratio between σ_1 and σ_2 as a constant is not suitable for all types of images. This ratio needs to be defined as an adaptive parameter and it needs to change according to input images. This kind of adaptive approach has been shown to increase success. The implementation of the adaptive difference of Gaussian (*ADoG*) is performed as shown in Eq. (9) and (10). The Gaussian low-pass filter and the input image are defined in the frequency domain as $G(u, v, \sigma_1)$ and $I(u, v)$, respectively. The parameters of u, v are defined as frequency indices and the adaptive standard deviation parameters for the Gaussian filter are calculated as shown in Eq. (11) and (12). The term of En is defined as total energy of input image.

$$\sigma_1 = \frac{\arg En\{G(u, v, \sigma_1)I(u, v)\}}{En\{I(u, v)\}} = T_1 \quad (11)$$

$$\sigma_2 = \frac{\arg En\{G(u, v, \sigma_2)I(u, v)\}}{En\{I(u, v)\}} = T_2 \quad (12)$$

The comparative results about *DoG* and *ADoG* are shown in Fig. 5. As shown in the figure, the intensity change becomes more obvious at the result of *ADoG*.

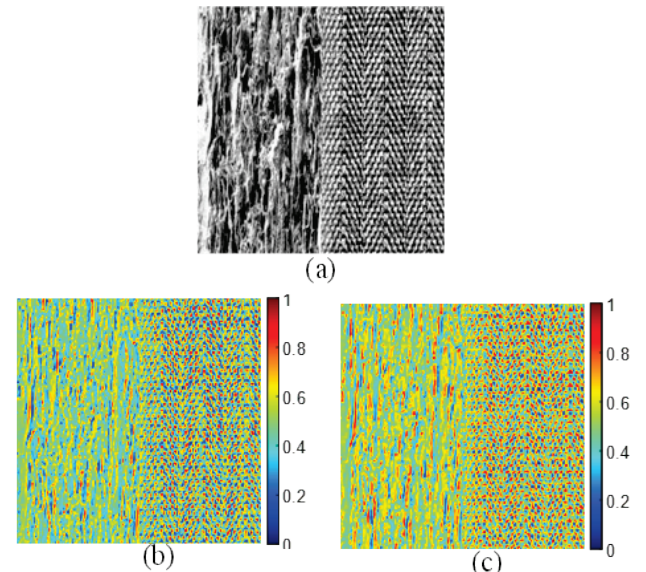


Figure 5 (a) Input Image, (b) *DoG* output (c) *ADoG* output

2.5 Regional Descriptors

The input image has to be divided into parts to increase success of texture segmentation algorithm. The users have to determine a sub-image or local window screen size for this split operation. This parameter is very important for most of the texture analyzing methods. This new sub-images should represent a pattern in the real image without any distortion and the size of screens has to be selected according to this information. The adaptive selection of sub-image screen size is performed with two parameters

and these are regional minima and maxima of the input image. The terms of regional min and max are not similar to local min and max [14]. The sub-images of the input image are obtained by sliding windows technique and the screen sizes of these sub-images are changed according to these input patterns for each input image. The term of regional is related to a connected component of pixels and they have examined the neighborhood relations. For instance, the regional minima are defined as local minimum parts on specific region at images with usage of neighborhood connectivity and the opposite of this term is defined as regional maxima and the sample outputs are shown in Fig. 6. As shown in the figure below, strong brightness transitions on the images can be modeled as local information with the pattern which is contained.

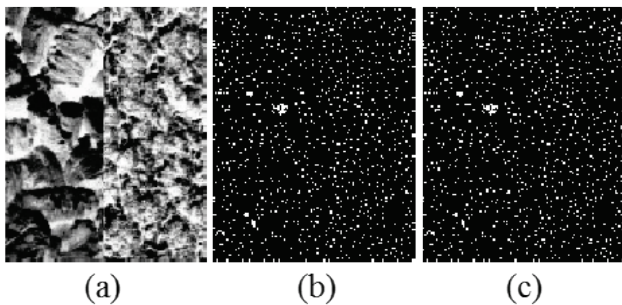


Figure 6(a) Input Image (b) Regional Minima (c) Regional Maxima

The optimal sub-image screen size is selected by these regional maxima and minima. The regional map about input image is obtained as Eq. (13).

$$R_{map}(i, j) = R_{max}(i, j) \oplus R_{min}(i, j) \tag{13}$$

All the points on the regional map can be defined as centroid and the adaptive screen size is calculated by the distance of centroid points. The centroid points on texture image are shown in Fig. 7.

