SELÇUK ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ

S.Ü. Müh. Bilim ve Tekn. Derg., c.6, s.2, ss. 175-187, 2018
Selcuk Univ. J. Eng. Sci. Tech., v.6, n.2, pp. 175-187, 2018
ISSN: 2147-9364 (Electronic)
DOI: 10.15317/Scitech.2018.124

FAST K-MEANS COLOR IMAGE CLUSTERING WITH NORMALIZED DISTANCE VALUES

[1]Nurdan AKHAN BAYKAN, [2]Ali SAGLAM

[1,2]*Selcuk University, Engineering Faculty, Computer Engineering, Konya, TURKEY*
[1] nurdan@selcuk.edu.tr, [2] alisaglam@selcuk.edu.tr

(Geliş/Received: 11.01.2017; Kabul/Accepted in Revised Form: 25.10.2017)

ABSTRACT: Image segmentation is an intermediate image processing stage in which the pixels of the image are grouped into clusters such that the data resulted from this stage is more meaningful for the next stage. Many clustering methods are used widely to segment the images. For this purpose, most clustering methods use the features of the image pixels. While some clustering method consider the local features of images by taking into account the neighborhood system of the pixels, some consider the global features of images. The algorithm of the K-means clustering method, that is easy to understand and simple to put into practice, performs by considering the global features of the entire image. In this algorithm, the number of cluster is given by users initially as an input value. For the segmentation process, if the distribution of the pixels over a histogram is used, the algorithm runs faster. The values in the histogram must be discrete in a certain range. In this paper, we use the Euclidean distance between the color values of the pixels and the mean color values of the entire image for taking advantage of the every color values of the pixels. To obtain a histogram that consists of discrete values, we normalize the distance value in a specific range and round the values to the nearest integers for discretization. We tested the versions of K-means with the gray-level histogram and the distance value histogram on an urban image dataset getting from ISPRS WG III/4 2D Semantic Labeling dataset. Comparing the two histograms, the distance value histogram proposed in this paper is better than the gray-level histogram.

Key Words: *Clustering, Histogram, Image segmentation, K-means*

Normalize Edilmiş Uzaklık Değerleri ile Hızlı K-ortalama Renkli Görüntü Segmentasyonu

ÖZ: Görüntü segmentasyonu, görüntü piksellerinin kümelere gruplandığı orta seviye bir görüntü işleme aşamasıdır. Öyle ki, bu şamadan elde edilen veri daha sonraki aşamalar için önceki veriye göre daha anlamlı hale gelmiş olur. Birçok kümeleme metodu, görüntü segmentasyonu amacıyla yaygın bir şekilde kullanılmaktadır. Bu amaçla, çoğu kümeleme metodu görüntü piksellerinin özelliklerini kullanmaktadır. Bazı kümeleme metotları piksellerin komşuluk sistemini kullanarak görüntünün yerel özelliklerini ele alırken, bazıları da görüntünün genel özelliklerini ele almaktadır. Anlaşılması kolay ve uygulaması basit olan K-ortalama algoritması, bütün görüntünün özelliklerini ele alarak segmentasyon yapmaktadır. Bu algoritmada, küme sayısı başlangıç giriş değeri olarak kullanıcı tarafından verilmektedir. Bu segmentasyon işlemi için, eğer piksellerin bir histogram üzerindeki dağılımı kullanılırsa algoritma daha hızlı çalışmaktadır. Bu histogram üzerindeki değerler belirli bir aralıkta ve ayrık olmak zorundadır. Bu çalışmada, piksellerin her bir renk değerinden faydalanmak için piksellerin renk değerleri ile görüntünün ortalama renk değerleri arasındaki Öklit uzaklığı kullanılmıştır. Ayrık değerlerden oluşan bir histogram elde etmek için, uzaklık değerlerini belirli değer aralığında normalize ettik ve bu değerleri ayrıklaştırmak için en yakınındaki tamsayıya yuvarladık. Bu K-means metodu, ISPRS WG III/4 2D Semantic Labeling veri setinden alınan kentsel görüntüler üzerinde gri seviye değerlerinin ve uzaklık mesafesi değerlerinin histogramları ile test edilmiştir.

Note: This paper has been presented at the Selcuk International Scientific Conference on Applied Sciences (ISCAS 2016) held in Antalya (Turkey), September 27-30, 2016.

İki histogram karşılaştırıldığında, uzaklık değeri histogramı gri seviye değer histogramından daha iyi sonuç vermiştir.

Anahtar Kelimeler: *Kümeleme, Histogram, Görüntü segmentasyonu, K-ortalama*

INTRODUCTION

The main goal of the clustering methods is to group the given data according to their similarities so that the elements of any group will be similar entities (de Amorim and Makarenkov, 2016). To solve many practical problems, such as image segmentation, the clustering methods are commonly used in image processing applications. Image segmentation is an important intermediate digital image-processing problem by this way the data becomes more useful and meaningful for next stages. Digital images are the discrete forms of images, such that it allows to make many processes on it easily and the storage of it in binary format. Image pixels are the smallest and the most fundamental elements of an digital image (Gonzalez and Woods, 2007).

Segmentation of an image mostly depends on the color values of image pixels and the neighborhood between them. The locally segmentation algorithms that respects to the neighborhood of pixels tend to achieving to the goal more quickly than globally segmentation algorithms that consider the features of the entire image for each operation, because the locally segmentation algorithms have inherently low computation complexities (Felzenszwalb and Huttenlocher, 2004; Saglam and Baykan, 2017). However, globally segmentation algorithms are favorable to analyze an image especially if the cluster number has been specified, because they regard the image elements as aggregate. The K-means algorithm, which is also commonly used to segment images, is the most popular clustering algorithm and it needs a specified cluster number. This algorithm is also known as Lloyd's algorithm (Lloyd, 1982).

