Gazi Üniv. Müh. Mim. Fak. Der.
Cilt 29, No 4, 655-664, 2014

Journal of the Faculty of Engineering and Architecture of Gazi University
Vol. 29, No 4, 655-664, 2014

ALT-BLOKLAR TEKNİĞİ VE KÜMELEME YÖNTEMLERİ İLE GÖRÜNTÜ BÖLÜTLEMENİN HIZLANDIRILMASI

Melike ŞİŞECİ[1*], Sedat METLEK[2], Bayram CETİŞLİ[3]
[1]Bucak Zeliha Tolunay Uyg. Teknoloji ve İşletmecilik Yüksekokulu, Mehmet Akif Ersoy Üniversitesi, Burdur
[2]Burdur MYO, Mehmet Akif Ersoy Üniversitesi, Burdur
[3]Bilgisayar Mühendisliği, Süleyman Demirel Üniversitesi, Isparta
melikesiseci@mehmetakif.edu.tr, sedatmetlek@hotmail.com, bayramcetisli@sdu.edu.tr

(Geliş/Received: 06.11.2012; Kabul/Accepted: 16.10.2014)

ÖZET

Bu çalışmada, kümeleme algoritmaları ile bölütlenen resimlerin daha hızlı bölütlenmesi için, alt-bloklama tekniğinin kullanılması amaçlanmaktadır. Genellikle, görüntü bilgisinin tamamının kümeleme algoritmalarına verilmesinden dolayı, kümeleme işlemi uzun sürmekte ve gerçek zamanlı bölütleme uygulamalarında gecikmeler olmaktadır. Bu çalışmada; gecikmeyi azaltmak için görüntü alt-bloklara ayrılarak, sadece alt-blok ortalamalarının kümelemeye sokulması önerilmektedir. Böylece kümeleme verisi oldukça azalmaktadır. Deneysel çalışmalarda traverten plaka resimlerinin yanı sıra, "Lena", "Baboon" gibi bilindik resimler de kullanılmıştır. Önerilen yöntem, K-ortalamalar, Bulanık C-ortalamalar, K-ortaylar ve Spektral kümeleme yöntemleri ile kıyaslanmış ve 2-4 kat hızlanma sağlanmıştır. Blok boyutunun küçük tutulması durumunda görüntü kalitesinin çok fazla değişmediği de gözlenmiştir.

Anahtar Kelimeler: Hızlandırılmış bölütleme, K-ortalamalar, Kümeleme, Alt-bloklar, Görüntü işleme, Gerçek zamanlı uygulamalar.

ACCELERATING THE IMAGE SEGMENTATION USING SUB-BLOCK TECHNIQUE AND CLUSTERING METHODS

ABSTRACT

In this study, it is aimed to use sub-block technique for the purpose of speeding up the segmentation of images within clustering algorithms. Due to the fact that all of the image data is given to clustering algorithms, generally the clustering process takes a lot of time, and it causes delays in real-time segmentation applications. In this study, in order to minimize the delays, the dividing of images into the sub-blocks, and using the average values of sub-blocks for the clustering process are proposed. As a result, the size of clustering data is relatively decreased. In the experimental studies, besides the images of travertine plates, well-known images such as "Lena" and "Baboon" were also used. The proposed method was compared with K-means, Fuzzy C-means, K-medoids and Spectral clustering methods, and its speed increased 2-4 times. Furthermore, it was observed that the image quality did not change too much in case of small size of blocks.

Key Words: Accelerated segmentation method, K-means, Clustering, Sub-blocks, Image processing, Real-time applications.

1. GİRİŞ (INTRODUCTION)

Görüntü bölütleme, sayısal görüntü işlemede uzun süredir kullanılan önemli ve aynı zamanda zor bir konudur. Bu işlem, resim içinde birbirinden kolayca ayrılabilen ve kendi içinde homojen olabilen farklı bölgelerin elde edilmesini amaçlar. Resimler üzerinde doku analizi veya tanıma yapılmak istenildiğinde ilk önce bölütleme (segmentasyon) işlemleri yapılır [1]. Bölütleme işlemi için ise genellikle kümeleme algoritmaları kullanılır. Kümeleme işlemi ötelemeli bir yöntem olup zaman almaktadır.

Gerçek zamanlı çalışmalarda bölütlemenin hızlı yapılabilmesi için kümeleme algoritmalarının da hızlı olması ya da bölütlenen verinin azaltılması gerekmektedir. Bu çalışma, endüstriyel bir uygulama olan doğal taş tanımanın bir parçasıdır. Şekil 1'de

TEYDEB 1511 Öncelikli Alanlar Araştırma Teknoloji Geliştirme ve Yenilik Projeleri Destekleme Programı'nda yer alan 1120167 nolu "Doğal Taş Plakaların Makine Görmesi ile Otomatik Sınıflandırılması" projesinde tasarlanan makine görme sistemi görülmektedir. 80 adet traverten plaka resimleri bu sistemden alınmıştır. Örnek traverten resimleri ise Şekil 2'de gösterilmektedir.