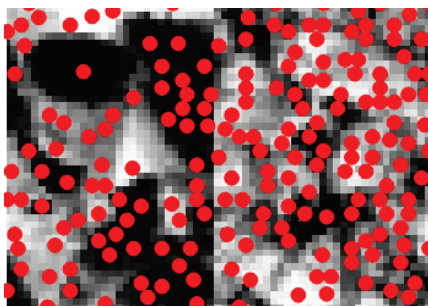


Figure 7 The map of regional pattern analysis

The Euclidean distance between all centroid points is calculated and the median value of these distances gives information about optimal screen size for sub-images. In addition to that, these distances are changing significantly for different texture patterns. Even with these distances, texture segmentation process can be achieved with acceptable success. The Euclidean distance between centroid points and calculation of optimal screen size are shown in Eq. (14) and (15), respectively. The coordinates of centroids are defined as (c_x, c_y) . This approach has been achieved in the light of information obtained by several experimental approaches on different types of images.

$$d_i = \sqrt{(c_{x_{i+1}} - c_{x_i})^2 + (c_{y_{i+1}} - c_{y_i})^2} \tag{14}$$

$$optimal\ screen\ size(s_i) = median(d_i) \tag{15}$$

After several operations, the adaptive screen size which is representing optimal pattern on images is obtained for texture segmentation. At the end of these processes, an unsupervised dimension reduction method is needed. The Principal Component Analysis (PCA) is suitable for this dimension reduction due to easy implementation and analysis of results.

2.6 PCA and Dimension Reduction

PCA was invented in 1901 by Karl Pearson. Today, it is mostly used as a tool for analyzing data and for making predictive models [15]. PCA is also generally utilized in image processing and computer vision area for face recognition, image classification, and image segmentation. The main advantage of PCA is representing very low dimensional data without loss or acceptable amount of loss.

The process of the algorithm is as follows:

The observation dataset or matrix is represented by I_{ij} .

$$I_{ij} = \begin{bmatrix} I_{11} & I_{12} & \dots & I_{1j} \\ I_{21} & I_{22} & \dots & I_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ I_{i1} & I_{i2} & \dots & I_{ij} \end{bmatrix} \tag{16}$$

$$\bar{I} = \frac{1}{N} \sum_{j=1}^N I_{1j} \quad \theta_j = I_j - \bar{I} \tag{17}$$

$$cov(\theta) = \theta^T \theta \tag{18}$$

The covariance matrix of the observation image is calculated by Eq. (18). The eigenvalues and eigenvectors are computed using this covariance matrix, and these eigenvectors and eigenvalues perform the process of dimension reduction and feature extraction. The main problem of dimension reduction algorithms is to select a new component size or better component representation. Therefore in this paper, this problem is eliminated by Horn's PA method, and the optimal number of components is obtained via PA method. The eigenvectors of the covariance matrix are represented by V and the new component set or the result of dimension reduction is shown by Ω in Eq. (19). The PCA is a powerful feature extraction algorithm.

$$w_j = V^T \theta \quad \Omega = [w_1 \ w_2 \ \dots \ w_j] \tag{19}$$

This section of the paper is about the application of Horn's PA on image segmentation. PCA gives information about better and smaller representation of images. LSD proposes sub-feature images and provides much successful segmentation on images. Despite these advantages, there is a major problem about some component selection process for optimal representation in compressed image space. There is a need to provide a component selection

that can be adaptively selected according to the distribution of the input image or dataset.

2.7 Horn's Parallel Analysis

In this paper, Horn's parallel analysis (PA) is suggested for optimal and adaptive component selection on the texture segmentation. Horn's PA[16] is one of the methods used for deciding the number of components or factors for reconstruction of the dataset. Researchers have proposed various methods for components of elements selection process. In general, the 95% of generic summation about eigenvalue is defined as a threshold. The number of eigenvalues and eigenvectors corresponding to this threshold is obtained and this number is used at the dimension reduction process on PCA.

PA provides the pre-information about optimal number for an explanation of dataset via eigenvalues. Realization of this method is similar to Monte-Carlo simulation model.

