


Coronavirüs Sürü Bağışıklığı Algoritması ile Otsu Tabanlı Optimal Çok Düzeyli Görüntü Eşiği

Araştırma Makalesi/Research Article

 İsmail KOÇ

Yazılım Mühendisliği Bölümü, Konya Teknik Üniversitesi, Konya, Türkiye

ismailkoc@ktun.edu.tr

(Geliş/Received:09.09.2022; Kabul/Accepted:07.11.2022)

DOI: 10.17671/gazibtd.1172909

Özet— Eşik seçimi, görüntü bölütlemede önemli bir rol oynamaktadır. Eşik seçimiyle ilgili en faydalı yöntemler olarak minimum hata yöntemi, iteratif yöntem, entropi yöntemi ve Otsu yöntemi bilinmektedir. Bu çalışmada eşikleme yöntemi olarak Otsu tekniği kullanılmaktadır. Eşik sayısının (K) artmasına bağlı olarak problemin karmaşıklık düzeyi üstel olarak artacağı için matematiksel yöntemler yerine sürü zekâsı algoritması kullanılması daha uygun görülmektedir. Bundan dolayı, bu çalışmada sürü zekâsı algoritması olarak da son yıllarda literatüre kazandırılmış olan Coronavirüs sürü bağışıklığı algoritması (CHIO) kullanılmaktadır. Deneysel çalışmalarda test verisi olarak altı farklı görüntü kullanılmaktadır. K değeri bu çalışmada 2, 3, 4 ve 5 olarak belirlenmektedir. Bu veri seti kullanılarak CHIO algoritması ile literatürde yer alan diferansiyel evrim (differential evolution: DE), gri kurt (gray wolf optimizer: GWO), parçacık sürü (particle swarm optimization: PSO) algoritmaları gibi başarılı algoritmalarla eşit koşullarda kıyaslanmaktadır. Elde edilen sonuçlara göre, CHIO algoritması kullanılarak 6 test verisi üzerinde yapılan çalışmalarda K=2 olduğunda verilerin %100, K=3 ve 4 iken %83 ve son olarak K=5 iken %50'sinde en iyi sonuçları yakaladığı görülmektedir. Bu sonuçlar ışığında, CHIO algoritmasının çözüm kalitesi açısından rekabet edici olduğu tespit edilmiştir. Sonuç olarak CHIO algoritması çok düzeyli görüntü eşiği problemi için alternatif bir algoritma olabilir.

Anahtar Kelimeler— coronavirüs sürü bağışıklığı algoritması, yapay zeka, optimizasyon, görüntü eşik seçimi

Otsu Based Optimal Multilevel Image Thresholding Using Coronavirus Herd Immunity Optimizer

Abstract— Thresholding selection plays an important role in image segmentation. Minimum error method, iterative method, entropy method and Otsu method are known as the most useful methods for thresholding selection. In this study, Otsu technique is used as thresholding method. Since the complexity of the problem would increase exponentially depending on the increase in the number of thresholds (K), it seems more appropriate to use a swarm intelligence algorithm instead of mathematical methods. Therefore, in this study, the Coronavirus herd immunity optimizer (CHIO), which has been introduced to the literature in recent years, is used as a swarm intelligence algorithm. In the experimental studies, six different images are used as test data in experimental studies. K value is determined as 2, 3, 4 and 5 in this study. Using this data set, the CHIO algorithm is compared with the successful algorithms in the literature such as differential evolution (DE), gray wolf (GWO), and particle swarm (PSO) algorithms in the equal conditions. According to the results obtained, it is seen that in studies conducted on 6 test data using the CHIO algorithm, the proposed algorithm achieves the best results in 100% of the images when K=2, 83% when K=3 and 4, and finally 50% when K=5. In the light of these results, it has been determined that the CHIO algorithm is competitive in terms of solution quality. As a result, the CHIO algorithm can be an alternative algorithm for the multilevel image thresholding problem.

Keywords— coronavirus herd immunity optimizer, artificial intelligence, metaheuristic, image threshold selection

1. GİRİŞ (INTRODUCTION)

Eşikleme, görüntü segmentasyonu için en önemli ve etkili araçlardan biridir. Eşikleme, video sıkıştırma, [1, 2], görüntü gürültü giderme [3], belge işleme [4] ve hedef tanıma [5] yaygın olarak kullanılan görüntü bölütlemenin ana yöntemi ve önemli bir dalıdır. Eşikleme işlemi bir eşik (th) değeri olarak çalıştığı için, yoğunluk değeri 'th' değerinden yüksek olan pikseller birinci sınıf olarak etiketlenirken geri kalanlar ikinci sınıf olarak etiketlenir [6]. Görüntü arka plan ve hedef nesne olmak üzere iki sınıfa ayrıldığında, görev iki seviyeli eşikleme (BT) olarak adlandırılır ve yalnızca bir 'th' değeri gerektirir. İki seviyeli eşiklemenin uzantısı çok seviyeli eşikleme (MT) olarak adlandırılır [7]. MT'de pikseller ikiden fazla sınıfa ayrılır ve bu da birden fazla 'th' değeri gerektirir [8]. Çok seviyeli eşikleme, gri seviyeli bir görüntüyü birden fazla eşik tespit ederek birkaç farklı bölgeye ayırır [9].

Eşik sayısı arttıkça, eşikleme problemindeki hesaplama karmaşıklığı da önemli ölçüde artmaktadır. Bu nedenle, hesaplama verimliliğini artırmak için bu probleme literatürde çok sayıda sürü zekası optimizasyon algoritması uygulanmaktadır [10, 11]. Akay (2012)'in çalışmasında parçacık sürü (PSO) ve yapay arı koloni (ABC) algoritmalarını kullanılmıştır. 12 farklı görüntü üzerinde yaptığı çalışmalarda ise eşikleme yöntemi olarak Otsu ve Kapur yöntemleri tercih edilmiştir. Oliva vd. çalışmalarında Harmoni arama algoritmasını 10 farklı MRI görüntü üzerinde kullanmışlardır. Algoritmayı ise BF, PSO ve genetik algoritma ile kıyaslamışlardır [8]. Raja vd. Brownian Dağılım rehberli Ateşböceği Algoritması kullanarak Otsu metodu yöntemiyle eşikleme işlemini gerçekleştirmişlerdir. Çalışmalarını 12 adet gri seviye görüntü kullanarak orijinal ateş böceği algoritması ile kıyaslamışlardır [12]. Aziz vd. balina ve güve alive optimizasyon algoritmalarıyla eşikleme problemine çözüm aramışlardır. 8 farklı görüntü üzerinde Otsu yöntemi kullanarak elde ettikleri çözümleri RMSE, PSNR, SSIM gibi kriterlere göre değerlendirmişlerdir [13]. Kotte vd. 10 farklı MRI görüntüsü kullanarak Kapur ve Otsu yöntemleriyle eşikleme işlemini gerçekleştirmişlerdir. Adaptif rüzgar sürücü algoritması (AWDO) ile yaptıkları çalışmada sonuçları PSO ve orijinal WDO algoritması gibi algoritmalarla kıyaslamışlardır [14]. Kahraman vd. renkli görüntülerin çok seviyeli eşiklenmesi ve sınıflandırılması üzerine çalışmışlardır [15]. Koc vd. gri kurt optimizasyon algoritması kullanarak Otsu yöntemiyle eşikleme problemine yeni çözümler üretmiştir. Eşikleme işlemini 6 görüntü üzerinde Otsu yöntemi kullanarak gerçekleştirmişlerdir [16]. Diz evrimsel çok amaçlı algoritma ile yaptıkları çalışmada 6 gri seviye görüntü kullanarak eşikleme işlemini gerçekleştirmişlerdir. Eşikleme tekniği olarak ise Kapur, Otsu ve bulanık C-means gibi yöntemler kullanılmıştır [17]. Çelik ve Demirel zatürre tespiti için Otsu ve Ridler-Calvard görüntü işleme yöntemini önermişlerdir [18]. Karasulu görüntülerde insan kulağı tespit ve bölütlemesini temel alan biyometrik yetkilendirme üzerine bir inceleme yapmıştır. Yapılan çalışmada eşikleme yöntemi kullanılmıştır [19].