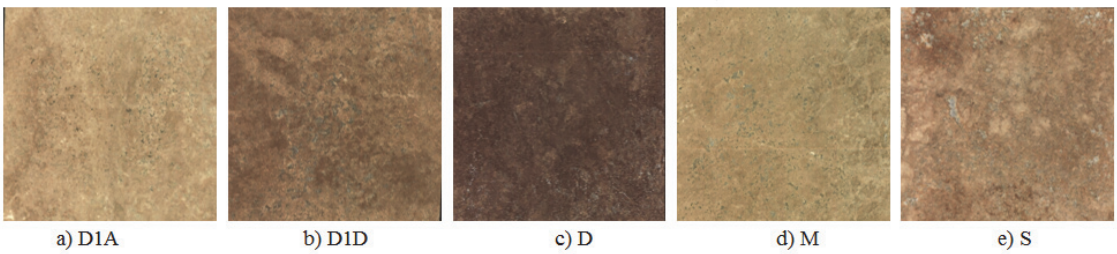
Gerçek zamanlı bir çalışmada, iyi bir tanıma için bölütleme önemlidir ve uzun süren bu sürecin kısaltılması gerekmektedir. Bunun için iki yol önerilebilir: Bunlardan ilki, resmi küçülterek kümelemek, diğeri ise kümelemeyi daha az veri ile yapmaktır. Resmi küçültmek, hem fazladan süreç gerektirdiğinden ve resmin özgünlüğü kaybolduğundan dolayı bu çalışma için uygun görülmemiştir. Ayrıca resmi küçültmek, alt-bloklar tekniğinden daha uzun sürmektedir. Bu anlamda, en mükemmel halini almış kümeleme algoritmalarının yapısını değiştirmek yerine, algoritmaya daha az verinin verilmesi uygun olacaktır. Alt-bloklar tekniği ile bütün pikseller yerine alt-blok ortalamalarını kullanmak, kümeleme verisini azaltacaktır. Ortalamaların kullanılması hızlandırmayı sağlarken, doğal olarak, bölütlenmiş resimdeki yumuşatma oranını arttıracaktır. Bu çalışmada; "Lena", "Baboon" ve traverten plaka resimleri kullanılarak, önerilen yöntemin hızlandırma ve yumuşatma etkileri incelenmektedir.

Bu çalışmada kümeleme için önerilen alt-bloklama tekniği birçok çalışmada farklı amaçlar için kullanılmıştır. Bunlardan bazıları; öznitelik çıkarma [2], tümör resimlerinde renk indirgeme ve gürültüyü azaltma [3], görüntü sıkıştırma [4], dörtlü ağaç bölümleme ile vektör kuantalama [5], kopyalanmış resim tespiti [6], hava resimlerinde kenar tespiti [7], takip eşleştirme özelliklerinin üretilmesi [8] yazının arka plan resminden ayrıştırılması [9], manzara resimlerini semantik kategorilere ayırma [10] ve mikroskobik hücre çekirdeği bölümlemesidir [11].

Bölütlemede; bölgelerin tespiti için genellikle K-ortalamalar (KO), Bulanık C-ortalamalar (BCO), K-ortaylar (KM) vb. gibi kümeleme algoritmaları kullanılmaktadır [1-11]. Kümeleme, öğreticisiz bir öğrenme tekniğidir; veri madenciliği ve örüntü tanıma gibi alanlarda sıklıkla kullanılmaktadır. Kümeleme algoritmaları, bölücü, hiyerarşik, yoğunluk, ızgara ve model tabanlı gibi farklı gruplara ayrılmaktadır [12]. Görüntülerin bölütlenmesinde öz örgütlemeli harita (SOM) ağları ve gri düzey eş oluşum matrislerinin de (GLCM) kullanıldığı görülmektedir. SOM bölütlemede iyi sonuçlar vermesine karşın KO kümeleme tekniğine göre daha karmaşık yapıya sahiptir ancak sınırlılıkları vardır [13]. KO algoritması küçük ve orta ölçekli problemler için mevcut algoritmalar içinde en hızlı ve en doğru sonuç üreten algoritmadır [14]. Bu nedenle, önerilen hızlandırma tekniği sadece KO algoritmasında ele alınmıştır.



Şekil 1. Doğal taş tanıma için tasarlanan makine görme sistemi (Designed machine vision system for natural stone recognition)



Şekil 2. Örnek traverten plaka sınıfları. (Class samples of travertine slabs.)

Bu çalışma, aşağıdaki gibi organize edilmiştir. Bölüm 2'de hiyerarşik olmayan kümeleme yöntemleri verilmektedir. Yeni geliştirilen alt-bloklara ayırarak kümeleme yöntemi Bölüm 3'te anlatılmaktadır. Deneysel çalışmalar, Bölüm 4'te; sonuçlar ise Bölüm 5'tedir.

2. HİYERARŞİK OLMAYAN KÜMELEME YÖNTEMLERİ (NON-HIERARCHICAL CLUSTERING METHODS)

Hiyerarşik olmayan kümeleme algoritmaları, $\mathbf{X} \in R^{Nxd}$ veri tabanında yer alan ve doğal sınıfları kesin olarak verilmeyen örnekleri belli bir uzaklık kavramıyla K adet gruba ayıran ve alt kümeleri oluşturan algoritmalardır [15,16]. Genellikle rastgele başlangıç değerleri ile sürekli küme içi dağılımları en küçük yapan ötelemeli algoritmalarla kümeler oluşturulur.

2.1. K-Ortalamalar Kümeleme (K-Means Clustering)

En eski kümeleme algoritmalarından olan K-ortalamalar (KO), MacQueen tarafından geliştirilmiştir [17]. En yaygın kullanılan ve öğreticisiz öğrenme yöntemlerinden birisi olan KOın atama mekanizması, her verinin sadece bir kümeye ait olabilmesine izin verir. Bu nedenle, keskin bir kümeleme algoritmasıdır. Küme merkezinin kümeyi temsil etmesi ana fikrine dayalı bir yöntemdir [15].

KO algoritmasında her bir nesnenin merkez noktalara uzaklığını hesaplamak için en çok kullanılan ölçütler Öklid, Manhattan, Minkowski ve Pearson metrikleridir [17].

KO algoritması N adet $\mathbf{x}_j$ $\{j = 1,2, \dots, N\}$ örneklerini K adet gruba G_i, $\{i = 1,2, \dots, K\}$ bölütler ve her grup için uzaklık ölçümü olarak tanımlanan amaç fonksiyonunu küçültme yoluyla $\mathbf{c}$ küme merkezlerini bulmaya çalışır. Eğer uzaklık ölçütü Öklid uzaklığı seçilirse, G_i grubunda yer alan $\mathbf{x}_k \in G_i$ örneklerinin $\mathbf{c}_i$ küme merkezine olan uzaklıkları hesaplanarak toplam uzaklık maliyeti şu şekilde tanımlanır:

$$J = \sum_{i=1}^{K} J_i = \sum_{i=1}^{K} (\sum_k \|\mathbf{x}_k - \mathbf{c}_i\|^2) \quad (1)$$

Burada J_i, i. grubun amaç fonksiyonunu; G_i ise i. gruba ait örnekler kümesini; $\mathbf{x}_k$ ise G_i'ye ait örnekleri göstermektedir. J_i, G_i kümesinin geometrik özelliklerine ve $\mathbf{c}_i$ küme merkezlerine bağlıdır [16].