PA is realized as follows:

- The random observation matrix (Eq. (20)) which has the same size as real input covariance matrix is created.
- The eigenvalues and eigenvectors are calculated from this random matrix (Eq. (21)).

$$R_{ij} = \begin{bmatrix} R_{11} & R_{12} & \dots & R_{1j} \\ R_{21} & R_{22} & \dots & R_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ R_{i1} & R_{i2} & \dots & R_{ij} \end{bmatrix} \quad (20)$$

$$\lambda_{\text{parallel}} = [e_1 \quad e_2 \quad \dots \quad e_j] \quad (21)$$

- This procedure is repeated k (iteration) times.
- The mean eigenvalue of random matrices is obtained (22).

$$\bar{\lambda} = \frac{1}{k} \sum_{i=1}^k \lambda_i \quad (22)$$

- At the end of the process, eigenvalues of PA are compared with the real eigenvalues.
- The eigenvalues which are bigger than those of PA are selected.

The result of PA gives information about the optimal number of components. As shown in Fig. 8, only one component is enough to represent sub-images for texture segmentation.

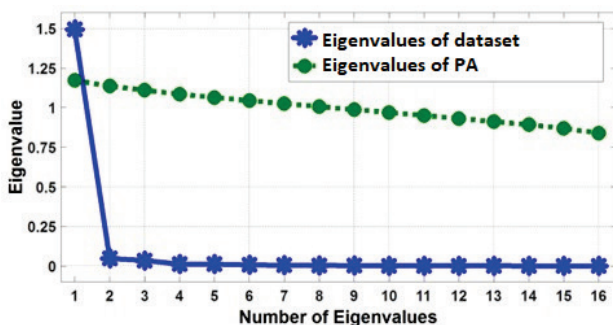


Figure 8 The comparison of PA and PCA Eigenvalues

As a result of these processes, the input image is represented in the best way. The clustering algorithms are applied to these outputs and texture segmentation is performed as a result of this process.

3 CLUSTERING ALGORITHMS AND IMAGE SEGMENTATION

There are several data clustering algorithms in the literature, such as k-means, k-medoids, hierarchical clustering, fuzzy c-means, etc. However, each method has its own advantages and disadvantages. In this work, the k-medoids algorithm has been selected for clustering process due to robust and satisfactory results for different types of images. In this section, we explain some of the data clustering methods commonly used in literature. In this study, the algorithms used frequently in the literature are explained, such as k-means, k-medoids, and fuzzy c-means. These algorithms are used in the basic applications of image segmentation problems.

3.1 k-Means Algorithm

The k-means algorithm is a well-known technique to perform data clustering [17]. Each cluster center is called the centroid. The centroid's coordinates are the mean point on the distribution of clusters.

The k-means algorithms measure the distance between all data points and centroid points and minimize error by changing centroid points. These procedures are repeated for all groups. While the selection of group number is not a significant problem for clusters which are coherent internally and are clearly different from each other, it is a major issue for high-dimensional datasets such as hyperspectral images. The computation time of segmentation will become a real problem and slows down clustering process.

3.2 k-medoids algorithm

The k-medoids algorithm which is similar to k-means is one of the data clustering techniques. It is more robust to noise compared to the k-means algorithm. In contrast to the k-means algorithm, k-medoids select one of the observation points for cluster centers called as medoids [18]. The user can increase number of clusters by using these medoids. This selection occurs randomly as k-means. The medoids are determined by finding object i within the cluster that minimizes Eq. (23). Cluster i is represented by C_i and the distance between C_i and C_j observations are represented by $d(i, j)$. There are many different types of methods for calculating the distance between these observation points such as Euclidean, Mahalanobis, and Minkowski etc.

$$Distance = \sum_{j \in C_i} d(i, j) \quad (23)$$

3.3 Fuzzy c-Means Algorithm

Another data clustering algorithm which is called c-means has been performed for texture segmentation problem. The fuzzy c-means has an additional parameter at

clustering process which is called the degree of membership and the application of fuzzy c-means is based on fuzzy logic [19]. The assignment of data to clusters is not like the other data clustering algorithms.

The ratio of fuzzy membership varies according to the distance. Hence the membership function (m) on fuzzy c-means approach has the smooth characteristic. The sample illustration of single space dataset is shown in Fig. 9. The difference between soft and hard clustering is shown clearly in the figure below.

