

**KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

META-SEZGİSEL YAKLAŞIMLARA DAYALI İMGE BÖLÜTLEME

YÜKSEK LİSANS TEZİ

Bilgisayar Müh. Büşranur KÜÇÜKUĞURLU

**HAZİRAN 2019
TRABZON**

**KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

META-SEZGİSEL YAKLAŞIMLARA DAYALI İMGE BÖLÜTLEME

Bilgisayar Müh. Büşranur KÜÇÜKUĞURLU

**Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsünde
"BİLGİSAYAR YÜKSEK MÜHENDİSİ"
Unvanı Verilmesi İçin Kabul Edilen Tezdir.**

**Tezin Enstitüye Verildiği Tarih : 20.05.2019
Tezin Savunma Tarihi : 19.06.2019**

Tez Danışmanı: Dr. Öğr. Üyesi. Eyüp GEDİKLİ

Trabzon 2019

**KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**Bilgisayar Mühendisliği Anabilim Dalında
Büşranur KÜÇÜKUĞURLU Tarafından Hazırlanan**

META-SEZGİSEL YAKLAŞIMLARA DAYALI İMGE BÖLÜTLEME

**başlıklı bu çalışma, Enstitü Yönetim Kurulunun 28 / 05 / 2019 gün ve 1806 sayılı
kararıyla oluşturulan jüri tarafından yapılan sınavda
YÜKSEK LİSANS TEZİ
olarak kabul edilmiştir.**

Jüri Üyeleri

Başkan : Prof. Dr. Mustafa ULUTAŞ

Üye : Doç. Dr. Rüya ŞAMLI

Üye : Dr. Öğr. Üyesi Eyüp GEDİKLİ

m. ulutaş
R. Şamli
E. Gedikli

**Prof. Dr. Asim KADIOĞLU
Enstitü Müdürü**

ÖNSÖZ

Bölütleme görüntüyü analiz etme ve anlamada önemli bir yere sahiptir. Günümüzde tıbbi görüntüleme, yürüyüş tanıma, yüz, parmak izi ve nesne tanıma, uydu görüntülerindeki nesnelere bulma gibi birçok görüntü ve video işleme alanlarında kullanılmaktadır.

Bölütleme, görüntüyü farklı bölgelere ayırmak amacıyla kullanılır. Eşikleme, önemli bölütleme tekniklerinden biridir ve genel olarak görüntüyü arka plan ve nesnelere ayırma işlemi şeklinde tanımlanabilir. Meta-sezgisel algoritmalar ise doğadan esinlenilerek geliştirilen, sayısı her geçen gün artan ve optimizasyon problemlerinde başarılı sonuçlar üreten uygulamalardır. Bölütleme işlemi meta-sezgisel algoritmalar ile birlikte bir optimizasyon problemi olarak ele alınabilir.

Bu çalışmada, bazı önemli meta-sezgisel algoritmalar ve henüz imge bölütlemesinde kullanılmamış ‘Simbiyotik Organizmalar Arama’ algoritması farklı amaç fonksiyonlarına dayalı olarak gerçekleştirilmeye ve başarımları ölçülmeye çalışılmıştır.

Çalışmamda emeği geçen değerli danışman hocam Dr. Öğr. Üyesi Eyüp GEDİKLİ’ye ilgi, destek ve tecrübelerinden dolayı teşekkürlerimi sunuyorum. Ayrıca, başta annem ve babam olmak üzere desteklerini esirgemeyen Dr. Öğr. Üyesi Özkan BİNGÖL hocama ve arkadaşlarıma çok teşekkür ediyorum.

Büşranur KÜÇÜKÜĞURLU

Trabzon 2019

TEZ ETİK BEYANNAMESİ

Yüksek Lisans Tezi olarak sunduğum “Meta-sezgisel Yaklaşımlara Dayalı İmge Bölütleme” başlıklı bu çalışmayı baştan sona kadar danışmanım Dr. Öğr. Üyesi Eyüp GEDİKLİ'nin sorumluluğunda tamamladığımı, verileri/örnekleri kendim topladığımı, deneyleri/analizleri ilgili laboratuvarlarda yaptığımı/yaptırdığımı, başka kaynaklardan aldığım bilgileri metinde ve kaynakçada eksiksiz olarak gösterdiğimi, çalışma sürecinde bilimsel araştırma ve etik kurallara uygun olarak davrandığımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiğimi beyan ederim. 19/06/2019

Büşranur KÜÇÜKUĞURLU

İÇİNDEKİLER

	<u>Sayfa No</u>
ÖNSÖZ.....	III
TEZ ETİK BEYANNAMESİ.....	IV
İÇİNDEKİLER.....	V
ÖZET.....	VII
SUMMARY	VIII
ŞEKİLLER DİZİNİ	IX
TABLolar DİZİNİ.....	IX
SEMBOLLER DİZİNİ	XII
1. GENEL BİLGİLER	1
1.1. Giriş.....	1
1.2. İmge Bölütleme.....	3
1.3. Eşikleme Yöntemleri.....	4
1.3.1. Kapur Entropisi	5
1.3.2. Otsu Metodu.....	7
1.4. Meta-Sezgisel Algoritmalar	8
1.4.1. Parçacık Sürü Optimizasyonu (Particle Swarm Optimization - PSO)	9
1.4.2. Ateşböceği Algoritması (Firefly Algorithm - FA)	10
1.4.3. Yapay Arı Koloni (Artificial Bee Colony - ABC)	11
1.4.4. Genetik Algoritma (Genetic Algorithm - GA).....	13
1.4.5. Simbiyotik Organizmalar Arama (Symbiotic Organisms Search-SOS)	14
1.5. Meta-Sezgisel Yaklaşımlara Dayalı İmge Bölütleme.....	17
2. YAPILAN ÇALIŞMALAR, BULGULAR VE İRDELEME.....	26
2.1. PSO Tabanlı İmge Bölütleme	27
2.2. FA Tabanlı İmge Bölütleme	30
2.3. ABC Tabanlı İmge Bölütleme	32
2.4. GA Tabanlı İmge Bölütleme.....	34
2.5. SOS Tabanlı İmge Bölütleme	36

2.6.	Performans Deęerlendirme Metrikleri	40
2.6.1.	Amaç (Objective) Fonksiyon Deęeri	40
2.6.2.	PSNR ve SSIM Deęerleri.....	40
2.6.3.	Standart Sapma Deęeri.....	41
2.6.4.	Cpu İşlem Süresi	42
2.7.	Deneysel Sonuçlar.....	42
2.7.1.	Amaç Fonksiyon Deęeri Deneysel Sonuçları	42
2.7.2.	PSNR ve SSIM Deęerleri Deneysel Sonuçları	60
2.7.3.	Standart Sapma Deęeri Deneysel Sonuçları	66
2.7.4.	CPU İşlem Süresi Deneysel Sonuçları.....	68
2.7.5.	Klasik Yöntemler ile Karşılaştırma.....	71
3.	SONUÇLAR VE TARTIŞMA	26
4.	ÖNERİLER.....	26
5.	KAYNAKLAR	76
ÖZGEÇMİŞ		

Yüksek Lisans Tezi

ÖZET

META-SEZGİSEL YAKLAŞIMLARA DAYALI İMGE BÖLÜTLEME

Büşranur KÜÇÜKUĞURLU

Karadeniz Teknik Üniversitesi

Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Dr. Öğr. Üyesi Eyüp GEDİKLİ

2019, 81 Sayfa

Literatürde yıllardan beri imge bölütleme problemine yönelik sayısız metot önerilmiş ve tartışılmıştır. Eşikleme, imge bölütlemeye en yaygın kullanılan, uygulanabilirliği kolay ve elverişli sonuçlar üreten bir tekniktir. Kapur Entropisi ve Otsu yöntemi literatürde sıkça kullanılan başarılı ispatlanmış yöntemlerin başında gelir.

Tez çalışmasında, literatürde var olan bazı meta-sezgisel algoritmalar ve yeni bir meta-sezgisel algoritma olan ‘Simbiyotik Organizmalar Arama’ algoritması ile Otsu ve Kapur Entropisi eşikleme yöntemleri bir arada kullanılarak birçok görüntünün bölütleme işlemi gerçekleştirilmiştir. Bu yöntemler ile işlemlerin hesaplama karmaşıklığı azaltılmış ve daha elverişli sonuçlar üretilmiştir. Ayrıca, gerçekleştirilen tüm yöntemlerin karşılaştırılması ve deneysel sonuçları rapor edilmiştir.

Anahtar Kelimeler: Bölütleme, Eşikleme, Meta-Sezgisel Algoritma, Otsu, Kapur Entropisi, Parçacık Sürü Optimizasyonu, Ateşböceği Algoritması, Yapay Arı Koloni, Genetik Algoritma, Simbiyotik Organizmalar Arama.

Master Thesis

SUMMARY

IMAGE SEGMENTATION BASED ON META-HEURISTIC APPROACHES

Büşranur KÜÇÜKUĞURLU

Karadeniz Technical University
The Graduate School of Natural and Applied Sciences
Computer Engineering Graduate Program
Supervisor: Assoc. Ass. Prof. Eyüp GEDİKLİ
2019, 81 Pages

Numerous methods for image segmentation have been proposed and discussed in the literature for many years. Thresholding is the most commonly used technique in image segmentation, which is easy to implement and produces favorable results. Kapur Entropy and Otsu method are proven methods, which frequently used in the literature.

In this thesis, segmentation of several images has been performed by using Kapur's Entropy and Otsu thresholding methods with existing meta-heuristic algorithms and a new meta-heuristic algorithm, Symbiotic Organisms Search, together. The computational complexity of the processes was reduced by these methods and more favorable results were obtained. In addition, comparison and experimental results of all methods were reported.

Keywords: Segmentation, Thresholding, Meta-Heuristic Algorithm, Otsu, Kapur's Entropy, Particle Swarm Optimization, Firefly Algorithm, Artificial Bee Colony, Genetic Algorithm, Symbiotic Organisms Search.

ŞEKİLLER DİZİNİ

	<u>Sayfa No</u>
Şekil 1. (a) Gürültülü parmak izi (b) Histogram (c) Global eşik ile bölütlenmiş sonuç görüntüsü [13].....	2
Şekil 2. (a) Tek eşik ile bölütlenecek görüntü histogramı (b) İki eşik ile bölütlenecek görüntü histogramı [13].....	4
Şekil 3. SOS Algoritması temel adımları [33].....	15
Şekil 4. Meta- Sezgisel Yönteme Dayalı İmge Bölütleme.....	17
Şekil 5. Performans değerlendirmeleri için kullanılan test görüntüleri.....	26
Şekil 6. PSO tabanlı imge bölütleme akış şeması	28
Şekil 7. Kapur Entropi ölçüsü ve Otsu tabanlı gerçekleştirilmiş PSO algoritması ile $m= 1, 2, 3, 4$ ve 5 için elde edilen eşik değerleri ve bu değerler ile bölütlenmiş görüntüler.....	29
Şekil 8. FA tabanlı imge bölütleme akış şeması.....	30
Şekil 9. Kapur Entropi ölçüsü ve Otsu tabanlı gerçekleştirilmiş FA algoritması ile $m= 1, 2, 3, 4$ ve 5 için elde edilen eşik değerleri ve bu değerler ile bölütlenmiş görüntüler.....	31
Şekil 10. ABC tabanlı imge bölütleme akış şeması	32
Şekil 11. Kapur Entropi ölçüsü ve Otsu tabanlı gerçekleştirilmiş ABC algoritması ile $m= 1, 2, 3, 4$ ve 5 için elde edilen eşik değerleri ve bu değerler ile bölütlenmiş görüntüler	33
Şekil 12. GA tabanlı imge bölütleme akış şeması	35
Şekil 13. Kapur Entropi ölçüsü ve Otsu tabanlı gerçekleştirilmiş GA algoritması ile $m= 1, 2, 3, 4$ ve 5 için elde edilen eşik değerleri ve bu değerler ile bölütlenmiş görüntüler	36
Şekil 14. SOS tabanlı imge bölütleme akış şeması	38
Şekil 15. Kapur Entropi ölçüsü ve Otsu tabanlı gerçekleştirilmiş SOS algoritması ile $m= 1, 2, 3, 4$ ve 5 için elde edilen eşik değerleri ve bu değerler ile bölütlenmiş görüntüler	39
Şekil 16. Kapur tabanlı algoritmalar ile eşiklenmiş Cameraman görüntüleri	46
Şekil 17. Kapur tabanlı algoritmalar ile eşiklenmiş Lena görüntüleri.....	46
Şekil 18. Kapur tabanlı algoritmalar ile eşiklenmiş Peppers görüntüleri.....	47
Şekil 19. Kapur tabanlı algoritmalar ile eşiklenmiş Airplane görüntüleri.....	47
Şekil 20. Kapur tabanlı algoritmalar ile eşiklenmiş Baboon görüntüleri	48
Şekil 21. Kapur tabanlı algoritmalar ile eşiklenmiş Butterfly görüntüleri	48

Şekil 22. Kapur tabanlı algoritmalar ile eşiklenmiş Ostrich görüntüleri	49
Şekil 23. Kapur tabanlı algoritmalar ile eşiklenmiş Starfish görüntüleri	49
Şekil 24. Kapur tabanlı algoritmalar ile eşiklenmiş Flower görüntüleri	50
Şekil 25. Kapur tabanlı algoritmalar ile eşiklenmiş Plane görüntüleri.....	50
Şekil 26. Kapur tabanlı algoritmalar ile eşiklenmiş Girl görüntüleri	51
Şekil 27. Otsu tabanlı algoritmalar ile eşiklenmiş Cameraman görüntüleri.....	55
Şekil 28. Otsu tabanlı algoritmalar ile eşiklenmiş Lena görüntüleri	55
Şekil 29. Otsu tabanlı algoritmalar ile eşiklenmiş Peppers görüntüleri	56
Şekil 30. Otsu tabanlı algoritmalar ile eşiklenmiş Airplane görüntüleri	56
Şekil 31. Otsu tabanlı algoritmalar ile eşiklenmiş Baboon görüntüleri.....	57
Şekil 32. Otsu tabanlı algoritmalar ile eşiklenmiş Butterfly görüntüleri.....	57
Şekil 33. Otsu tabanlı algoritmalar ile eşiklenmiş Ostrich görüntüleri	58
Şekil 34. Otsu tabanlı algoritmalar ile eşiklenmiş Starfish görüntüleri.....	58
Şekil 35. Otsu tabanlı algoritmalar ile eşiklenmiş Flower görüntüleri.....	59
Şekil 36. Otsu tabanlı algoritmalar ile eşiklenmiş Plane görüntüleri	59
Şekil 37. Otsu tabanlı algoritmalar ile eşiklenmiş Girl görüntüleri.....	60
Şekil 38. Kapur tabanlı algoritmaların ortalama Cpu işlem süresi.....	70
Şekil 39. Otsu tabanlı algoritmaların ortalama Cpu işlem süresi	71

TABLolar DİZİNİ

Sayfa No

Tablo 1. Kapur Entropisi tabanlı algoritmaların ortalama amaç fonksiyon deęerleri ..	43
Tablo 2. Kapur Entropisi tabanlı algoritmalarından elde edilen ortalama eşik deęerleri.	44
Tablo 3. Otsu tabanlı algoritmaların ortalama amaç fonksiyon deęerleri	52
Tablo 4. Otsu tabanlı algoritmalarından elde edilen ortalama eşik deęerleri	53
Tablo 5. Kapur Entropisi tabanlı algoritmalarından elde edilen ortalama PSNR deęerleri	61
Tablo 6. Kapur Entropisi tabanlı algoritmalarından elde edilen ortalama SSIM deęerleri	62
Tablo 7. Otsu yöntemi tabanlı algoritmalarından elde edilen ortalama PSNR deęerleri	63
Tablo 8. Otsu yöntemi tabanlı algoritmalarından elde edilen ortalama SSIM deęerleri .	64
Tablo 9. Kapur amaç fonksiyonuna dayalı algoritmaların standart sapma deęerleri ...	66
Tablo 10. Otsu amaç fonksiyonuna dayalı algoritmaların standart sapma deęerleri.....	67
Tablo 11. Farklı tabanlarda gerçekleştirilmiş algoritmaların minimum (s) CPU işlem süresi	69
Tablo 12. Kapur Entropisi ve Kapur tabanlı PSO algoritması karşılaştırması	72
Tablo 13. Otsu yöntemi ve Otsu tabanlı PSO algoritması karşılaştırması	72

SEMBOLLER DİZİNİ

GA	Genetic Algorithm (Genetik Algoritma)
SA	Simulated Annealing (Benzetilmiş Tavlama)
TS	Tabu Search (Tabu Araştırma)
ACO	Ant Colony Optimization (Karıncı Koloni Optimizasyonu)
PSO	Particle Swarm Optimization (Parçacık Sürü Optimizasyonu)
DE	Differential Evolution (Diferansiyel Gelişim)
HS	Harmony Search (Harmoni Arama)
ABC	Artificial Bee Colony (Yapay Arı Koloni)
FA	Firefly Algorithm (Ateşböceği Algoritması)
SOS	Symbiotic Organisms Search (Simbiyotik Organizma Arama)
ATC	Automatic Thresholding Criterion (Otomatik Eşikleme Kriteri)
LGA	Learning Genetic Algorithm (Öğrenmeli Genetik Algoritma)
SLGA	Self-Learning Genetic Algorithm (Öz Öğrenmeli Genetik Algoritma)
HCOCLPSO	Hybrid Cooperative-Comprehensive Learning Particle Swarm Optimization (Kapsamlı-İşbirlikçi Hibrit Öğrenmeli Parçacık Sürü Optimizasyonu)
CQPSO	Cooperative Quantum Particle Swarm Optimization (Kapsamlı Kuantum Parçacık Sürü Optimizasyonu)
QPSO	Quantum Particle Swarm Optimization (Kuantum Parçacık Sürü Optimizasyonu)
HBMO	Honey Bee Mating Optimization (Bal Arısı Çiftleşme Optimizasyonu)
PSNR	Peak Signal Noise Ratio (Tepe Sinyal Gürültü Oranı)
SSIM	Structural Similarity Index Measure (Yapısal Benzerlik İndeks Ölçütü)
FSIM	Feature Similarity Index Measure (Özellik Benzerlik İndeks Ölçütü)
MSE	Mean Square Error (Ortalama Karesel Hata)
SD	Standard Deviation (Standart Sapma)

FODPSO	Fractional-Order Darwinian Particle Swarm Optimization (Kesirli Sıralı Darvinsel Parçacık Sürü Optimizasyonu)
DPSO	Darwinian Particle Swarm Optimization (Darvinsel Parçacık Sürü Optimizasyonu)
SBX	Simulated Binary Crossover (Temsili İkili Çaprazlama)
ABF	Adaptive Bacterial Foraging (Adaptif Bakteriyel Yem Arama)
BF	Bacterial Foraging (Bakteriyel Yem Arama)
AI	Adaptive Inertia (Adaptif Durağanlık)
AP	Adaptive Population (Adaptif Popülasyon)
FFA	Fuzzy Firefly Algorithm (Bulanık Ateşböceği Algoritması)
BFA	Brownian Search Firefly (Brown Arama Ateşböceği)
LFA	Lévy Search Firefly (Levy Arama Ateşböceği)
JADE	Adaptive Differential Evolution (Adaptif Diferansiye Gelişim)
CS	Cuckoo Search (Guguk Kuşu Arama)
WDO	Wind Driven Optimization (Rüzgar Güdümlü Optimizasyon)
ELR	Egg Laying Radius (Yumurta Yayma Yarıçapı)
ES	Evolution Strategy (Gelişim Stratejisi)
BFO	Bacterial Foraging Optimization (Bakteriyel Yem Arama Optimizasyonu)
GWO	Grey Wolf Optimizasyon (Gri Kurt Optimizasyonu)
SCA	Sine Cosine Algorithm (Sinüs Kosinüs Algoritması)
SSO	Social Spider Optimization (Sosyal Örümcek Optimizasyonu)

1. GENEL BİLGİLER

1.1. Giriş

İmge bölütleme görüntü işlemede temel bir problemdir. Şimdiye kadar imge bölütleme için birçok yöntem geliştirilmiş ve sunulmuştur ancak bu yöntemlerin çoğu probleme özgü olarak tasarlanmıştır. Tüm görüntü tipleri için uygulanabilecek genel ve etkili çözüm sunan bir bölütleme tekniği bulunmamaktadır. Bir bölütleme tekniği düşük seviye görüntü işleme tekniği olarak kabul edilir ve bölütleme sonuçları daha yüksek seviye görüntü işleme tekniklerine (örüntü tanıma, nesne tanıma ve izleme vb.) girdi olarak verilebilir [1].