Parçalanan gruplar genellikle $K \times N$ boyutundaki $\mathbf{U}$ ikili üyelik matrisi ile tanımlanır. Eğer $\mathbf{x}_j$ verisi i. grup altında kalırsa u_{ij} değeri 1'dir, değilse 0'dır.

Eşitlik (2)'de $\mathbf{c}_k$; i. grup dışındaki kümelerin merkezlerini göstermektedir. Bütün veriler bir gruba dâhil edildiğinden dolayı $\mathbf{U}$ matrisi Eşitlik 3'teki özellikleri sağlar:

$$u_{ij} = \begin{cases} 1, Eğer\ \|\mathbf{x}_j - \mathbf{c}_i\|^2 \le \|\mathbf{x}_j - \mathbf{c}_k\|^2, \text{her bir } k \neq i\ için \\ 0, Diğer\ durumlarda \end{cases} \quad (2)$$

$$\sum_{i=1}^{K} u_{ij} = 1, \forall j = 1, \dots, n, \text{ ve } \sum_{i=1}^{K} \sum_{j=1}^{N} u_{ij} = n \quad (3)$$

Eğer u_{ij} sabitlenirse en iyi küme merkezi $\mathbf{c}_i$ Eşitlik (1)'i küçültür ve $\mathbf{c}_i$'nin yeni değeri:

$$\mathbf{c}_i = \frac{1}{|G_i|} \sum_{k, \mathbf{x}_k \in G_i} \mathbf{x}_k, \quad (4)$$

Burada $|G_i|$ i. gruba ait eleman sayısını gösterir.

Ötelemeli bir algoritma olan KO, belirlenen sayıdaki küme merkezlerini ve aitlik matrisini her adımda yeniler. Amaç fonksiyonu yeteri kadar küçültüldüğünde ya da verilerin aitliği değişmediğinde algoritma sonlandırılır. Aşağıda KO algoritması verilmektedir.

1. K küme sayısını belirle. Görüntünün her renkteki en küçük ($\mathbf{m}_n$) ve en büyük ($\mathbf{m}_x$) piksel değerlerine göre aralığını belirle.
2. $\mathbf{c}$ küme merkezlerinin başlangıç değerlerini $\mathbf{m}_n - \mathbf{m}_x / K$ ile eşit aralıkta ile belirle. I adım sayısını belirle. $\mathbf{J} = [\mathbf{0}]_{K,N}$, $\mathbf{U} = [\mathbf{0}]_{K,N}$.
3. For i=1:I
4. For j=1:K
5. For k=1:N
6. $J(j,k) = \sum_{l=1}^{d} (x(k,l) - c(j,l))^2$
7. End
8. End
9. For j=1:N
10. [yer,$deger$]=min(J(:,j)) ve U(yer,j)=1.
11. End
12. For j=1:K
13. **indis**=**U**(j,:)==1, **Küme_verisi** = **X**(**indis**,:)
14. $\mathbf{c}(j,:) = ortalama(\mathbf{Küme_verisi})$
15. End
16. End

Algoritmada verilen toplama, çıkarma, kuvvet ve kıyas işlemleri eşdeğer kabul edilirse algoritma karmaşıklığı (AK) yaklaşık olarak [18];

$$AK \approx I \cdot \Bigg(\underbrace{K \cdot N \cdot d}_{\substack{\text{örneklerin kümelere} \\ \text{uzaklık hesabı}}} + \underbrace{K \cdot N}_{\text{üyelik hesabı}} + K \cdot N \cdot d \text{küme merkezlerinin yeni değer hesabı} \approx IKN2d+1, \quad (5)$$

hesaplanır. Sonuç olarak K-ortalamaların hesabı I adım sayısına, K küme sayısına, N örnek sayısına ve d öznitelik sayısına bağlı olduğu görülür. Büyük-O

gösterimine göre KO karmaşıklığı, eğer $I = K = N$ kabul edilirse, $O(N) = N^3$ olarak belirlenir [18].

2.2. Bulanık C-Ortalamalar Kümeleme (Fuzzy C-Means Clustering)

Bulanık C-ortalamalar (BCO), bulanık kümeleme tekniklerinden en iyi bilinen ve yaygın olarak kullanılan ötelemeli bir algoritmadır [19]. BCO algoritması amaç fonksiyonunu küçültmeyi hedefler. BCO, nesnelerin iki veya daha fazla kümeye ait olabilmesine izin verir. Bulanık mantık prensibi gereği her veri, kümelerin her birine [0-1] arasında değişen birer üyelik değeri ile aittir. Nesne hangi küme merkezine yakın ise o kümeye ait olma üyeliği diğer kümelere ait olma üyeliğinden daha büyük olacaktır. Amaç fonksiyonun belli bir değere yakınsamasıyla kümeleme işlemi tamamlanır [16].

2.3. K-Ortaylar Kümeleme (K-Medoids Clustering)

K-ortaylar (KM) kümelemenin temeli, verinin çeşitli yapısal özelliklerini temsil eden *K* tane temsilci örneği bulmaya dayanmaktadır [20]. Küme merkezleri; ortalama yerine küme içindeki merkeze en yakın örnekler olarak belirlenir [20]. Küme ortayları, veri seti içerisindeki *K* tane alt kümeyi temsil ederler. Ortay örnek, diğer örneklere olan ortalama uzaklığı minimum yapan kümenin en merkezi örneğidir. Bu yöntem, her bir örnek ve onun referans noktası arasındaki benzersizliklerin toplamını küçültmeyi amaçlamaktadır.

2.4. Spektral Kümeleme (Spectral Clustering)

Spektral kümeleme, görüntü özvektörlerine ve ikili yakınlık matrisinden türetilen NxN boyutundaki özdeğerlere dayanmaktadır. Bu özvektörler, piksellerin düşük boyutlu alt uzaylara ayrılmasını sağlamaktadır. Daha sonra KO gibi basit bir kümeleme metoduyla son bölümleme yapılır [21].