Bu çalışmada son yıllarda geliştirilmiş olan Coronavirus sürü bağışıklığı algoritması (CHIO) kullanarak çok düzeyli görüntü eşik seçimi gerçekleştirilmiştir. Yapılan çalışmada 6 farklı görüntü kullanılmış ve eşik seçimi için amaç fonksiyonu olarak Otsu tekniği tercih edilmiştir. Elde edilen sonuçlar GWO, DE gibi güçlü algoritmalar başta olmak üzere toplam altı farklı algoritma ile kıyaslanmıştır. Karşılaştırmalı sonuçlara göre CHIO algoritmasının diğer algoritmalara göre rekabet edici olduğu çok açık bir şekilde söylenebilir.

Bu çalışmanın genel organizasyonu şu şekildedir: Bölüm 2'de Otsu eşikleme metodu ve CHIO algoritmasına ait detaylar sunulmaktadır. Bölüm 3'te ise deneysel çalışmalar karşılaştırmalı verilmektedir. Son olarak Bölüm 4'te ise çalışmaya ilişkin tartışma kısmı yer almaktadır.

2. MATERYAL VE METOD (MATERIAL AND METHOD)

2.1. Çoklu Eşikleme Problemi (Multilevel Thresholding Problem)

Çok seviyeli eşikleme, literatürde çok sıklıkla tercih edilen görüntü segmentasyon yöntemlerinden birisidir [20]. Çok seviyeli eşikleme problemi, bir uygunluk kriterini $I(x,y)$ maksimuma çıkaran $[0, L-1]$ gri ölçek aralığındaki optimal eşikleri bulmakla ilgilenmektedir. Eşik değerlerini bulmak için Otsu'nun sınıflar arası varyans fonksiyonu kullanılır. Optimizasyon probleminin arama boyutu, dikkate alınan eşik sayısına göre atanır [12].

2.2. Otsu Eşikleme Metodu (Otsu Thresholding Method)

Literatürde var olan klasik ve optimizasyon algoritması tabanlı eşikleme yöntemleri, bazı yol gösterici parametreleri karşılayarak parçalı histogramda mümkün olan en iyi eşiği bulmak için kullanılmaktadır. Otsu tabanlı görüntü eşikleme ilk olarak 1979'da önerilmiştir. Bu yöntem, amaç fonksiyonunu maksimize ederek optimal değerleri sunmaktadır. Bu çalışmada Otsu'nun sınıflar arası varyans olarak bilinen parametrik olmayan bölütleme yöntemi ele alınmaktadır [12, 21]. Gri seviye eşikleme yöntemine için amaç fonksiyonuna ait matematiksel modeller aşağıda verilmektedir:

$$t^* = \operatorname{argmax}[f_b(t)] \quad (1)$$

$$f_b(t) = \sigma_0 + \sigma_1 \quad (2)$$

$$\sigma_0 = \omega_0(\mu_0 - \mu_t)^2 \text{ and } \sigma_1 = \omega_1(\mu_1 - \mu_t)^2 \quad (3)$$

$$\mu_0 = \frac{1}{\omega_0} \sum_{i=0}^{t-1} i x p_i \text{ and } \mu_1 = \frac{1}{\omega_1} \sum_{i=t}^{L-1} i x p_i \quad (4)$$

$$\omega_0 = \sum_{i=0}^{t-1} p_i \text{ and } \omega_1 = \sum_{i=t}^{L-1} p_i \quad (5)$$

$$p_i = \frac{x_i}{X} \quad (6)$$

buradaki x_i , I düzeyindeki piksel sayısını, X ise her düzeydeki toplam piksel sayısını ifade etmektedir. p_i

Denklem (6)'daki i . gri seviyenin olasılık değeridir. ω_0 ve ω_1 , Denklem (5)'deki 0. ve 1. segment oluşumunun tahmin edilen olasılığıdır. Denklem (4)'teki μ_0 ve μ_1 , 0. sınıf ve 1. sınıf ortalama yoğunluğudur. μ_t , Denklem(3)'deki orijinal görüntünün ortalama değerine karşılıktır. Denklem (2)'de σ_0 , 0. sınıfın varyansındır ve σ_1 , 1. sınıfın varyansındır.

Sınıflar arası varyansın temel denklemler ışığında, sınıflar arası varyansa dayalı görüntü eşikleme, aşağıdaki gibi verilen çok düzeyli eşikleme genişletilmektedir. Optimum eşik değeri t^* ise Denklem (7)'deki gibi hesaplanmaktadır.

$$t^* = \operatorname{argmax}[f_m(t)] \quad (7)$$

$$f_b(t) = \sum_{i=0}^n \sigma_i \quad (8)$$

$$\sigma_0 = \omega_0(\mu_0 - \mu_t)^2, \sigma_1 = \omega_1(\mu_1 - \mu_t)^2, \dots, \sigma_n = \omega_n(\mu_n - \mu_t)^2 \quad (9)$$

$$\mu_0 = \frac{1}{\omega_0} \sum_{i=0}^{t_0-1} i x p_i, \mu_1 = \frac{1}{\omega_1} \sum_{i=t_0}^{t_1-1} i x p_i, \dots, \mu_n = \frac{1}{\omega_n} \sum_{i=t_n}^{L-1} i x p_i \quad (10)$$

$$\omega_0 = \sum_{i=0}^{t_0-1} p_i, \omega_1 = \sum_{i=t_0}^{t_1-1} p_i, \dots, \omega_n = \sum_{i=t_n}^{L-1} p_i \quad (11)$$

2.3. Coronavirüs Sürü Bağışıklığı Algoritması (Coronavirus Herd Immunity Optimizer)

Coronavirüs sürü bağışıklığı algoritması (CHIO), 2020 yılında Al-Betar vd. tarafından önerilen yeni bir metasezgisel algoritmadır [22]. Diğer birçok sürü zekası algoritmasına benzer şekilde, doğal bir varlığın davranışını taklit eder, bu durumda ilhamını patojenik bir koronavirüsten almaktadır. CHIO, bulaşıcı hastalıklardan bağışıklık elde etme yöntemlerinden biri olarak bilinen sürü psikolojisinin uygulanması yoluyla bir sürüde doğal bağışıklık elde etme sürecini taklit etmektedir.