İmge bölütleme görüntüdeki nesnelere ve kenar bilgilerini bulmaya yönelik bir yöntemdir [2]. Tam olarak tanıma; görüntünün gri seviye değerlerine, renk, doku, şekil, boyut veya konuma göre görüntüyü birbiri ile çakışmayan bölgelere ayırmak şeklinde verilebilir [3]. İmge bölütleme yöntemleri genel olarak dört gruba ayrılabilir; (1) histogram eşiklemeye dayalı yöntemler, (2) kenar bulmaya yönelik yöntemler, (3) bölge büyütme dayalı yöntemler (4) kümelemeye dayalı yöntemler. Eşikleme bu yöntemler içerisinde en basit ve en yaygın kullanılan yöntemdir [4]. Eşikleme işleminin bölütleme yöntemlerinin başında gelmesinin sebepleri; hesaplama azlığı, üstün performansı, kolay uygulanabilir olması, sonuçlarının kesinliği ve güvenilirliği şeklinde sıralanabilir [1, 5, 6].

Bir görüntü için tek eşik değeri seçimi, gri seviye dağılımında iki tepe noktası arasındaki kesişimin minimum değeri ile elde edilir [7]. Eşik seçimini özyinelemeli olarak otomatik kestirebilen ilk algoritma 1978'de Riddler ve Calvard [8] tarafından geliştirilmiştir. Sonrasında; 1996'da Leung ve Lam [9], 1979'da Trussel [10], 1994'te Yanni ve Horne [11] tarafından algoritma daha da geliştirilerek aşağıda adımları verilen algoritma elde edilmiştir [12, 13, 14].

Adım 1: Başlangıç için bir T küresel eşik seviyesi seç; bu eşik seviyesi görüntüdeki ortalama gri seviye değeri olarak seçilebilir.

Adım 2: T ile görüntüyü bölütleyin, oluşan iki piksel grubu için μ_1 ve μ_2 ortalama gri seviye değerlerini hesapla.

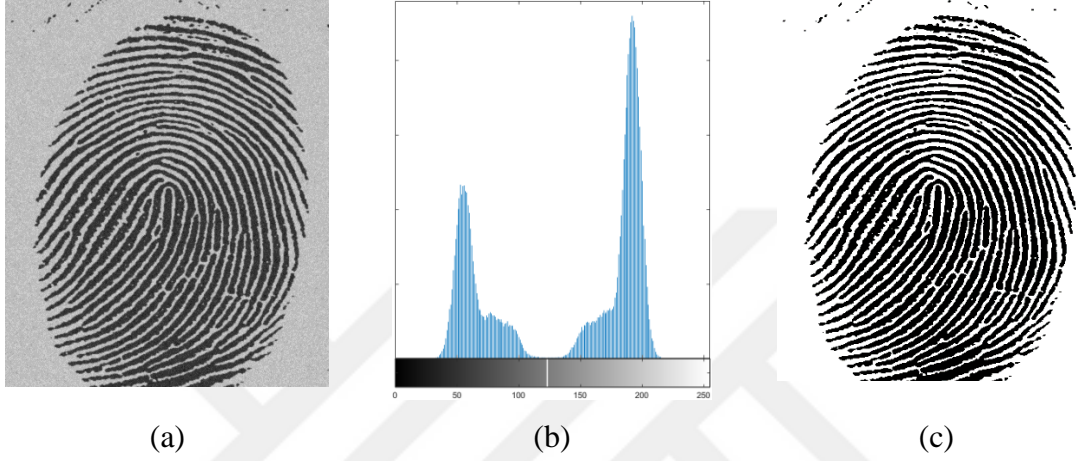
Adım 3: Yeni eşik değerini Eşitlik (1) 'i kullanarak hesapla.

$$T = \frac{\mu_1 + \mu_2}{2} \quad (1)$$

Adım 4: Bu adımları T ardışık değerleri arasındaki değişim durana kadar tekrarla.

Şekil 1' de geliştirilen algoritma ile bölütlenmiş bir parmak izi görüntüsü verilmiştir.

Algoritma sonucunda eşik değeri $T=125$ olarak bulunmuştur.



Şekil 1. (a) Gürültülü parmak izi (b) Histogram (c) Global eşik ile bölütlenmiş sonuç görüntüsü [13]

Görüntüyü tek eşik seviyesi üzerinden iki sınıfa ayırmak bölütlemeye en basit yoldur. Ancak görüntüye bağlı olarak birçok durumda bölütlemeye daha detaylı ve tutarlı sonuç elde etmek gerekir. Bu durumda problem daha karmaşık bir hal alır. Çok seviyeli eşikleme işlemi bu sebeple geliştirilmiştir [6, 15]. Tek seviyeli eşiklemede pikseller iki gruba ayrılır; bunlardan biri belirlenen eşik seviyesinin üstünde gri seviye değerlerine sahip olanlar, diğeri geriye kalan piksellerdir. Çok seviyeli eşikleme ise pikselleri birkaç sınıfa böler. Her sınıf, birkaç eşik seviyesi ile belirlenen aralıklardaki gri seviye değerlerine sahiptir [16].

Hem çok seviyeli hem de tek seviyeli eşikleme yöntemleri parametrik ve parametrik olmayan yaklaşımlar olarak sınıflandırılabilir [16]. Parametrik yaklaşımlarda, her bir sınıfın gri seviye dağılımı, genellikle Gauss dağılımına uyduğu varsayılan bir olasılık yoğunluk fonksiyonuna sahiptir. Bu yöntemler en küçük kareler yöntemini kullanarak verilen histogram verilerine en uygun dağılım parametrelerinin bir tahminini bulmaya çalışır. Bu da maliyetli ve zaman alıcı bir hesaplamaya sebep olur. Parametrik olmayan

yaklaşımlar ise görüntünün gri seviye bölgelerini, sınıf varyansı, entropi gibi bazı ayırt edici kriterlere dayanarak en uygun şekilde ayıran eşikleri bulur [16, 17].

Tek eşikli bölütleme için birçok parametrik ve parametrik olmayan yöntem önerilmiştir [16]. Bu yöntemler çok seviyeli eşikleme için genişletilebilirler ancak; eşik sayısındaki artışla birlikte meydana gelen kapsamlı arama hesaplama maliyetinin ve toplam çalışma süresinin de artmasına sebep olur [15].

Literatürde sıkça Otsu (sınıflar arası varyans) metodu [18], Kapur Entropisi [19], Tsallis Entropisi [20], Fuzzy Entropi [21], Minimum Çapraz Entropi [22], Renyi Entropi [23] gibi en bilinen eşikleme yöntemleri, hesaplama süresi problemini çözmek için meta-sezgisel yöntemlerde bir optimizasyon problemi olarak ele alınmış ve başarılı sonuçlar elde edilmiştir.

Bu çalışmada literatürde en sık kullanılan iki eşikleme yönteminden Kapur ve Otsu yaklaşımları Parçacık Sürü Optimizasyonu (Particle Swarm Optimization - PSO), Ateşböceği Algoritması (Firefly Algorithm - FA), Simbiyotik Organizmalar Arama (Symbiotic Organisms Search - SOS), Yapay Arı Koloni (Artificial Bee Colony - ABC) ve Genetik Algoritma (Genetic Algorithm - GA) ile optimize edilerek görüntülerin çok seviyeli eşikleme işlemleri gerçekleştirilmiştir.

1.2. İmge Bölütleme

İmge bölütleme görüntüyü birbiri ile kesişmeyen ve homojen bölgelerden oluşan parçalara ayırmayı amaçlar [24]. İmge bölütleme işlemi aşağıdaki gibi tanımlanabilir [4]:

Bir I görüntüsü ve bir P homojenlik şartı verilmiş olsun. I görüntüsünün K $\{R_1, R_2, \dots, R_K\}$ bölgeye ayrılması için aşağıdaki şartlar sağlanmalıdır:

- Görüntüdeki her piksel bir bölgeye atanmalıdır, $\bigcup_{k=1}^K R_k = I$
- Her bir piksel sadece bir bölgeye atanır, $R_k \cap R_{kk} = \emptyset$, burada $k \neq kk$
- Her bölge homojenlik şartını sağlar, $P(R_k) = \text{doğru}$, $k = 1, \dots, K$
- İki farklı bölge homojenlik şartını sağlayamaz, $P(R_k \cup R_{kk}) = \text{yanlış}$, burada $k \neq kk$

Bölütleme yöntemleri genel olarak dört gruba ayrılabilir; eşikleme (thresholding), kenar (edge) bulma, bölge büyütme (region growing) ve kümeleme (clustering) [4, 13]. Bu yöntemler aşağıdaki gibi açıklanabilir:

Eşikleme teknikleri: En basit ve uygulanabilirliği en kolay bölütleme yöntemidir. En basit haliyle, belirlenen bir eşik seviyesi üzerinden görüntüyü nesne ve arka plan olmak üzere iki sınıfa ayırmak şeklinde tanımlanabilir.

Kenar bulma teknikleri: Bölütleme işlemi görüntüdeki kenarlar bulunarak gerçekleştirilir. Görüntüdeki yerel değişimler genellikle belirlenen bir maske ile görüntü üzerinde gezdirilerek bulunur.

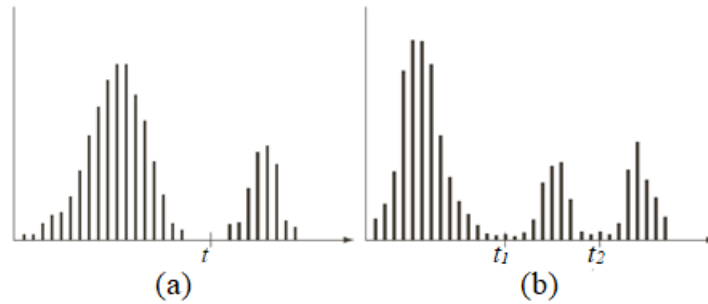
Bölge büyütme teknikleri: Bu teknikte bir tohum piksel kümesi seçilir. Belirlenen bir homojenlik şartına göre tohum piksel kümesinin komşuluk pikselleri kümeye dahil edilir. Bu işlem daha fazla piksel kümeye dahil olmayana dek devam eder.

Kümeleme teknikleri: İmge bölütleme işlemi bir kümeleme problemi olarak ele alınır. Her piksel bir desene ve bir bölge bir sınıfa denk gelir.

Bu yöntemler arasında en sık kullanılanlar eşikleme yöntemleri olduğu için bir sonraki bölümde eşikleme yöntemleri ayrıntılı olarak açıklanmıştır.

1.3. Eşikleme Yöntemleri

Gerçekleştirmesinin basitliği, hesaplama hızı ve sonuçların tutarlı olmasından dolayı görüntü eşikleme en sık kullanılan bölütleme yöntemlerinin başında gelir. Eşikleme işlemi, görüntü histogramındaki gri seviye değerlerine dayalı olarak görüntüyü doğrudan bölgelere ayırır. Örnek olarak Şekil 2 (a) 'da histogramı verilen siyah arkaplan üzerine parlak nesnelere olan bir I görüntüsünü verilsin. Görüntüde nesne ve arkaplan piksellerinin gri seviye değerleri iki ayrı baskın grup oluşturacaktır. Nesnelere arkaplandan ayırmak için uygun t eşik seviyesinin seçilmesi bölütleme için yeterli olacaktır.



Şekil 2. (a) Tek eşik ile bölütlenecek görüntü histogramı (b) İki eşik ile bölütlenecek görüntü histogramı [13]

Şekil 2 (b) 'de siyah arkaplan üzerine iki tip parlak cisim bulunan bir görüntünün histogramı verilmiştir. Nesnelere ve arkaplana ait gri seviye değerleri üç baskın grup oluşturduğundan çoklu eşikleme ile iki ayrı eşik seviyesi t_1 ve t_2 seçilmesi problem için daha uygun olacaktır [13].

L gri seviyeli bir görüntü için tek seviyeli (bi-level) eşikleme işlemi Eşitlik (2) ile ifade edilebilir:

$$\begin{aligned} M_0 &= \{g(x, y) \in I \mid 0 \leq t - 1\} \\ M_1 &= \{g(x, y) \in I \mid 0 \leq L - 1\} \end{aligned} \quad (2)$$

Çoklu eşikleme (multi-level) için genişletilmiş formül Eşitlik (3) ile ifade edilir:

$$\begin{aligned} M_0 &= \{g(x, y) \in I \mid 0 \leq g(x, y) \leq t_1 - 1\} \\ M_1 &= \{g(x, y) \in I \mid t_1 \leq g(x, y) \leq t_2 - 1\} \\ M_i &= \{g(x, y) \in I \mid t_i \leq g(x, y) \leq t_{i+1} - 1\} \\ M_m &= \{g(x, y) \in I \mid t_m \leq g(x, y) \leq L - 1\} \end{aligned} \quad (3)$$

burada, $g(x, y)$ pikselin gri seviye değeri, t_i ($i = 1, \dots, m$) i . eşik seviye değeri ve m eşik sayısıdır. M_m ($m = 1, \dots, m$), bölütlenen sınıflardır.

Literatürde eşikleme problemi için birçok yöntem önerilmiştir. Bu çalışmada en yaygın kullanılan ve başarısı ispatlanmış olan iki önemli eşikleme yöntemi Kapur Entropisi ve Otsu (between class variance) metodu ele alınmıştır.

1.3.1. Kapur Entropisi

Kapur fonksiyonu [19] sınıflar arası ayrılabilirliği ölçer. Görüntünün gri seviyelerinin olasılıksal dağılımına bağlı olarak entropi ölçüsü hesaplar. Sınıflar arası entropi ölçüleri maksimum olduğunda optimum eşik seviyeleri elde edilir. Bu nedenle, amaç en yüksek entropi değerini veren eşik seviyelerine ya da amaç fonksiyon değerine ulaşmaktır [25, 26]. Orijinal Kapur Entropisi tek seviyeli eşikleme için geliştirilmiştir ancak çok seviyeli eşikleme için de genişletilebilir. Her i gri seviye değerinin olasılığı, i . gri seviyenin oluşum frekansı, gri seviye değerlerinin toplamı ile normalize edilerek Eşitlik (4)' teki gibi hesaplanır:

$$p_i = \frac{h(i)}{\sum_{i=0}^{L-1} h(i)}, \quad i = 0, \dots, L-1 \quad (4)$$

Tek seviyeli eşikleme için Kapur Entropi ölçüsü Eşitlik (5) ile tanımlanabilir:

$$H_0 = - \sum_{i=0}^{t-1} \frac{p_i}{\omega_0} \ln \frac{p_i}{\omega_0}, \quad \omega_0 = \sum_{i=0}^{t-1} p_i \quad (5)$$

$$H_1 = - \sum_{i=t}^{L-1} \frac{p_i}{\omega_1} \ln \frac{p_i}{\omega_1}, \quad \omega_1 = \sum_{i=t}^{L-1} p_i$$

Eşitlik (6)'daki sınıf entropileri toplamı maksimum yapıldığında elde edilen eşik seviye değeri görüntü için en optimum eşik seviye değeri olacaktır.

$$t^* = \operatorname{argmax}(H_0 + H_1) \quad (6)$$

Çok seviyeli eşikleme için genişletilmiş Kapur Entropi ölçüsü [27] Eşitlik (7) 'de gösterildiği gibidir:

$$H_0 = - \sum_{i=0}^{t_1-1} \frac{p_i}{\omega_0} \ln \frac{p_i}{\omega_0}, \quad \omega_0 = \sum_{i=0}^{t_1-1} p_i$$

$$H_1 = - \sum_{i=t_1}^{t_2-1} \frac{p_i}{\omega_1} \ln \frac{p_i}{\omega_1}, \quad \omega_1 = \sum_{i=t_1}^{t_2-1} p_i$$

$$H_2 = - \sum_{i=t_2}^{t_3-1} \frac{p_i}{\omega_2} \ln \frac{p_i}{\omega_2}, \quad \omega_2 = \sum_{i=t_2}^{t_3-1} p_i \quad (7)$$

$$H_m = - \sum_{i=t_m}^{L-1} \frac{p_i}{\omega_m} \ln \frac{p_i}{\omega_m}, \quad \omega_m = \sum_{i=t_m}^{L-1} p_i$$

Eşitlik (8)'deki amaç fonksiyon değeri maksimize edilerek m optimum eşik değerleri elde edilir.

$$(t)^* = \operatorname{argmax} \sum_{i=0}^m H_i \quad (8)$$

1.3.2. Otsu Metodu

Otsu (between class variance) metodu [18] sınıflar arası varyans yöntemi olarak bilinen, parametrik olmayan bir bölütleme yöntemidir. Otsu yöntemi' nin amacı sınıflar arası varyans ölçüsünü maksimize ederken, her sınıftaki pikseller arasındaki sınıf içi varyansı ise minimize edebilmektir [24]. Otsu yöntemi sınıflar arası varyans ölçüsünü her sınıfın sigma fonksiyonlarının toplamı olarak Eşitlik (9)' daki gibi tanımlar:

$$f(t) = \sigma_0 + \sigma_1 \quad (9)$$

$$\sigma_0 = \omega_0(\mu_0 - \mu_T)^2, \quad \sigma_1 = \omega_1(\mu_1 - \mu_T)^2 \quad (10)$$

burada, μ_T orijinal görüntünün ortalama gri seviye değeridir. Her sınıf için ortalama gri seviye değeri (μ_i) Eşitlik (11) ile elde edilir.

$$\mu_0 = \sum_{i=0}^{t-1} \frac{ip_i}{\omega_0}, \quad \mu_1 = \sum_{i=t}^{L-1} \frac{ip_i}{\omega_1} \quad (11)$$

Optimum eşik seviye değeri Eşitlik (12) kullanılarak sınıflar arası varyansın maksimize edilmesiyle elde edilir.

$$t^* = \operatorname{argmax}(f(t)) \quad (12)$$

Sınıflara arası varyans yöntemi Eşitlik (13) kullanılarak çok seviyeli eşikleme için genişletilebilir [25]:

$$f(t) = \sum_{i=0}^m \sigma_i \quad (13)$$

Her sınıf için sigma fonksiyonları ve ortalama gri seviye değerleri sırasıyla Eşitlik (14) ve (15) ile hesaplanır:

$$\begin{aligned} \sigma_0 &= \omega_0(\mu_0 - \mu_T)^2 \\ \sigma_1 &= \omega_1(\mu_1 - \mu_T)^2 \\ \sigma_2 &= \omega_2(\mu_2 - \mu_T)^2 \\ \sigma_m &= \omega_m(\mu_m - \mu_T)^2 \end{aligned} \quad (14)$$

$$\mu_0 = \sum_{i=0}^{t_1-1} \frac{ip_i}{\omega_i}, \quad \mu_1 = \sum_{i=t_1}^{t_2-1} \frac{ip_i}{\omega_i}, \quad \mu_2 = \sum_{i=t_2}^{t_3-1} \frac{ip_i}{\omega_i}, \quad \mu_m = \sum_{i=t_m}^{L-1} \frac{ip_i}{\omega_i} \quad (15)$$

Eşitlik (16) ile sınıflar arası varyans maksimize edilerek çok seviyeli eşikleme için kullanılacak m optimum eşik seviyesi elde edilir.

$$(t)^* = \operatorname{argmax} \left(\sum_{i=0}^m \sigma_i \right) \quad (16)$$

1.4. Meta-Sezgisel Algoritmalar

“Sezgisel” kelimesi genel olarak “denemek”, “deneme yanılma yolu ile keşfetmek” anlamına gelir. Bu yöntemler ile bir optimizasyon problemi için çözümler makul bir sürede bulunabilir ancak en iyi çözümlerin bulunacağına garanti yoktur. Problemin karmaşık olması mümkün olabilecek tüm çözümlerin denenmesini imkansız kılar; bu sebeple amaç kabul edilebilir bir sürede en uygun çözümlere ulaşmaktır. “meta-“ ise “ötesinde” veya “üst düzey” anlamına gelir ve genellikle sezgisel aramalardan daha iyi performans gösterirler. Tüm meta-sezgisel algoritmalar rastgeleleştirme ve yerel arama kullanır. Rastgeleleştirme, yerel aramadan küresel ölçekte aramaya geçiş yapmayı sağlar. Bu nedenle, neredeyse tüm meta-sezgisel algoritmalar küresel çözümü bulmayı amaçlamaktadır.