3. HIZLANDIRILMIŞ BÖLÜTLEME (ACCELERATED SEGMENTATION)

Resim bölütlemesinde kullanılan klasik kümeleme yöntemlerinin girişleri, resmin bütün pikselleridir. Bu nedenle, renkli ve büyük boyutlu resimlerin kümeleme süresi oldukça uzundur. Hızlı bölütleme yöntemindeki veriler, resim üzerinden *n*x*n* {*n*=1,2,3,4,5} boyutlarda ortalama maskesi kullanılarak yapılan konvolüsyon ile üretilir [1]. Küçük boyutlu ortalama maskesi kullanılmasının nedeni, blok içindeki dağılımın genellikle homojen olmasıdır [22,23]. Sonuç olarak blok içinde bütün pikseller birbirine benzemekte ve blok ortalama değeri bu bloğu temsil edebilir niteliktedir [1].

$\mathbf{I} \in \mathbb{N}^{\boldsymbol{MxNx3}}$ renkli bir görüntü ifadesi olsun. Görüntü, *n*x*n* bloklarına ayrıldığında oluşan satır ve sütundaki yaklaşık blok sayıları sırasıyla $\boldsymbol{m_R}$ **ve** $\boldsymbol{m_C}$ ile ifade edilebilir:

$$\boldsymbol{m_R} \cong \frac{\boldsymbol{M}}{\boldsymbol{n}},\ \boldsymbol{m_C} \cong \frac{\boldsymbol{N}}{\boldsymbol{n}}. \tag{6}$$

Her bir blok için $\boldsymbol{O(i,j,k)}$ özniteliği Eşitlik (7) ile hesaplanır.

$$\boldsymbol{O(i,j,k)} = \frac{\mathbf{1}}{\boldsymbol{n^2}} \sum_{\boldsymbol{\alpha=n(i-1)+1}}^{\boldsymbol{ni}} \sum_{\boldsymbol{\beta=n(j-1)+1}}^{\boldsymbol{nj}} \boldsymbol{I(\alpha,\beta,k)}, \tag{7}$$

$$\textbf{Burada} \begin{cases} \boldsymbol{i = 1,2,\dots,m_R} \\ \boldsymbol{j = 1,2,\dots,m_C} \\ \boldsymbol{k = 1,2,3} \end{cases}.$$

Bu açıklamaların sonucundaki hızlı bölütleme algoritması aşağıda verilmektedir.

1. Bölütlenecek görüntüyü, n blok boyutu değerini ve K küme sayısını belirle. m_R ve m_C değerlerini hesapla.
2. $O(i,j,k)$ ifadesini Eşitlik (7)'ye göre hesapla.
3. $\mathrm{O}_{(m_R m_C)\mathrm{x}3}$ matrisini elde ederek K için KO kümeleme yöntemini uygula.

Hızlı bölütleme algoritmasının karmaşıklığı hesaplanmak istenirse, her resimde bir kez yapılan bir alt-bloklama ve blokların ortalama hesabı vardır. Toplam yapılan ortalamalardaki işlem sayısı $m_R \cdot m_C \cdot n^2$'dir. Kümelemeye sokulan toplam örnek sayısı N adet yerine $N/(n^2)$olarak azaltıldığından yaklaşık algoritma karmaşıklığı (AK);

$$AK \approx m_R \cdot m_C \cdot n^2 + IKN(2d+1)/(n^2), \tag{8}$$

olarak hesaplanır [17].

$n \ll N$ olduğundan, yaklaşık $AK \approx IKN(2d+1)/(n^2)$ biçiminde düzenlenebilir. Eğer $I = K = N$ kabul edilirse, hızlı bölütleme algoritmasının büyük-O gösterimi $O(N) = (N^3/n^6)$ olarak hesaplanır [18].

4. DENEYSEL ÇALIŞMALAR (EXPERIMENTAL STUDIES)

Önerilen yöntemin değerlendirilebilmesi için çok bilinen ve kullanılan "Lena", "Baboon" resimleri ve 80 adet traverten taş blok resimleri üzerinde deneysel çalışmalar yapılmıştır. Deneysel çalışmalar, 2,66 GHz lik çift çekirdek ve 3GB RAM belleğe sahip bir bilgisayarda MATLAB yazılımı kullanılarak yapılmıştır.

Önerilen yöntemin zamandan elde ettiği kazancın yanı sıra, resim kalitesi de çalışmalarda değerlendirilmiştir. Ayrıca kümeleme yöntemleri arasındaki farklılıklar da ortaya konmuştur. Yöntemlerin kıyası için Ortalama Kare Hatasının Karekökü (OKHK) ve Doruk Sinyalin

Gürültüye Oranı (DSGO) ölçütleri kullanılmıştır. Bunlar sırasıyla Eşitlik (9) ve (10) ile ifade edilmektedir.

$$OKHK = \sqrt{\frac{1}{3MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{k=1}^{3}[I(i,j,k) - I'(i,j,k)]^2} \quad (9)$$

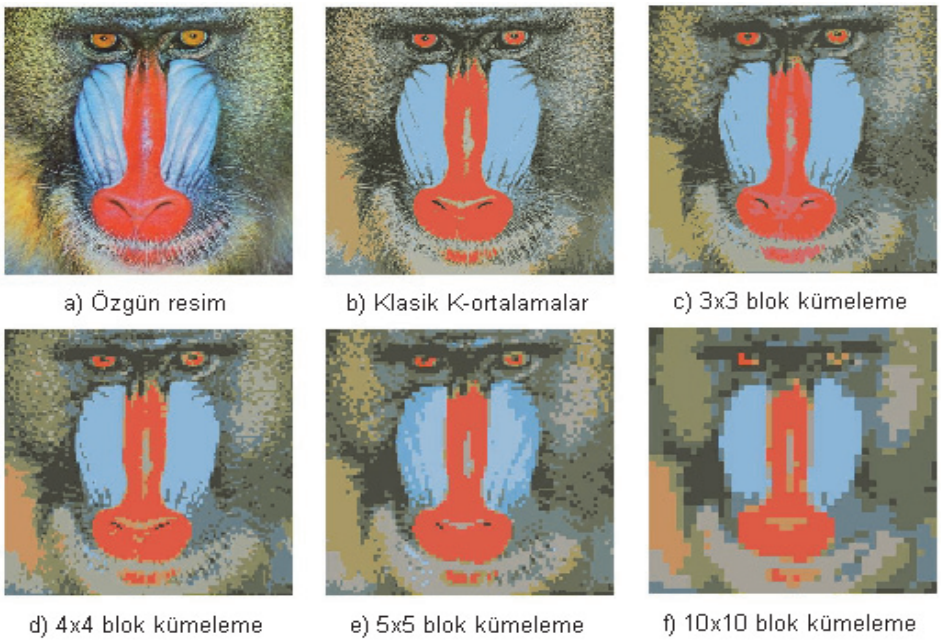
$$DSGO = 20\log_{10}\left(\frac{m_x}{OKHK}\right) \quad (10)$$