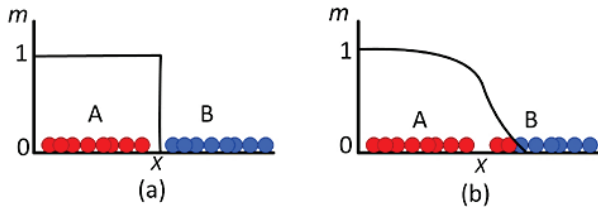


Figure 9 (a) Hard clustering, (b) Soft Clustering

Generally, the implementation of fuzzy c-means is performed via minimization of Eq. (24). In other words, this equation is defined as a cost function of this clustering algorithm.

$$J_m = \sum_{i=1}^N \sum_{j=1}^C u_{ij}^m \|x_i - c_j\|^2 \quad (24)$$

where: u_{ij} is fuzzy membership of x_i in the cluster j , x_i and c_j represent d-dimensional observation data and cluster of center in space, respectively. The fuzzy membership and cluster center are defined as Eq. (25) and (26), respectively.

$$u_{ij} = \frac{1}{\sum_{k=1}^c \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}} \quad (25)$$

$$c_j = \frac{\sum_{i=1}^N u_{ij}^m x_i}{\sum_{i=1}^N u_{ij}^m} \quad (26)$$

Besides, the users have to define an error criterion (ϵ), such as other clustering algorithm and the algorithm is repeated until error criteria are satisfied which is shown in Eq. (27).

$$\|u_{ij}^{k+1} - u_{ij}^k\| < \epsilon \quad (27)$$

All the algorithms have their own cons and pros, such as computation time, better convergence, robustness etc.

4 PROPOSED ALGORITHM

In this section, we explain the implementation of a proposed algorithm for unsupervised image texture segmentation. There are several criteria for the success of a texture segmentation method. First of all, the algorithm needs to give reliable results for different types of texture images and it must be robust against noise. Secondly,

boundaries of different textures must be determined successfully. Moreover, the proposed algorithm should require minimum or no prior information from the user and this prior information can be defined as seed points on images. The proposed algorithm consists of several steps as follows:

- Investigation of patterns on input image and selection of optimal screen size for sub-images.
- The feature extraction process on the image, such as entropy, local range, $ADoG$, etc.
- The extracted features are converted to sub-images in the light of adaptive screen size selection algorithm.
- Dimension reduction with PCA and determination of the optimal number of components via Parallel analysis at the same time.
- The application of adaptive mask for minimizing merges of patch error.
- Clustering of the optimal representation of input image and obtaining the result of the proposed algorithm.

The flowchart of proposed algorithm is shown in Fig. 10.

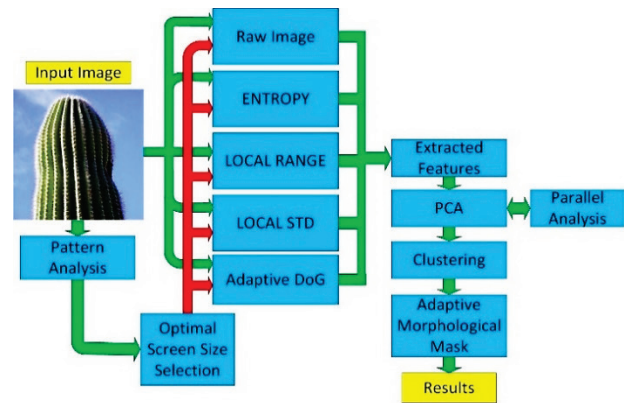


Figure 10 Flowchart of Proposed method

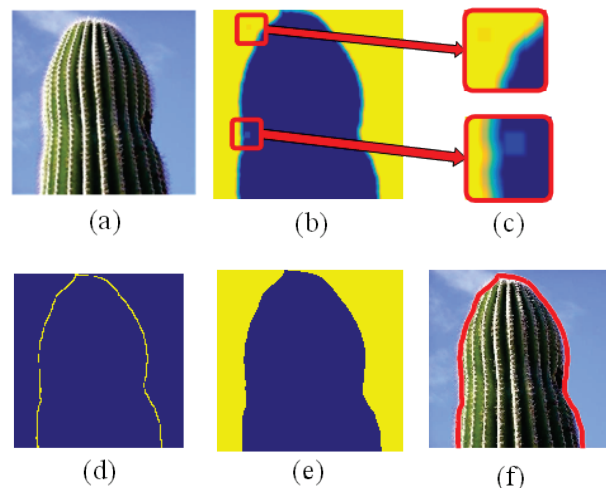


Figure 11 (a) Input Image, (b) Raw Output, (c) Local Segmentation Errors, (d) Adaptive Morphological Mask, (e) Result of Algorithm, (f) Boundaries

In the process of merging patches after clustering, some local errors occur on output, especially in edge regions. Therefore, an adaptive morphological mask is defined to minimize these errors. This mask includes several morphological operations, such as erosion, dilation, filling holes [20]. The basic procedure of adaptive morphological mask is shown in Fig. (11).