Sezgisel algoritmaları ilk kullanan kişi Alan Turing olarak kabul edilebilir. II. Dünya Savaşı sırasında Alman gizli şifrelerini çözmek için “sezgisel arama” adını verdiği yöntem ile her seferinde başarıya ulaşmasa da muazzam sonuçlar elde etmiştir. 1960 ve 1970 yılları arasında evrimsel algoritmaların en önemli gelişmesi olan Genetik Algoritma (GA) John Holland tarafından geliştirilmiştir. meta-sezgisel algoritmaların en önemli gelişme evrelerinden biri de 1983 yılında S. Kirkpatrick ve arkadaşları tarafından, metalleri tavlama işleminden esinlenilerek geliştirilen benzetilmiş tavlama (Simulated Annealing - SA) algoritmasının geliştirilmesidir. Yüksek sıcaklıkta tahmini bir başlangıç çözümüyle başlayan ve kademeli olarak sistemi soğutan yörünge tabanlı bir algoritmadır. Yeni çözüm daha iyi ise kabul edilir, değilse sistemi yerel optimuma takılmadan kurtarması olasılığı ile kabul edilir. Sistem yeteri kadar yavaş bir şekilde soğutulunca optimum çözüme ulaşılabilir. Bu algoritmalar dışında; 1986’ da Fred Glover Tabu Araştırma (Tabu Search - TS), 1992’ de Dorigo ve arkadaşları tarafından Karınca Koloni Optimizasyonu (Ant

Colony Optimization - ACO), 1995 yılında Kennedy ve arkadaşları tarafından Parçacık Sürü Optimizasyonu (Particle Swarm Optimization - PSO), 1997 yılında Storn ve Price tarafından Diferansiyel Gelişim (Differential Evolution - DE), 2001 yılında Harmoni Arama (Harmony Search - HS), 2005 yılında Karaboğa tarafından Yapay Arı Koloni (Artificial Bee Colony - ABC), 2008 yılında Yang tarafından Ateşböceği Algoritması (Firefly Algorithm - FA) geliştirilmiştir ve bu algoritmalar halen geliştirilmekte, sayıları her geçen gün artmaktadır [28].

Bu tezde, literatürde sıkça kullanılan PSO [29], FA [30], ABC [31] ve GA [32] algoritmaları ile çok seviyeli eşikleme problemine henüz uygulanmamış ‘Simbiyotik Organizmalar Arama (Symbiotic Organisms Search - SOS)’ [33] optimizasyon algoritmasına yer verilmiştir.

1.4.1. Parçacık Sürü Optimizasyonu (Particle Swarm Optimization - PSO)

PSO algoritması belirli bir pozisyon değerine ulaşmak için bir bütün olarak hareket edebilen biyolojik canlıların sosyal davranışları örnek alınarak geliştirilen bir algoritmadır. Bir besin kaynağına ulaşmaya çalışan bir kuş sürüsü PSO algoritmasının örnek aldığı bir biyolojik canlı örneği olarak verilebilir. Bu arama davranışı gerçek değerli bir arama alanındaki lineer olmayan denklemlerin çözümlerini arama problemleri ile ilişkilendirilmiştir. Parçacıklar her zaman bireysel olarak buldukları en iyi çözüm ile kendi aralarında buldukları en iyi çözüme doğru yönelim halindedirler [34].

PSO algoritmasının temel adımları [35] aşağıdaki gibi sıralanabilir:

Adım 1: Her parçacık için başlangıç hız ve konum değerleri ile birlikte algoritma başlangıç parametrelerini ata.

Adım 2: Her parçacık için uygunluk fonksiyonunu değerlendir.

Adım 3: Yerel ($p(t)$) ve küresel ($g(t)$) en iyi çözümleri belirle.

Adım 4: Her parçacık için, Eşitlik (18) ve (19)’u kullanarak yeni hız ve pozisyon değerlerini güncelle.

Adım 5: Sonlanma şartını kontrol et, sağlanmıyorsa Adım 2’ den devam et.

Her parçacık için başlangıç hız değeri sıfır veya probleme göre rastgele belirlenmiş bir değer olarak alınabilir [36]. Başlangıç konumları Eşitlik (17) kullanılarak rastgele belirlenir.

$$x_{ij} = x_j^{min} + rand(0,1)(x_j^{max} - x_j^{min}) \quad (17)$$

burada, x_i vektörü i . parçacığın konumudur, $i = 1, \dots, N$, N : sürü sayısıdır. $j = 1, \dots, d$, d : problemin boyutudur. x_j^{min} ve x_j^{max} vektörleri her bir boyuttaki minimum ve maksimum değerleri temsil eder.

Algoritmada her parçacığın pozisyonu amaç (uygunluk) fonksiyonuna göre belirlenir. Popülasyonda her parçacık için bulunan en iyi çözüm (pozisyon) yerel çözüm, tüm parçacıkların içinde bulunan en iyi çözüm küresel çözümdür. Tüm parçacıklar için hız güncellemesi Eşitlik (18) ile hesaplanır:

$$v(t+1) = \omega v(t) + \phi_1 rand(0,1)(p(t) - x(t)) + \phi_2 rand(0,1)(g(t) - x(t)) \quad (18)$$

burada, $p(t)$ yerel en iyi çözüm, $g(t)$ küresel en iyi çözümdür. $x(t)$ güncel çözüm değeridir. $x(t)$ değerinin $p(t)$ yerel ve $g(t)$ küresel çözüm değerleri ile farkı ϕ_1 bilişsel ve ϕ_2 sosyal sabitleri ile ağırlıklandırılır. ω eylemsizlik ağırlığıdır ve önceki hızın $v(t)$, yeni hıza katkısını kontrol eder. Ayrıca, hız güncelleme işleminden sonra parçacığın hızının aşırı artmasını veya azalmasını engellemek, parçacığın hızının belirlenen sınırlar içerisinde kalmasını kontrol etmek için probleme göre bir V_{min} ve V_{max} değeri belirlenir. Parçacık için yeni çözüm değeri Eşitlik (19)'da gösterildiği gibi, parçacığın eski konumuna yeni hız değerinin eklenmesiyle hesaplanır [25, 36].

$$x(t+1) = x(t) + v(t+1) \quad (19)$$

1.4.2. Ateşböceği Algoritması (Firefly Algorithm - FA)

Ateşböceği algoritması ateşböceklerinin birbirlerinin parlaklık karakteristiklerine göre davranışlarını modelleyen bir algoritmadır. Algoritmada üç temel kural vardır: (1) Tüm ateşböcekleri tek cinstir. Bu nedenle cinsiyetten bağımsız olarak birbirlerine doğru hareket ederler. (2) Çekicilik parlaklıkla doğru orantılıdır, daha az parlak olan daha çok

parlak olana doğru hareket eder. İki ateşböceği arasındaki mesafe arttıkça çekicilik azalır. Kendinden daha parlak ateşböceği bulamayan ateşböceği rastgele hareket eder.(3) Bir ateşböceğinin çekiciliği amaç fonksiyon değeriyle belirlenir [28]. FA algoritmasının temel adımları ([28]'den uyarlanmıştır) aşağıdaki gibi sıralanabilir:

Adım 1: Başlangıç popülasyonunu oluştur.

Adım 2: Her ateşböceği için parlaklık değerini hesapla.

Adım 3: Daha az parlak olan ateşböceğini parlaklığı fazla olana hareket ettir.

Adım 4: Uzaklığa bağlı olarak çekicilikleri değiştir.

Adım 5: Yeni çözümleri değerlendir ve parlaklık değerlerini güncelle.

Adım 6: Ateşböceklerini sırala ve şimdiye kadarki en iyi çözümü bul.

Adım 7: Sonlanma şartı sağlanıncaya kadar Adım 3'e git.

Ateşböceklerinin çekiciliği Eşitlik (20) ile hesaplanır:

$$\beta = \beta_0 e^{-\gamma r^2} \quad (20)$$

Burada β_0 , i . ve j . ateşböcekleri arası uzaklık $r = 0$ olduğu durumdaki çekicilik değeri, γ ışık emilim katsayısıdır. Herhangi x_i ve x_j ateşböcekleri arasındaki r_{ij} uzaklığı Eşitlik (21) ile hesaplanır. Burada $k = 1, \dots, d$, d : problemin boyutudur.

$$r_{ij} = \|x_i - x_j\| = \sqrt{\sum_{k=1}^d (x_{i,k} - x_{j,k})^2} \quad (21)$$

i . ateşböceğinin çekiciliği daha yüksek olan j . ateşböceğine hareketi Eşitlik (22) ile hesaplanır. Burada $\alpha \in [0,1]$ arasında sabit bir sayı, $\epsilon_i = R - 1/2$ ve R , $[0,1]$ arasında rastgele bir sayıdır.

$$x_i = x_i + \beta(x_j - x_i) + \alpha\epsilon_i \quad (22)$$

1.4.3. Yapay Arı Koloni (Artificial Bee Colony - ABC)

Arı koloni algoritması bal arılarının yiyecek arama davranışları örnek alınarak geliştirilmiş bir algoritmadır. Algoritmada [25] her yiyecek kaynağı bir çözüme denk gelir. Her arı bir yiyecek kaynağı seçer ve bu kaynağın komşuluklarındaki çözümleri araştırır.

Üç çeşit arı vardır: işçi arılar, gözcü arılar ve kaşif arılar. Arılar yiyecek kaynağını bulma şekillerine göre sınıflandırılmışlardır. Bir işçi arı hafızasındaki kaynağın komşuluklarını araştırır ve daha iyi çözüm bulduğunda hafızasındaki çözümü günceller. Ayrıca araştırdığı kaynak sayısını da sayar. Gözcü arılar hafızasında bir kaynak tutmaz ve yararlı olma olasılığı olan bir yiyecek kaynağı seçer. Bu olasılık bilgisini işçi arılardan gelen bilgiye göre elde eder. Bir gözcü arı bir kaynak seçtiğinde kaynağın komşuluklarını araştırır. Daha iyi bir çözüm bulduğunda kaynağın konum bilgisini günceller. Bir kaynağın araştırılma sayısının bir ‘limit’ değerine ulaşması kaynağın tükendiğini gösterir. Kaynağın arısı kaynağı araştırmayı bırakır ve kaşif arı olur. Kaşif arı araştırmak için rastgele yeni bir kaynak seçer. Arı koloni algoritmasında işçi arı, gözcü arı ve kaşif arı aşamaları sırasıyla gerçekleştirilir. Algoritma adımları [37] genel olarak aşağıdaki gibi sıralanabilir:

Adım 1: Başlangıç kaynaklarını üret.

Adım 2: İşçi arıları kaynaklara gönder.

- a) kaynağın komşuluklarını belirle.
- b) nektar miktarlarını değerlendir.

Adım 3: Gözcü arılar ile işçi arılardan gelen bilgiye göre bir kaynak seç.

- a) kaynağın komşuluklarını belirle.
- b) nektar miktarlarını değerlendir.

Adım 4: Bırakılacak kaynakları belirle.

- a) kaşif arılar tarafından bulunan kaynaklar ile bu kaynakları değiştir.

Adım 5: Şimdiye kadar bulunan en iyi kaynağı hafızaya al.

Adım 6: Sonlanma şartı sağlanana kadar Adım 2 ye git.

Başlangıç aşamasında başlangıç kaynakları Eşitlik (17) ile belirlenir. Daha iyi bir kaynak bulunduğunda kaynağın pozisyonun güncellenme işlemi Eşitlik (23) ile sağlanır:

$$x'_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \quad (23)$$

Burada $k \in [1, N]$ rastgele seçilen komşu kaynak, N , kaynak sayısı veya sürü boyutu, $j = 1, \dots, d$, d : problemin boyutudur. ϕ_{ij} , $[-1,1]$ arasında x_{ij} mevcut kaynak ile x_{kj} komşu kaynak arasındaki farkı ağırlıklandıran rastgele bir sayıdır. Bu işlem sonucunda üretilen çözümün daha önceden belirlenmiş olan sınırları aşması durumunda çözüm alt veya üst sınır değerlerine Eşitlik (24)' teki gibi ötelenir [36]:

$$x'_{ij} = \left\{ \begin{array}{ll} x_j^{min} , & x'_{ij} < x_j^{min} \\ x'_{ij} , & x_j^{min} \leq x'_{ij} \leq x_j^{max} \\ x_j^{max} , & x'_{ij} > x_j^{max} \end{array} \right\} \quad (24)$$

İşçi arılar kaynaklardan elde ettiği bilgileri gözcü arılarla paylaşırlar. Bu işlem Eşitlik (25) ' te gösterildiği gibi, her kaynağa bir olasılık tabanlı normalize edilmiş uygunluk değeri atayarak sağlanır. Her gözcü arı rulet tekerleği yöntemine göre bir kaynak seçimi yapar.

$$p_i = \frac{fitness_i}{\sum_{i=1}^{SS} fitness_i} \quad (25)$$

'Limit' değerine ulaşmış kaynağın yerine yeni kaynak bulunması Eşitlik (17) ile kaşif arılar tarafından gerçekleştirilir [25].

1.4.4. Genetik Algoritma (Genetic Algorithm - GA)

Genetik algoritma doğadaki adaptasyon işlemini modelleyen evrim tabanlı bir algoritmadır. Orijinal genetik algoritmada ikili (binary) dizileri kullanarak karmaşık yapıları kodlar. Yapılar (kromozomlar, bireyler) çözülecek problem için çözümleri temsil eder. Başlangıç popülasyonu oluşturulduktan sonra tabii seleksiyon, çaprazlama ve mutasyon genetik operatörleri ile gelecek jenerasyondaki çözümler üretilir. Birbirini takip eden jenerasyonların geliştirilmesi ve değerlendirilmesi optimal çözüm bulununcaya kadar devam eder.

Genetik algoritmada başlangıç popülasyonu genellikle rastgele sayı üretilerek oluşturulur. Tabii seleksiyon (tekrar üreme) operatörü ile kalitesi yüksek bireylerin hayatta kalmaları ve sayılarının artması, kalitesi düşük bireylerin ise sayılarının azalarak kaybolması amaçlanır. Bu operatörün uygulanması için kullanılan en basit yöntem rulet tekerleği tekniğidir. Çaprazlama operatörü eski iki yapıdan yeni iki yapı meydana getirmek için kullanılır. En basit hali ile eski yapılar üzerinde rastgele birer nokta seçilerek kesilir, kesilen yapıların kuyrukları yeni iki yapı üretmek için birbirleri arasında yer değiştirilir. Çaprazlama işlemi iki iyi çözümden daha iyi iki yeni çözüm üretmek amacıyla gerçekleştirilir. Çaprazlama oranı ile çaprazlama operatörünün frekansı belirlenir. Mutasyon işlemi mutasyon oranına göre çözümlerdeki bit değerlerinin tersine

dönüştürülerek bireylerin değiştirilmesi aşamasıdır. Bu işlem ile yeni ve araştırılmamış çözüm elemanlarının bulunması amaçlanır [36]. Genetik algoritmanın temel adımları [38] aşağıdaki gibi sıralanabilir:

Adım 1: Rastgele oluşturulmuş N kromozomlu bir popülasyon ile başla.

Adım 2: Her kromozomun uygunluk değerini hesapla.

Adım 3: Sonlanma kriteri sağlanıncaya kadar aşağıdaki adımları tekrarla

- a) Popülasyondan bir çift ebeveyn kromozomu seç.
- b) İki yeni birey oluşturmak için çaprazlama oranına göre seçilen kromozomları rastgele seçilen bir noktadan çaprazla.
- c) Mutasyon oranına göre bireyleri mutasyonla değiştir.

Adım 4: Adım 2' ye git.

Bu çalışmada orijinal GA yerine gerçek kodlu GA kullanılmıştır. Gerçek kodlu GA sayısal fonksiyonların optimizasyonunda daha avantajlı hesaplama sunar. Kromozomlar '1' ve '0' şeklinde ikili dizilere dönüştürülmez, direkt gerçek rakamlarla işlem yapılır. Bu durum algoritmanın verimliliğini ve hızını artırır. Algoritmada çaprazlama yöntemi olarak 'Aritmetik Çaprazlama' kullanılmıştır. S_x ve S_y çözümleri için aritmetik çaprazlama eşitlik (26) ile gösterilmiştir.

$$yenis_x = as_y + (1 - a)s_x \quad (26)$$

$$yenis_y = as_x + (1 - a)s_y$$

Burada $\alpha \in [0,1]$ arasında rastgele bir sayıdır. Mutasyon, kromozomdaki genlerin belirlenen aralıkta rastgele yeni sayılar üretilerek değiştirilmesiyle sağlanmıştır [5].

1.4.5. Simbiyotik Organizmalar Arama (Symbiotic Organisms Search - SOS)

SOS, Min-Yuan Cheng ve Doddy Prayogo tarafından 2014 yılında geliştirilen meta-sezgisel bir algoritmadır. Ekosistemdeki organizmaların hayatta kalmak ve yayılmak amacıyla aralarında kurdukları simbiyotik etkileşimleri modellemektedir. Temel olarak üç ana aşamadan oluşur: (1) mutualizm fazı, (2) kommensalizm fazı ve (3) parazitizm fazı. Şekil 3' te algoritmanın temel adımları gösterilmiştir.

Şekil 3. SOS Algoritması temel adımları [33].

-
- 1: Başlat
 - 2: **repeat**
 - 3: mutualizm faz
 - 4: kommensalizm faz
 - 5: parazitizm faz
 - 6: **until** (sonlanma şartı sağlanana dek)
-

Mutualizm fazı: Bu aşama, karşılıklı olarak birbirlerine faydalı olan iki farklı tür arasındaki ilişkiyi modellemektedir. Örnek olarak arılar ve çiçekler arasındaki ilişki verilebilir. Arılar çiçeklerden bal için gereken nektarı toplarlar, bu süreçte polenleri dağıtırlar, böylece çiçeklerin üremesi için gerekli tozlaşmayı kolaylaştırırlar. Bu aşamada, X_i organizması kendisiyle etkileşmek üzere ekosistemden rastgele seçilen X_j üyesi ile eşleştirilir. X_i ve X_j için yeni çözümler Eşitlik (27) ve (28) 'deki gibi organizmalar arasındaki mutual ilişkiye göre hesaplanır.

$$X_{i_{new}} = X_i + rand(0,1) * (X_{best} - Mutual_vector * BF_1) \quad (27)$$

$$X_{j_{new}} = X_j + rand(0,1) * (X_{best} - Mutual_vector * BF_2) \quad (28)$$

$$Mutual_vector = \frac{X_i + X_j}{2} \quad (29)$$

burada, BF_1 ve BF_2 organizmaların birbirlerine sağladıkları yarar seviyesini gösteren 'yarar faktörleri' dir. Bu faktörler 1 veya 2 olmak üzere rastgele seçilirler. X_{best} adaptasyon için en iyi çözümdür. Eşitlik (29)' daki $Mutual_vector$ X_i ve X_j arasındaki ilişki karakteristiğini gösterir. Denklemlerdeki $(X_{best} - Mutual_vector * BF_1)$ kısım iki organizmanın hayatta kalabilmek için gösterdikleri çabayı temsil eder [33].

Kommensalizm fazı: Kommensalizm aşaması, biri yarar sağlarken diğerinin bu durumdan etkilenmediği iki farklı tür arasındaki ilişkiyi modeller. Remora balıkları ile köpekbalıkları arasındaki ilişki bir kommensal ilişkidir. Remora balığı kendini köpekbalığına yapıştırır ve köpek balığından kalan yemek artıklarını yer. Köpekbalığı bu durumdan zarar görmez veya yarar sağlamaz. X_i organizması ile X_j organizması arasındaki ilişkiye göre, yeni $X_{i_{new}}$ çözümü Eşitlik (30) ile hesaplanır:

$$X_{i_{new}} = X_i + rand(-1,1) * (X_{best} - X_j) \quad (30)$$

Denklemin $(X_{best} - X_j)$ kısmı X_j organizmasının X_i organizmasına hayatta kalabilmesi için sağladığı yararı temsil eder [33].

Parazitizm fazı: Bu fazda, biri ilişkiden fayda sağlarken, diğeri bu ilişkiden zarar görür. Örnek olarak, sıtma sivrisinekleri ile insan arasındaki ilişki gösterilebilir. Sivrisinek insan vücuduna yerleşir ve çoğalmaya başlar. Kişi sıtma hastalığına yakalanır ve bu ölümüne neden olabilir.