Koronavirüs sürü bağışıklığı fikri, CHIO adlı teorik bir optimizasyon algoritması oluşturmak için Al-Betar tarafından matematiksel olarak modellenmiştir. Model, enfekte olmayan çaresiz nüfusun büyük kısmını sağlam bir nüfusa dönüştürerek insanlığı hastalığa karşı en iyi şekilde korumanın bir yolunu bulmaya dayanmaktadır [22]. Sonuç olarak, geriye kalan tüm hassas vakalar etkilenmeyecek ve dirençli popülasyon artık hastalığı bulaştırmayacaktır. Modelde, sürü bağışıklığı olan bireylerden oluşan popülasyon üç kategoride sınıflandırılmaktadır: duyarlı, kontamine (veya doğrulanmış) ve bağışıklık kazanmış (veya yeniden kazanmış) kişiler [22, 23]. Algoritmaya ait temel denklemler Denklem (12)-(14)'te verilmiştir.

$$C(X_i^j(t)) = X_i^j(t) + r x (X_i^j(t) - X_i^c(t)) \quad (12)$$

$$N(X_i^j(t)) = X_i^j(t) + r x (X_i^j(t) - X_i^m(t)) \quad (13)$$

$$R(X_i^j(t)) = X_i^j(t) + r x (X_i^j(t) - X_i^m(t)) \quad (14)$$

Algoritma aşağıdaki adımlardan oluşmaktadır:

Adım 1: Algoritmanın başlangıç parametrelerini ayarla.
Parametreleri başlat. (N , HIS , S_r , $MaxIterasyon$, $MaxYas$)

Adım 2: CHIO popülasyonunu üret.

Her bir elamanın maliyetini uygunluk fonksiyonuna göre hesapla.

Adım 3: Populasyon konum güncelle

while $t \leq MaxIterasyon$

for $j=1$ to HIS **do**

Corona($X^j(t)$)=false

for $i=1$ to N

if $r < \frac{1}{3} x BR_r$

$X_i^j(t+1) = C(X_i^j(t))$ [Bakınız: Denklem (12)]

else if $r < \frac{2}{3} x BR_r$

$X_i^j(t+1) = N(X_i^j(t))$ [Bakınız: Denklem (13)]

else if $r < BR_r$

$X_i^j(t+1) = R(X_i^j(t))$ [Bakınız: Denklem (14)]

else

$X_i^j(t+1) = X_i^j(t)$

end if

end for

Adım 4: Populasyondaki elemanları güncelle

if $f(X^j(t+1)) \geq f(X^j(t))$: maksimizasyon işlemi

$X^j(t+1) = X^j(t)$

else

$A_j = A_j + 1$

end if

if $f(X^j(t+1)) < \frac{f(X^j(t+1))}{\Delta f(x)} \wedge S_j = 0 \wedge \text{Corona}(X^j(t+1))$

$S_j = 1$

$A_j = 1$

end if

if $f(X^j(t+1)) > \frac{f(X^j(t+1))}{\Delta f(x)} \wedge S_j = 1$

$S_j = 2$

$A_j = 0$

end if

Adım 5: Ölümcül Durumlar

if $A_j \geq MaxYas \wedge (S_j == 1)$

Yeni birey üret.

$S_j = 0$

$A_j = 0$

end if

end for

$t = t+1$

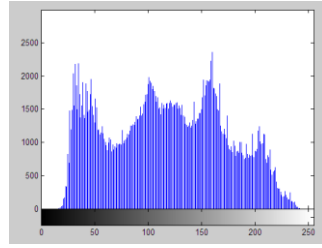
end while

Algoritma ilk olarak probleme uygun olarak parametrelerinin ayarlanması ile başlamaktadır. Adım 2'de popülasyon verilen parametrelere göre üretilir ve problemin amaç fonksiyonu yardımıyla her bir çözümün maliyeti hesaplanır. Adım 3'te verilen iterasyon sayısına göre her bir bireyin önerilen algoritmanın denklemleri yardımıyla mevcut pozisyonları güncellenir. Adım 4'te ise Adım 3'te üretilen yeni pozisyonlara ait çözümler amaç

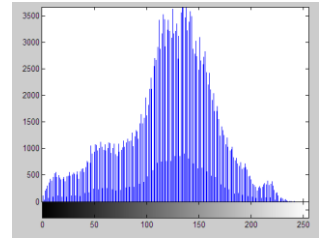
fonksiyonuna gönderilir ve çözümler daha iyiye yeni çözüm mevcut çözüm ile yer değiştirir (güncellenir). Adım 5'te ise söz konusu çözümde bir iyileşme görülmezse mevcut çözüm yerine arama uzayında yeniden çözüm üretilir. Bu işlem ile global arama güçlendirilir. Tüm bu işlemler maksimum iterasyon boyunca devam ettirilir. Sonuç olarak, elde edilen en iyi sonuç raporlanır.

3. DENEYSEL ÇALIŞMALAR (EXPERIMENTAL STUDIES)

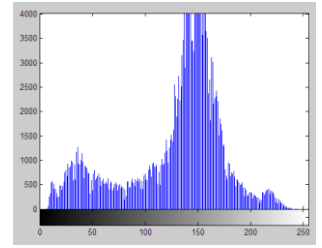
Bu çalışmada çok seviyeli eşikleme yöntemi olarak Otsu tekniği tercih edilmiştir. Eşikleme yöntemlerinde test etmek amacıyla farklı özelliklere sahip 6 görüntü kullanılmıştır [16]. Optimal eşik seviyesi belirlemek amacıyla da literatürde son yıllarda önerilmiş olan Coronavirüs sürü bağışıklık algoritması (CHIO) kullanılmıştır. Görüntülere ait özellikler Şekil 1'de verilmiştir. Ayrıca Otsu yöntemiyle elde edilen sonuçlar da eşik sayısına göre Tablo 1'de sırasıyla verilmiştir. Tüm deneysel çalışmalar Intel Core i7 2.80 GHz CPU, 16 GB RAM ve Windows 10 64-bit işletim sistemi üzerinde Matlab R2021a kullanılarak gerçekleştirilmiştir.