These processes are as follows: filling small holes on raw output on initial clustering result. After that, the information of edges is obtained via Sobel edge detection algorithm and this information is applied to input images, image dilation and erosion are applied respectively. As a result of this process, the local error is removed and the boundaries become more apparent on output image.

5 ALGORITHM RESULTS

In this section, the results of the proposed algorithm are compared to the other image texture segmentation algorithm and human-based segmentation. The results of comparison are shown in Fig. 12. As shown in the figure below, the proposed method has much better performance compared to other methods. In addition to that, the most successful result was found adaptively with the help of parallel analysis as shown in Tab. 1.

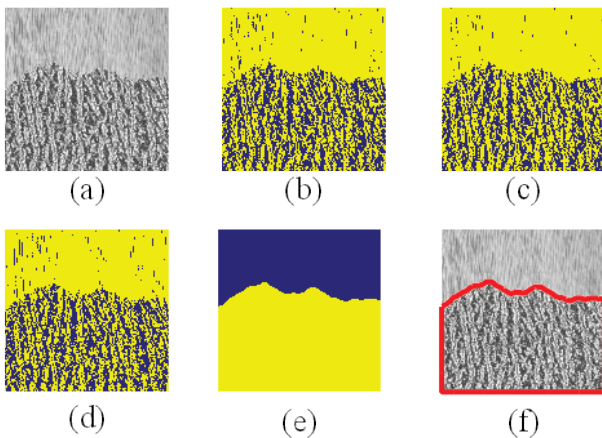


Figure 12 (a) Input Image, (b) k-means, (c) k-medoids, (d) fuzzy c-means, (e) Ground Truth, (f) Object Boundaries of Proposed Algorithm

Firstly optimal screen size is determined, and thereafter the result of PCA and it is compared with the previously computed parallel analyses as shown in Fig. 13.

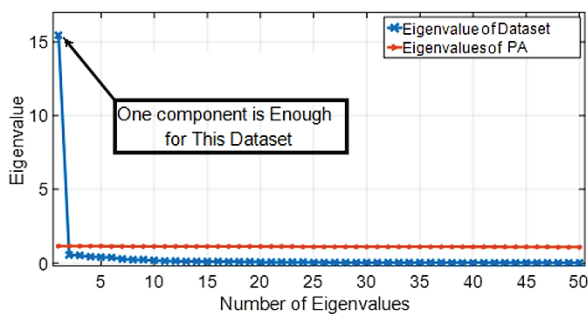


Figure 13 PA result for the input image in Fig. 12

As shown in Tab. 1, the highest percentage of success is obtained via first component and this result has been already pointed by PA before segmentation process in Fig. 12. For instance, the value of optimal screen size and component number for input image in Fig. 12 are determined as $s_i = 9 \times 9$ and 1, respectively. The result of single object segmentation is shown in Fig. 14. As shown in the figure below, the result of our proposed algorithm is quite similar to Human segmentation result and boundaries of the algorithm are fairly successful for this kind of mid-level segmentation problem [21].

As shown in the figure above, the first component is suitable for optimal representation of the image in Fig. 12. The error percentage of image segmentation process is calculated via the ratio between the total area of segmentation error and area of input image. Percentage of error according to selected component is shown in Tab. 1.

Table 1 Comparison of success for different number of components

Number of Components	Error Percentage (%)
1	0.904
2	1.762
3	3.432

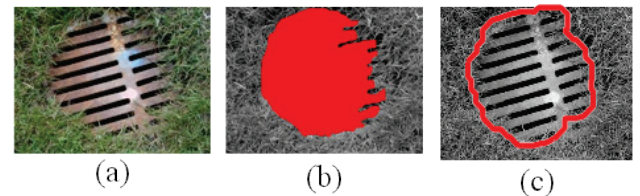


Figure 14 (a) Input Image, (b) Human Segmentation, (c) Proposed Algorithm

The performance of the proposed algorithm has been tested on a different type of images, such as medical tissue, hyperspectral and the results of proposed texture segmentation algorithm performance were compared with the other important texture segmentation methods in the literature [22]. These methods are defined in Tab. 2.