The K-means algorithm initially chooses the cluster centers by randomly or some specified ways. A cluster center represents the related cluster which covers it. K-means makes progress iteratively. In the first iteration, each element in the whole data becomes a member of a cluster according to its closeness to the cluster centers. In the next iteration, each cluster center is recalculated by taking into account the average feature value (or values if the feature has a vector) of the elements in the cluster. After the calculation of the cluster centers, the memberships of the every element in the data are updated according to the values of the new cluster centers. If there is no changes in the cluster centers, the algorithm ends (Jain et al., 1999).

For the measurement of the closeness between the elements and the cluster centers in the clustering algorithm, the Euclidean distance has been used widely in the literature (Lin et al., 2014; Cheng et al., 2001; Rupali and Shweta, 2014). When the K-means algorithm is applied to a digital image for the segmentation purpose, the distances between many pixels and the k cluster centers must be calculated in each iteration of the algorithm. This means that it consumes a much processing time. To reduce the time complexity, the histogram of the pixel values such as grayscale values and brightness values is commonly used. In digital image processing, for a gray-level image, each pixel has a discrete gray-level integer value, which is ranging from 0 to 255 (Lin et al., 2014). For this reason, a gray-level histogram has 256 bins such that each represents a gray scale value. In a gray-level histogram, a bin indicates an integer number that represents the number of the pixels that have the same gray value; the value also equals the bin number. When the closest cluster center of a bin number is calculated, all of the pixels whose gray-level values equal the bin number are assigned to the same cluster.

Many image segmentation methods mostly use the color values of color images instead of their gray-level values to obtain more accurate results. However, if using the histograms of the every color elements is intended, too many probabilities turn up ($256 \times 256 \times 256$ for RGB color space). This needs very much processing time, therefore, in these cases, the histogram-based approach is not used, and the distance values between every pixels and every cluster centers are calculated in each iteration (Lin et al., 2014).

In this paper, to use the color values and to take advantage of the histogram-based fast approach, we use the distance values between the color values of pixels and the mean color values of all pixels of the entire images (the mean color values are normalized in a specific range and rounded to the nearest integer values) instead of the gray-level values of the pixels. The RGB color space has been used as color space, and the Euclidean distance has been used as distance measurement (Mignotte 2008; Cheng et al. 2001). We applied the fast gray-level histogram-based approach and the proposed approach based on the histogram of the color distances on the images obtained by airborne sensors acquired from ISPRS WG III/4 2D Semantic Labeling dataset (Axelsson 2000; Labeling and Vaihingen 2016). The ground truth of the images is also available in the data set. As the two histogram-based approaches are compared, it seems that the color distance histogram that is proposed in this paper is better than the gray-level histogram.

MATERIAL AND METHOD

Let $I = \{p_1, p_2, \dots, p_n\}$ be the pixel set of the input image I where the number of pixels of the input image is n.

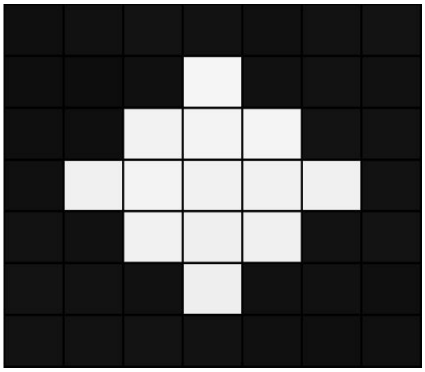
Traditional K-means Algorithm for Gray Level Image Segmentation

Let $I_{gray} = \{p_1^G, p_2^G, \dots, p_n^G\}$ be the set of gray-level values of image pixels of the input image I, such that p_i^G is the gray-level value of the pixel p_i, where $i = 1,2, \dots, n$, and $C^G = \{c_1^G, c_2^G, \dots, c_k^G\}$ be the cluster centers in the gray-level space where the specified number of cluster is k (Jain and Dubes 1988; Jain 2010). The distance value $d(p_i^G, c_j^G)$ between the gray value of ith pixel and jth cluster center is calculated as in Eq.1.

$$d\left(p_i^G, c_j^G\right) = |p_i^G - c_j^G| \tag{1}$$

Normally, the K-means algorithm ends when any changing at the positions of the cluster centers does not happen. However, for image segmentation, achieving this case may need too much loops because of being many pixels in the image. Due to this reason, a tolerance value can be applied to change the positions of the cluster centers (Lin et al. 2014; Likas et al. 2003). In this paper, we limit the number of loops to a upper limit value m instead of applying a tolerance value. Nevertheless, in our experiments, all cluster centers came to unchanging state for all images in the dataset before without reaching the upper limit of loops. The graphical demonstrations of the total changes of the cluster centers values for each image appear in the section result and discussion.

The output set I_{labels} defines the segmented image that consist of the label set $I_{labels} = \{p_1^L, p_2^L, \dots, p_n^L\}$ such that p_i^L corresponds to the label number of the ith pixel of the input image I, where $i = 1,2, \dots, n$, $p_i^L = 1,2, \dots, k$ and k is the number of cluster. The labels corresponding to the pixels within the same segment have the same label number (Figure 1), e.g. if p_i and p_j are resulted within the same segment, p_i^L would be equal to p_j^L.