Eşitlik (9)'da $I(i,j,k)$ ve $I'(i,j,k)$ sırasıyla özgün ve bölütlenmiş resimleri, Eşitlik (10)'da m_x ise görüntü içindeki en büyük piksel değerini göstermektedir [23].

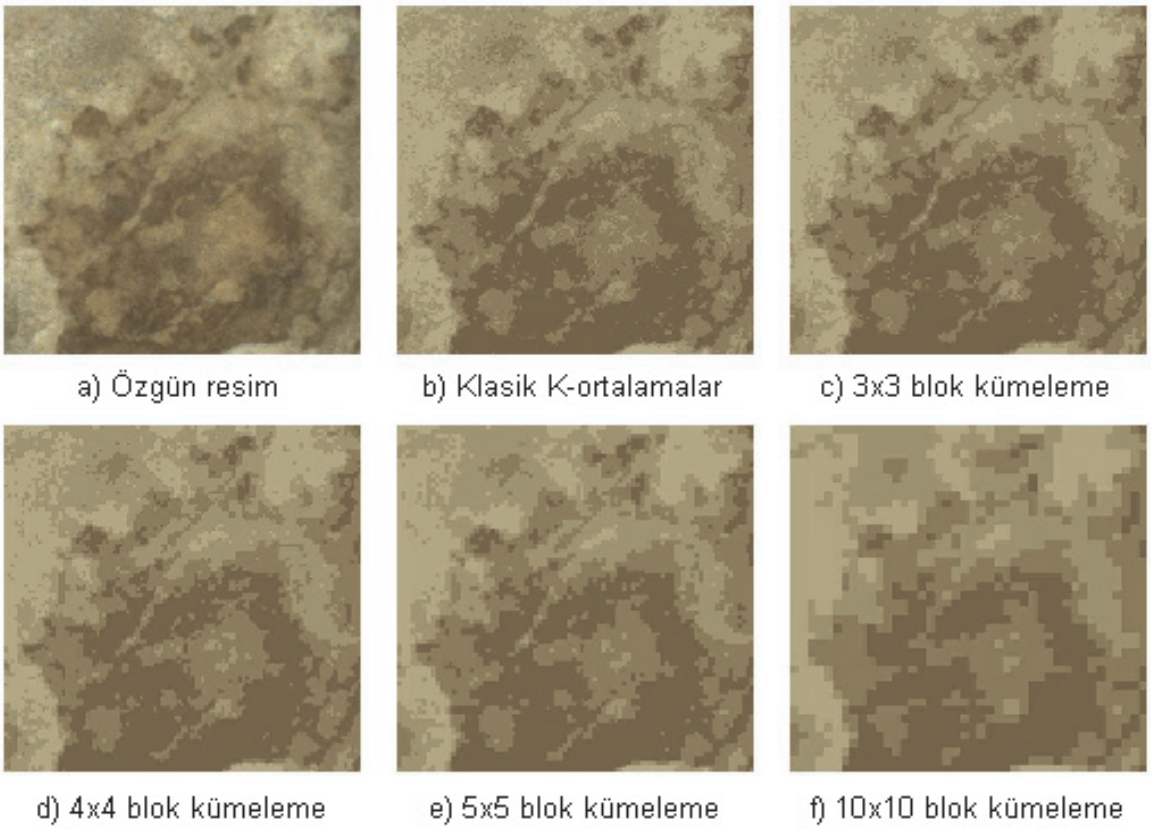
Deneysel çalışmada ilk önce en iyi blok boyutu araştırılmıştır. Bu nedenle, sırasıyla Lena ve Baboon resimlerinin farklı blok boyutları ve K=8 için kümeleme sonuçları Şekil 3 ve Şekil 4'de gösterilmektedir. Traverten örneğinin ise farklı blok boyutları ve K=4 için kümeleme sonuçları Şekil 5'te gösterilmektedir.



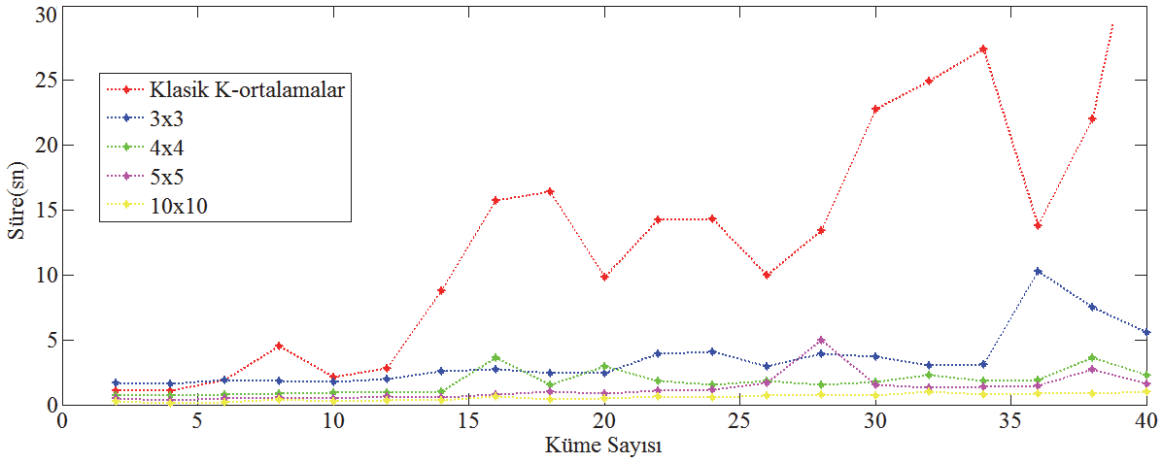
Şekil 3. Lena'nın farklı blok boyutlarıyla 8'e kümelenmesi (Clustering of Lena into 8 clusters by using different block sizes)



