

KONTROLSÜZ SINIFLANDIRMADA DİFERANSİYEL GELİŞİM ALGORİTMASININ KULLANIMI

Ü.H. Atasever¹, C. Özkan¹, F. Sunar²

¹Erciyes Üniversitesi Mühendislik Fakültesi, Harita Mühendisliği Bölümü, Uzaktan Algılama Anabilim Dalı, Kayseri, uhatasever, cozkan@erciyes.edu.tr

²İstanbul Teknik Üniversitesi İnşaat Fakültesi, Geomatik Mühendisliği Bölümü, Maslak, İstanbul, fsunar@itu.edu.tr

ÖZET

Fen bilimlerinde karşılaşılan birçok problem doğrusal veya doğrusal olmayan bir optimizasyon problemi ile temsil edilir. Doğrusal olan optimizasyon problemlerinin çözümünde klasik deterministik yöntemler kullanılmasına rağmen, doğrusal olmayan problemlerin çözümünde artan değişken sayısı, kullanılan veri tipi gibi birtakım zorluklarla karşılaşmaktadır. Bu bağlamda, doğrusal olmayan optimizasyon problemlerinin çözümü için günümüzde Diferansiyel Gelişim Algoritması, Genetik Algoritma, Yapay Arı Kolonisi gibi birçok sezgisel algoritma geliştirilmiştir. Bu çalışmada Aster, Landsat ve Ikonos verileri, kullanımı son yıllarda giderek artan yapay zeka optimizasyon algoritmalarından popülasyon tabanlı sezgisel bir algoritma olan Diferansiyel Gelişim Algoritması (DEA) ile sınıflandırılmıştır. Yöntemin uydu görüntü verilerinin sınıflandırılmasındaki etkinliğinin değerlendirilmesi için klasik kontrolsüz sınıflandırma yöntemlerinden K-means ve Fuzzy C-means yöntemleriyle karşılaştırma yapılmıştır.

Anahtar Sözcükler: Optimizasyon, Sınıflandırma, Diferansiyel Gelişim Algoritması, K-means, Fuzzy C-means

THE USE OF DIFFERENTIAL EVOLUTION ALGORITHM IN UNSUPERVISED CLASSIFICATION

ABSTRACT

Many of the problems encountered in all disciplines represent linear or non-linear optimization problems. Although classical deterministic methods are being used for resolving linear optimization problems, however, there are some difficulties such as increase number of variables, data types etc. for the solution of non-linear problems. Therefore, many intuitional algorithms like Artificial Bee Colony Algorithm, Genetic Algorithm and Differential Evolution Algorithm were developed for the solution of non-linear optimization problems. In this study, Aster, Landsat and Ikonos image data were classified by using the Differential Evolution Algorithm, one of the most commonly used artificial intelligence optimization algorithms, which is a population based intuitional algorithm. Its effectiveness in image classification was evaluated by comparing it with K-means and Fuzzy C-means methods which are classical unsupervised methods.

Keywords: Optimization, Classification, Differential Evolution Algorithm, K-Means, Fuzzy C-Means