Algoritmada bu durum, ekosistemdeki X_i organizmasının kopyalanıp değiştirilmesiyle bir *Parasite_vector* elde edilerek gerçekleştirilir. X_i organizması ekosistemden rastgele seçilir ve *Parasite_vector* için bir konak görevi görür. Daha sonra her iki organizma değerlendirilir. *Parasite_vector* daha iyi uygunluk değerine sahipse, X_i organizmasını öldürür ve ekosistemdeki yerini alır. SOS algoritmasının adımları aşağıdaki gibi sıralanabilir [33]:

Adım 1: Ekosistem başlangıç parametrelerini ata

Adım 2: En iyi çözümü bul

Adım 3: Mutualizm fazını gerçekleştir:

- a) X_i organizması dışında rastgele bir X_j organizması seç
- b) *Mutual_vector* 'ü hesapla
- c) X_{inew} ve X_{jnew} yeni çözümlerini üret
- d) Yeni çözümler ve eski çözümler arasından uygunluk değeri en iyi olanları seç

Adım 4: Kommensalizm fazını gerçekleştir:

- a) X_i organizması dışında rastgele bir X_j organizması seç
- b) X_{inew} yeni çözümünü üret
- c) Yeni çözüm ve eski çözüm arasından uygunluk değeri en iyi olanı seç

Adım 5: Parazitizm fazını gerçekleştir:

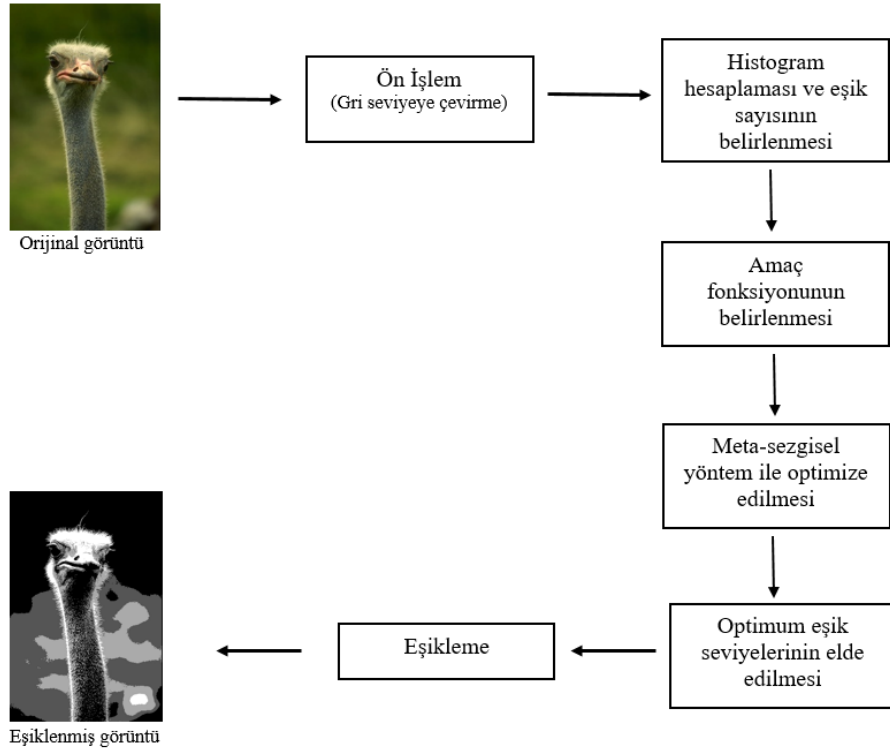
- a) X_i organizması dışında rastgele bir X_j organizması seç
- b) X_i organizmasını kullanarak *Parasite_vector* 'ü üret
- c) *Parasite_vector* ve X_j organizmasından uygunluğu en iyi olanı seç

Adım 6: Şimdiye kadarki uygunluğu en iyi olan organizmayı hafızaya al

Adım 7: Sonlanma kriteri sağlanana kadar Adım 3' e git.

1.5. Meta-Sezgisel Yaklaşımlara Dayalı İmge Bölütleme

Eşikleme işleminin çok seviyeye genişletilmesinde artan hesap yükünü azaltmak için birçok yöntem önerilmiştir. İlk çalışmalarda iteratif yöntemler önerilirken ilerleyen zamanlarda meta-sezgisel yöntemlere yönelim görülmüştür. meta-sezgisel yöntemlerde ‘amaç fonksiyonu’ optimize edilmesi istenen problemdir ve probleme bağlı olarak maksimize ya da minimize edilebilir. Bölütleme problemlerinde literatürde çeşitli entropi ölçüleri, Otsu yöntemi, Snake fonksiyonları gibi birçok yöntem amaç fonksiyonu olarak kullanılmış ve meta-sezgisel algoritmalar ile optimize edilmişlerdir. Bu çalışmada Kapur Entropi ölçüsü ve Otsu yaklaşımları amaç fonksiyonu olarak seçilmiştir. meta-sezgisel algoritmalar olarak PSO, FA, ABC, GA ve SOS algoritmaları kullanılmıştır. Gri seviyeye ($R*0.3+G*0.59+B*0.11$) çevrilmiş renkli görüntülerin histogram bilgileri algoritmalara girdi verisi olarak verilmiştir. Eşikleme işleminin kaç seviyeye kadar olacağı belirlendikten sonra amaç fonksiyonu optimize edilmiştir. Bu çalışmada 1, 2, 3, 4 ve 5 eşik seviyesine kadar görüntüler meta-sezgisel yöntemler ile eşiklenmiştir. Optimizasyon sonucunda elde edilen en iyi eşik seviye değerlerine göre çok seviyeli eşikleme işlemi gerçekleştirilmiştir. Meta-sezgisel yöntemler ile eşikleme işleminin temel adımları Şekil 4’ te gösterilmiştir:



Şekil 4. Meta- Sezgisel Yönteme Dayalı İmge Bölütleme

Literatürde Yin (1997) ve arkadaşları Otsu ve Kapur fonksiyonlarını rastgele bir eşik değeriyle başlamak üzere iteratif hesaplama yapan hızlı bir yöntem önermiştir [39]. Yöntemde fonksiyon, her adımda amaç fonksiyonunu iyileştirmeye çalışır ve ardışık iki amaç fonksiyon değeri arasında bir artış olmayana dek hesaplama devam eder. Hesaplama sonucunda optimum eşik seviyeleri elde edilir.

Yin (1999) amaç fonksiyonu Kapur ve Otsu fonksiyonlarına benzer GA ‘ ya dayalı hesaplama yapan hızlı bir yöntem önermiştir [40]. Yöntemde $[0, L-1]$ arasındaki eşik seviye çözümleri ikili diziler ile kodlanmıştır. Önerilen yöntemin tekdüzelik ölçütü (uniformity measure) ve işlemci süresine göre Gauss-yumuşama ve simetri/ikililik yöntemlerinden daha başarılı sonuç verdiği gözlenmiştir. Benzer yöntem Jinsong (1999) ve arkadaşları tarafından Kapur fonksiyonuna benzer bir fonksiyon kullanılarak çok seviye eşik değerleri elde edilmiştir [41].

Liao ve arkadaşları (2001) Otsu fonksiyonunu bir look-up tablosuna göre değiştirerek hızlı hesaplama yapan çok seviyeli ‘Fast-Otsu’ iteratif arama algoritmasını önermişlerdir [42].

Kim ve arkadaşları (2003) wavelet dönüşümünü kullanarak histogram boyutunu azaltmış ve azaltılan boyut üzerinden kapsamlı bir hesaplama ile Otsu fonksiyonunu optimize ederek uygun eşik değerlerini elde etmişlerdir [43].

Yang ve arkadaşları (2003) relatif entropi tabanlı GA gerçekleştirmiş ve Lena görüntüsünü üç eşik seviyesine kadar eşiklemişlerdir [44]. Standart Kapur ve Otsu fonksiyonlarından daha hızlı sonuç elde edildiği görülmüştür.

Du Feng ve arkadaşları (2005) kızılötesi görüntüleri bölütleme için 2D maksimum entropi tabanlı PSO ile tek eşik seviyesine dayalı bölütleme gerçekleştirmişlerdir [45]. Hata oranı metriğine göre başarılı sonuç elde edildiği kanıtlanmıştır.

Peng Yeng Yin (2006) minimum çapraz entropi tabanlı PSO algoritmasını geliştirmiş ve standart görüntüler üzerinde dört seviyeye kadar eşikleme işlemi gerçekleştirmiştir [46]. Kapsamlı aramalar ile işlemci süresi bakımından karşılaştırıldığında üstün başarı elde edildiği görülmüştür.

Hammouche ve arkadaşları (2007) Kim tarafından önerilen [43] yönteme benzer yöntemi, Yen tarafından geliştirilen ‘Automatic Thresholding Criterion’ (ATC) [47] fonksiyonuna dayalı GA ile gerçekleştirerek standart görüntüler için optimum eşik seviye değerlerine ulaşmışlardır [16]. Tekdüzelik ölçütüne (uniformity measure) göre sonuçların başarılı olduğu görülmüştür.

Madhubanti Maitra ve arkadaşları (2007) yüksek boyut probleminden kurtulmak ve erken yakınsamayı engellemek için kapsamlı-işbirlikçi hibrit öğrenme içeren Hybrid Cooperative- Comprehensive Learning PSO (HCOCLPSO) algoritmasını geliştirmişlerdir [48]. Kapur Entropisi tabanlı HCOCLPSO yöntemi ile standart görüntüler çok seviyeli olarak eşiklenmiştir. Sonuçlar amaç fonksiyonu ve tekdüzelik ölçütleri (uniformity measure) ile değerlendirilmiş önerilen yöntemin Kapur tabanlı Learning-GA (LGA), Gauss-yumuşama (gauss-smoothing) ve simetri/ikililik (symmetry/duality) metotlarından daha başarılı sonuç verdiği görülmüştür.

Li Cao ve arkadaşları (2007) GA' daki seçim, çaprazlama ve mutasyon aşamalarını bir öğrenme operatörü ile iyileştirerek öz öğrenmeli Self-Learning GA (SLGA) algoritmasını geliştirmişlerdir [6]. Algoritma Kapur ve Otsu tabanlı olarak standart görüntülerin bölütlenmesinde kullanılmıştır. Sonuçlar öğrenmeli GA, standart otsu ve Kapur tabanlı GA yöntemleri ile işlemci hızı ve yakınsama oranı ölçütleri üzerinden karşılaştırılmış ve önerilen yöntemin daha başarılı olduğu sonucuna varılmıştır.

Linyi Li ve arkadaşları (2008) fuzzy entropi tabanlı PSO algoritmasını geliştirmiştir. İşlemci süresi olarak sonuçların kapsamlı aramadan çok daha hızlı elde edildiği görülmüştür [49].

Hao Gao ve arkadaşları (2009) işbirlikçi-kuantum tabanlı Cooperative Quantum PSO algoritmasını (CQPSO) Otsu amaç fonksiyonu ile standart görüntüleri çok seviyeli eşiklemede kullanmışlardır [50]. CQPSO ile PSO' nun hızlı yakınsama oranı engellenmeye, işlemci süresi kısaltılmaya ve boyut problemi ortadan kaldırılmaya çalışılmıştır. Sonuçlar HCOCLPSO, Quantum PSO (QPSO), PSO, learning GA (LGA) ve ACO ile amaç fonksiyon değeri, standart sapma ve ortalama işlemci süresi metrikleri ile karşılaştırılmış ve CQPSO' nun üstün performans gösterdiği sonucuna varılmıştır.

Ming Huwi Horng (2009) Honey Bee Mating Optimization (HBMO) algoritması ile maksimum entropi ölçüsünü maksimize ederek gri seviyeli standart görüntülerin beş eşik seviyesine kadar bölütleme işlemini gerçekleştirmiştir [51]. Algoritmanın performansını değerlendirmek için sonuçlar PSO, HCOCLPSO ve Fast Otsu metotları ile hesaplama süresi, PSNR (Peak Signal-to-Noise Ratio) ve uygunluk değeri metrikleri açısından karşılaştırılmıştır. Önerilen algoritmanın yakınsama hızının karşılaştırılan algoritmalarından daha hızlı olduğu kanıtlanmıştır. Ayrıca diğer metriklerde de başarılı sonuçlar elde edilmiştir.

Akhilesh Chander ve arkadaşları (2010) PSO algoritmasındaki hız güncelleme fonksiyonundaki momentum bileşenlerini parçacığın hareketine göre adaptif olarak değişmesini sağlayan bir yöntem önermiştir [52]. Yöntemde, parçacığın daha başarılı sonuca yöneldiği bir adımda parçacığın hız güncelleme fonksiyonundaki ‘momentum’ kısmı, parçacığın daha başarısız bir çözüme yöneldiği bir adımda ise ‘sosyal’ kısmı daha fazla w ağırlık katsayısı verilerek adaptif olarak çözüme ulaşılır. Bu algoritma Otsu tabanlı olarak standart görüntülerin bölütlenmesinde kullanılmıştır. Sonuçlar Gauss-yumuşama, simetri/ikililik ve GA yöntemleri ile işlemci süresi ve tekdüzelik ölçütüne (uniformity measure) göre karşılaştırılmıştır. Önerilen yöntemin daha üstün performans gösterdiği ispatlanmıştır.

Ming Huwi Horng (2011) maksimum entropi tabanlı ABC algoritmasını geliştirmiş ve standart görüntüleri çok seviyeli eşiklemiştir [53]. Sonuçlar HCOCLPSO, ‘Fast-otsu’ metodu ve Honey Bee Mating Optimization (HBMO) yöntemleri ile amaç fonksiyon değeri, Peak Signal Noise Ratio (PSNR), hesaplama süresi ölçütlerine dayalı karşılaştırılmış ve önerilen yöntemin başarılı sonuç verdiği gözlemlenmiştir.

Ming Huwi Horng ve arkadaşları (2011) minimum entropi tabanlı FA algoritmasını geliştirmiştir [17]. Standart görüntülerin çok seviyeli eşiklemesi minimum entropi tabanlı PSO, QPSO, HBMO algoritmaları ile de gerçekleştirilmiş, hazırlanan yöntemin hesaplama süresi, PSNR ve amaç fonksiyon değeri metriklerine göre daha başarılı olduğu görülmüştür.

Kezong Tang ve arkadaşları (2011) minimum çapraz entropi (minimum cross entropy) tabanlı gerçek kodlu GA algoritmasını geliştirmişlerdir [54]. Önerilen algoritma ile standart görüntüler dört eşik seviyesine kadar bölütlenmiştir. Sonuçlar tekdüzelik ölçütü (uniformity measure) ve Cpu işlem süresine göre kapsamlı minimum çapraz entropi yöntemi ile karşılaştırılmıştır. Tekdüzelik ölçütüne göre algoritma sonuçlarının birbirine çok yakın olduğu, ancak önerilen yöntemin işlem süresinin çok daha kısa olduğu gözlenmiştir.

Bahriye Akay (2012) Kapur Entropisi ve Otsu tabanlı gerçekleştirdiği ABC ve PSO algoritmalarının performanslarını Berkeley Veri Tabanı’ ndaki en bilinen görüntüleri kullanarak karşılaştırmıştır [25]. Performansları karşılaştırmak için amaç fonksiyon değerleri, PSNR, SSIM (Structural Similarity Index Measure), işlemci süresi ve bu değerlerden bazılarının ortalama ve standart sapma değerleri metrikleri kullanılmıştır. Sonuç olarak Kapur tabanlı ABC algoritmasının çoklu eşiklemede iyi sonuç verdiği ve 2

eşik değerine kadar tüm algoritmaların aynı eşik seviye değerlerini verdiği sonucuna varılmıştır.

Pedram Ghamisi ve arkadaşları (2012) Fractional-Order Darwinian Particle Swarm Optimization (FODPSO) ve Darwinian Particle Swarm Optimization (DPSO) algoritmalarını Otsu tabanlı olarak standart görüntülerin bölütlenmesinde kullanmışlardır [55]. DPSO, yerel optimum değerine denk gelince arama uzayını değiştirme, parçacığı silme veya sürüyü silme işlevlerini içerir. FODPSO ise yakınsama oranını kontrol etmek için ‘fractional’ hesabı kullanır. İki algoritma amaç fonksiyon değeri, standart sapma ve işlemci süresi açısından karşılaştırılmıştır. Sonuç olarak FODPSO algoritması daha üstün gelmiştir.

Ashish Kumar Bhandari ve arkadaşları (2013) çok seviyeli eşikleme işlemini gerçekleştirmek için Cuckoo Search (CS), Wind Driven Optimization (WDO) ve Egg Laying Radius (ELR)- CS optimasyon algoritmalarında amaç fonksiyonu olarak Kapur Entropi ölçüsünü kullanmışlardır [26]. Geliştirilen algoritmalar ile uydu görüntülerinin bölütlemesi gerçekleştirilmiştir. Çalışma sonucunda CS algoritmasının optimum eşik seviyesi bulma ve kenar kaybetmeme özellikleri bakımından diğer algoritmalarından daha performanslı olduğu sonucuna varılmıştır. Ayrıca WDO algoritmasının hesaplama karmaşıklığının diğer algoritmalarından daha az olduğu ispatlanmıştır.

Manikandan ve arkadaşları (2013) Kapur Entropisi tabanlı gerçek kodlu genetik algoritmayı geliştirerek MR görüntülerini bölütlemeye kullanmışlardır [56]. Çalışmada çaprazlama işlemi için simulated binary crossover (SBX) kullanılmıştır. Sonuçlar Adaptive Bacterial Foraging (ABF), Bacterial Foraging (BF) ve Nelder–Mead simplex metot algoritmaları ile amaç fonksiyon değeri, standart sapma ve tekdüzelik (uniformity measure) ölçütleri kullanılarak karşılaştırılmış hazırlanan yöntem üstün performans göstermiştir.

Bhandari ve arkadaşları (2014) uydu görüntülerini bölütlemek amacıyla değiştirilmiş Modified ABC algoritmasını geliştirmişlerdir [57]. Algoritmada başlangıç değerleri hem kaotik sistem hem de karşılaştırma tabanlı öğrenme kullanılarak atanmıştır ve arama aşamasında yalnızca bir önceki iterasyonun en iyi değeri etrafında araştırma yapılmıştır. Algoritma Otsu, Kapur ve Tsallis tabanlı olarak uydu görüntülerini bölütlemeye uygulanmış, karşılaştırmak için Otsu, Kapur ve Tsallis tabanlı PSO, ABC ve GA algoritmaları kullanılmıştır. Mean Square Error (MSE), SD (Standard Deviation), Feature Similarity Index (FSIM), SSIM ve PSNR metriklerine göre Otsu, Kapur ve Tsallis tabanlı

Modified ABC algoritması çok seviyeli eşikleme için diğer yöntemlere göre üstün başarı göstermiştir.

Yi Liu ve arkadaşları (2014) Adaptive Inertia (AI) ve Adaptive Population (AP) kullanarak Modified PSO elde etmişlerdir [58]. AI aramanın durumuna göre durağanlık ağırlığını (w) değiştirir. Örnek olarak; küresel en iyi değer k adım boyunca hiç değişmemiş ise ağırlık 0.1 azaltılır değişmiş ise 0.1 artırılır. AP ise popülasyon boyutunu değiştirerek yerel optimuma takılmayı önler. Algoritma Otsu tabanlı olarak standart görüntülerin bölütlenmesinde kullanılmıştır. Sonuçlar Global PSO ve standart GA ile Benchmark fonksiyonları kullanılarak 12 standart görüntüde karşılaştırılmış ve hazırlanan yöntemin üstün performans gösterdiği kanıtlanmıştır.

Tahereh Hassanzadeh ve arkadaşları (2014) Standart FA algoritmasındaki ateşböceğinin yerel optimum çözümlere göre hareket etmesi yerine küresel çözüme göre hareket etmesini amaçlayarak Fuzzy FA geliştirmişlerdir [15]. FFA' ya göre ateşböceği en iyi k adet ateşböceğini göz önüne alarak fuzzy denklemine göre hareket eder. Sonuçlar standart FA ile karşılaştırılmış ve Otsu tabanlı FFA' nın Benchmark fonksiyonlarının daha iyi sonuç verdiği görülmüştür. FFA algoritması otsu tabanlı olarak standart görüntülerin bölütlenmesinde kullanılmış; amaç fonksiyon değeri, tekdüzelik (uniformity measure) ölçütü metriklerine göre HCOCLPSO, LGA-Kapur, FA-Otsu algoritmalarından daha üstün performans gösterdiği sonucuna varılmıştır.