15	17	19	18	16	18	19
14	13	16	245	17	18	17
15	15	242	243	244	19	18
16	240	243	241	241	240	18
18	17	241	239	240	18	19
19	18	17	238	17	17	15
18	18	19	18	16	15	16

1	1	1	1	1	1	1
1	1	1	2	1	1	1
1	1	2	2	2	1	1
1	2	2	2	2	2	1
1	1	2	2	2	1	1
1	1	1	2	1	1	1
1	1	1	1	1	1	1

a b c

Figure 1. (a) Input image (b) The gray-level values of the input image (c) Label values of the segmentation result of the input image

The algorithm of the traditional K-means for gray-levels is shown in Figure 2. According to the algorithm, the initial values of the cluster centers are assigned from 0 to 255 randomly. On the other hand, in literature, the process of assigning initial values of the cluster centers is an individual problem. Many researcher introduce various techniques that are relevant to this problem (Tian et al. 2013; Gingles and Celebi 2014). Because, the initializing of the cluster centers values can considerably influence the results of K-means. In each loop, each data chooses the closest cluster center to itself, and its corresponding label number is assigned as the selected cluster center index. After this process, all of the cluster centers are updated, and it continues to the next loop. These process is realized until the fact that there is no change at the values of the cluster centers or the number of loops are up to the limit value m.

INPUT: $I_{gray} = \{p_1^G, p_2^G, \dots, p_n^G\},\ k, m, n$
OUTPUT: $I_{labels} = \{p_1^L, p_2^L, \dots, p_n^L\}$

$t \leftarrow 0$
$C^{G,t} = \{c_1^{G,t}, c_2^{G,t}, \dots, c_k^{G,t}\}$
$C^{G,t} \leftarrow rand(0{,}255)$
repeat
  for $i \leftarrow 1$ **to** n **do**
    $min \leftarrow |p_i^G - c_1^{G,t}|$
    $p_i^L \leftarrow 1$
    for $j \leftarrow 2$ **to** k **do**
      $d(p_i^G, c_j^{G,t}) = |p_i^G - c_j^{G,t}|$
      if $d(p_i^G, c_j^{G,t}) < min$
        $min \leftarrow d(p_i^G, c_j^{G,t})$
        $p_i^L \leftarrow j$
      end if
    end for
  end for
  $t \leftarrow t + 1$
  for $a \leftarrow 1$ **to** k **do**
    $c_a^{G,t} \leftarrow mean(p_x^G)$, where $x = 1{,}2, \dots, n$ and $p_x^L = a$
  end for
until $t = m$ **or** $sum(C^{G,t} - C^{G,t-1}) = 0$

Figure 2. The algorithm of the traditional K-means for gray-level image segmentation

Fast K-means Algorithm Based on the Gray-Level Histogram

An ordinary image consists of many pixels. According to the traditional K-means, the distance value between every pixel and every clustering center must be calculated in each loop. This case consumes a lot of time. Reducing the time consuming, using the gray-level histogram of the image is a favorable method (de Amorim and Makarenkov 2016; Lin et al. 2014). A gray-level histogram of an digital image has certain number of bins such that each refers an integer gray-level value from 0 to 255. In the histogram, there are corresponding integer values for every bin, which denote the number of pixels having the same gray-level value (Figure 3). The algorithm of the fast K-means based on gray-level histogram can be seen in Figure 4.

The advantage of the algorithm, which ensures to accelerate the process, is that it calculates the distance values to the cluster centers 256 times instead of n times for each loop.

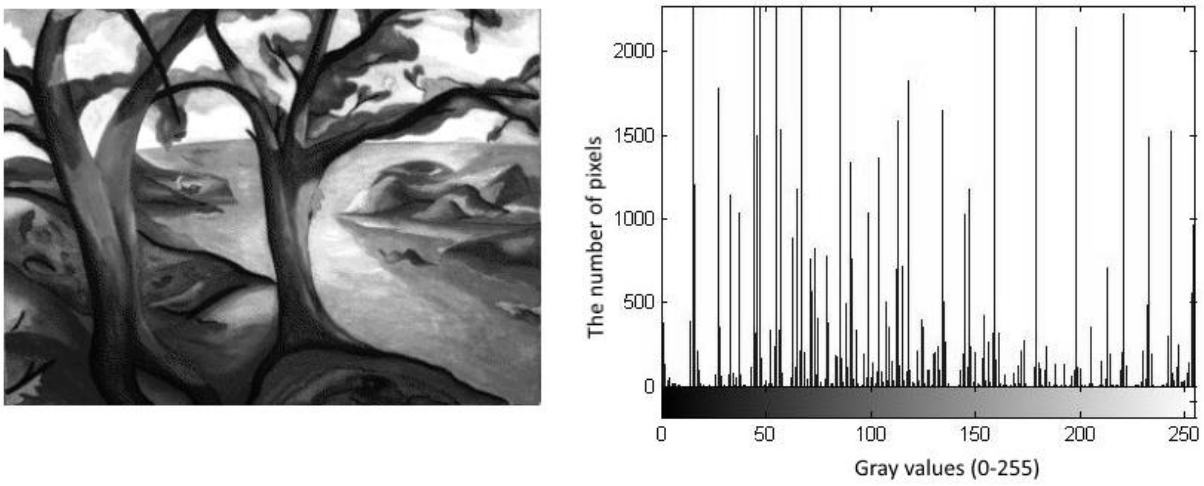


Figure 3. A gray-level image and its gray-level histogram

INPUT: $I_{gray} = \{p_1^G, p_2^G, \dots, p_n^G\},\ k, m$
OUTPUT: $I_{labels} = \{p_1^L, p_2^L, \dots, p_n^L\}$

$t \leftarrow 0$
$C^{G,t} = \{c_1^{G,t}, c_2^{G,t}, \dots, c_k^{G,t}\}$
$C^{G,t} \leftarrow rand(0{,}255)$
repeat
  for $r \leftarrow 0$ **to** 255 **do**
    $min \leftarrow |r - c_1^{G,t}|$
    $label \leftarrow 1$
    for $j \leftarrow 2$ **to** k **do**
      $d(r, c_j^{G,t}) = |r - c_j^{G,t}|$
      if $d(r, c_j^{G,t}) < min$
        $min \leftarrow d(r, c_j^{G,t})$
        $label \leftarrow j$
      end if
    end for
    $p_x^L \leftarrow label$, where $x = 1{,}2, \dots, n$ and $p_x^G = r$
  end for
  $t \leftarrow t + 1$
  for $a \leftarrow 1$ **to** k **do**
    $c_a^{G,t} \leftarrow mean(p_x^G)$, where $x = 1{,}2, \dots, n$ and $p_x^L = a$
  end for
until $t = m$ **or** $sum(C^{G,t} - C^{G,t-1}) = 0$