Şekil 4. Baboon'nun farklı bloklarla 8'e kümelenmesi (Clustering of Baboon into 8 clusters by using different block sizes)



Şekil 5. Traverten örneğinin farklı bloklarla 4'e kümelenmesi (Clustering of travertine into 4 clusters by using different block sizes)



Şekil 6. Lena'nın farklı blok boyutları ve KO ile elde edilen zaman-küme sonuçları (Time-cluster results of Lena by K-means clustering and different sub-block sizes)

Şekil 3-5 incelendiğinde; klasik kümelemeyle elde edilen resimlerle n=3 ve 4 ile yapılan kümeleme resimlerinin çok yakın olduğu ve özgün resimleri temsil edebildiği görülmektedir. Ayrıca blok boyutu büyüdükçe bölütlenen resmin görüntü kalitesinin de azaldığı gözle görülmektedir.

Farklı blok boyutları ve KO ile kümelenen Lena resmine ait zaman-küme sayısı sonuçları Şekil 6'da görülmektedir. Şekil 6'da her durum için elde edilen sonuçlar, üçer denemelerin ortalamasıdır.

Şekil 6 incelendiğinde, süre açısından 3x3, 4x4 ve 5x5 arasında düşük küme değerleri için belirgin bir fark oluşmamaktadır. Klasik KO ile alt-blok kullanımı ile elde edilen sonuçlara bakıldığında, alt-bloklara ayırmanın, küme sayısı arttıkça, zaman gereksinimi açısından, klasik KO'ya göre üstünlüğü ortaya çıkmaktadır. Ancak en uygun blok boyutunun belirlenmesi için DSGO ve OKHK değerlerine de bakılmalıdır.

Tablo 1. Lena'nın farklı yöntem ve küme sayılarıyla elde edilen sonuçları (Results of Lena by using different methods and cluster numbers)

Küme	Yöntem	Süre(sn)	OKHK	DSGO	Oran
4	KKO	1,06	14,57	25,12	1,83
	BCO	6,58	14,57	25,12	11,35
	KM	2,55	14,57	25,13	4,40
	S	5,08	14,69	25,07	8,67
	HKO	0,65	17,93	23,08	0,84
8	KKO	4,52	10,36	28,47	12,42
	BCO	19,19	10,53	28,42	51,79
	KM	3,63	10,51	28,02	9,68
	S	6,26	11,52	27,28	14,82
	HKO	0,86	15,10	24,75	1,41

Tablo 2. Lena ve Baboon resimlerinin KKO ve HKO ile kıyaslanması (The comparison results of KKO and HKO for Lena and Baboon)

Resim	Yöntem	Süre(sn)	OKHK	DSGO	Oran
Lena	KKO	1,05	14,56	24,86	1,79
	HKO	0,62	17,93	23,05	0,79
Baboon	KKO	1,76	25,42	20,02	1,38
	HKO	0,62	31,70	18,10	0,35

Tablo 3. Traverten seti sonuçları (Results of travertine set)

Küme	Yöntem	Süre(sn)	OKHK	DSGO	Oran
4	KKO	1,16	4,26	35,70	9,80
	BCO	8,22	4,26	35,69	68,81
	KM	0,87	4,35	35,50	7,13
	S	3,97	4,63	34,95	29,98
	HKO	0,67	5,31	33,75	4,29
8	KKO	3,15	2,92	38,90	41,92
	BCO	19,36	2,93	38,87	256,31
	KM	2,19	3,09	38,43	27,28
	S	5,89	3,99	36,20	53,46
	HKO	1,09	4,42	35,32	8,76

Sırasıyla Tablo 1 ve Tablo 2'de ise Lena ve Baboon resimleri için farklı küme değerleriyle elde edilen Klasik K-ortalamalar (KKO), Bulanık C-ortalamalar (BCO), K-ortaylar (KM), Spektral (S) ve Hızlandırılmış K-ortalamalar (HKO) yöntemlerinin OKHK, DSGO ve süre kıyaslamaları yapılmaktadır. Kıyas için önerilen ölçüt;

$$Oran = Süre \times DSGO/OKHK \quad (11)$$

Tablo 1'e göre, KKO ve BCO yöntemleri OKHK ve DSGO değerlerine göre benzer sonuçlar vermektedir. Ancak küme sayısı arttıkça KKO'nun daha iyi bir yöntem olduğu anlaşılmaktadır. Bu nedenle alt-bloklama sadece KKO yönteminde denenmiştir. Ayrıca yöntemlerin OKHK ve DSGO değerlerine bakıldığında, küme sayısı arttıkça bu değerlerin düştüğü görülmektedir. Sonuç olarak, 4x4 alt-bloklama en iyi sonucu vermektedir ve bundan sonraki deneylerde HKO için n=4 alınmaktadır. Tablo 2'de ise Lena ve Baboon resimlerinin K=4 için KKO ve HKO ile elde edilen sonuçları gösterilmektedir.

Tablo 2'ye göre, KKO ile HKO arasında çok fazla görüntü kalitesi farklılığı yoktur. Ancak işlem süresi akımından HKO oldukça üstündür. Tablo 3'de ise 80 adet traverten plaka resimlerinin farklı yöntemlerle bölütlenmesine ait sonuçlar verilmektedir. Tablodaki sonuçlar her yöntem ve her küme sayısı için on defa çalıştırılmış ve ortalama değerleri göstermektedir.