Tuba Kurban ve arkadaşları (2014) evrimsel tabanlı meta-sezgisel algoritmalar ile sürü tabanlı meta-sezgisel algoritmaların görüntülerin çok seviyeli eşikleme işlemlerinde başarımlarını ölçmek için performans karşılaştırması yapmışlardır [59]. Çalışmada evrimsel tabanlı algoritmalar Evolution Strategy (ES), GA, DE, Adaptive Differential Evolution (JADE); sürü tabanlı meta-sezgisel algoritmalar PSO, ABC, CS ve DE algoritmaları Kapur Entropisi tabanlı olarak renkli görüntülerin çok seviyeli eşiklenmesinde kullanılmıştır. Uygunluk değeri, SSIM, FSIM ve Cpu işlem süresi metrikleri ile yapılan ölçümlerde sürü tabanlı algoritmaların çok seviyeli eşikleme için daha tutarlı, evrim tabanlı algoritmaların ise işlem süresinin daha kısa olduğu ispatlanmıştır.

Liang Huang ve arkadaşları (2015) farklı yıllarda iki uzaktan kontrol görüntüsü alınmış bir yerin fark görüntüsüne 2D Otsu tabanlı FA uygulayarak bölütlemişlerdir [60]. Sonuçlar 2d maximum entropy ve 2d maximum fuzzy entropy sonuçları ile hata oranı

metriğine göre karşılaştırılmış 2d otsu tabanlı daha yüksek doğruluk oranına sahip olduğu görülmüştür.

Bhandari ve arkadaşları (2015) çok seviyeli eşikleme problemi için Tsallis entropi tabanlı CS algoritmasını geliştirmişlerdir [61]. Geliştirilen algoritma renkli uydu görüntülerini bölütlemek için kullanılmıştır. Sonuçların başarımını ölçmek için önerilen algoritma Tsallis entropi tabanlı DE, WDO, PSO ve ABC algoritmaları ile uygunluk değeri, Cpu işlem süresi, MSE, SSIM, FSIM ölçütleri bakımından karşılaştırılmıştır. Elde edilen sonuçlara göre algoritmaların kararlılık sıralaması $CS > DE > ABC > PSO > WDO$; işlem süresi sıralaması $DE < ABC < CS < PSO < WDO$ şeklinde elde edilmiştir.

Pare ve arkadaşları (2015) uydu görüntülerini bölütlemek için Kapur, Otsu ve Tsallis tabanlı GA algoritmalarını uygulamışlardır [5]. Kapur tabanlı GA algoritmasının PSNR, MSE, SSIM, FSIM metrikleri ve işlemci süresi ölçütü ile daha üstün performans gösterdiği kanıtlanmıştır.

Kai Chen ve arkadaşları (2016) FA algoritmasına çeşitliliği artırmak için Cauchy mutasyon eklemiş, algoritmanın küresel arama yeteneğini arttırmak ve yaklaşmayı hızlandırmak için komşuluk aramada iyileştirmeler yaparak ImprovedFA algoritmasını elde etmişlerdir [62]. Algoritma Otsu tabanlı olarak standart görüntülerin bölütlenmesinde kullanılmıştır. Sonuçlar Hybrid Differential Evolution Optimization (HDE), DPSO ve standart FA ile işlemci süresi, ortalama ve standart sapma ölçütleri ile karşılaştırılmış ve yeni yöntemin üstün performansı kanıtlanmıştır.

Abdel-Khalek ve arkadaşları (2016) çok seviyeli eşikleme için GA ile Tsallis ve Renyi entropilerini birlikte kullanarak farklı beyin MR görüntülerinin ve standart birkaç görüntünün bölütlenmesini gerçekleştirmişlerdir [38]. İşlem için girdi olarak 2D histogram bilgisi kullanılmıştır. Renyi entropi tabanlı GA algoritmasını PSNR ölçütü bakımından Tsallis tabanlı GA'dan daha başarılı sonuç verdiği ispatlanmıştır.

Pare ve arkadaşları (2016) renkli görüntülerin çok seviyeli bölütleme problemi için Kapur Entropisi, Tsallis Entropi ve Otsu metodlarını CS ve ELR-CS algoritmalarında amaç fonksiyonu olarak kullanmışlardır [63]. Algoritmalara girdi verisi olarak görüntü histogramı yerine görüntü enerji eğrisi kullanılmıştır. Standart test görüntüleri karşılaştırma için kullanılmış, Cpu işlem süresi, PSNR, MSE, SSIM, FSIM ölçütleri bakımından enerji eğrisi tabanlı CS-Kapur yönteminin bölütleme işleminde en üstün performansı sağladığı ispatlanmıştır.

Pare ve arkadaşlarının (2017) bir sonraki çalışmasında minimum çapraz entropi (minimum cross entropy) tabanlı CS algoritması standart renkli görüntülerin çok seviyeli eşiklemesinde kullanılmıştır [64]. Ayrıca CS, ABC, DE, WDO ve Bacterial Foraging Optimization (BFO) algoritmaları between class variance (Otsu) ve Tsallis entropi tabanlı olarak da gerçekleştirilerek performans karşılaştırması için kullanılmışlardır. Performans değerlendirmesi için, PSNR, MSE, SSIM, FSIM, Cpu işlem süresi, amaç fonksiyon değeri, ortalama ve standart sapma metrikleri kullanılmıştır. Çalışma sonucunda minimum çapraz entropi tabanlı CS algoritması en başarılı algoritma olarak bulunmuştur.

Lifang He ve arkadaşları (2017) FA algoritmasının başlangıç aşamasında Chaotic Map kullanarak çeşitliliği artırılmış ModifiedFA algoritmasını elde etmişlerdir [3]. Geliştirilen algoritma Kapur, Otsu ve minimum kross entropi olarak üç farklı tabanda renkli görüntülerin bölütlenmesinde kullanılmıştır. Sonuçlar, Brownian Search Firefly (BFA) ve Lévy Search Firefly (LFA) algoritmalarının Kapur, Otsu ve minimum kross entropi tabanlı versiyonlarının PSNR ve SSIM değerleri ile karşılaştırılmıştır. Sonuç olarak ModifiedFA algoritmasının daha üstün performans gösterdiği kanıtlanmıştır. Ayrıca Kapur tabanlı ModifiedFA gerçekleştiriminin Otsu ve minimum kross entropi tabanlı ModifiedFA versiyonlarından daha performanslı olduğu görülmüştür.

Abdul Kayom ve arkadaşları (2017) gri kurtların sosyal davranışlarını ve avlanma şekillerini örnek olarak geliştirilen Grey Wolf Optimizasyon (GWO) algoritmasını Kapur ve Otsu fonksiyonları ile birlikte kullanarak standart gri seviyeli görüntülerin çok seviyeli bölütlenmesi problemine uygulamışlardır [65]. Sonuçların kalitesini ölçmek için ortalama SSIM ve amaç fonksiyon değerleri kullanılmıştır. PSO ve BFO algoritmaları ile sonuçları karşılaştırılan önerilen algoritmanın daha yüksek kalite değerlerine sahip olduğu görülmüştür. Ayrıca BFO algoritmasından daha hızlı ancak PSO algoritmasından daha yavaş sonuç ürettiği de bulunan sonuçlar arasındadır.

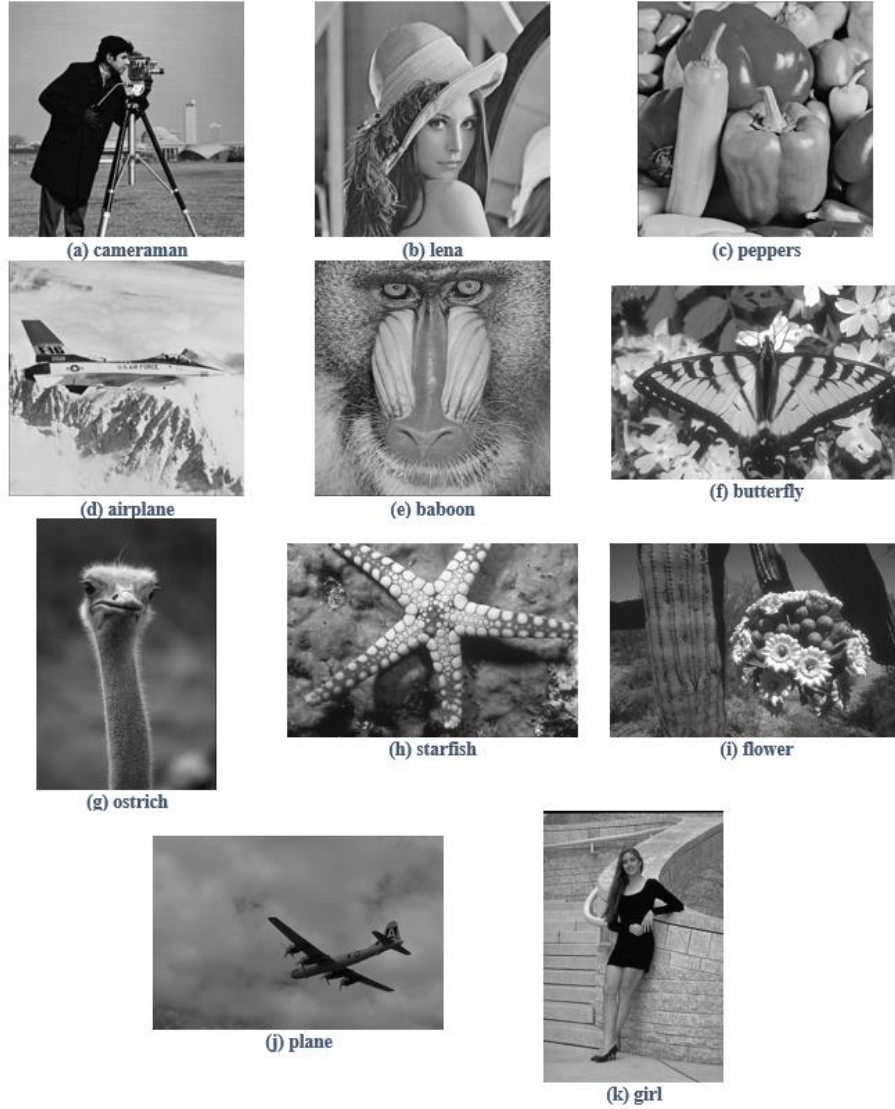
Mohamed Abd El Aziz ve arkadaşları (2017) Whale Optimization Algoritması (WOA) ve Moth-Flame Optimization (MFO) algoritmalarını kullanarak gri seviyeli görüntüleri beş eşik seviyesine kadar bölütlemiştir [66]. WOA algoritması kambur balinaların avlanırken gösterdikleri işbirlikçi davranışları örnek alınarak geliştirilmiş bir algoritmadır. MFO algoritması ise güvelerin ay ışığından yararlanarak geceleri yön bulma davranışlarından esinlenilerek geliştirilmiştir. Bu iki algoritma Otsu amaç fonksiyonu ile birlikte kullanılmış ve sonuçlar Sine Cosine Algorithm (SCA), HS, Social Spider Optimization (SSO), FASSO ve FA algoritmaları ile karşılaştırılmıştır. Elde edilen

sonularda nerilen iki yntemin de dięer sr algoritmalarından daha stn performans gsterdięi ayrıca MFO algoritmasının WOA algoritmasından daha bařarılı sonulara sahip olduęu kanıtlanmıřtır.

Pare ve arkadaşları (2018) komřu entropi llerinin farkını hesaba katan deęiřtirilmiř bir fuzzy entropisi nermiřlerdir [67]. Ayrıca FA algoritmasına Levy flight arama stratejisi ekleyerek rastgelelik oranını artırmıřlardır. Geliřtirilen fuzzy entropi ls tabanlı Levy flight FA algoritması standart renkli grntlerin bltlenmesinde kullanılmıřtır. Sonular Adaptive Differential Evolution (JADE), PSO, Cuckoo Search (CS) ve ABC algoritmalarının ama fonksiyon deęeri, SSIM, FSIM, PSNR, MSE ve iřlemci sresi deęerleri ile karřılařtırılmıř nerilen yntemin stn performans gsterdięi kanıtlanmıřtır.

2. YAPILAN ÇALIŞMALAR, BULGULAR VE İRDELEME

Bu çalışmada çok seviyeli eşiklemeye ilişkin literatürde var olan meta-sezgisel yöntemlerden birkaçı incelenmiş ve Şekil 5' te gösterilen USC-SIPI (URL-1, 2019) ile Berkeley [68] veri setlerinden alınan on bir görüntü üzerinde başarımları değerlendirilmiştir. Ayrıca yakın zamanda geliştirilmiş ve henüz bölütleme problemine uygulanmamış bir meta-sezgisel algoritma olan SOS algoritması çok seviyeli eşikleme problemi için uygulanmıştır. Algoritmalarda amaç fonksiyonu olarak Kapur Entropisi ve Otsu yaklaşımları kullanılmış ve optimize edilmişlerdir.



Şekil 5. Performans değerlendirmeleri için kullanılan test görüntüleri

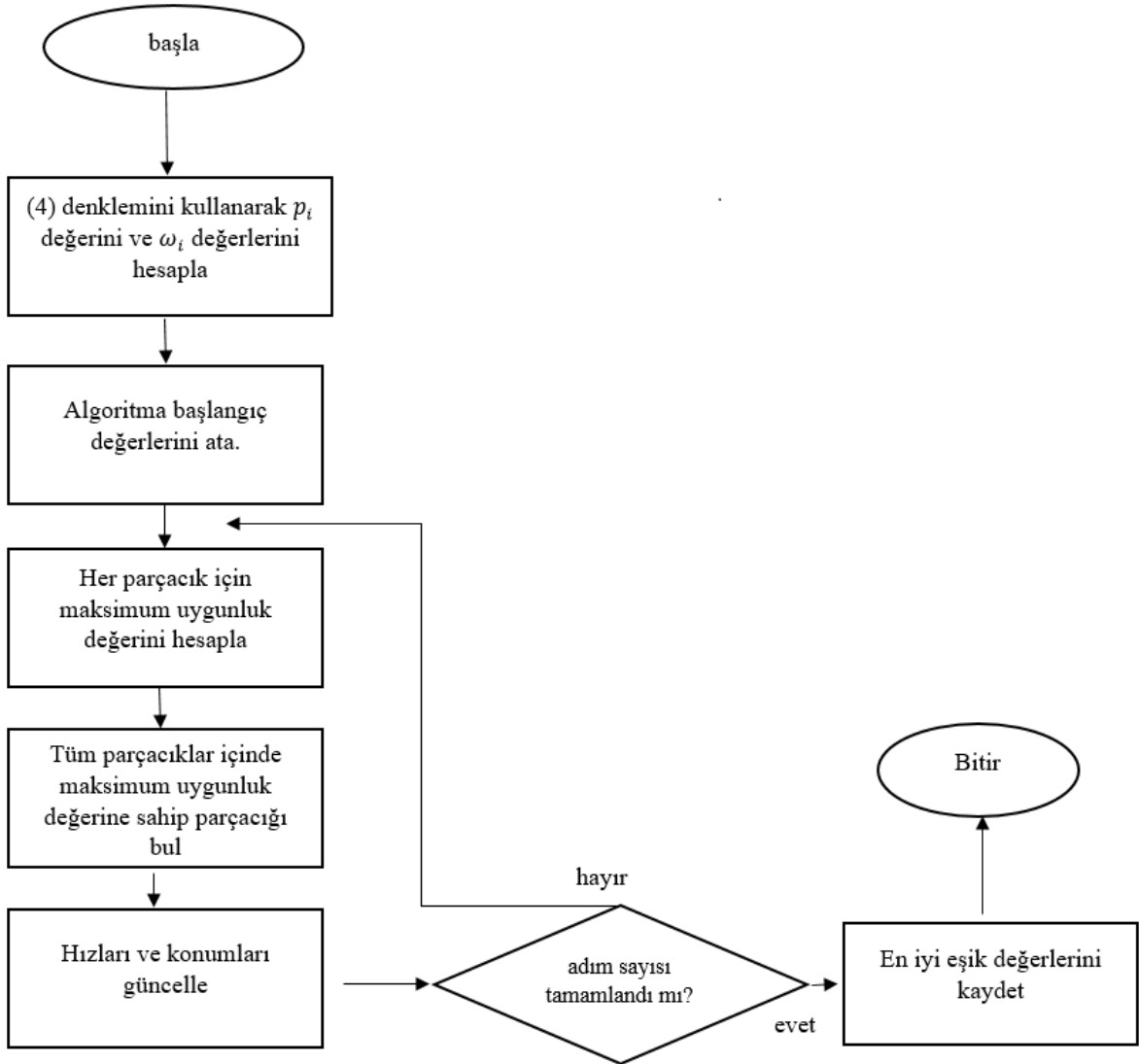
İlk çalışma olarak PSO algoritması, sonrasında FA, ABC ve GA algoritmaları Kapur Entropisi ve Otsu (between class variance) tabanlı olarak gerçekleştirilmiştir. PSO ve FA algoritmalarının ABC ve GA algoritmalarından bazı değerlendirme metriklerine göre daha başarılı sonuç verdikleri görülmüştür. Son olarak SOS algoritması Kapur Entropisi ve Otsu tabanlı olarak gerçekleştirilmiş, bazı görüntülerde PSO ve FA algoritmaları ile eşit, bazı görüntülerde ise daha başarılı performansa sahip olduğu görülmüştür.

Çalışmada algoritmaların başlangıç popülasyonunu oluşturma işlemi rastgele sayılar üretilerek gerçekleştirilmiştir. Bu sayılar gri seviyeli görüntülerin histogram bilgisindeki minimum ve maksimum değerler olan [0,255] değerleri arasında üretilen sayılardır. Bir gri seviye değerinin olasılığı tüm gri seviye değerlerinin toplamına bölünmesiyle Eşitlik (4) ile elde edilmiştir. Bu olasılık değerlerinin kümülatif toplamları ile ω_m değerleri elde edilmiştir. Bu değerler kullanılarak Kapur ve Otsu fonksiyonları hesaplanmıştır. Tüm görüntüler $m=1, 2, 3, 4$ ve 5 (m : eşik sayısı) eşik seviyesine kadar bölütlenmiştir. PSO algoritması için parçacık sayısı, FA algoritması için ateşböceği sayısı, ABC algoritması için sürü sayısı, GA algoritması için kromozom sayısı, SOS için ekosistem boyutu seçimi, eşiklenilmesi istenilen eşik sayısı değerinin 10 katı olarak belirlenmiştir (Örn; eşik sayısı:2, popülasyon: $2*10$). Literatürde 100 adım sayısı bölütleme problemleri için sonuca ulaşmada yeterli görülmüştür [25]. Bu sebeple çalışmada tüm algoritmalar için adım sayısı 100 olarak alınmıştır.