Figure 4. The algorithm of the fast K-means based on gray-level histogram

The Proposed Method Based on the Color Distance Histogram

Using more features of images, like color values of pixels for color images, provides better results in terms of accuracy than using gray-level values. In the RGB color spaces, each pixel in a color image has three-color values as red, green, and blue values (Peng et al. 2013; Cheng et al. 2001; Dai et al. 2015; Mignotte 2008). Therefore, the cluster centers have a vector that consists of three values instead of one value. For measuring the distance value between the color value vectors and the cluster centers' vector, the Euclidean distance is generally used, and its equation is that as seen in Eq. 2 (Lin et al. 2014). $p_{i,R}^{RGB}$, $p_{i,G}^{RGB}$ and $p_{i,B}^{RGB}$ refer to the red, the green and the blue values (respectively) of the ith pixel of the given color image. $c_{j,R}^{RGB}$, $c_{j,G}^{RGB}$ and $c_{j,B}^{RGB}$ refer to the vector values of the center of the jth cluster. The distance value $d(p_i^{RGB}, c_j^{RGB})$ refers to the Euclidean distance between the color vector p_i^{RGB} of the ith pixel and the jth cluster center c_j^{RGB}.

$$d(p_i^{RGB}, c_j^{RGB}) = \sqrt{\left(p_{i,R}^{RGB} - c_{j,R}^{RGB}\right)^2 + \left(p_{i,G}^{RGB} - c_{j,G}^{RGB}\right)^2 + \left(p_{i,B}^{RGB} - c_{j,B}^{RGB}\right)^2} \quad (2)$$

However, if the histograms for every color values are used, too much probability of the number of bins turns up to combine them into one histogram ($256 \times 256 \times 256$ for RGB color space). To take advantage of the color features of a color image and the speed of the histogram-based approach, we propose the method that bases on the distance values between the color values of the pixels and the mean color values of the entire image (Figure 5). Let the mean color values be $M_{RGB} = \{\mu_R, \mu_G, \mu_B\}$ for the means of the red, green, and blue values respectively. To use the distance values on a histogram, the values need to be discrete and normalized. Therefore, we firstly calculate the distance values between the color values of the pixels and the mean color values of the entire input image, and then, we

normalize the distance values from 0 to 255 according to the minimum and maximum distance values. Let $D = \{d_1, d_2, \dots, d_n\}$ be the set of the distance values of the color values of the pixels to the mean color values, and $D^N = \{d_1^N, d_2^N, \dots, d_n^N\}$ be the set of normalized form of D. The normalized distance value d^N is calculating as seen in Eq. 3.

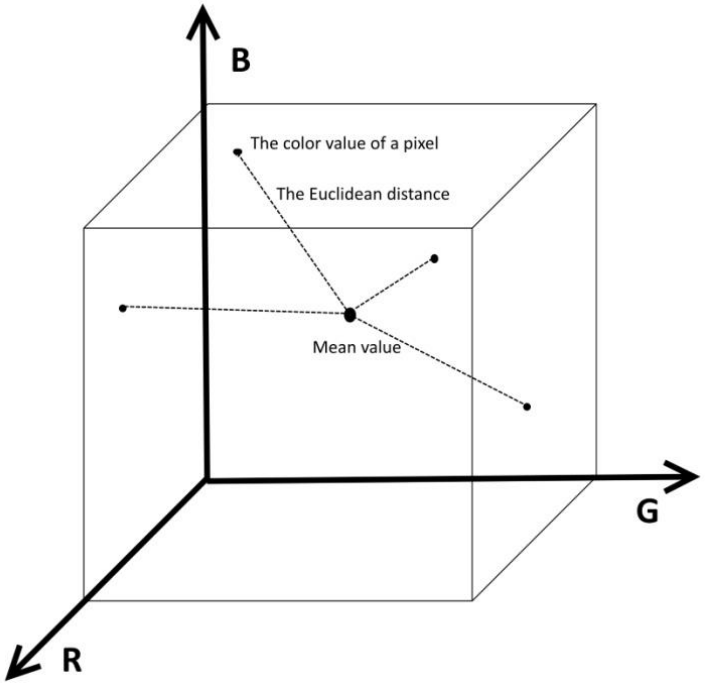


Figure 5. The Euclidean distance in the RGB color space between color values of a pixel and the mean color values of the entire given color image

INPUT: $= \{p_1^{RGB}, p_2^{RGB}, \dots, p_n^{RGB}\},\ n$
OUTPUT: $[D^N] = \{[d_1^N], [d_2^N], \dots, [d_n^N]\}$

for $i \leftarrow 1$ **to** n **do**
$d_i \leftarrow d(p_i^{RGB}, M_{RGB})$

$$= \sqrt{\left(p_{i,R}^{RGB} - \mu_R\right)^2 + \left(p_{i,G}^{RGB} - \mu_G\right)^2 + (p_{i,B}^{RGB} - \mu_B)^2}$$

end for
$min \leftarrow \min(D)$
$\max \leftarrow \max(D)$
for $i \leftarrow 1$ **to** n **do**
$d_i^N \leftarrow \frac{d_i - \min}{\max - \min} . 255$
$[d_i^N] \leftarrow round(d_i^N)$
end for

Figure 6. The algorithm of obtaining the normalized distancee values between the RGB color values of pixels and the mean RGB color values of the entire image

$$d_i^N = \frac{d_i - \min(D)}{\max(D) - \min(D)} . 255 \quad (3)$$

After the normalizing process, we round these normalized values to nearest integer numbers to discrete them. $[d_i^N]$ refers the discrete form of d_i^N, and the normalized distance value set $[D^N]$ is the discrete expression of D^N. The algorithm of these processes is shown in Figure 6. An example demonstration of a distance value histogram is presented in Figure 7.