Her üç tablodan da görünen şudur ki, resim içindeki küme sayısı arttıkça DSGO ile gösterilen görüntü kalitesinin de arttığı açıktır. Kalitenin yanı sıra, küme sayısıyla beraber kümeleme süresi de artmaktadır. Özgün resim ile kümeleme sonucu elde edilen

filtrelenmiş resim arasındaki farkı gösteren OKHK değeri, küme sayısı arttıkça düşmektedir. OKHKnün tablolarda büyük çıkmasının nedeni, oldukça küçük küme sayısı ile çalışmaktan kaynaklanmaktadır. Örneğin K=4 değeri kullanıldığında, 256/4= 64 piksel aralığı tek bir küme ile temsil edilmekte ve OKHK değeri doğal olarak 32 hesaplanmaktadır. Eğer renk aralığı traverten resimlerinde olduğu gibi dar aralıkta seyrederse OKHK değeri küçük çıkar. Ancak, Baboon resminde olduğu gibi, renk aralığı çok geniş ise OKHK değeri yüksek çıkacaktır.

Bu çalışmada yapılan denemelerden bir diğeri ise kümelemeye alt-blokların renk ortalaması yerine renk standart sapmasının kullanılmasıdır. Şekil 7, Lena resminin K=4, 8 ve n=4 için HKO ile elde edilen bölütleme sonucunu göstermektedir. Lena resminin renk ortalamaları ve standart sapma ile bölütlenmesi sonuçları ise Tablo 4'te verilmektedir.

Tablo 4. Lena'nın renk ortalaması ve standart sapma değerleriyle bölütlenmesi (The segmentation of Lena with color mean and standard deviations.)

Küme Sayısı	Yöntem	OKHK
4	Ortalama	17,93
	Standart S.	32,43
8	Ortalama	15,10
	Standart S.	31,81

Blok renk ortalamalarının kullanıldığı Şekil 3 ve blok standart sapmalarının kullanıldığı Şekil 7 kıyaslandığında, Şekil 3'ün çok çok daha iyi sonuçlar verdiği açıktır.

Tablo 4'e göre blok renk ortalamalarının blok standart sapmalarına göre bariz üstünlüğü bulunmaktadır. Bu nedenle blok standart sapmalarının bölütlemede ortalamalar yerine kullanılması fikri çok anlam ifade etmemektedir.

En son olarak resimlere gürültü eklendiğinde bölütleme başarımlarının nasıl olduğuna bakmak gerekir. Eğer pikseller olduğu gibi kullanılırsa kümeleme yöntemlerinin bu durumdan oldukça etkileneceği açıktır. Ancak hızlı bölütlemenin gürültüye karşı duyarlılığını ölçmek gerekir. Bu nedenle Lena resmine 4x4 blok boyutu, 4 ve 8 küme sayısı ile kümeleme uygulandığında Şekil 8'deki görüntüler elde edilir. Özgün resme eklenen gürültü, "Tuz-Biber" gürültüsü olup yoğunluk değeri 0.05'tir.

Şekil 8'e göre, HKO yöntemi blok ortalaması kullandığından gürültüden oldukça az etkilenmektedir. Çünkü ortalama işlemi bir alçak geçiren filtre olduğundan yüksek frekanslar içeren gürültüyü süzmektedir. Buna karşın, KKO yöntemi gürültüden etkilenmekte ve elde edilen resimlerde gürültüler olduğu gibi korunmaktadır. Bu durum Tablo 5'te verilen OKHK değerleriyle de ispatlanmaktadır.



K=4 K=8

Şekil 7. Lena'nın 4x4 blok standart sapmaları ve HKO ile bölütlenmesi (Segmentation of Lena with 4x4 block standard deviations using AKM)



Şekil 8. Gürültülü Lena'nın HKO ve KKO ile bölütlenmesi (The segmentation of noised Lena using AKM and CKM)

Tablo 5. Gürültülü Lena'nın bölütlenme sonuçları (The segmentation results of noised Lena.)

Küme	Yöntem	Gürültüsüz OKHK	Gürültülü OKHK	Fark
4	KKO	14,57	25,17	10,60
	HKO	17,93	18,74	0,81
8	KKO	10,36	24,17	13,81
	HKO	15,10	16,04	0,94

5. SONUÇ (CONCLUSION)

Gerçek zamanlı dokusal görüntü analizlerinde elde edilen bölgesel özniteliklerin çıkarımı, nesneleri tanıma ve tasnif etme için önemlidir. Hızlı bir karar verme için öznitelik çıkarımı hızlı olmalı, gerekirse görüntü kalitesinden taviz verilebilmelidir. Bölgesel öznitelikleri çıkarmak için kullanılan kümeleme algoritmaları ne kadar hızlı olursa olsun, gerçek zamanlı uygulamalarda istenen sürelerde sonuç üretemezler.

Bu çalışmada farklı görüntüler üzerine çalışmalar yapılmış ve 4x4 blok boyutunun alt-blok ayrımı için en uygun değer olduğu tespit edilmiştir. Bu yöntemle algoritma karmaşıklığı N^3 değerinden N^3/n^6 değerine indirgenmiştir. Kümeleme hızındaki 2-4 kat kazanç, kümeleme algoritmalarının gerçek zamanlı uygulamalarda kullanabilirliğini sağlamıştır. Geliştirilen yöntemle, gerçek zamanlı çalışmalarda, traverten görüntülerinin bölütlemesi 1 saniyenin altına indirilmiştir.

Alt-blok ortalamaların, alt-blok standart sapmalarına göre çok daha iyi bölütleme sağladığı deneysel çalışmalarla gösterilmiştir. Ayrıca önerilen yöntem, resimdeki gürültülerden etkilenmemekte ve gürültüyü yumuşatabilmektedir.

Önerilen yöntem, bir ön işlem olduğu için herhangi bir kümeleme yöntemine rahatlıkla uyarlanabilir. Yöntemin hızlanmada getirdiği avantajla birlikte görüntü kalitesinde büyük bir kayıp oluşmamıştır.

TEŞEKKÜR (ACKNOWLEDGMENT)

Bu çalışma TÜBİTAK TEYDEB 1511-1120167 nolu proje tarafından desteklenmiştir. Desteklerinden dolayı Bilim, Sanayi ve Teknoloji Bakanlığı'na ve değerli yorum ve katkılarından dolayı hakem heyetine teşekkür ederiz.