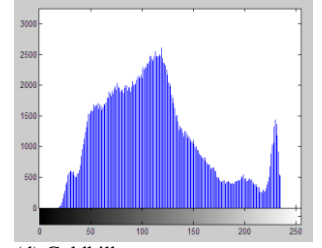
(a) Barbara



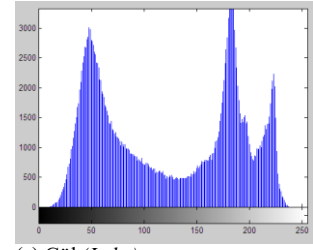
(b) Oturma odası (Living room)



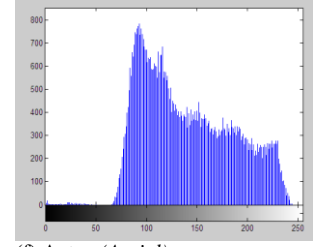
(c) Tekneler (Boats)



(d) Goldhill



(e) Göl (Lake)



(f) Anten (Aerial)

Şekil 1. Test verileri
(Test data)

Tablo 1. Otsu yöntemine göre elde edilen test veri sonuçları
(Test data results obtained according to the Otsu method)

Resimler	K	Eşik Değerleri	En iyi değer
Barbara	2	82, 147	2608.611
	3	75, 127, 176	2785.163
	4	66, 106, 142, 182	2856.262
	5	57, 88, 118, 148, 184	2890.977
Oturma odası	2	87, 145	1627.909
	3	76, 123, 163	1760.103
	4	56, 97, 132, 168	1828.864
	5	49, 88, 120, 146, 178	1871.991
Tekneler	2	93, 155	1863.347
	3	73, 126, 167	1994.536
	4	65, 114, 147, 179	2059.866
	5	51, 90, 126, 152, 183	2092.776
Goldhill	2	94, 161	2069.510
	3	83, 126, 179	2220.373
	4	69, 102, 138, 186	2295.380
	5	63, 91, 117, 147, 191	2331.157
Göl	2	85, 154	3974.738
	3	78, 140, 194	4112.631
	4	67, 110, 158, 198	4180.886
	5	57, 88, 127, 166, 200	4216.944
Anten	2	125, 178	1808.171
	3	109, 147, 190	1905.411
	4	104, 134, 167, 202	1957.0180
	5	99, 123, 148, 175, 205	1980.657

DeneySEL sonuçların adil bir şekilde karşılaştırılabilmesi için her bir eşik değeri için 25 bağımsız çalışma gerçekleştirilmiştir. Popülasyon sayısı olarak 40 belirlenirken, iterasyon sayısı olarak ise 2000 belirlenmiştir. Elde edilen sonuçlar literatürde yaygın olarak kullanılan PSO, Diferansiyel evrim (DE), Guguk

kuşu arama (CS), Ateş böceği (FA), Yarasa (BA) ve Gri kurt (GWO) algoritmalarının sonuçlarıyla kıyaslanmıştır. Bu algoritmaların sonuçları doğrudan Koc vd. çalışmasından alınmıştır [16]. Sonuçlar ortalama (Ort.) ve standart sapma (Std. Sap.) açısından karşılaştırmalı olarak sunulmuştur. *Ort ve Std. Sap.* Değerleri sırasıyla 25 bağımsız çalışmaya ait Otsu metodu kullanılarak edilen amaç fonksiyonunun sonuçlarının ortalama değerini ve standart sapma değerlerini ifade etmektedir. Tablo 2, 3, 4 ve 5 sırasıyla K=2, 3, 4 ve 5 değerleri için elde edilen

deneysel sonuçları sunmaktadır. Tablo 2-5'te koyu renkle ifade edilen yerler önerilen algoritmanın ilgili test verisi için en iyi olduğunu ifade etmektedir.

Tablo 2 incelendiğinde önerilen CHIO algoritmasının 6 görüntünün tamamında *Ort.* değer açısından en iyi değere ulaştığını göstermektedir. *Std. Sap.* açısından ise sadece Göl isimli görüntüde en iyiyi yakaladığı görülmektedir. Sonuç olarak CHIO yöntemi K=2 olduğu durumda çok başarılıdır.

Tablo 2. Otsu kriterine göre K=2 için algoritmalarından elde edilen sonuçların karşılaştırılması
(Comparison of the results obtained from the algorithms according to the Otsu criterion for K=2)

Yöntem		Barbara	Oturma odası	Tekneler	Goldhill	Göl	Anten
Otsu	En İyi	2608.611	1627.909	1863.347	2069.510	3974.738	1808.171
PSO	Ort.	2608.611	1627.909	1863.347	2069.510	3974.738	1808.171
	Std. Sap.	1.82E-12	0.00E+00	0.00E+00	4.55E-13	3.64E-12	2.27E-13
DE	Ort.	2608.611	1627.909	1863.347	2069.510	3974.738	1808.171
	Std. Sap.	1.82E-12	0.00E+00	0.00E+00	4.55E-13	3.64E-12	2.27E-13
CS	Ort.	2608.611	1627.909	1863.347	2069.510	3974.738	1808.171
	Std. Sap.	1.82E-12	0.00E+00	0.00E+00	4.55E-13	3.64E-12	2.27E-13
FA	Ort.	2608.611	1627.909	1863.347	2069.510	3974.738	1808.171
	Std. Sap.	1.82E-12	0.00E+00	0.00E+00	4.55E-13	3.64E-12	2.27E-13
BA	Ort.	2608.611	1627.909	1863.347	2069.510	3974.738	1808.171
	Std. Sap.	1.36E-12	2.27E-13	0.00E+00	4.55E-13	4.09E-12	2.27E-13
GWO	Ort.	2608.611	1627.909	1863.347	2069.510	3974.738	1808.171
	Std. Sap.	1.84E-12	6.89E-13	1.84E-12	1.84E-12	2.30E-12	6.89E-13
CHIO	Ort.	2608.898	1627.909	1863.347	2069.510	3974.738	1808.171
	Std. Sap.	1.65E-01	4.64E-13	6.96E-13	1.39E-12	1.39E-12	6.96E-13

Tablo 3. Otsu kriterine göre K=3 için algoritmalarından elde edilen sonuçların karşılaştırılması
(Comparison of the results obtained from the algorithms according to the Otsu criterion for K=3)

Yöntem		Barbara	Oturma odası	Tekneler	Goldhill	Göl	Anten
Otsu	En İyi	2785.163	1760.103	1994.536	2220.373	4112.631	1905.411
PSO	Ort.	2785.163	1760.103	1994.536	2220.373	4112.631	1905.411
	Std. Sap.	2.27E-12	2.27E-13	1.59E-12	1.36E-12	4.55E-12	1.14E-12
DE	Ort.	2785.162	1760.103	1994.535	2220.373	4112.631	1905.411
	Std. Sap.	8.31E-03	2.27E-13	7.26E-03	1.36E-12	4.55E-12	1.14E-12
CS	Ort.	2785.163	1760.103	1994.536	2220.373	4112.631	1905.411
	Std. Sap.	2.27E-12	2.27E-13	1.59E-12	1.36E-12	4.55E-12	1.14E-12
FA	Ort.	2785.163	1760.103	1994.536	2220.373	4112.631	1905.411
	Std. Sap.	2.27E-12	2.27E-13	1.59E-12	1.36E-12	4.55E-12	1.14E-12
BA	Ort.	2785.163	1760.103	1994.536	2220.373	4112.631	1905.411
	Std. Sap.	1.36E-12	2.27E-13	1.14E-12	1.36E-12	3.64E-12	1.14E-12
GWO	Ort.	2785.612	1760.103	1994.536	2220.373	4112.631	1905.411
	Std. Sap.	1.38E-12	6.89E-13	1.15E-12	1.38E-12	5.51E-12	9.19E-13
CHIO	Ort.	2785.383	1760.103	1994.536	2220.373	4112.631	1905.411
	Std. Sap.	1.10E-01	6.96E-13	2.32E-13	1.39E-12	0.00E+00	2.32E-13