1. GİRİŞ

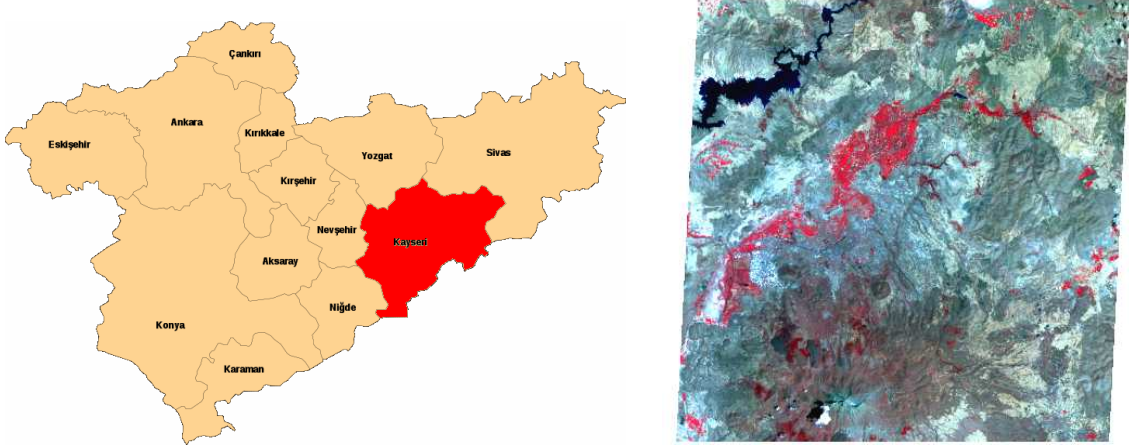
Modern uydu teknolojilerinin günümüzdeki en önemli kullanım alanlarından biri uzaktan algılama ile arazi örtüsü/kullanımının haritalanmasıdır. Uydu görüntü verilerinden tematik mekânsal bilginin çıkarılmasında en önemli aşama sınıflandırma aşamasıdır. Sınıflandırma işlemi, bir görüntüdeki benzer örüntülerin komşuluk ilişkilerini açıklayan matematiksel bağıntılar yardımıyla gruplandırması olarak tanımlanabilir.

Sınıflandırma işleminde kontrollü ve kontrolsüz olmak üzere iki temel yaklaşım vardır. Kontrollü sınıflandırmada sınıfları temsil eden görüntüye ait pikseller kullanılarak ayırt edici eğitim ve karar fonksiyonları hesaplanır. Kontrollü sınıflandırma uygulamalarında En Kısa Uzaklık, Maksimum Olabilirlik, Mahalanobis Uzaklığı gibi klasik yöntemlerin yanında son yıllarda Destek Vektör Makineleri, Karar Ağaçları ve Yapay Sinir Ağları sıklıkla kullanılan diğer yöntemlerdir. Kontrolsüz sınıflandırmada ise, sınıfları temsil eden eğitim örnekleri bulunmamakta, bunun yerine spektral parlaklık değerlerine göre kümeleme yapılmaktadır. Uzaktan algılamada kontrolsüz sınıflandırma uygulamalarında en çok kullanılan yöntemler K-means, Isodata ve Fuzzy C-Means'dır. Günümüzde ise, bu problemin çözümü için yapay zeka optimizasyon algoritmaları kullanılmaya başlanmıştır. Literatürde, Genetik Algoritma (Mandyopadhyay and Maulik, 2002; Chiou and Lan, 2001; Garai and Chaudhuri, 2004); Diferansiyel Gelişim Algoritması (Das and Konar, 2009; Maulik and Saha, 2009; Paterlini and Krink, 2006), Yapay Arı Kolonisi (Karaboga and Ozturk, 2010; Zhang and Ouyang, 2010). Genel olarak problem, küme merkezleri ile piksellerin uzaklıklarının çözümü için oluşturulan

fonksiyonun minimizasyonu olarak tanımlanır ve optimizasyon probleminin çözümü, yani küme merkezlerinin bu yönde bulunması amaçlanır.

2. ÇALIŞMA ALANI VE VERİ

Kayseri, İç Anadolu'nun güney bölümü ile Toros Dağlarının birbirine yaklaştığı bir yerde Orta Kızılırmak bölümünde yer alır. Yaklaşık olarak yüzölçümü 16917 km² olan il 35° 45' ile 38° 18' kuzey enlemleri ve 34° 56' ile 36° 58'dakika doğu boylamları arasında bulunmaktadır. Doğu ve kuzeydoğusu Sivas, kuzeyi Yozgat, batısı Nevşehir, güneybatısı Niğde, güneyi ise Adana ve Kahramanmaraş illeri ile çevrilidir (Şekil 1) (URL-1). Bu çalışmada kullanılan uydu görüntüleri Erciyes Üniversitesi Bilimsel Araştırmalar Birimi'nden "Kentsel Gelişim Analizi Örneği: Melikgazi Belediyesi" projesi için temin edilmiş 2006 yılına ait 15m çözünürlüklü çok kanallı Aster uydu görüntüsü, 2002 yılı Kayseri iline ait Landsat görüntüsü ve 2008 yılına ait Ikonos görüntüsüdür.