2.1. PSO Tabanlı İmge Bölütleme

Literatürde sıkça PSO algoritması ile çeşitli amaç fonksiyonlarına dayalı olarak imge bölütlemesi gerçekleştirilmiştir. Bu çalışmada algoritmanın amaç fonksiyonu, ilk olarak Kapur Entropisi ikinci olarak Otsu fonksiyonu seçilmiş ve en optimum eşik değerleri elde edilmiştir. Son olarak elde edilen eşik değerlerine göre imge bölütlemesi gerçekleştirilmiştir. Algoritmada parçacık değerleri gri seviye değerlerine denk gelmektedir. Eşikleme işlemi için algoritma başlangıç parametreleri atandıktan sonra başlangıç popülasyonu rastgele oluşturulmuştur. Tüm parçacıklar için amaç (uygunluk) fonksiyonu Kapur tabanlı eşikleme için Eşitlik (8), Otsu tabanlı eşikleme için Eşitlik (16) kullanılarak her adımda hesaplanmıştır. Parçacıklar için her adımda hız güncellemesi Eşitlik (18), yeni çözümler Eşitlik (19) ile hesaplanmıştır. Tüm hesaplama boyunca her parçacık için en iyi çözüm değeri ile tüm parçacıklar içinde en iyi çözüm değerine sahip

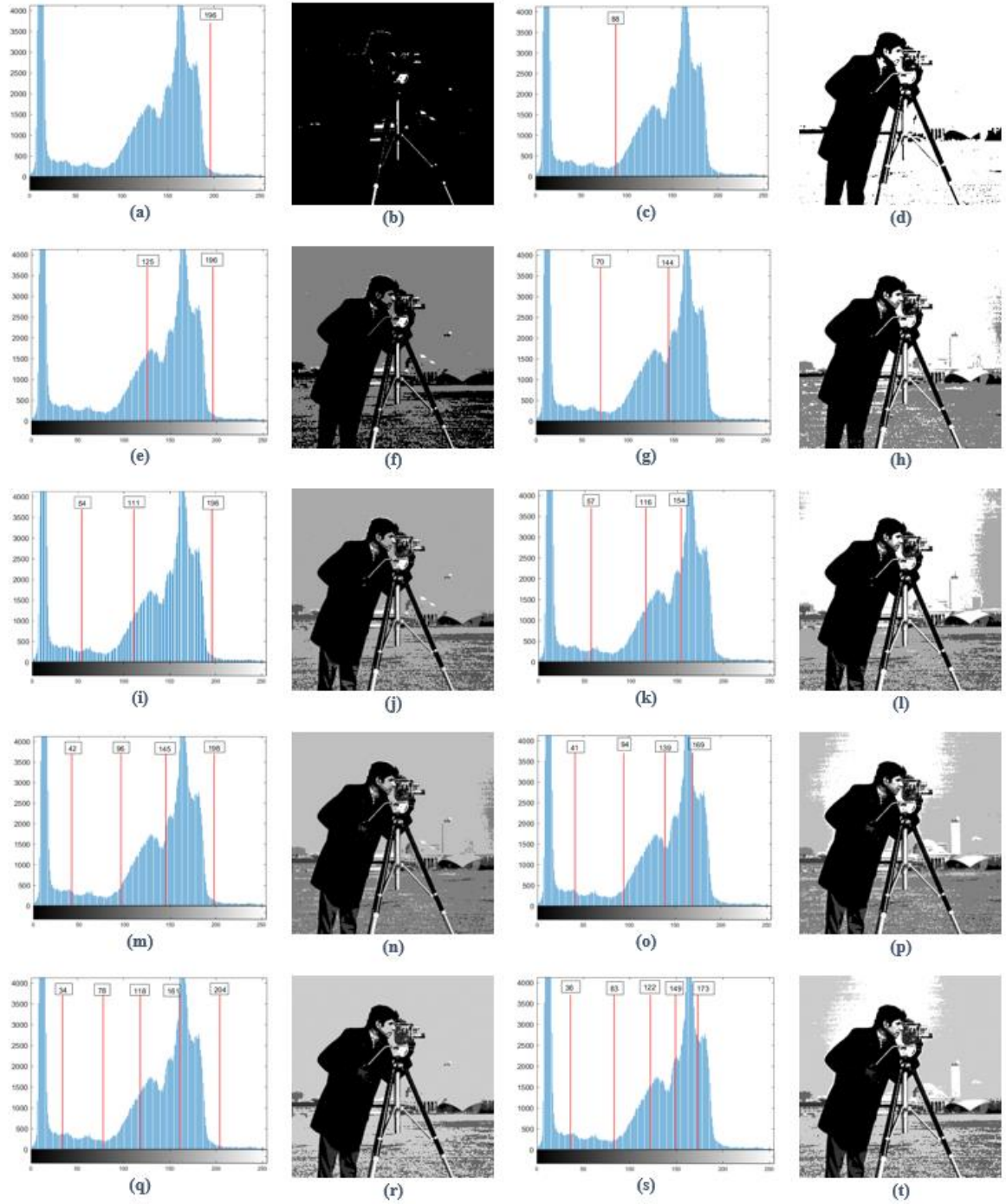
parçacık ayrı ayrı hesaplanmıştır. Hesaplama işlemi bittiğinde tüm parçacıklar içerisinde en iyi uygunluk (amaç) değerine sahip parçacık en optimum eşik değerlerini barındıran parçacıktır. PSO algoritmasındaki parametrelerin seçimi amaç fonksiyon değerine göre belirlenmiştir. Amaç fonksiyon değerinin en yüksek olmasını sağlayan parametreler en uygun değerler olarak alınmıştır. Bu çalışmada başlangıç hızı 0 (sıfır), ϕ_1 ve ϕ_2 değerleri 2, w değeri 0.6, V_{max} ve V_{min} sırasıyla 4 ve -4 olarak belirlenmiştir. PSO tabanlı imge bölütleme adımları [46] Şekil 6' da gösterilmiştir.



Şekil 6. PSO tabanlı imge bölütleme akış şeması

Şekil 7' de elde edilen eşiklere göre bölütlenmiş 'Cameraman' sonuç görüntüleri verilmiştir. Şekilde birinci sütun Kapur Entropisi ile sırasıyla $m=1, 2, 3, 4$ ve 5 için

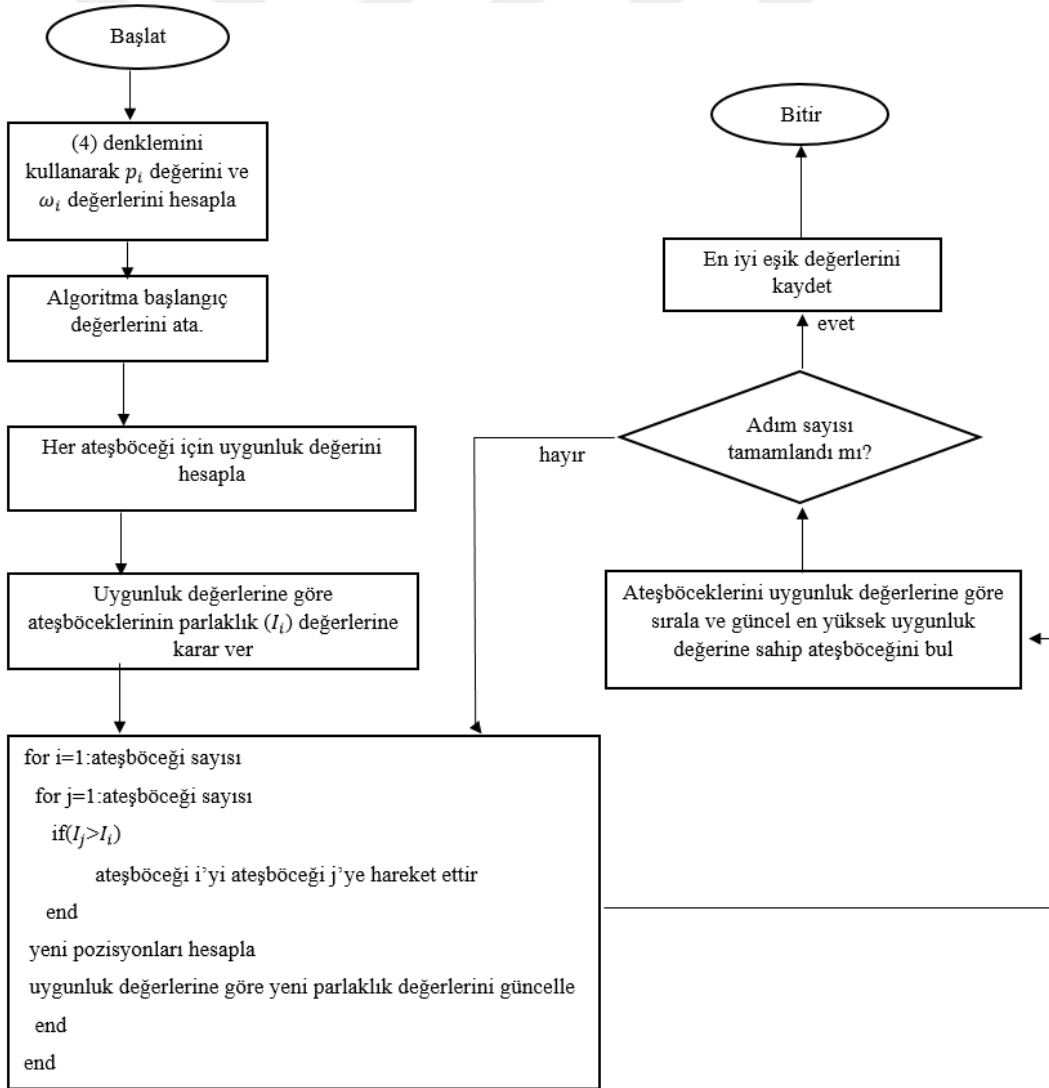
bulunan eşik seviye değerlerini, ikinci sütun bu eşik seviyeleri ile eşiklenmiş sonuç görüntülerini; üçüncü sütun Otsu yöntemi ile bulunan eşik seviye değerlerini, dördüncü sütun bu eşik seviyeleri ile bölütlenmiş sonuç görüntülerini göstermektedir.



Şekil 7. Kapur Entropi ölçüsü ve Otsu tabanlı gerçekleştirilmiş PSO algoritması ile $m= 1, 2, 3, 4$ ve 5 için elde edilen eşik değerleri ve bu değerler ile bölütlenmiş görüntüler.

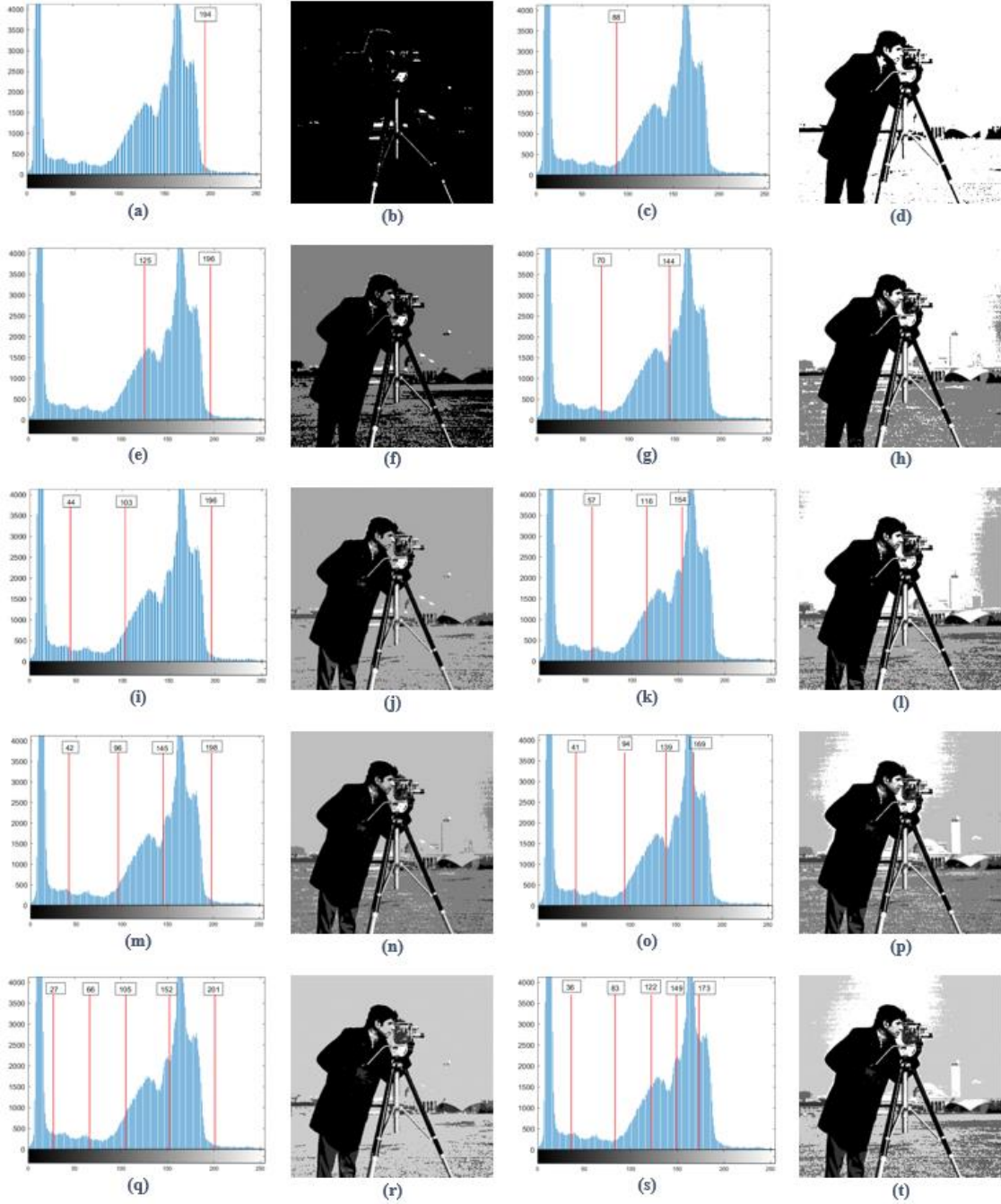
2.2. FA Tabanlı İmge Bölütleme

İmge bölütleme problemleri için farklı amaç fonksiyonlarına dayalı FA algoritması literatürde sıkça uygulanmıştır. Algoritmadaki her ateşböceği gri seviye değerine denk gelmektedir. Algoritma parametreleri atandıktan ve başlangıç popülasyonu rastgele oluşturulduktan sonra her ateşböceği için parlaklık (uygunluk) değeri hesaplanır. Uygunluk (amaç) değeri ilk bölütleme işleminde Kapur Entropisi ikinci bölütleme işleminde Otsu fonksiyonu seçilerek hesaplanmıştır. Her adımda uygunluk (çekicilik) değeri daha düşük olan ateşböcekleri uygunluk değeri daha yüksek olan ateşböceğine hareket ettirilerek belirlenen amaç fonksiyonu optimize edilmiştir. FA tabanlı çok seviyeli eşikleme işlemi adımları Şekil 8 'de [67] gösterilmiştir.



Şekil 8. FA tabanlı imge bölütleme akış şeması

Ateşböceklerinin çekiciliği Eşitlik (20), birbirlerine hareket etmeleri sonucu çözüm değerlerinin güncelleme işlemi Eşitlik (22) ile hesaplanmıştır. Her adımda yerel ve küresel çözümler ayrı ayrı hesaplanmaktadır. Optimizasyon işlemi sonlanma kriterine ulaştığında en iyi çözüm en iyi eşik değerlerine sahip ateşböceğidir. Son olarak elde edilen eşik değerleri üzerinden görüntü eşikleme işlemi gerçekleştirilmiştir.

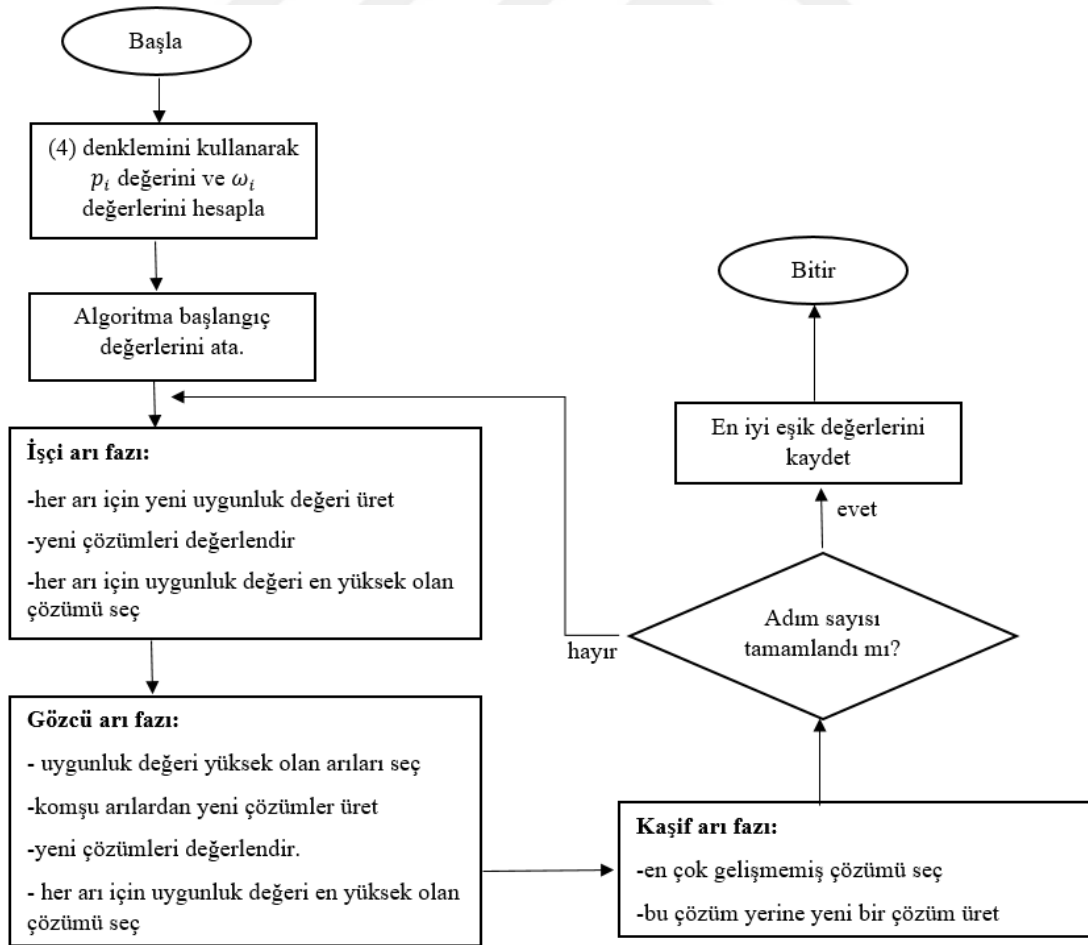


Şekil 9. Kapur Entropi ölçüsü ve Otsu tabanlı gerçekleştirilmiş FA algoritması ile $m= 1, 2, 3, 4$ ve 5 için elde edilen eşik değerleri ve bu değerler ile bölütlenmiş görüntüler.

Şekil 9’ da elde edilen eşiklere göre bölütlenmiş ‘Cameraman’ sonuç görüntüleri verilmiştir. Şekilde birinci sütun Kapur Entropisi ile sırasıyla $m=1, 2, 3, 4$ ve 5 için bulunan eşik seviye değerlerini, ikinci sütun bu eşik seviyeleri ile eşiklenmiş sonuç görüntülerini; üçüncü sütun Otsu yöntemi ile bulunan eşik seviye değerlerini, dördüncü sütun bu eşik seviyeleri ile bölütlenmiş sonuç görüntülerini göstermektedir. Algoritmadaki β_0 , α ve γ parametre değerleri 1 olarak alınmıştır. Bu değerler amaç fonksiyonları olarak seçilen Kapur Entropisi ve Otsu fonksiyon değerlerini en yüksek yapan parametre değerleridir.

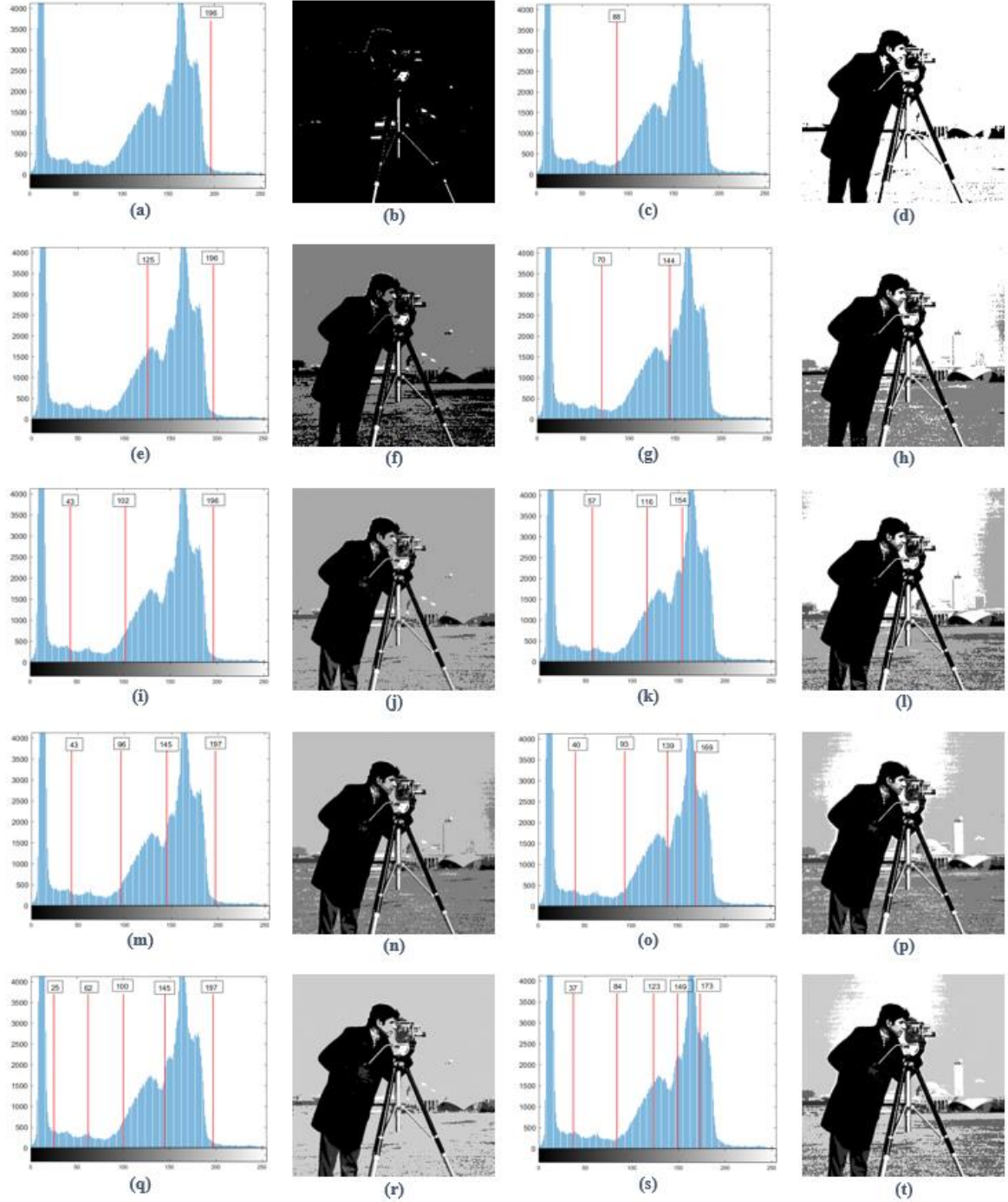
2.3. ABC Tabanlı İmge Bölütleme

ABC algoritması ile eşikleme işleminde başlangıç popülasyonu rastgele oluşturulmuştur. Popülasyondaki arılar gri seviye değerlerine denk gelmektedir. Algoritmada amaç fonksiyonu olarak sırasıyla Kapur ve Otsu yaklaşımları seçilmiştir.