Tablo 4. Otsu kriterine göre K=4 için algoritmalarından elde edilen sonuçların karşılaştırılması
(Comparison of the results obtained from the algorithms according to the Otsu criterion for K=4)

Yöntem		Barbara	Oturma odası	Tekneler	Goldhill	Göl	Anten
Otsu	En İyi	2856.262	1828.864	2059.866	4180.886	4180.886	1957.0180
PSO	Ort.	2856.261	1828.864	2059.866	2295.380	4180.884	1955.086
	Std. Sap.	6.66E-03	1.59E-12	4.22E-04	1.48E-03	7.41E-03	7.65E + 00
DE	Ort.	2856.261	1828.860	2059.865	2295.380	4180.884	1955.086
	Std. Sap.	2.80E-03	1.30E-02	6.85E-03	1.48E-03	7.41E-03	7.65E + 00
CS	Ort.	2856.262	1828.864	2059.866	2295.380	4180.886	1957.018
	Std. Sap.	2.45E-03	1.59E-12	1.36E-12	2.27E-12	0.00E+00	0.00E+00
FA	Ort.	2856.262	1828.864	2059.866	2295.380	4180.886	1957.018
	Std. Sap.	4.55E-13	1.59E-12	1.36E-12	2.27E-12	0.00E+00	0.00E+00
BA	Ort.	2856.262	1828.864	2059.866	2295.380	4180.886	1957.018
	Std. Sap.	4.55E-13	2.27E-12	9.09E-13	2.27E-12	0.00E+00	2.27E-13
GWO	Ort.	2856.699	1828.864	2059.866	2295.380	4180.886	1957.018
	Std. Sap.	4.59E-13	1.61E-12	2.30E-12	2.30E-12	0.00E+00	2.77E-04
CHIO	Ort.	2856.463	1828.864	2059.866	2295.380	4180.886	1957.018
	Std. Sap.	1.10E-01	4.64E-13	0.00E+00	1.39E-12	0.00E+00	1.16E-12

Tablo 5. Otsu kriterine göre K=5 için algoritmalarından elde edilen sonuçların karşılaştırılması
(Comparison of the results obtained from the algorithms according to the Otsu criterion for K=5)

Yöntem		Barbara	Oturma odası	Tekneler	Goldhill	Göl	Anten
Otsu	En İyi	2890.977	1871.991	2092.776	2331.157	4216.944	1980.657
PSO	Ort.	2890.976	1871.985	2092.771	2331.156	4216.943	1979.170
	Std. Sap.	5.05E-02	2.29E-02	8.36E-03	3.56E-04	3.99E-03	2.51E + 00
DE	Ort.	2890.971	1871.977	2092.767	2331.156	4216.943	1979.170
	Std. Sap.	2.05E-02	2.34E-02	2.71E-02	3.56E-04	3.99E-03	2.51E + 00
CS	Ort.	2890.977	1871.990	2092.776	2331.155	4216.944	1980.651
	Std. Sap.	4.85E-04	2.70E-03	1.03E-03	4.76E-03	9.09E-13	1.16E-02
FA	Ort.	2890.977	1871.991	2092.774	2331.157	4216.944	1980.657
	Std. Sap.	3.64E-12	0.00E+00	3.57E-03	2.27E-12	9.09E-13	9.09E-13
BA	Ort.	2890.977	1871.991	2092.773	2331.157	4216.944	1979.514
	Std. Sap.	2.73E-12	0.00E+00	3.78E-03	2.27E-12	3.64E-12	2.29E + 00
GWO	Ort.	2891.491	1871.985	2092.773	2331.156	4216.943	1980.656
	Std. Sap.	6.96E-03	1.91E-02	6.43E-03	3.42E-03	1.97E-03	2.57E-03
CHIO	Ort.	2891.116	1871.990	2092.774	2331.153	4216.943	1980.589
	Std. Sap.	9.59E-02	3.86E-03	3.32E-03	7.81E-03	7.05E-04	1.01E-01

Tablo 3 detaylı incelendiğinde ortalama değerlere göre Barbara hariç diğer tüm 5 görüntüde CHIO algoritmasının en iyi eşik değerlerini yakaladığı görülmektedir. Standart sapma açısından ise Barbara, Göl ve Goldhill hariç diğer 3 görüntüde en iyi değerleri ürettiği görülmektedir. Sonuç olarak K=3 değerinde önerilen yöntemin çözüm kalitesi açısından diğer algoritmalarla beraber çok başarılı olduğu görülmektedir.

K=4 değeri için Tablo 4'e ait sonuçlar incelendiğinde ortalama değer açısından K=3 olduğu duruma benzer şekilde Barbara verisi hariç diğer tüm görüntülerde en iyi değere ulaştığı görülmektedir. Standart sapma açısından önerilen algoritma değerlendirildiğinde sadece Barbara ve Anten verisinde en iyi değere ulaşamadığı diğer 4 görüntüde kararlı sonuçlar ürettiği görülmektedir. Sonuç olarak K=4 değeri için CHIO algoritması çözüm kalitesi ve kararlılık açısından başarılı bir algoritmadır.

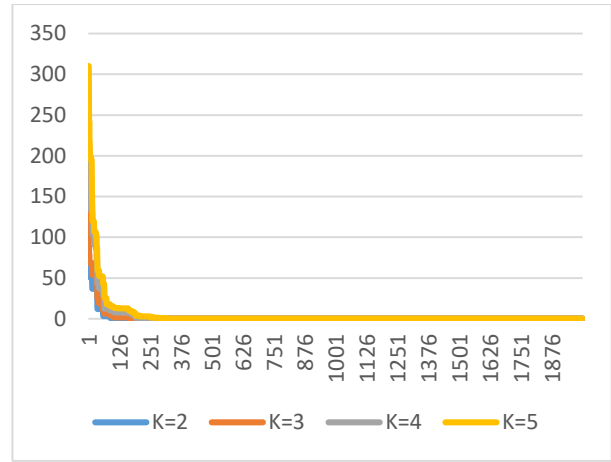
Son olarak Tablo 5 incelendiğinde ise çözüm kalitesi açısından Oturma odası, tekne ve göl görüntülerinde en iyi sonucu yakaladığı görülürken, diğer görüntülerde ise optimal değere çok yaklaştığı görülmektedir. Standart sapma açısından incelendiğinde ise önerilen yöntemin çok başarılı bir performans sergileyemediği görülmektedir. Sonuç olarak CHIO yöntemi K=5 olduğu durumda çözüm kalitesi açısından kayda değer bir başarı yakalarken, standart sapma açısından ise çok kararlı gözükmemektedir.