Şekil 1. Çalışma alanının haritası ve Aster uydu görüntüsü.

2.1. K-Means Algoritması

Uydu görüntülerinin kontrolsüz sınıflandırılmasında kullanılan en eski algoritmalarından biri olan K-means, 1967 yılında J.B. MacQueen tarafından geliştirilmiştir. Bu algortmada sınıflandırılacak her bir eleman, kendisine en yakın küme merkezine atanmaktadır. K-means Algoritması sınıflandırılacak her bir elemanın yalnızca bir kümeye atanmasına izin verir ve temelde merkez noktanın kümenin tamamını temsil etmesi esasına dayanır (Han and Kamber, 2006). Piksellerin küme merkez noktalarına olan uzaklıkları,

$$\sum_{i=1}^k \sum_{j=1}^n dist^2(m_i, x) \quad (1)$$

eşitliği ile hesaplanmaktadır. Burada;

m_i : Küme merkezlerini,

x : Her bir pikseli,

k : Sınıf sayısını,

n : Piksel sayısını tanımlamaktadır.

Algoritma adımları:

- Sınıf sayısı seçimi (k)
- k sayıdaki sınıfların rastgele üretilmesi ve sınıf merkezlerinin hesaplanması yada rastgele k noktalarının sınıf merkezleri olarak üretilmesi,
- Her bir elemanın en yakın sınıf merkezine atanması,
- Yeni sınıf merkezlerinin tekrar hesaplanması,
- Önceki iki adımın yakınsama kriterleri sağlanana kadar tekrarlanması.

Bu algoritmanın en önemli avantajı, basit ve büyük veri setlerine hızlı biçimde uygulanabilmesidir. Dezavantajı ise, kümeleme sonuçlarının temelini oluşturan başlangıç değerlerinin rastgele üretilmesinden dolayı kodun her koşuturulmasında aynı sonuçları sağlamamasıdır (URL-2).

2.2. Bulanık C-Means Algoritması

Bulanık C-means (FCM) Algoritması 1973 yılında Dunn tarafından ortaya atılan, şu anki hali 1981 yılında Bezdek tarafından üretilmiş olan ve uzaktan algılama alanında en çok kullanılan kontrolsüz sınıflandırma yöntemlerinden biridir. Bulanık kümelemede, bulanık mantıkta olduğu gibi, sınıflandırılacak her bir piksel tamamen bir sınıfa ait olmayıp, tüm sınıflara aitlik dereceleri kadar üye olmaktadır. Kural olarak, her bir elemanın, tüm sınıflara olan üyelik değerlerinin toplamı 1 olmalıdır. Her piksel, üyelik derecesinin en büyük olduğu sınıfa atanır. Proses belirlenen minimum ilerleme değerine yakınsaması durumunda tamamlanır (Işık ve Çamurcu, 2007).

Algoritma aşağıda verilen fonksiyonun minimizasyonunu temel alarak çalışmaktadır (Höppner et al., 2000).

$$J_m = \sum_{i=1}^n \sum_{j=1}^c u_{ij}^m x_i - c_j^2, \quad 1 \leq m < \infty \quad (2)$$

Bir sonraki adımda, u üyelik matrisi rastgele seçilerek, 3 nolu eşitlik ile sınıf merkezleri hesaplanır (Höppner et al., 2000).

$$c_j = \frac{\sum_{i=1}^n u_{ij}^m x_i}{\sum_{i=1}^n u_{ij}^m} \quad (3)$$

(3) nolu eşitlik ile hesaplanan küme merkezlerine göre, u matrisi (4) nolu eşitlik kullanılarak tekrar hesaplanır. Eski u ve güncel u matrisi arasındaki fark ϵ 'dan küçük olana kadar işlem devam eder (Moertini, 2002).

$$u_{ij} = \frac{1}{\left(\sum_{k=1}^c \frac{x_i - c_i}{x_i - c_k} \right)^{2/m-1}} \quad (4)$$

2.3. Diferansiyel Gelişim Algoritması

Diferansiyel gelişim algoritması, Price ve Stone tarafından geliştirilen ve özellikle sürekli verilerin kullanıldığı problemlerde etkin sonuçlar üretebilen, popülasyon tabanlı sezgisel bir optimizasyon tekniğidir (Storn and Price, 1995; Karaboga, 2004).