Table 2 Abbreviations of Methods

Abbreviation	Explanations
Method 1	Method based on contour [23]
Method 2	Method by based on fuzzy logic [24]
Method 3	Method by based on binary tree [25]
Method 4	Method by based on mutual information [22]
Method 5	Method by based on local histogram [22]

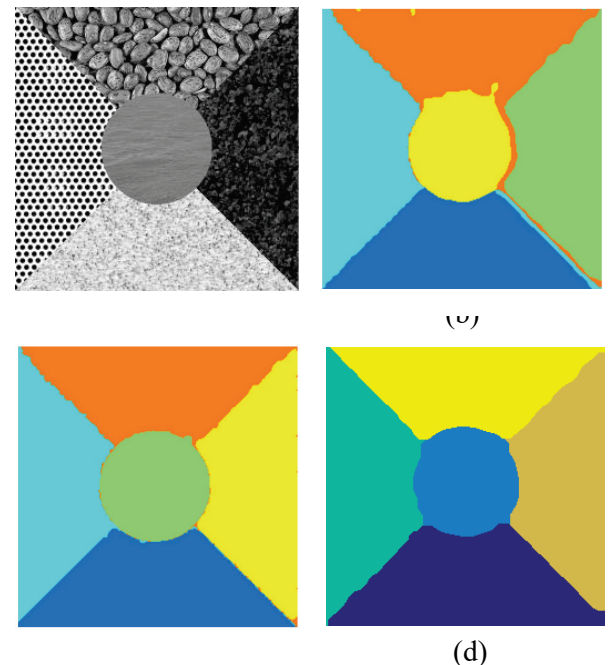


Figure 15 (a) Input Image, (b) Result of Method 4, (c) Result of Method 5, (d) Proposed Algorithm

The success of the method is compared with the methods which are important in the literature via an image containing 5 different textures in Fig. 15.

Experimental results for medical tissue are shown in Fig. 16. As shown in the figure below, the proposed algorithm is providing textures as a whole part. In this way, it avoids the local segmentation error.