After the algorithm shown in Figure 6, we use the set $[D^N] = \{[d_1^N], [d_2^N], \dots, [d_n^N]\}$ as the input value set instead of the gray-level value set $I_{gray} = \{p_1^G, p_2^G, \dots, p_n^G\}$ in the algorithm shown in Figure 4.

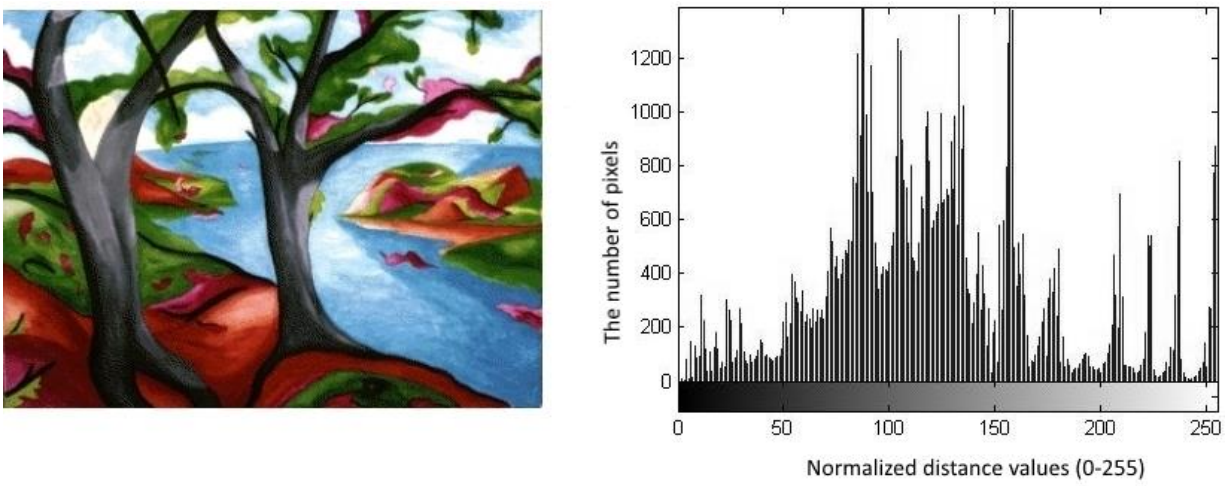


Figure 7. A color image and its distance value histogram

RESULT AND DISCUSSION

We use the 2D Semantic Labeling - Vaihingen data to test the accuracy of the proposed method. This data set was captured over the region of Vaihingen in Germany, such that the data set consists of the sub-parts of the data used for the test of digital aerial cameras carried out by the German Association of Photogrammetry and Remote Sensing (DGPF) (Axelsson 2000; Labeling and Vaihingen 2016). The data set contains 33 area images of different sizes and each consists of a true orthophoto (TOP) extracted from a larger TOP mosaic as seen in Figure 8. But, the ground truths of only 16 images of them are available in the data set. Hence, we use the 16 images which have a ground truth.

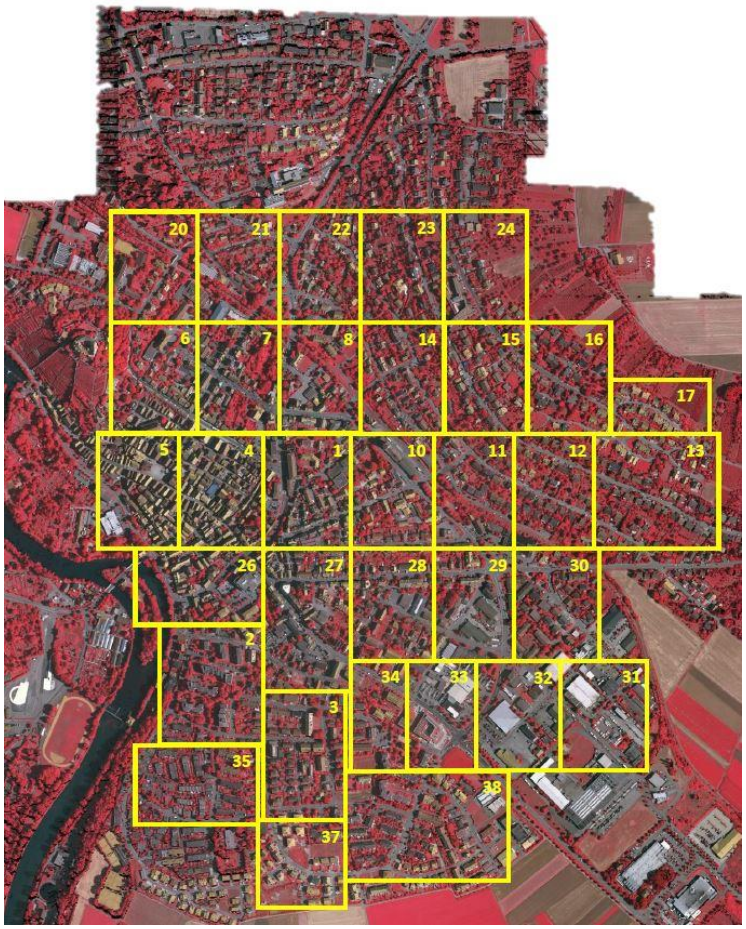


Figure 8. The larger true orthophoto mosaic and the areas

The data set actually is prepared for the automatic extraction of urban objects such as buildings, roads, or trees. This process includes the classification and the object recognition issues. We use the data set for clustering; therefore, it is no matter that which segment defines which object. The main problem of the clustering process is separating the objects from each other. Among the 16 ground truth, 11 ground truth images consist of 5 clusters, while 5 ground truth images consist of 6 clusters. Thus, we determine the cluster number parameter k according to these numbers for each image.