Yapı itibari ile genetik algoritmaya dayandığından, genetik algoritmadaki çaprazlama, mutasyon ve seçim operatörleri diferansiyel gelişim algoritmasında da kullanılmaktadır. Diferansiyel Gelişim Algoritması'nın parametreleri,

- Np : Popülasyon büyüklüğü
- D : Parametre sayısı (gen sayısı)
- Cr : Çaprazlama oranı
- G : Jenerasyon sayısı (adım sayısı) ve
- F : Adım büyüklüğüdür.

Algoritmanın temel amacı, problem kısıtlarını sağlayan en iyi çözümü bulmaktır. Bu algoritma genellikle sürekli değişkenlerle beraber kullanılmasına rağmen, kesikli değişkenler ile çalışanları da bulunmaktadır (Lin and Wang, 2004) Algoritmanın temel adımları aşağıda verilmektedir (Karaboga, 2004):

1. Adım: D , G , $NP(>3)$, F ve Cr değerleri ve parametre sınırları olan x^{min} , x^{max} değerlerinin atanması.

2. Adım: Başlangıç popülasyonunun üretilmesi.

$$\forall m \leq NP \wedge \forall n \leq D : x_{n,m,G=0} = x_n^{min} + rand_n [0,1] * (x_n^{max} - x_n^{min}) \quad (5)$$

$$m = (1, 2, 3, \dots, NP), \quad n = (1, 2, \dots, D), \quad G = 0, \quad rand_n \in [0, 1] \quad (6)$$

3. Adım: Mutasyon ve Rekombinasyon

$$s_1, s_2, s_3 \in \{1, 2, 3, \dots, NP\}, \quad s_1 \neq s_2 \neq s_3 \neq m \quad (7)$$

$$n_{rand} \in \{1, 2, 3, \dots, D\}$$

$$\forall n \leq D, v_{n,m,G+1} = \begin{cases} x_{n,s3,G} + F'(x_{n,s1,G} - x_{n,s2,G}) \\ \text{eğer}(rand_n [0,1] < Cr \vee n = n_{rand}) \\ \text{farklı durumlarda} \\ x_{n,m,G} \end{cases} \quad (8)$$

4. Adım: Seleksiyon

$$\bar{x}_{m,G+1} = \begin{cases} \bar{v}_{m,G+1} \text{ eğer } f(\bar{v}_{m,G+1}) \leq f(\bar{x}_{m,G}) \\ \bar{x}_{m,G} \text{ farklı durumlarda} \end{cases} \quad (9)$$

2.3.1. Başlangıç Popülasyonu

Diferansiyel gelişim algoritmasında (DGA) sabit büyüklükte popülasyon kullanılır (Karaboga, 2004). Popülasyonun üretilmesi için parametrelerin alt ve üst sınırları belirlenmiş olmalıdır. Başlangıç sınırları belirlendiğinde bir rastgele sayı üretici, belirlenen aralık içinde, her vektörün her parametresine bir değer atar.

$$x_{j,i,0} = rand_j(0,1) \cdot (b_{j,U} - b_{j,L}) + b_{j,L} \quad (10)$$

Buradaki rastgele sayı üretici, yani $rand_j(0,1)$, (0,1) aralığında uniform sayılar üretir. Denklemdaki b_{ij} ve b_L alt ve üst sınırları gösteren D boyutlu başlangıç vektörleridir.