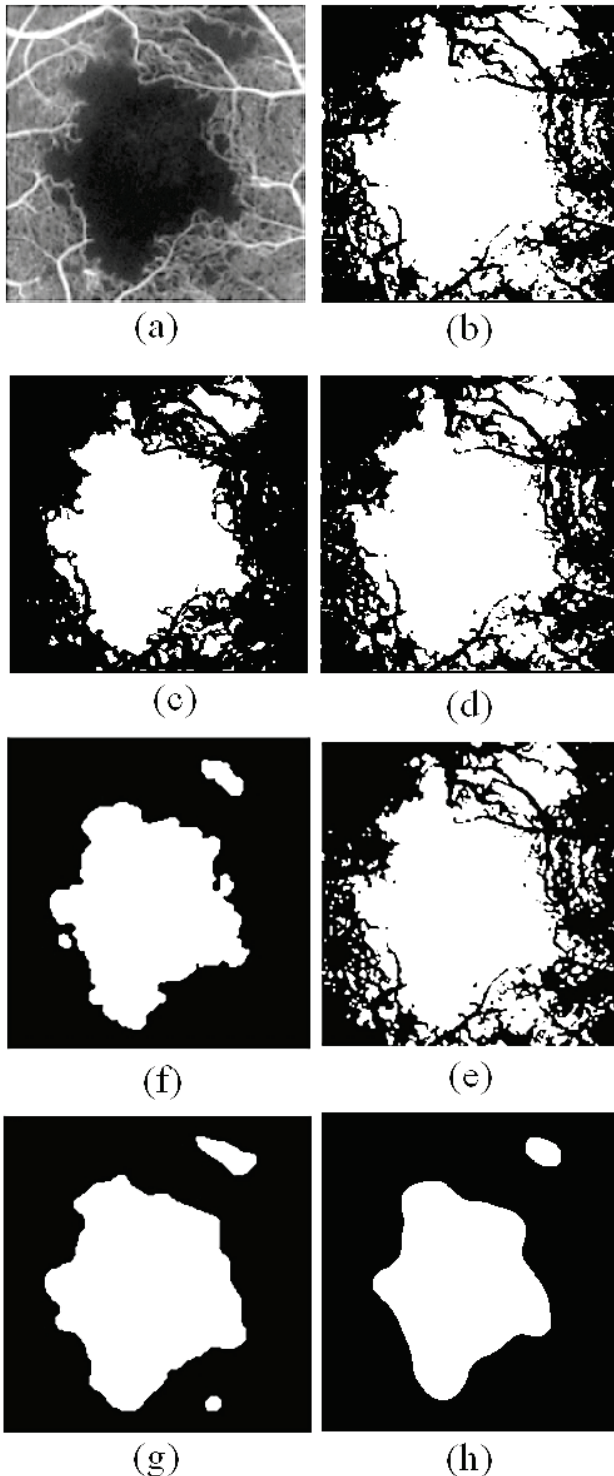


Figure 16 (a) Input Image, (b) k-means, (c) k-medoids, (d) fuzzy c-means, (e) Method 1, (f) Method 2, (g) Method 3, (h) Proposed Algorithm

6 CONCLUSION

In this work, an adaptive and robust texture segmentation algorithm is proposed. The algorithm combines several features of texture segmentation adaptively. This algorithm performs segmentation process unsupervised. The users do not have to define an initial

seed point for different regions; the proposed algorithm divides them into regions according to their own texture features. Firstly, the proposed algorithm performs pattern analysis with regional minimum and maximum information. This information assures real distribution of pattern at input images. Using this information, the sub-image sizes are determined without loss of the actual distribution of the texture pattern. In addition to that, an adaptive difference of Gaussian filter is applied to the input image to improve the performance of the proposed algorithms. The dimensions of these obtained features are reduced with the combination of PCA and Horn's parallel analysis. This combination is providing an optimal representation of dataset before the clustering process. Furthermore, a particular adaptive morphological mask has been developed for this work. Thanks to this adaptive mask, local errors on raw output are minimized and the region boundaries of the textures have been made more evident. The results of the proposed method were compared with several studies in the literature. Moreover, the results of the proposed method were compared with human segmentation. The proposed method has achieved more successful results and it is much more robust against noise. As a result of this work, a novel algorithm which is unsupervised and robust against noise has been proposed successfully.

7 REFERENCES

- [1] Min, H., Jia, W., Wang, X. F., Zhao, Y., Hu, R. X., Luo, Y. T., ... & Lu, J. T. (2015). An intensity-texture model based level set method for image segmentation. *Pattern Recognition*, 48(4), 1547-1562. <https://doi.org/10.1016/j.patcog.2014.10.018>
- [2] Dhanachandra, N., Mangle, K., & Chanu, Y. J. (2015). Image segmentation using K-means clustering algorithm and subtractive clustering algorithm. *Procedia Computer Science*, 54(2015), 764-771. <https://doi.org/10.1016/j.procs.2015.06.090>
- [3] Arbelaez, P., Maire, M., Fowlkes, C., & Malik, J. (2011). Contour detection and hierarchical image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 33(5), 898-916. <https://doi.org/10.1109/TPAMI.2010.161>
- [4] Khan, W. (2013). Image segmentation techniques: A survey. *Journal of Image and Graphics*, 1(4), 166-170. <https://doi.org/10.12720/joig.1.4.166-170>
- [5] Peng, B., Zhang, L., & Zhang, D. (2013). A survey of graph theoretical approaches to image segmentation. *Pattern Recognition*, 46(3), 1020-1038. <https://doi.org/10.1016/j.patcog.2012.09.015>
- [6] Wu, Q. & Yang, W. (2017). A Texture Segmentation Algorithm and Its Application to Target Recognition. In *Examining Information Retrieval and Image Processing Paradigms in Multidisciplinary Contexts* (pp. 51-72). IGI Global. <https://doi.org/10.4018/978-1-5225-1884-6.ch003>
- [7] Catalbas, M. C., Yildirim, M., Gulden, A., & Kurum, H. (2016, September). Optimal component selection for image segmentation via Parallel Analysis. In *Intelligent Systems (IS), 2016 IEEE 8th International Conference on* (pp. 499-502). IEEE. <https://doi.org/10.1109/IS.2016.7737468>
- [8] Camps-Valls, G., Tuia, D., Bruzzone, L., & Benediktsson, J. A. (2014). Advances in Hyperspectral Image Classification: Earth Monitoring with Statistical Learning Methods. *IEEE Signal Processing Magazine*, 31(1), 45-54. <https://doi.org/10.1109/MSP.2013.2279179>