Şekil 10. ABC tabanlı imge bölütleme akış şeması

Seçilen amaç fonksiyonları algoritma ile optimize edilmiştir. Sonlanma kriterine ulaşıldığında en yüksek uygunluk değerine sahip arı en optimum eşik değerlerini barındıran çözümdür ve bu eşik değerleri üzerinden imge bölütlemesi gerçekleştirilmiştir. İşlem adımları [69] Şekil 10' da gösterilmiştir.



Şekil 11. Kapur Entropi ölçüsü ve Otsu tabanlı gerçekleştirilmiş ABC algoritması ile $m=1, 2, 3, 4$ ve 5 için elde edilen eşik değerleri ve bu değerler ile bölütlenmiş görüntüler.

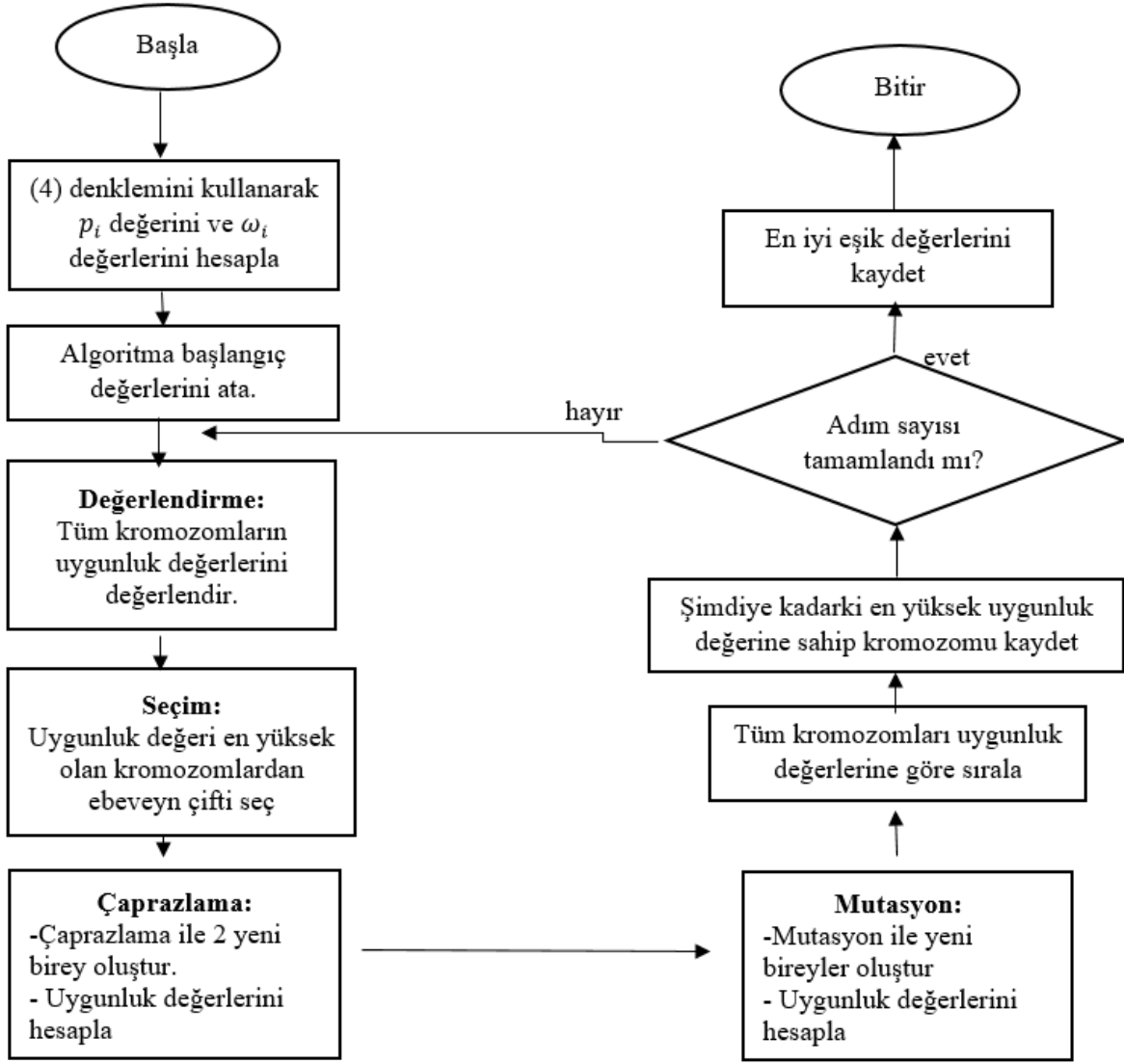
Algoritmada işçi arı, gözcü arı ve kaşif arı fazları her adımda art arda işletilerek en optimum eşik değerleri elde edilmeye çalışılmıştır. İşçi ve gözcü arı fazlarında hesaplanan yeni çözümler ile eski çözümler değerlendirilip Kapur veya Otsu değeri daha yüksek olan arılar seçilmiştir. Kaşif arı fazında ise geliştirilememiş çözümler yerine yeni çözümler üretilmiştir. Algoritmadaki 'limit' parametresi $0.6 * N * d$ olarak alınmıştır. Burada N sürü sayısı, d problemin boyutu olan eşik seviye sayısıdır.

Şekil 11' de elde edilen eşiklere göre bölütlenmiş 'Cameraman' sonuç görüntüleri verilmiştir. Şekilde birinci sütun Kapur Entropisi ile sırasıyla $m=1, 2, 3, 4$ ve 5 için bulunan eşik seviye değerlerini, ikinci sütun bu eşik seviyeleri ile eşiklenmiş sonuç görüntülerini; üçüncü sütun Otsu yöntemi ile bulunan eşik seviye değerlerini, dördüncü sütun bu eşik seviyeleri ile bölütlenmiş sonuç görüntülerini göstermektedir.

2.4. GA Tabanlı İmge Bölütleme

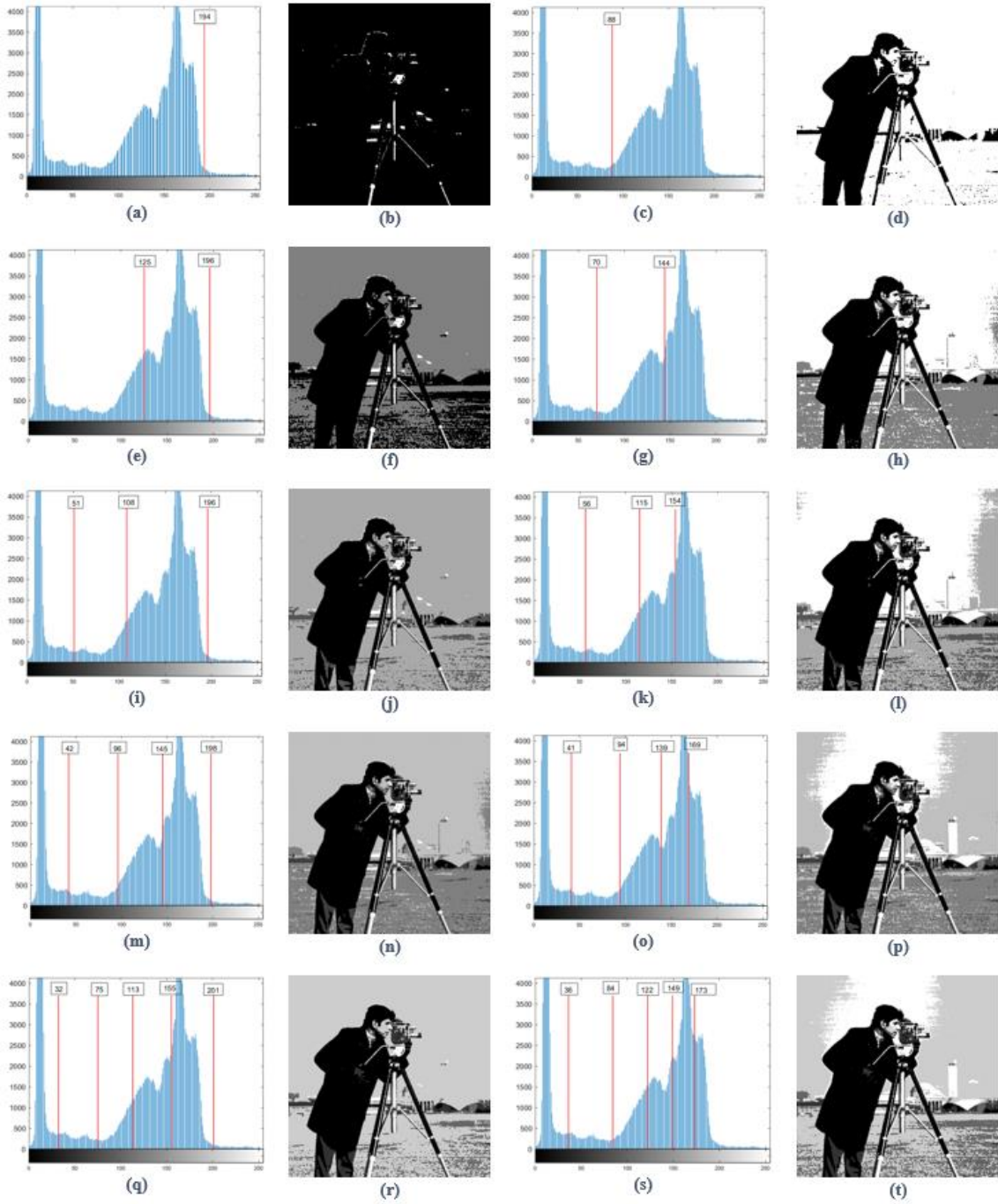
GA birçok optimizasyon ve bölütleme problemlerine sıkça uygulanmış evrimsel tabanlı bir algoritmadır. Orijinal GA' da bireyler '1' ve '0' değerlerinden oluşan dizilerle ifade edilir. Bu çalışmada bireyler ikili sayılar yerine gerçek numerik sayılar kullanılarak ifade edilmiştir. Algoritmada her birey (kromozom) gri seviye değerine denk gelir. Amaç fonksiyonu ilk olarak Kapur ikinci olarak Otsu fonksiyonu seçilip GA algoritma ile optimize edilerek görüntüler eşiklenmiştir. Algoritmada seçim, çaprazlama ve mutasyon aşamaları art arda gerçekleştirilmiştir. Ebeveyn seçim işlemi 'rulet tekerleği' yöntemi ile uygunluk değerine bağlı olarak seçilme olasılıkları en yüksek olan bireyler arasından seçilerek sağlanmıştır. Çaprazlama işlemi Eşitlik (26) kullanılarak iki yeni birey elde etmeyle gerçekleştirilmiştir. Mutasyon aşaması bireylerdeki genlerin rastgele yeni sayılar üretilip değiştirilmesiyle sağlanmıştır. Her adımda seçilen amaç fonksiyonu optimize edilmiş, sonlanma şartı sağlanınca en iyi uygunluk değerine sahip birey en optimum eşik değerlerini barındıran çözüm olarak alınmıştır. Son olarak eşik değerleri kullanılarak bölütleme işlemi gerçekleştirilmiştir. Çalışmada amaç fonksiyon değerinin en yüksek olmasını sağlayan çaprazlama oranı ve mutasyon oranı parametre değerleri, sırasıyla 0.9 ve 0.1 olarak belirlenmiştir.

Algoritmanın temel adımları Şekil 12 'de gösterilmiştir.



Şekil 12. GA tabanlı imge bölütleme akış şeması

GA ile bölütlenmiş ‘Cameraman’ görüntüsünün beş eşik seviyesine kadar bölütlenmiş sonuç görüntüleri Şekil 15’ te gösterilmiştir. Şekilde birinci sütun Kapur Entropisi ile sırasıyla $m=1, 2, 3, 4$ ve 5 için görüntünün histogramı üzerinde bulunan eşik seviye değerlerini, ikinci sütun bu eşik seviyeleri üzerinden eşiklenmiş sonuç görüntülerini; üçüncü sütun Otsu yöntemi ile histogram üzerinde bulunan eşik seviye değerlerini, dördüncü sütun bu eşik seviyeleri ile bölütlenmiş sonuç görüntülerini göstermektedir.



Şekil 13. Kapur Entropi ölçüsü ve Otsu tabanlı gerçekleştirilmiş GA algoritması ile $m=1, 2, 3, 4$ ve 5 için elde edilen eşik değerleri ve bu değerler ile bölütlenmiş görüntüler.

2.5. SOS Tabanlı İmge Bölütleme

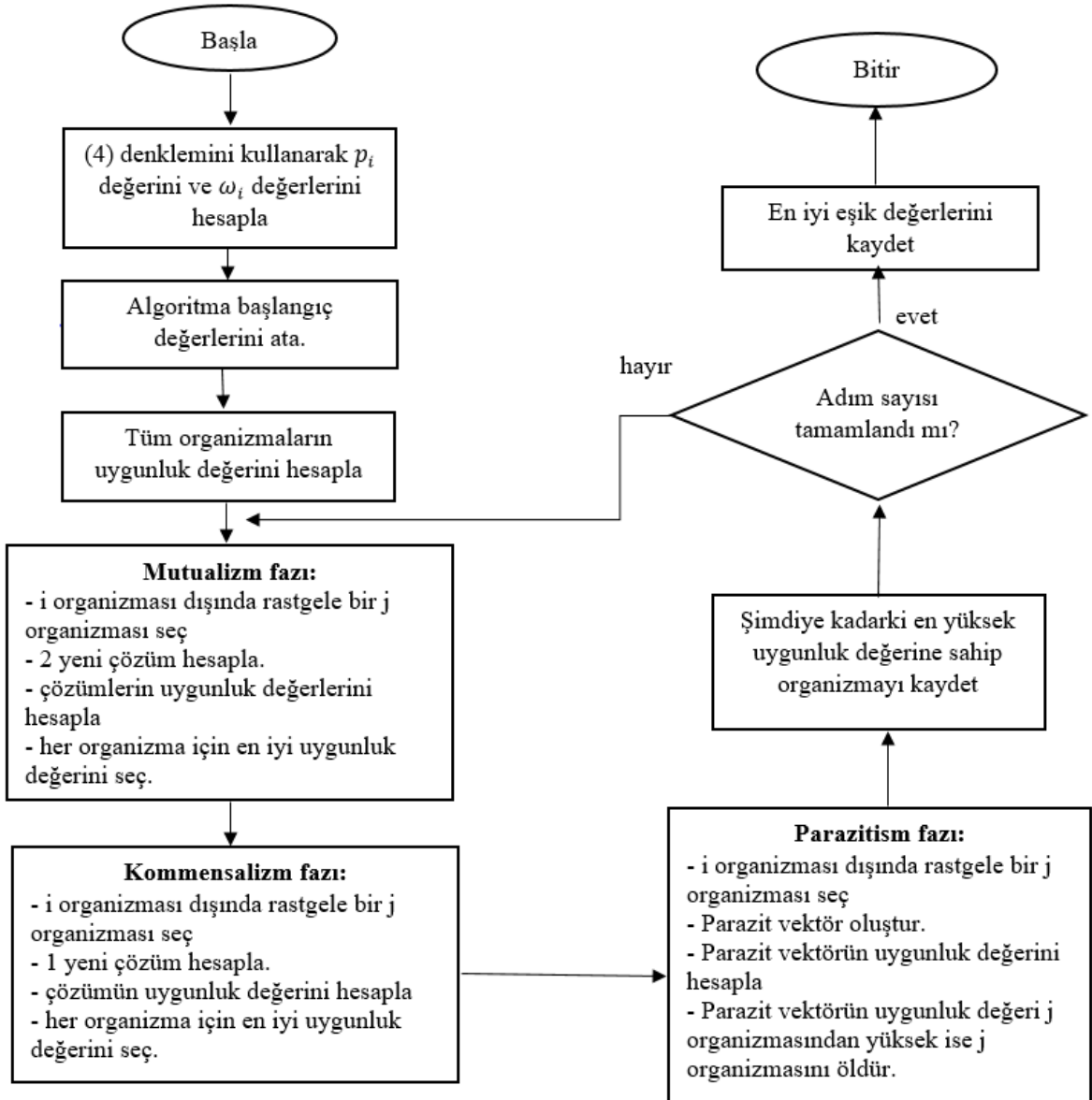
Bu çalışmada amaç eşikleme problemine henüz uygulanmamış SOS algoritmasının başarımını değerlendirmektir. SOS algoritması diğer popülasyon tabanlı algoritmalar ile birçok benzer özelliğe sahiptir. Küresel çözüm hesaplanırken mevcut çözümler üzerinde

iteratif olarak belirli operatörler kullanılır. Birçok evrimsel tabanlı algoritmalarından farklı olarak yeni nesiller üretmez fakat birçok optimizasyon algoritması gibi bireysel etkileşimler üzerine yoğunlaşır. Bu benzerliğe rağmen mutualizm, kommensalizm ve parazitizm fazlarında kullanılan stratejiler ile algoritmanın diğer optimizasyon algoritmalarından önemli ölçüde farklılık göstermesi sağlanmıştır. Mutualizm fazında, diğer algoritmalarından farklı olarak yeni çözümler, en iyi çözüm ile iki farklı çözümün ortalama değeri (mutual vektör) arasındaki fark kullanılarak hesaplanır. Bu özellik, iki çözümün birbirinden çok uzak konumda olması durumunda yeni bölgeleri keşfetmede bir avantaj sağlar. Ayrıca bu iki birey için yeni çözümler aynı anda güncellenir. Kommensalizm fazında, çözümün bir diğer çözüme göre güncellenme işlemi birçok algoritmada kullanılan bir yöntemdir. Ancak SOS algoritmasında farklı olarak, en iyi çözümün yakınındaki bölgeleri araştırabilmek için, en iyi çözüm referans noktası olarak alınır. Bu özellik yakınsama hızını artırmaya yardımcı olur. Parazitizm fazı ise SOS algoritmasına özgü bir mutasyon operatörüdür. Parazit vektör kullanılarak mevcut çözüm, yerine geçebilecek rastgele üretilmiş yeni bireyler ile yarışırılır. Parazitizm fazının avantajları şu şekilde sıralanabilir: (1) parazit vektör ile konak çözümün küçük bir parçası değiştirilmez, rastgele boyutlarda rastgele sayılar üretilerek değiştirilir, (2) tüm boyutlarda yapılan değişiklikler; erken yakınsamayı önler ve çeşitliliği artırır böylece rastgelelik artar, (3) tamamen farklı bölgelerde bulunabilecek benzersiz çözümler üretir [33]. Belirtilen avantajlar sebebi ile SOS algoritmasının imge bölütleme problemlerindeki performansı literatürde kullanılmış diğer algoritmalar ile karşılaştırılmıştır.