We compare the gray-level histogram-based K-means and the distance value histogram-based K-means. For the two methods, the initial cluster center values have an important effect to the clustering results. Thus, we determine the specific number for the initial cluster centers to keep stability and true comparing. To specify the initial cluster center numbers for the gray-level histogram-based method, we use the mean gray-level values of pixels of the intended regions for each cluster by looking up the ground truth, and similarly, for the distance value histogram-based method the mean distance values of pixels of the intended regions. As the upper limit value m for the loop number, we chose the value 50. All of the cluster centers achieve to stability for all images in the dataset before without reaching the upper limit 50. The graphical projections of the total changes of the cluster centers values for each image can be seen in Figure 9 for the gray-level histogram-based method and in Figure 10 for the normalized distance value histogram-based method. After segmentation process, to remove small cluster particles, we apply the median filter (3×3) on the output label values as post-processing.

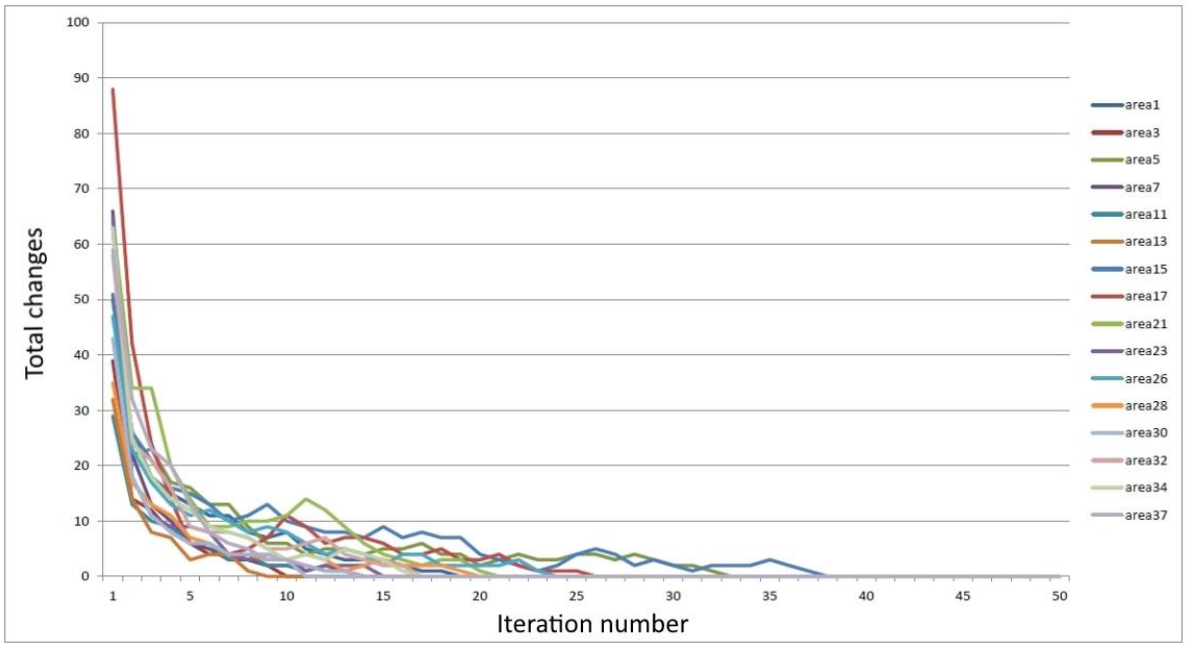


Figure 9. Total changes of the cluster centers values for each image for the gray-level histogram-based method