2.3.2. Mutasyon

Mutasyon işlemi, var olan bir amaç vektörüne bir ya da birden fazla parametreleri üzerine, ölçeklenmiş, rastgele örneklenmiş bir artışın ilave edilmesi anlamına gelir. Algoritmada, mutasyon işlemine tabi tutulacak kromozom (amaç vektörü) dışında üç kromozom (rastgele belirlenmiş değer) seçilir (r_1, r_2, r_3), ilk ikisinin farkı alınır ve adım büyüklüğü (F) ile çarpılarak ilk kromozoma eklenir. (11) nolu eşitlikte rastgele seçilen üç vektörün, bir mutant vektör oluşturmak için nasıl kullanıldığı gösterilmektedir (Storn and Price, 1995; Karaboga, 2004).

$$v_{i,g} = x_{r0,g} + F \cdot (x_{r1,g} - x_{r2,g}) \quad (11)$$

2.3.3. Rekombinasyon

Rekombinasyon ya da diğer adıyla çaprazlama işlemi mevcut vektör parametrelerinden, yeni vektörler oluşturarak araştırmanın başarılı olmasına yardımcı olan tamamlayıcı bir işlemdir.

$$u_{i,g} = u_{j,i,g} = \begin{cases} v_{j,i,g} \text{ eğer } (rand_j(0,1) \leq Cr \text{ ya da } j = j_{rand}) \\ x_{j,i,g} \text{ diğer durumlarda} \end{cases} \quad (12)$$

Burada Cr , çaprazlama oranı olarak adlandırılan, kullanıcı tarafından tanımlanan ve işlem boyunca sabit tutulan kontrol parametresidir. Araştırma eğer bir binom uygulama ise, rastgele üretilen bir sayının Cr 'den küçük olması durumlarında elemanlar mutasyona uğramış çözümden değilse ebeveynden alınır. Uygulama üstel ise bu işlemin tersi uygulanır. Durum ne olursa olsun, algoritma da üretilen vektörün en az bir parametresinin ebeveynden farklı olması sağlanır (Karaboga, 2004).

2.3.4. Seleksiyon

Bu adımda, yeni üretilen vektörlerin hangi durumlarda popülasyona dahil edilebileceği belirlenir. Strateji gereği, çocuk-ebeveyn popülasyonunda en iyi performans gösteren üyelerin gelişimine izin verilir. Diferansiyel gelişim algoritmasının seleksiyon işleminde, yeni üretilen vektör, bir önceki

vektör ile aynı gelişme seviyesinde ya da ondan daha gelişmiş değil ise, ebeveyn vektör popülasyonda kalmaya devam etmektedir.

$$x_{i,g+1} = \begin{cases} u_{i,g} & \text{eğer } f(u_{i,g}) \leq f(x_{ig}) \\ x_{ig} & \text{diğer durumlarda} \end{cases} \quad (13)$$

3. DİFERANSİYEL GELİŞİM ALGORİTMASI İLE GÖRÜNTÜ SINIFLANDIRMA

Bu çalışmada, Diferansiyel gelişim algoritmasının kolay kullanım, esneklik, verimlilik vb. özellikleri ile görüntü sınıflandırma problemi irdelenmeye çalışılmıştır. Çalışmada diferansiyel gelişim algoritmasının tercih edilmesinin nedeni, en başta tanımlanan algoritma üst ve alt sınırları (x^{min}, x^{max}) arasındaki arama uzayında ki en iyi çözüm olan global optimuma yakınsama hızının çok yüksek olmasıdır. Görüntü sınıflandırmada, diferansiyel gelişim algoritmasının kullanımı için tasarlanan algoritma şu şekildedir:

- 1. Adım:** Sınıf merkezleri için başlangıç popülasyonu üretilir,
- 2. Adım:** Görüntüdeki pikseller, kendilerine en yakın sınıfa atanır,
- 3. Adım:** Tüm bireylerin kalite değerlerinin bulunması için, sınıf merkezleri ile sınıflara ait pikseller arasındaki uzaklık bulunur,
- 4. Adım:** Tüm bireyler için, popülasyondan rastgele 3 birey seçilir ve bu bireylerden yeni bir birey üretilir,
- 5. Adım:** Önceki ve yeni üretilen bireylerin kalite ölçütleri kıyaslanır, kaliteli olan birey popülasyona dahil edilir,
- 6. Adım:** En başta kullanıcı tarafından tanımlanan jenerasyon sayısına ulaşıldıysa ya da hata belirlenen değerin altına düşmüşse işlem tamamlanır aksi taktirde 2. Adıma dönülür.
- 7. Adım:** Oluşturulan kümelerle hiçbir piksel atanmaması halinde optimizasyon fonksiyonunun değeri kullanıcı tarafından belirlenen maksimum değere eşitlenir ve çözüm doğruluğu bir önceki jenerasyona göre artmadığı için algoritmanın 1. Adıma tekrar dönmesi sağlanır (Maksimum değer $1e+20$ olarak kabul edilmiştir).

Problemin çözümü için kullanılan fonksiyonda amaç, sınıf merkezleri ile sınıf elemanları arasındaki uzaklığın minimum olmasını sağlamaktır.

x_i : i . piksel

x_j : j . sınıfın merkezi

C_j : j . sınıfta bulunan pikseller

$$C_j = (x_i | \min(x_i - x_j)) \quad (14)$$

$$\operatorname{argmin} \sum c_j - x_j \rightarrow \min \quad (15)$$

4. PERFORMANS DEĞERLENDİRMESİ

Literatürde kümeleme algoritmalarının performans değerlendirilmesi için genelde mse, Kappa katsayısı değerleri kullanılmaktadır. Bu çalışmada performans kriteri olarak niceleme hatası (quantization error) kullanılmıştır (Törmä, 1993). Niceleme hatası, kümelerdeki her bir pikselin küme merkezlerine olan uzaklıkların toplanıp, kümelerdeki piksel sayılarına oranının, sınıf sayısına bölünmesiyle elde edilir ve aşağıdaki eşitlik ile tanımlanır:

$$\frac{\sum_{s=1}^S [\sum_{p \in C_s} d(p, m_s)] / n_s}{S} \quad (16)$$

Burada;

p : Kümedeki her bir piksel

m_s : Küme merkezi

n_s : Kümedeki piksel sayısı

d : Metrik

S : Sınıf sayısını tanımlamaktadır.

5. UYGULAMA

Önerilen DGA temelli yöntemi test etmek için Aster, Landsat ve Ikonos uydularına ait üç farklı görüntü kullanılmıştır. Test görüntülerinin değerlendirilebilmesi için kullanılan üç yöntemde 1000 iterasyon yapılmıştır. Önerilen yöntemin tutarlı olup olmadığının belirlenmesi için her görüntü 30'ar kez sınıflandırma işlemine tabi tutulmuştur. Landsat görüntüsü 7, Ikonos görüntüsü 8, Aster görüntüsü 5 sınıf oluşturulacak şekilde sınıflandırma işlemi yapılmış ve kullanılan parametreler Tablo 1'de verilmiştir.

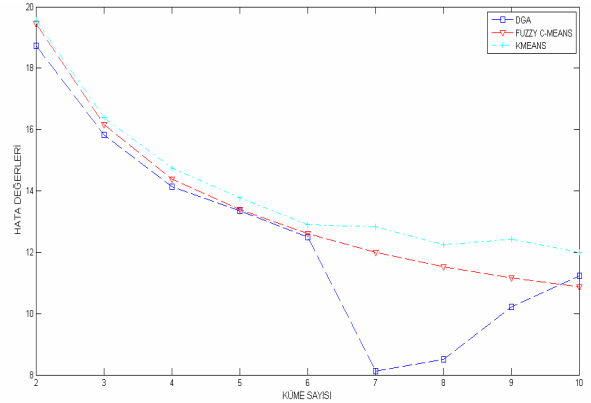
Tablo 1. Sınıflandırma işleminde kullanılan parametreler.

Görüntü	Boyut	Mekansal Çözünürlük	Spektral Bant Sayısı	İterasyon/ Jenerasyon	NP*	CR*	F*	Sınıf Sayısı
ASTER	185 x 216	30 m	6	1000	15	0.4	0.4	5
LANDSAT	276 x 408	90 m	3	1000	15	0.4	0.4	7
IKONOS	272x340	1 m	3	1000	15	0.4	0.4	8

Yapılan testler sonucunda elde edilen hata değerleri Tablo 2, Tablo 3 ve Tablo 4'te verilmektedir.