3.1. Yakınsama Grafikleri (Convergence Curves)

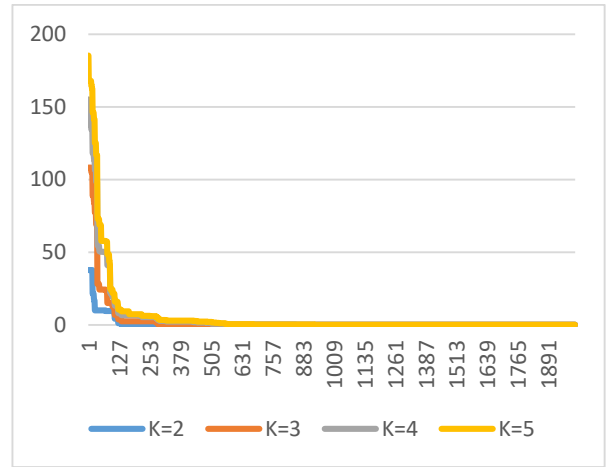
Önerilen CHIO algoritmasına ait 25 bağımsız çalışmadan elde edilen en iyi yakınsama sonuçları her bir görüntü için ayrı ayrı sunulmuştur. Burada CHIO algoritmasıyla elde edilen sonuçlar ile Tablo 1'deki Otsu metoduna ait bilinen en iyi değerlerinin farkı alınmıştır. Böylece hata oranları tespit edilmektedir. K değerleri açısından her bir görüntüye ait sonuçlar kıyaslamalı olarak Şekil 2'de verilmektedir.

Şekil 2(a) incelendiğinde Barbara görüntüsünde K=2, 3, 4 ve 5 olduğu durumlarda yaklaşık ilk 200 iterasyon içerisinde optimal değere ulaşıldığı görülmektedir. Şekil 2(b) incelendiğinde oturma odası'na ait yakınsama sonuçlarında K sayısına bağlı olarak K'nın artması yakınsama süresini uzattığı görülmektedir. K=2 olduğunda neredeyse ilk 100 iterasyonda yakınsamayı tamamlarken, K=5 olduğunda ise yaklaşık 500. iterasyonlarda yakınsamanın tamamlandığı görülmektedir.

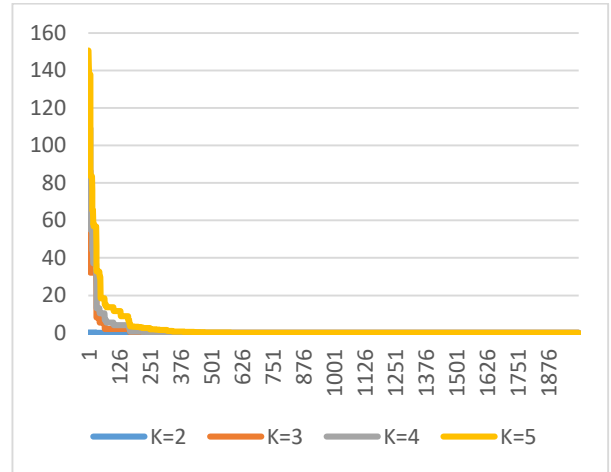
Şekil 2(c)'ye ait yakınsama sonuçları incelendiğinde K=2 olduğu durumda hemen ilk iterasyonlarda yakınsama başarıyla tamamlanırken, K=3, 4 ve 5 olduğu durumlarda ise yaklaşık olarak 150-300. iterasyonlar arasında optimal değere ulaştıkları görülmektedir.



(a) Barbara

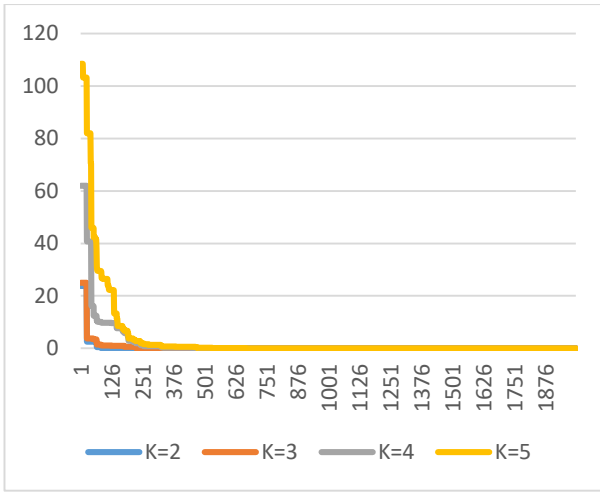


(b) Oturma odası



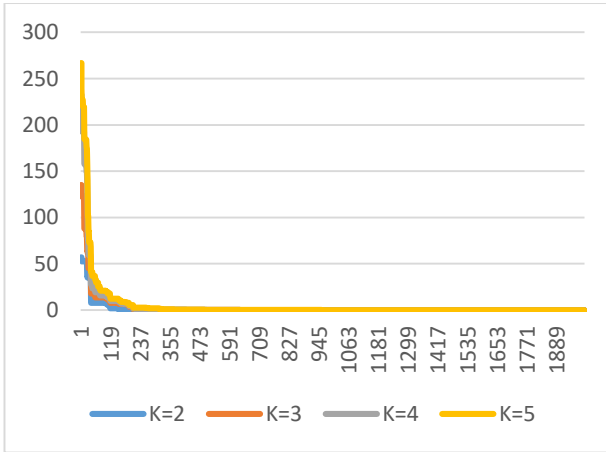
(c) Tekneler

Şekil 2(d)'ye ait görüntünün iterasyon sonuçları analiz edildiğinde K=2 için çok hızlı bir yakınsama görülmektedir. K=3 olduğunda yaklaşık ilk 200 iterasyonda işlem sonuçlanırken, K=4 ve K=5 durumunda yaklaşık olarak ilk 300 iterasyonda yakınsama işlemi tamamlanmasına karşın K=5 durumunda 250-300. iterasyonlar arasında daha hızlı bir yakınsamaya rastlanmıştır.



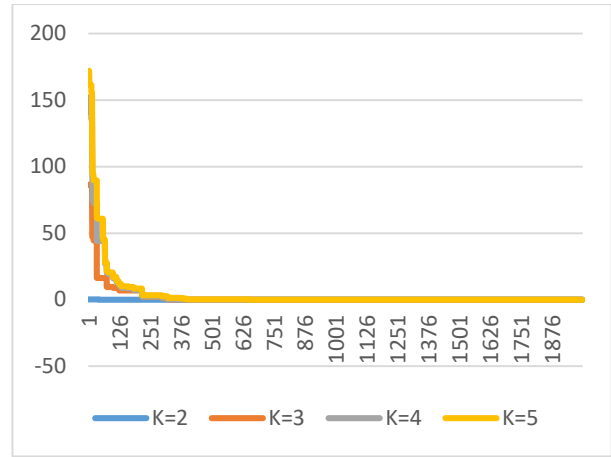
(d) Goldhill

Şekil 2(e)'ye ait görüntünün sonuçları incelendiğinde ise yaklaşık olarak ilk 200 iterasyon içerisinde tüm K (2, 3, 4 ve 5) değerleri için algoritmanın yakınsamayı başarıyla tamamladığı açıkça görülmektedir. K değerlerinin artması sadece ilk aşamada optimal değere olan uzaklığı artırırken algoritmanın başarılı yakınsama kabiliyetiyle beraber K değerinin yüksek olması sonucu negatif etkilememektedir.