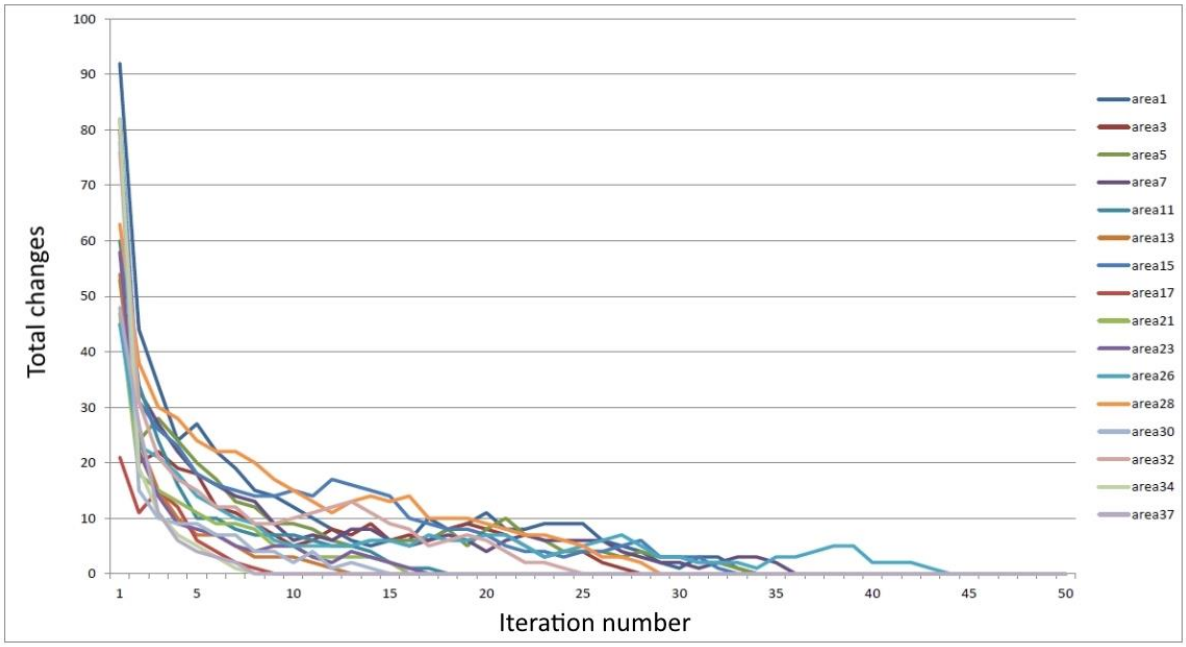
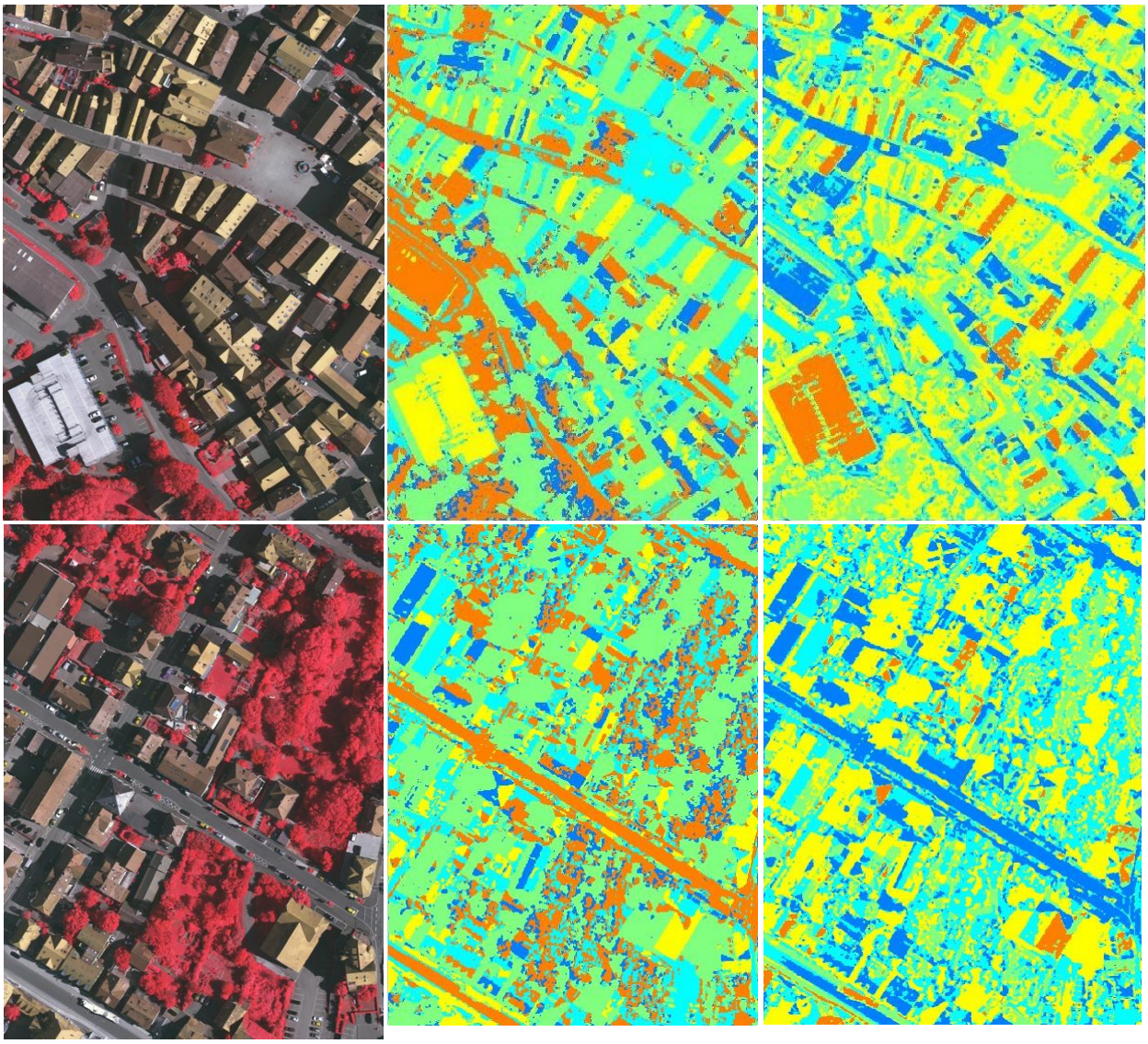


Figure 10. Total changes of the cluster centers values for each image for the normalized distance value histogram-based method



a b c

Figure 11. (a) Input images "area5" (top) and "area7" (bottom), (b) their colored segmentation results for the gray-level histogram-based method and (c) for the distance value histogram-based (the proposed) method

In Figure 11, an example image and its colored segmentation results for the two methods are demonstrated. We use MATLAB to test the algorithms. In Table 1, the PSNR results of the two methods can be seen. The PSNR is the abbreviation of "peak signal to noise ratio" and it can be widely used as a measure of segmentation quality (Punjab and Punjab 2012). The signal in this data set is the segmentation results and the ground truth images for these segmentation methods. A higher PSNR value means that the segmentation result is of higher quality. Looking the results, it seems that the results of the distance value histogram-based method are better than the gray-level histogram-based method for all of the images.