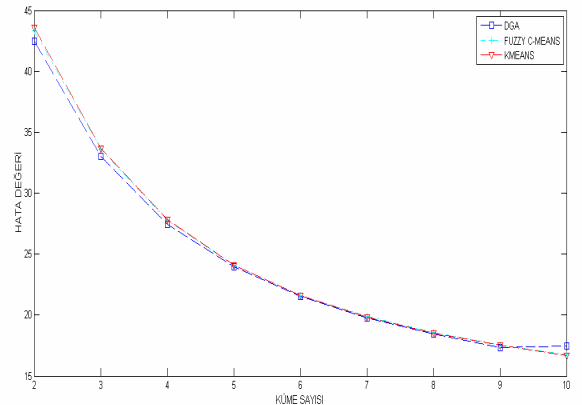
Tablo 2. ASTER görüntüsüne ait hata değerleri.

Küme Sayısı	DGA	FUZZY C-MEANS	KMEANS
2	18.7267	19.4461	19.5581
3	15.8241	16.1627	16.3967
4	14.1397	14.3757	14.7478
5	13.3379	13.3907	13.7641
6	12.4804	12.5985	12.8940
7	8.1199	11.9864	12.8386
8	8.5163	11.5260	12.2458
9	10.2176	11.1676	12.4321
10	11.2314	10.8705	12.0008



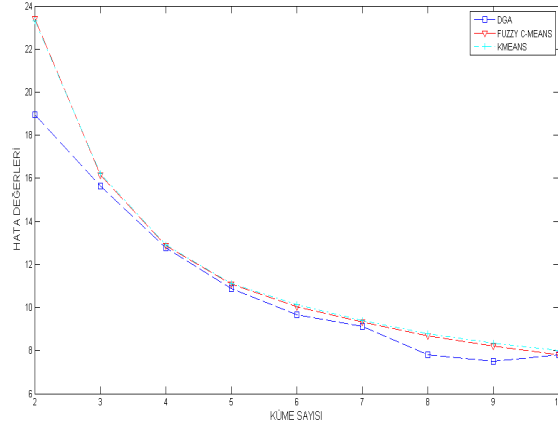
Tablo 3. LANDSAT görüntüsüne ait hata değerleri.

Küme Sayısı	DGA	FUZZY C-MEANS	KMEANS
2	42.4827	43.5583	43.3410
3	33.0229	33.6983	33.5949
4	27.4534	27.8429	27.7494
5	23.9779	24.1066	24.0796
6	21.5106	21.5700	21.5927
7	19.7760	19.8171	19.8788
8	18.4320	18.4913	18.5872
9	17.3091	17.5019	17.5189
10	17.4420	16.7027	16.7671



Tablo 4. IKONOS görüntüsüne ait hata değerleri.

Küme Sayısı	DGA	FUZZY C-MEANS	KMEANS
2	18.9609	23.3960	23.3059
3	15.6393	16.1603	16.2223
4	12.7714	12.8675	12.9069
5	10.8905	11.0667	11.1285
6	9.6717	10.0278	10.1477
7	9.1365	9.3145	9.3960
8	7.8204	8.6691	8.7740
9	7.4923	8.2131	8.3624
10	7.8050	7.8187	8.0170



Tablo 2, Tablo 3 ve Tablo 4'te görüldüğü gibi önerilen metot, genel olarak tüm veri setleri için Fuzzy C-Means ve K-means yöntemlerine göre daha başarılı sonuçlar vermiştir. Önerilen yöntem; Aster ve Landsat verilerinde, 10 sınıf için yapılan işlemler dışındaki tüm testlerde daha düşük hata değerleri üretmiştir.