(e) Göl

Son olarak Şekil 2(f)'teki yakınsama sonuçları analiz edildiğinde ise K=2 durumunda algoritmanın çok kısa bir sürede optimal sonuca ulaştığı görülmektedir. K=3 olduğu duruma bakıldığında yaklaşık olarak ilk 300 iterasyon içerisinde yakınsama işleminin tamamlandığı görülmektedir. K=4 ve K=5 durumunda ise ilk 400 iterasyon içerisinde algoritmanın en iyi sonuca ulaştığı görülmektedir.



(f) Anten

Şekil 2. Görüntülere ait yakınsama grafikleri
(Convergence curves of the test images)

Şekil 3-8'de K değerlerine göre segmente edilmiş tüm sonuçlar verilmiştir. Önerilen algoritma ile elde edilen K değerlerine göre üretilen sonuç görüntüler birlikte sunulmaktadır. Bu sonuçlara K=2 olduğu durumda elde edilen görüntü ile K=5 olduğu durumda elde edilen sonuç görüntü arasında segmentasyon açısından önemli bir fark olduğu açıkça görülmektedir.



(K=2) 82, 147



(K=3) 75, 128, 176



(K=4) 67, 106, 141, 182



(K=5) 55, 87, 116, 148, 184

Şekil 3. Barbara için K değerlerine göre elde edilmiş görüntüler

(Images obtained by K values for Barbara)



(K=2) 87, 145



(K=3) 76, 123, 163



(K=4) 56, 97, 132, 168



(K=5) 49, 88, 120, 146, 178

Şekil 4. Oturma odası için K değerlerine göre elde edilmiş görüntüler
(Images obtained by K values for Oturma Odası)



(K=2) 94, 161



(K=3) 83, 126, 179



(K=4) 69, 102, 138, 186



(K=5) 63, 91, 117, 147, 191

Şekil 6. Goldhill için K değerlerine göre elde edilmiş görüntüler
(Images obtained by K values for Goldhill)



(K=2) 93, 155



(K=3) 73, 126, 167



(K=4) 65, 114, 147, 179



(K=5) 51, 90, 126, 152, 183

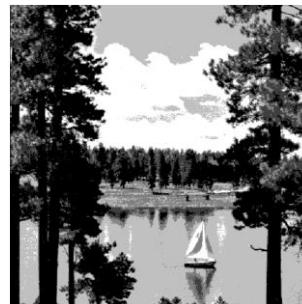
Şekil 5. Tekneler için K değerlerine göre elde edilmiş görüntüler
(Images obtained by K values for Tekneler)



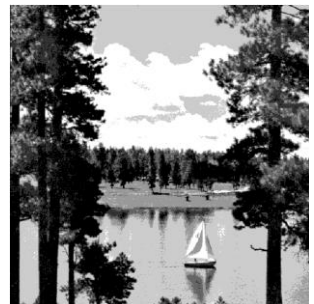
(K=2) 85, 154



(K=3) 78, 140, 194

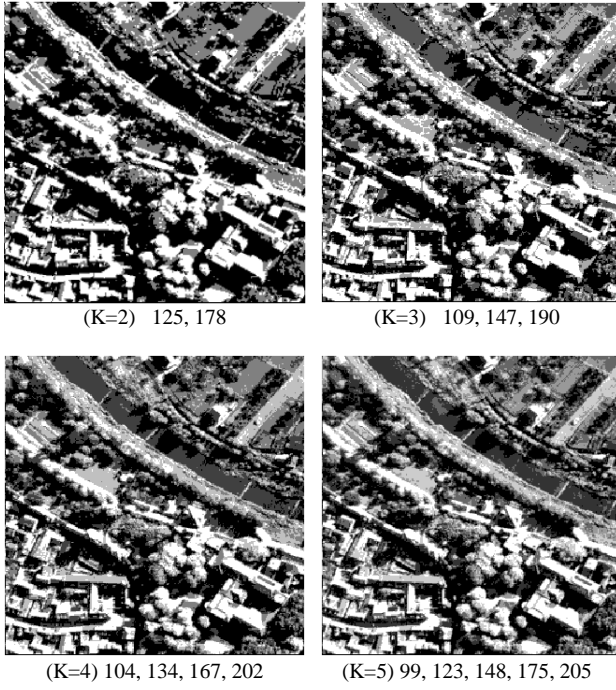


(K=4) 67, 110, 158, 198



(K=5) 57, 88, 127, 166, 200

Şekil 7. Göl için K değerlerine göre elde edilmiş görüntüler
(Images obtained by K values for Göl)



Şekil 8. Anten için K değerlerine göre elde edilmiş görüntüler
(Images obtained by K values for Anten)

4. SONUÇ (CONCLUSION)

Bu çalışmada çok düzeyli eşik seçimi için literatürde en çok kullanılan tekniklerden biri olan Otsu metodu tercih edilmiştir. Otsu metodu ile elde edilen değerler amaç fonksiyonunun çıktı değerleri olarak kabul edilmiş ve bu değerler bir optimizasyon algoritması ile maksimize edilmeye çalışılmıştır. Bu çalışmada sürü zekâsı algoritması olarak 2021 yılında önerilmiş olan Coronavirüs sürü bağışıklık algoritması (CHIO) kullanılmıştır. Eşik sayısı olarak ise 2 ile 5 değerleri bu çalışmada tercih edilmiştir. Deneysel çalışmalarda test veri seti olarak altı görüntü kullanılmıştır. CHIO algoritması altı farklı algoritma ile adil şekilde kıyaslanmıştır.

K değerlerine bağlı olarak elde edilen çözüm kalitesine ait sonuçlar incelendiğinde K=2 olduğu durumda önerilen algoritma çok iyi sonuçlar üretmektedir. K=3 ve K=4 olduğu durumlarda önerilen algoritma K=2 durumuna göre daha az başarılı görülmektedir. K=5 olduğu duruma bakıldığında ise bu kez diğer durumlara nazaran CHIO algoritmasının başarı grafiğinin belli ölçüde düştüğü görülmektedir. Bu durum dikkate alınarak K değerinin artmasına karşın CHIO yönteminin diğer algoritmalarla rekabet düzeyinin ihmal edilmez düzeyde düştüğü görülmektedir. Standart sapma değerlerine göre incelendiğinde algoritmanın genel olarak daha az başarılı olduğu görülmektedir. Buna rağmen ürettiği değerler makul bir seviyede olduğu düşünüldüğünde algoritmanın kararlı olduğu söylenebilir. Yakınsama grafikleri açısından genel olarak incelendiğinde ise yine K değerlerindeki artışla beraber yakınsama sürelerinin yaklaşık %25