SEMBOLLER (NOTATIONS)

J_i	: i. grubun amaç fonksiyonu
G_i	: i. gruba ait örnekler kümesi
x_k	: G_i'ye ait örnekler
c_i	: Küme merkezleri
U	: İkili üyelik matrisi
c_k	: i. grup dışındaki kümelerin merkezleri
$\|G_i\|$	: i. gruba ait eleman sayısı
m_n	: Görüntünün her renkteki en küçük piksel değeri
m_x	: Görüntünün her renkteki en büyük piksel değeri
K	: Küme sayısı
I	: Adım sayısı
AK	: Algoritma karmaşıklığı
N	: Örnek sayısı
d	: Öznitelik sayısı
I	: Renkli bir görüntü

I' : Bölütlenmiş görüntü
m_R : Satır bazında blok sayısı
m_C : Sütun bazında blok sayısı
O : Her bir blok için öznitelik
n : Blok boyutu değeri

KAYNAKLAR (REFERENCES)

1. Şişeci, M., **Traverten Plaka Taşlarda Sınıfların Kümeleme Yöntemleri ile Belirlenmesi**, Y. Lisans Tezi, Süleyman Demirel Ünv., Fen Bilimleri Ens., 2012.
2. Coleman, G.B. ve Andrews, H.C., "Image Segmentation By Clustering", **Proc. of IEEE**, Cilt 67, No 5, 773-785, 1979.
3. Umbaugh, S.E., Moss, R.H. ve Stoecker, W.V., "Automatic Color Segmentation of Images with Application to Detection of Variegated Coloring in Skin Tumors", **IEEE Eng. in Medicine and Biology Magazine**, Cilt 8, No 4, 43-50, 1989.
4. Vaisey, J. ve Gersho, A., "Image Compression with Variable Block Size Segmentation", **IEEE Trans. on Signal Processing**, Cilt 40, No 8, 2040-20,60, 1992.
5. Hu, Y.C. ve Chang, C.C., "Variable Rate Vector Quantization Scheme Based on Quadtree Segmentation", **IEEE Trans. on Consumer Electronics**, Cilt 45, No 2, 310-317, 1999.
6. Kim, C., "Content-Based Image Copy Detection", **Signal Processing: Image Communication**, Cilt 18, No 3, 169-184, 2003.
7. Tsai, V.J.D., "Automatic Shadow Detection and Radiometric Restoration on Digital Aerial Images", **In Proceedings of the IEEE International Geoscience and Remote Sensing Symposium**, Toulouse, France, Cilt 2, 732-733, 21-25 Temmuz 2003.
8. Shi, R., Feng, H., Chua, T.S. ve Lee, C.H., "An Adaptive Image Content Representation and Segmentation Approach to Automatic Image Annotation", **Third Int. Conf. on Image and Video Retrieval (CIVR)**, Dublin, Ireland, 545-554, 2004.
9. Wu, B.-F., Chiu, C.-C. ve Chen, Y.-L., "Algorithms for Compressing Compound Document Images with Large Text Background Overlap", **IEEE Vision, Image and Signal Processing**, Cilt 151, No 6, 453-459, 2004.
10. Bosch, A., Muñoz, X. ve Martí, R., "Which is the Best Way to Organize/Classify Images by Content?", **Image and Vision Computing,** Cilt 25, No 6, 778-791, 2007.
11. Ko, B., Seo, M. ve Nam, J., "Microscopic Cell Nuclei Segmentation Based on Adaptive Attention Window", **J. of Digital Imaging, Springer**, Cilt 22, No 3, 259-274, 2009.
12. Clausi, D.A., "K-means Iterative Fisher (KIF) Unsupervised Clustering Algorithm Applied to Image Texture Segmentation", **Pattern Recognition, Science Direct**, Cilt 35, No 9, 1959-1972, 2002.
13. Demirhan, A. ve Guler, I., "Image Segmentation Using Self-Organizing Maps and Gray Level Co-Occurrence Matrices", **Journal of the Faculty of Engineering and Architecture of Gazi University,** Cilt 25, No 2, 285-291, 2010.
14. Chuchra, R., Sood, S. ve Kaur, K., "Performance Analysis & Comparison b/w Enhanced K-Means & Orthogonal Partitioning (OC), based on proposed New Approach: "DRID" ", **Int. J. of Computer Science and Network Security (IJCSNS)**, Cilt 12, No 3, 61-63, 2012.
15. Xu, R. ve Wunsch, D., **Clustering**, Wiley-IEEE Press, 2008.
16. Jang, J.S.R., Sun, C.-T. ve Mizutani, E., **Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence**, Prentice Hall, 1997.
17. MacQueen, J.B., "Some Methods For Classification and Analysis of Multivariate Observations", **Proc. of the Fifth Berkeley Symp. on Mathematical Statistics and Probability**, Berkeley, USA, Cilt 1, 281–297, 1967.
18. Cormen, T.H., Leiserson, C.E., Rivest, D.L., Stein, C., **Introduction to Algorithms, 2nd Revised Edition**, MIT Press, Cambridge, 1180, 2001.
19. Dunn, J.C., "A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact, Well-Separated Clusters", **J. of Cybernetics**, Cilt 3, No 3, 32-57, 1973.
20. Kaufman, L. ve Rousseeuw, P., "Clustering by Means of Medoids", **Statistical Data Analysis Based on The L1-Norm and Related Methods, Edited by Y. Dodge,** North-Holland, 405-416, 1987.
21. Fowlkes, C., Belongie, S., Chung, F. ve Malik, J., "Spectral Grouping Using the Nystrom Method", **IEEE Trans. on Pattern Analysis and Machine Intelligence**, Cilt 26, No 2, 214-225, 2004.
22. Shi, J. ve Malik, J., "Normalized Cuts and Image Segmentation", **IEEE Trans. on Pattern Analysis and Machine Intelligence**, Cilt 22, No 8, 888-905, 2000.
23. Woods, J.W., **Multidimensional Signal, Image, and Video Processing and Coding**, Academic Press, Second Edition, 2011.