6. SONUÇ

Bu çalışmada çok kanallı uydu görüntülerinin arazi örtüsü/kullanımı haritalamasında sıklıkla kullanılan Kmeans ve Fuzzy C-means algoritmaları, Diferansiyel Gelişim Algoritması tabanlı sınıflandırma yöntemi ile karşılaştırılmıştır. Yapılan testlerde önerilen yöntemin, çok büyük farklılıklar olmamasına karşın; sınıf sayısının 10 olduğu testler dışında klasik yöntemlerden daha iyi sonuçlar verdiği gözlenmiştir. Diferansiyel Gelişim Algoritması'nın bir kontrolsüz sınıflandırma yöntemi olarak kullanımında karşılaşılan problemler, algoritma parametrelerinin kestirimi ve yöntemin çok fazla kaynak tüketmesi; dolayısıyla diğer klasik yöntemlere göre daha uzun sürmesidir. Hata oranları, değişen veri setlerine göre çok büyük farklılıklar gösterebilmektedir. Kullanılan veri setine göre, algoritmada alt ve üst sınırların daraltılması ve kullanılan parametrelerin (Np , Cr ve F) uygun seçilmesi çıktı sonuçlarının daha doğruluklu elde edilmesinde etkin olmaktadır.

KAYNAKLAR

- Bandyopadhyay, S. and Maulik, U.**, 2002, Genetic Clustering for Automatic Evolution of Clusters and Application to Image Classification, *Pattern Recognition* 35(6): 1197-1208.
- Chiou, Y.C. and Lan, L.W.**, 2001, Genetic Clustering Algorithms, *European Journal of Operational Research* 135(2): 413-427.
- Das, S. and Konar, A.**, 2009, Automatic Image Pixel Clustering With an Improved Differential Evolution, *Applied Soft Computing* 9(1): 226-236.
- Garai, G. and Chaudhuri, B.B.**, 2004, A Novel Genetic Algorithm for Automatic Clustering. *Pattern Recognition Letters* 25(2): 173-187.
- Han, J., and Kamber, M.**, 2006, *Data Mining Concepts and Techniques*, (San Francisco: Morgan Kaufmann Publishers).
- Höppner, F., Klawonn, F., Kruse, R., and Runkler, T.**, 2000, *Fuzzy Cluster Analysis*, (Chichester: John Wiley&Sons)
- Işık, M. and Çamurcu, A.,Y.** 2007, K-Means, ve Bulanık C-Means Algoritmalarının Uygulamalı Olarak Performanslarının Tespiti, *İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi*, Yıl: 6, Sayı:11, s. 31-45.
- Karaboğa, D.**, 2004, *Yapay Zeka Optimizasyon Algoritmaları*, 38, 169-191, (İstanbul: Koza Yayın)
- Karaboga, D. and Ozturk, C.**, A Novel Clustering Approach: Artificial Bee Colony (ABC) algorithm. *Applied Soft Computing* In Press, Corrected Proof.
- Lin, Y.C., and Wang, F.S.**, 2004, A Mixed-Coding Scheme of Evolutionary Algorithms to Solve Mixed-Integer Nonlinear Programming Problems. *Computers and Mathematics with Applications*, 47, 1295-1307.
- Maulik, U. and Saha, I.**, 2009, Modified Differential Evolution Based Fuzzy Clustering for Pixel Classification in Remote Sensing Imagery. *Pattern Recognition* 42(9): 2135-2149.
- Paterlini, S. and Krink, T.**, 2006, Differential Evolution and Particle Swarm Optimisation in Partitional Clustering. *Computational Statistics & Data Analysis*, 50(5): 1220-1247.

- Storn, R., and Price, K.,** 1995, Differential Evolution: A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces, in International Computer Science Institute. Berkeley.
- Törmä, M.,** 1993, Comparison Between Three Different Clustering Algorithms. Photogrammetric Journal of Finland.13(2), 85-94.
- Zhang, C., Ouyang D and Ning J.,** 2010, An Artificial Bee Colony Approach for Clustering. Expert Systems with Applications, 37(7): 4761-4767.
- URL-1** <http://www.kayseri.gov.tr>
- URL-2** http://en.wikipedia.org/wiki/Cluster_analysis