oranında uzadığı görülmektedir. Bununla beraber genel olarak tüm test görüntülerinde CHIO algoritmasının neredeyse ilk 300 iterasyon içerisinde yakınsama işlemini başarıyla tamamladığı görülmektedir. Bu da algoritmanın yakınsama açısından etkili bir yöntem olduğunu göstermektedir. Bununla beraber K değerlerindeki artış durumunda K=3 ve K=4 durumunda önerilen yöntemin Barbara verisi için en başarılı sonucu bulamadığı görülmektedir. K=5 olduğu durumda ise Goldhill ve Anten verisinde de en iyi sonucu bulamamasına karşın sonuçların diğer algoritmalarla elde edilen sonuçlara çok yakın olduğu görülmektedir. Örneğin Goldhill için bulunan en iyi değer 2331.157 iken CHIO algoritması ile 2331.153 değeri bulunmuştur. Söz konusu farkın 0.004 olduğu göz önüne alındığında bu durumun ihmal edilebilir olduğu söylenebilir. Anten verisi için en iyi değer olarak 1980.657 sonucu bulunmuş olup CHIO ile 1980.589 değeri bulunmuştur. Yine bu sonuçlardan hareketle ortalama açısından farkın 0.068 olduğu görülmektedir. Bu sonucun da çok büyük bir fark olmadığı açıktır. Sonuç olarak, tüm bu değerlendirmeler ışığında önerilen yöntemin Barbara verisi hariç diğer tüm veri setleri için başarılı olduğu söylenebilir.

Gelecek çalışmalar için CHIO algoritması daha kararlı bir hale getirilerek K değerlerindeki artışa rağmen daha başarılı sonuçlar üretilebilir. Ayrıca, CHIO algoritması doğrudan ayrık bir uzayda temsil edilerek bu probleme uyarlanabilir ve sonuçlar mukayese edilebilir.

KAYNAKLAR (REFERENCES)

- [1] M. Omari and S. O. Jaafri, "Application of image compression to multiple-shot pictures using similarity norms with three level blurring", *Computers, Materials and Continua*, 58(2), 753-775, 2019.
- [2] Z. Pan, X. Yi, Y. Zhang, B. Jeon, and S. Kwong, "Efficient in-loop filtering based on enhanced deep convolutional neural networks for HEVC", *IEEE Transactions on Image Processing*, 29, 5352-5366, 2020.
- [3] K. Jin and S. Wang, "Image denoising based on the asymmetric Gaussian mixture model", *J. Internet Things*, 2(1), 1-11, 2020.
- [4] S. Susan and K. Rachna Devi, "Text area segmentation from document images by novel adaptive thresholding and template matching using texture cues", *Pattern Analysis and Applications*, 23(2), 869-881, 2020.
- [5] S. Bandyopadhyay, S. Das, and A. Datta, "A hybrid fuzzy filtering-fuzzy thresholding technique for region of interest detection in noisy images", *Applied Intelligence*, 50(4), 1112-1132, 2020.
- [6] K. Sowjanya and S. K. Injeti, "Investigation of butterfly optimization and gases Brownian motion optimization algorithms for optimal multilevel image thresholding", *Expert Systems with Applications*, 182, 115286, 2021.
- [7] B. Akay, "A study on particle swarm optimization and artificial bee colony algorithms for multilevel thresholding", *Applied Soft Computing*, 13(6), 3066-3091, 2013.

- [8] D. Oliva, E. Cuevas, G. Pajares, D. Zaldivar, and M. Perez-Cisneros, "Multilevel thresholding segmentation based on harmony search optimization", *Journal of Applied Mathematics*, 2013, 2013.
- [9] J. N. Kapur, P. K. Sahoo, and A. K. Wong, "A new method for gray-level picture thresholding using the entropy of the histogram", *Computer vision, graphics, and image processing*, 29(3), 273-285, 1985.
- [10] S. Pare, A. Kumar, G. K. Singh, and V. Bajaj, "Image segmentation using multilevel thresholding: a research review", *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, 44(1), 1-29, 2020.
- [11] L. Li, L. Sun, Y. Xue, S. Li, X. Huang, and R. F. Mansour, "Fuzzy multilevel image thresholding based on improved coyote optimization algorithm", *IEEE Access*, 9, 33595-33607, 2021.
- [12] N. Sri Madhava Raja, V. Rajinikanth, and K. Latha, "Otsu based optimal multilevel image thresholding using firefly algorithm", *Modelling and Simulation in Engineering*, 2014, 2014.
- [13] M. Abd El Aziz, A. A. Ewees, and A. E. Hassanien, "Whale optimization algorithm and moth-flame optimization for multilevel thresholding image segmentation", *Expert Systems with Applications*, 83, 242-256, 2017.
- [14] S. Kotte, R. K. Pullakura, and S. K. Injeti, "Optimal multilevel thresholding selection for brain MRI image segmentation based on adaptive wind driven optimization", *Measurement*, 130, 340-361, 2018.
- [15] A. S. Kahraman, T. R. Farshi, and R. Demirci, "Renkli görüntülerin çok seviyeli eşiklenmesi ve sınıflandırılması", *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, 6(4), 846-859, 2018.
- [16] I. Koc, O. K. Baykan, and I. Babaoglu, "Gri kurt optimizasyon algoritmasına dayanan çok seviyeli imge eşik seçimi", *Politeknik Dergisi*, 21(4), 841-847, 2018.
- [17] M. Abd Elaziz and S. Lu, "Many-objectives multilevel thresholding image segmentation using knee evolutionary algorithm", *Expert systems with Applications*, 125, 305-316, 2019.
- [18] A. Çelik and S. Demirel, "Otsu ve Ridler-Calvard Görüntü İşleme Yöntemlerinin Zatürre Tespitinde Kullanılması", *Muş Alparslan Üniversitesi Fen Bilimleri Dergisi*, 10(1), 917-923.
- [19] B. Karasulu, "Görüntülerde insan kulağı tespit ve bölütlemesini temel alan biyometrik yetkilendirme üzerine bir inceleme", *Bilişim Teknolojileri Dergisi*, 9(2), 97, 2016.
- [20] Y. Ölmez, A. Sengur, and G. Ozmen Koca, "Multilevel thresholding with metaheuristic methods", *Journal of the Faculty of Engineering and Architecture of Gazi University*, 36(1), 213-224, 2020.
- [21] N. Otsu, "A threshold selection method from gray-level histograms", *IEEE transactions on systems, man, and cybernetics*, 9(1), 62-66, 1979.
- [22] M. A. Al-Betar, Z. A. A. Alyasseri, M. A. Awadallah, and I. Abu Doush, "Coronavirus herd immunity optimizer (CHIO)", *Neural Computing and Applications*, 33(10), 5011-5042, 2021.
- [23] J. S. Lavine, A. A. King, and O. N. Bjørnstad, "Natural immune boosting in pertussis dynamics and the potential for long-term vaccine failure", *Proceedings of the National Academy of Sciences*, 108(17), 7259-7264, 2011.