- [9] Yuan, J., Wang, D., & Cheriyyadat, A. M. (2015). Factorization-based texture segmentation. *IEEE Transactions on Image Processing*, 24(11), 3488-3497. <https://doi.org/10.1109/TIP.2015.2446948>
- [10] Bailey, D. G. & Hodgson, R. M. (1985). Range filters: Local intensity subrange filters and their properties. *Image and Vision Computing*, 3(3), 99-110. [https://doi.org/10.1016/0262-8856\(85\)90058-7](https://doi.org/10.1016/0262-8856(85)90058-7)
- [11] Chang, D. C. & Wu, W. R. (1998). Image contrast enhancement based on a histogram transformation of local standard deviation. *IEEE transactions on medical imaging*, 17(4), 518-531. <https://doi.org/10.1109/42.730397>
- [12] Dai, T., Lu, W., Wang, W., Wang, J., & Xia, S. T. (2017). Entropy-based bilateral filtering with a new range kernel. *Signal Processing*, 137, 223-234. <https://doi.org/10.1016/j.sigpro.2017.02.005>
- [13] Gao, H. Y. & Lam, K. M. (2014, June). Saliency detection based on adaptive DoG and distance transform. In *Circuits and Systems (ISCAS), 2014 IEEE International Symposium on* (pp. 534-537). IEEE. <https://doi.org/10.1109/ISCAS.2014.6865190>
- [14] Faraji, M., Shanbehzadeh, J., Nasrollahi, K., & Moeslund, T. B. (2015). Extremal regions detection guided by maxima of gradient magnitude. *IEEE Transactions on Image Processing*, 24(12), 5401-5415. <https://doi.org/10.1109/TIP.2015.2477215>
- [15] Abdi, H. & Williams, L. J. (2010). Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4), 433-459. <https://doi.org/10.1002/wics.101>
- [16] Dinno, A. (2009). Exploring the sensitivity of Horn's parallel analysis to the distributional form of random data. *Multivariate behavioral research*, 44(3), 362-388. <https://doi.org/10.1080/00273170902938969>
- [17] Aggarwal, C. C. & Reddy, C. K. (Eds.). (2013). *Data clustering: algorithms and applications*. CRC press. <https://doi.org/10.1201/b15410>
- [18] Park, H. S. & Jun, C. H. (2009). A simple and fast algorithm for K-medoids clustering. *Expert systems with applications*, 36(2), 3336-3341. <https://doi.org/10.1016/j.eswa.2008.01.039>
- [19] Zheng, Y., Jeon, B., Xu, D., Wu, Q. M., & Zhang, H. (2015). Image segmentation by generalized hierarchical fuzzy C-means algorithm. *Journal of Intelligent & Fuzzy Systems*, 28(2), 961-973.
- [20] Soille, P. (2013). *Morphological image analysis: principles and applications*. Springer Science & Business Media.
- [21] Alpert, S., Galun, M., Brandt, A., & Basri, R. (2012). Image segmentation by probabilistic bottom-up aggregation and cue integration. *IEEE transactions on pattern analysis and machine intelligence*, 34(2), 315-327. <https://doi.org/10.1109/TPAMI.2011.130>
- [22] Zheng, Y., & Chen, K. (2012). A hierarchical algorithm for multiphase texture image segmentation. *ISRN Signal Processing*, 2012. <https://doi.org/10.5402/2012/781653>
- [23] Chan, T. F. & Vese, L. A. (2001). Active contours without edges. *IEEE Transactions on image processing*, 10(2), 266-277. <https://doi.org/10.1109/83.902291>
- [24] Lia, F. & Ng, M. K. (2010). Kernel density estimation based multiphase fuzzy region competition method for texture image segmentation. <https://doi.org/10.1.1.702.1049>
- Ni, K., Hong, B. W., Soatto, S., & Chan, T. (2009). Unsupervised multiphase segmentation: A recursive approach. *Computer Vision and Image Understanding*, 113(4), 502-510. <https://doi.org/10.1016/j.cviu.2008.12.006>

Contact information:

Mehmet Cem CATALBAS, Assistant Prof. Dr
OSTIM Technical University, Ankara, Turkey
mccatalbas@gmail.com