Table 1. Segmentation results

Images Name	k	Gray-level histogram-based (PSNR)	Distance value histogram-based (PSNR)
area1	5	42.5407	42.9248
area3	5	42.3794	43.6914
area5	5	43.3873	44.1835
area7	5	42.8795	43.8389
area11	5	43.0932	44.4425
area13	5	41.6641	43.9590
area15	6	42.5164	43.4017
area17	5	41.0036	44.8187
area21	5	42.4975	44.9825
area23	6	42.5538	43.4264
area26	6	43.3340	43.9569
area28	6	43.2629	43.5551
area30	5	41.5260	43.3239
area32	6	41.7520	44.3433
area34	5	42.0649	44.3584
area37	5	40.4618	44.9234
AVERAGE		**42.3073**	**44.0082**

CONCLUSION

In this paper, we demonstrate the traditional K-means clustering algorithm, its fast form based on gray-level histogram, and the proposed method based on the distance value histogram for color images. The histogram-based K-means algorithm is quite fast compared to the traditional K-means algorithm when they are applied on gray-level images. Color image pixels have a feature vector, which consists of commonly three values, instead of a feature value. This case complicates the implementation of the histogram-based approach for color images. To utilize the fast approach, color images are generally converted to the gray-level images before using the histogram-based algorithm; however, the contribution of the color values to the segmentation process is lost. In this study, we aim to take advantage of color values; and for this purpose, we use the normalized distance values of color images for the histogram-based approach. In a normalized distance value histogram, the distance refers to the Euclidean distance between the color value vectors of pixels and the mean color value vector of the entire image. The results show that the proposed method provides better segmentation results than the gray-level histogram-based method.

ACKNOWLEDGE

This study was supported by "Scientific Research Projects of Selcuk University". This study has been presented as an oral presentation at ISCAS'2016 conference held in Antalya (Turkey), 27-30 September 2016 and selected for the journal of SUJEST (Selcuk University Journal of Engineering, Science and Technology).

REFERENCES

de Amorim, R.C., Makarenkov, V., 2016, "Applying Subclustering and Lp Distance in Weighted K-Means with Distributed Centroids", *Neurocomputing*, Vol. 173, pp. 700–707.

Axelsson, P., 2000, "DEM Generation from Laser Scanner Data Using adaptive TIN Models", *International Archives of Photogrammetry and Remote Sensing*, Vol. 33(Part B4), pp.110–117.

Cheng, H.D., Jiang, X. H., Sun, Y., Wang, J., 2001, "Color Image Segmentation: Advances and Prospects", *Pattern Recognition*, Vol. 34(12), pp.2259–2281.

Dai, S., Lu, K., Dong, J., Zhang, Y., Chen, Y., 2015, "A Novel Approach of Lung Segmentation on chest CT Images using Graph Cuts", *Neurocomputing*, Vol. 168, pp.799–807.

Felzenszwalb, P.F., Huttenlocher, D.P., 2004, "Efficient Graph-Based Image Segmentation", *International Journal of Computer Vision*, Vol. 59(2), pp.167–181.

Gingles, C., Celebi, M.E., 2014, "Histogram-Based Method for Effective Initialization of the K-Means Clustering Algorithm", *The Twenty-Seventh International Flairs Conference* In *FLAIRS Conference*, Florida, 21-23 May 2014.

Gonzalez, R.C., Woods, R.E., 2007, *Digital Image Processing (3rd Edition)*, Pearson International Edition.

Jain, A.K., 2010, "Data Clustering: 50 Years Beyond K-means", *Pattern Recognition Letters*, Vol. 31(8), pp.651–666.

Jain, A.K., Dubes, R.C., 1988, *Algorithms for Clustering Data*, Prentice Hall, 355, p.320.

Jain, A.K., Murty, M.N., Flynn, P.J., 1999, "Data Clustering: a Review", *ACM Computing Surveys*, Vol. 31(3), pp.264–323.

Labeling, S., Vaihingen, B., 2016, *Use of the Stair Vision Library within the ISPRS Use of the Stair Vision Library within the ISPRS 2D*,

Likas, A., Vlassis, N., J. Verbeek, J., 2003, "The Global k-means Clustering Algorithm", *Pattern Recognition*, Vol. 36(2), pp.451–461.

Lin, C.-H., Chen, C. C., Lee, H. L., Liao, J. R., 2014, "Fast K-means Algorithm Based on a Level Histogram for Image Retrieval", *Expert Systems with Applications*, Vol. 41(7), pp.3276–3283.

Lloyd, S.P., 1982, "Least Squares Quantization in PCM", *IEEE Transactions on Information Theory*, Vol. 28(2), pp.129–137.

Mignotte, M., 2008, "Segmentation by Fusion of Histogram-Based K-means Clusters in Different Color Spaces", *IEEE Transactions on Image Processing*, Vol. 17(5), pp.780–787.

Peng, B., Zhang, L., Zhang, D., 2013, "A survey of Graph Theoretical Approaches to Image Segmentation", *Pattern Recognition*, Vol. 46(3), pp.1020–1038.

Punjab, P., Punjab, P., 2012, "Performance Analysis of Segmentation Techniques", *International Journal of Computer Applications*, Vol. 45(23), pp.18–23.

Rupali, N., Shweta, J., 2014, "Color Image Segmentation With K Means", *Saiom publications*, Vol. 1(5), pp.389–397. Available at:

Saglam, A., Baykan, N.A., 2017, "Sequential Image Segmentation Based on Minimum Spanning Tree Representation", *Pattern Recognition Letters*, Vol. 87, pp.155–162.

Tian, M., Yang, Q., Maier, A., Schasiepen, I., Maass, N., Elter, M., 2013, "Automatic Histogram-Based Initialization of K-means Clustering in CT". In *Bildverarbeitung für die Medizin 2013*, Springer Berlin Heidelberg, pp. 277–282.