Bölütleme işlemi için başlangıç popülasyonu diğer algoritmalarda olduğu gibi rastgele gri seviye değerleri üretilerek oluşturulmuştur. Popülasyondaki organizmalar gri seviye değerlerine denk gelmektedir. Optimize edilecek olan amaç fonksiyonu ilk olarak Kapur Entropisi ve sonrasında Otsu yöntemi seçilmiştir. Mutualizm, kommensalizm ve parazitizm fazları art arda yürütülerek amaç fonksiyonları optimize edilmeye çalışılmıştır. Mutualizm fazında Eşitlik (27) ve (28) kullanılarak iki yeni çözüm, kommensalizm fazında Eşitlik (30) kullanılarak tek yeni çözüm elde edilmiştir. Elde edilen yeni çözümlerin uygunluk değerleri eski çözümlerin uygunluk değerleriyle karşılaştırılarak daha yüksek entropi veya sınıflar arası varyansa sahip çözümler seçilmiştir. Parazitizm fazında üretilen parazit vektör ile ekosistemden rastgele seçilen bir organizmanın uygunluk değerleri karşılaştırılmış ve daha yüksek uygunluk değerine sahip organizma ekosistemdeki yerini almıştır. Tüm hesaplamalardan sonra sonlama kriteri sağlanıyorsa en yüksek entropi

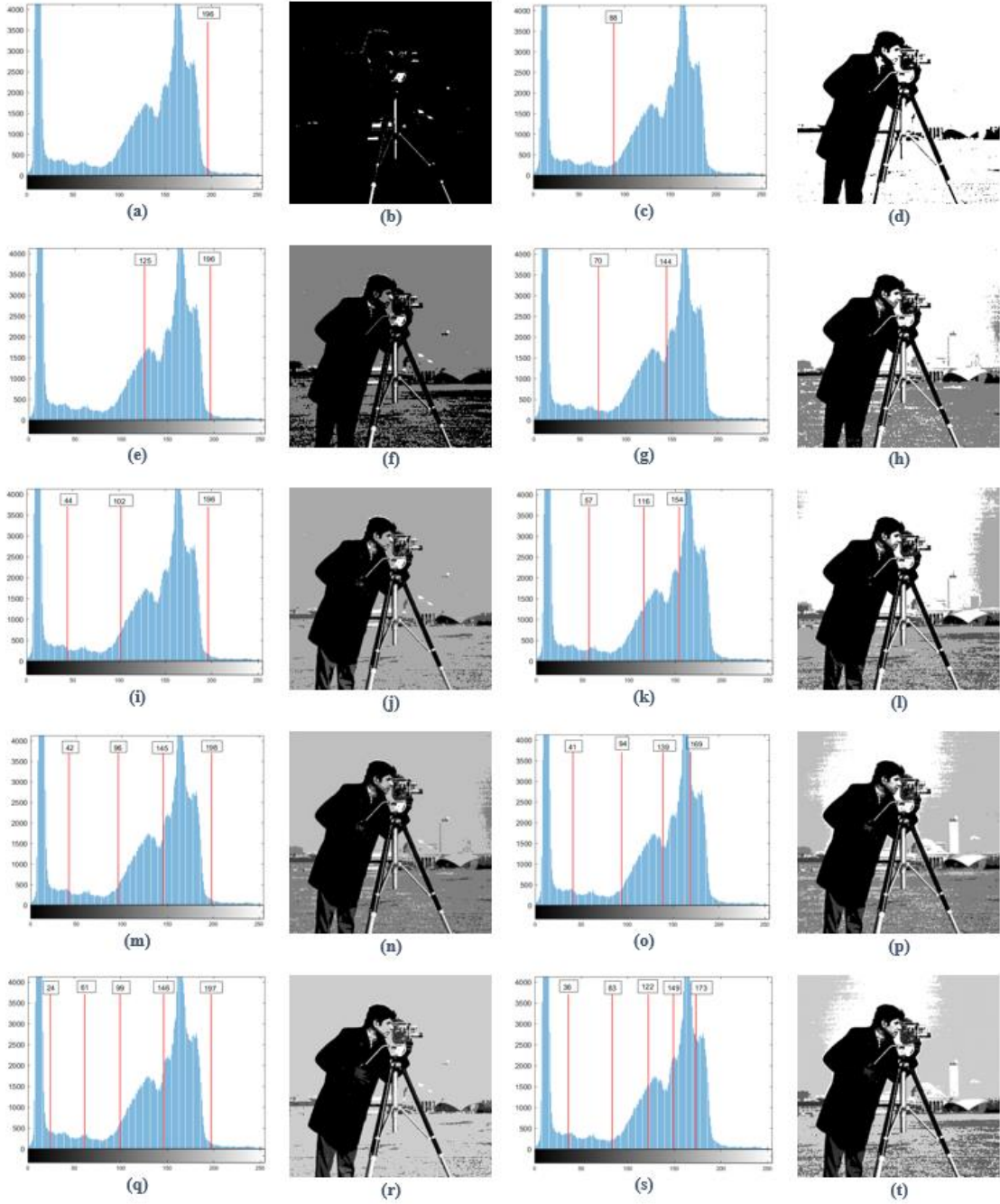
değerine veya sınıflar arası varyans değerine sahip organizma en optimal eşik değerlerini barındıran organizma olarak kaydedilmiştir. Son olarak elde edilen eşik değerleri üzerinden görüntü beş eşik seviyesine kadar bölütlenmiştir.

Algoritmanın orijinalinde 1 veya 2 olmak üzere her adımda rastgele belirlenen BF_1 ve BF_2 yarar faktör parametrelerinin seçimi eşikleme işleminde maksimum entropi veya otsu uygunluk değerlerinin hesaplanmasını sağlayacak şekilde 1 olarak belirlenmiştir. Algoritmanın temel adımları Şekil 14'te gösterilmiştir.



Şekil 14. SOS tabanlı imge bölütleme akış şeması

‘Cameraman’ görüntüsü için SOS ile bölütlenmiş sonuç görüntüleri Şekil 15’ te gösterilmiştir. Şekilde birinci sütun Kapur Entropisi ile sırasıyla $m=1, 2, 3, 4$ ve 5 için bulunan eşik seviye değerlerini, ikinci sütun bu eşik seviyeleri üzerinden eşiklenmiş sonuç görüntülerini; üçüncü sütun Otsu yöntemi ile bulunan eşik seviye değerlerini, dördüncü sütun bu eşik seviyeleri ile bölütlenmiş sonuç görüntülerini göstermektedir.



Şekil 15. Kapur Entropi ölçüsü ve Otsu tabanlı gerçekleştirilmiş SOS algoritması ile $m = 1, 2, 3, 4$ ve 5 için elde edilen eşik değerleri ve bu değerler ile bölütlenmiş görüntüler.

2.6. Performans Değerlendirme Metrikleri

Bu bölümde tez kapsamında yapılan çalışmaların performans analizinin yapılabilmesi için kullanılan performans değerlendirme metrikleri incelenecektir. Hazırlanan tüm meta-sezgisel tabanlı eşikleme yöntemleri amaç fonksiyon değeri, PSNR, SSIM ve Cpu işlem süresi bakımından karşılaştırılmışlardır.

2.6.1. Amaç (Objective) Fonksiyon Değeri

Bir görüntü üzerinde iyi bir eşikleme işlemi gerçekleştirmek için en optimum eşik seviyelerini elde etmek gerekir. Eşik seviyesi bulmak için geliştirilmiş yöntemler bir amaç fonksiyonunu maksimize veya minimize ederek görüntü için en uygun eşik seviyelerini araştırır. Çalışmada kullanılan Kapur ve Otsu yöntemleri, sınıfların homojenliğini sağlamak için sırasıyla, sınıflar arası entropi ölçüsünü ve varyansı maksimize etmeyi amaçlar. Optimizasyon işlemi ise bir süreci daha iyi hale getirme işlemidir. Optimizasyon işleminde girdi birçok değerden oluşur, gerçekleştirilecek işlem veya fonksiyon; maliyet (cost) fonksiyonu, amaç (objective) fonksiyonu veya uygunluk (fitness) fonksiyonu olarak adlandırılır [26 ,70]. Eşikleme sürecini iyileştirmek için Eşitlik (8) ve Eşitlik (16) ile ifade edilen Kapur ve Otsu yöntemleri meta-sezgisel optimizasyon algoritmaları ile maksimize edilmeye çalışılmıştır. Hesaplanan çözümün kalitesi veya uygunluğu objektif fonksiyonun değeri ile orantılı olabileceği gözlenmiştir. Bu sebeple en yüksek amaç fonksiyon değerini hesaplayan meta-sezgisel algoritma en başarılı algoritma olarak kabul edilmiştir.

2.6.2. PSNR ve SSIM Değerleri

PSNR (Peak Signal-to-Noise Ratio) [46, 66] her pikselin MSE (Mean-Squared Error) değerine bağlı olarak bölütlenmiş görüntü ile orijinal görüntü arasındaki farkı ölçmek için kullanılır. Görüntüdeki gri seviye değerlerine bağlıdır ve yeniden oluşturulan görüntünün kalitesini yansıtır. SSIM (Structural Similarity Index) [66, 71] ise orijinal görüntü ile bölütlenmiş sonuç görüntüsü arasındaki benzerliği değerlendirmek için kullanılır.

PSNR ve MSE yeni oluşturulmuş görüntünün doğruluğunu ve kesinliğini ifade eder. SSIM yeni oluşturulmuş görüntü ile orijinal görüntü arasındaki karşılaştırmayı sağlayan,

yapısal bilginin bozulmasına dayanan bir kalite değerlendirmesidir. PSNR ve MSE ölçütleri Eşitlik (31) ve Eşitlik (32) kullanılarak hesaplanır [61].

$$MSE = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N [I(i,j) - I'(i,j)]^2 \quad (31)$$

$$PSNR = 10 \log_{10} \left(\frac{255^2}{MSE} \right) \quad (32)$$

burada M ve N , I orijinal görüntüsünün boyutu ve I' eşiklenmiş sonuç görüntüsüdür. SSIM ölçütü Eşitlik (33) kullanılarak hesaplanır [61].

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad (33)$$

burada, μ_x ve μ_y sırasıyla x ve y görüntülerinin ortalama gri seviye değerleridir. σ_x ve σ_y , x ve y görüntülerinin standart sapmalarıdır. σ_{xy} , x ve y arasındaki yerel korelasyon katsayısı, c_1 ve c_2 değerleri $c_1 = c_2 = 0.065$ olan sabitlerdir.

PSNR ve SSIM değerlerinin yüksek olması bölütleme algoritmasının yüksek performans gösterdiğini ispatlar.

2.6.3. Standart Sapma Değeri

Daha önce belirtildiği gibi, tüm optimizasyon algoritmaları rastgelelik özelliğine sahiptir. Bu nedenle algoritmaların doğruluğunu ve kararlılığını ölçmek için ortalama ve standart sapma metrikleri kullanılmıştır. Ortalama ve standart sapma değerleri sırasıyla Eşitlik (34) ve Eşitlik (35) kullanılarak hesaplanır [27].

$$\mu = \frac{\sum_{i=1}^k S_i}{k} \quad (34)$$

$$std = \sqrt{\frac{1}{k} \sum_{i=1}^k (S_i - \mu)^2} \quad (35)$$

Burada, μ ve std sırasıyla amaç fonksiyon deęerinin ortalama ve standart sapma deęerlerini gstermektedir. k deęeri her bir algoritmanın alıřtırılma sayısını, S_i deęeri i . alıřtırılmada elde edilen en iyi uygunluk deęerini temsil etmektedir.

2.6.4. Cpu İřlem Sresi

Gerek zamanlı uygulamalar yksek performansın yanı sıra kısa hesaplama sresi gerektirir. Bu nedenle tek seviyeli eřikleme iřleminin ok seviyeli eřiklemeye geniřletilmesiyle artan hesaplama sresini azaltmak iin alıřmada nerilen tm meta-sezgisel tabanlı yntemler Cpu iřlem sresi bakımından incelenmiřtir.

2.7. Deneysel Sonular

Tez kapsamında hazırlanan tm meta-sezgisel algoritma tabanlı eřikleme yntemlerinin USC-SIPI ve Berkeley veri setlerinden alınan on bir standart grnt zerinde gerekleřtirilmiř deneysel sonuları bu blmde verilmiřtir. Deneyler Intel i7 Core 2.70 GHz iřlemcili, 8 GM RAM, Windows 10 iřletim sistemli dizst bilgisayarında Matlab yazılımında kodlanmasıyla gerekleřtirilmiřtir.

Algoritmaların karřılařtırılması iin 30 alıřtırma sonucu algoritmalarından elde edilen amaç fonksiyon deęeri, eřik deęerleri, PSNR ve SSIM deęerlerinin ortalaması alınmıřtır.

2.7.1. Ama Fonksiyon Deęeri Deneysel Sonuları

Hazırlanan tm yntemlerde Kapur Entropisi ve Otsu yaklařımları amaç fonksiyonu seilerek meta-sezgisel algoritmalar ile optimize edilmiřlerdir. Ama fonksiyonunu en yksek yapan algoritma en bařarılı algoritma olarak kabul edilir.

Tablo 1 'de Kapur Entropisi tabanlı gerekleřtirilmiř algoritmaların her bir grnt iin ortalama amaç fonksiyon deęerleri, Tablo 2'de elde edilen ortalama eřik seviye deęerleri verilmiřtir. Tablolarda ilk stn grnty (Im), ikinci stn eřik sayısını (m) belirtmektedir.

Tablo 1. Kapur Entropisi tabanlı algoritmaların ortalama amaç fonksiyon deęerleri

Im	m	Amaç fonksiyon				
		PSO	FA	SOS	ABC	GA
Cameraman	1	8.7868	8.7748	8.7868	8.7868	8.7747
	2	12.2865	12.2865	12.2865	12.2865	12.2865
	3	15.3744	15.3928	15.3943	15.3927	15.3810
	4	18.5567	18.5563	18.5567	18.5445	18.5564
	5	21.2809	21.3213	21.3254	21.2756	21.2792
Lena	1	8.9413	8.9413	8.9413	8.9413	8.9413
	2	12.3466	12.3466	12.3466	12.3466	12.3466
	3	15.3183	15.3183	15.3183	15.3179	15.3176
	4	18.0016	18.0090	18.0121	18.0019	18.0011
	5	20.6095	20.6093	20.6073	20.5760	20.6017
Peppers	1	9.1423	9.1423	9.1423	9.1423	9.1423
	2	12.6346	12.6346	12.6346	12.6346	12.6346
	3	15.6887	15.6887	15.6887	15.6885	15.6883
	4	18.5216	18.5354	18.5392	18.5238	18.5229
	5	21.2730	21.2817	21.2818	21.2446	21.2755
Airplane	1	8.6191	8.6191	8.6191	8.6191	8.6191
	2	12.2111	12.2115	12.2115	12.2115	12.2114
	3	15.5039	15.5038	15.5039	15.5035	15.5036
	4	18.3120	18.3119	18.3120	18.3070	18.3099
	5	20.9087	20.9085	20.9081	20.8773	20.9000
Baboon	1	8.8984	8.8984	8.8984	8.8984	8.8984
	2	12.2178	12.2178	12.2178	12.2178	12.2178
	3	15.2786	15.2790	15.2792	15.2787	15.2782
	4	18.1246	18.1246	18.1265	18.1188	18.1198
	5	20.7895	20.7889	20.7888	20.7612	20.7755
Butterfly	1	9.1170	9.1170	9.1170	9.1170	9.1170
	2	12.6308	12.6308	12.6308	12.6308	12.6308
	3	15.7082	15.7086	15.7105	15.7070	15.7077
	4	18.7850	18.7850	18.7850	18.7763	18.7846
	5	21.4382	21.4503	21.4488	21.3989	21.4360
Ostrich	1	9.0648	9.0648	9.0648	9.0648	9.0648
	2	12.5935	12.5935	12.5935	12.5935	12.5935
	3	15.6550	15.6550	15.6550	15.6540	15.6547
	4	18.5555	18.5555	18.5563	18.5476	18.5528
	5	21.3769	21.4604	21.4613	21.3940	21.4068
Starfish	1	9.3759	9.3759	9.3759	9.3759	9.3759
	2	12.9682	12.9682	12.9682	12.9682	12.9682
	3	16.1254	16.1254	16.1254	16.1251	16.1249
	4	19.0578	19.0576	19.0579	19.0526	19.0548
	5	21.8088	21.8086	21.8084	21.7829	21.8038
Flower	1	9.2252	9.2252	9.2252	9.2252	9.2252
	2	12.6227	12.6227	12.6227	12.6227	12.6227
	3	15.7331	15.7369	15.7369	15.7364	15.7364
	4	18.6951	18.6949	18.6951	18.6896	18.6936
	5	21.3700	21.3716	21.3719	21.3488	21.3670
Plane	1	8.1580	8.1580	8.1580	8.1580	8.1580
	2	11.0739	11.0774	11.0774	11.0774	11.0758
	3	13.8912	13.9522	13.9586	13.9571	13.9406
	4	16.6455	16.6648	16.6705	16.6311	16.6390
	5	19.1482	19.1448	19.1478	19.0740	19.1279
Girl	1	8.6091	8.6091	8.6091	8.6091	8.6091
	2	11.9353	11.9340	11.9353	11.9353	11.9353
	3	15.0761	15.0761	15.0761	15.0751	15.0760
	4	17.8740	17.8733	17.8735	17.8607	17.8727
	5	20.6819	20.6940	20.6977	20.6371	20.6716

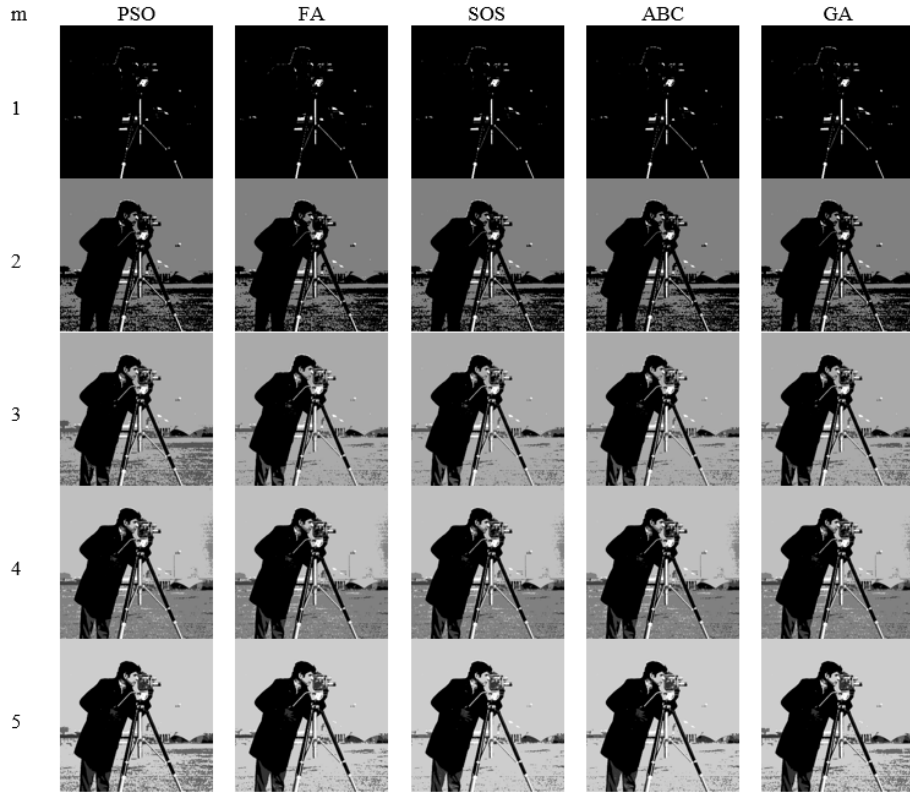
Tablo 2. Kapur Entropisi tabanlı algoritmalarından elde edilen ortalama eşik değerleri

Im	m	Eşik değerleri				
		PSO	FA	SOS	ABC	GA
Cameraman	1	196	194	196	196	194
	2	125 196	125 196	125 196	125 196	125 196
	3	54 111 196	44 103 196	44 102 196	43 102 196	51 108 196
	4	42 96 145 198	42 96 145 198	42 96 145 198	43 96 145 197	42 96 145 198
	5	34 78 118 161 204	27 66 105 152 201	24 61 99 146 197	25 62 100 145 197	32 75 113 155 201
Lena	1	123	123	123	123	123
	2	97 164	97 164	97 164	97 164	97 164
	3	82 126 175	82 126 175	82 126 175	82 126 175	83 127 176
	4	70 107 146 183	66 100 139 180	64 97 137 179	66 99 139 179	70 107 144 181
	5	63 94 128 163 194	63 94 128 163 194	63 94 128 163 194	63 94 128 162 193	64 94 127 161 192
Peppers	1	117	117	117	117	117
	2	75 147	75 147	75 147	75 147	75 147
	3	61 113 165	61 113 165	61 113 165	61 113 165	61 113 164
	4	55 99 141 186	58 105 147 193	58 105 148 194	57 105 148 194	54 99 142 188
	5	42 77 113 153 193	42 77 114 154 195	42 77 114 154 195	43 77 114 155 195	43 78 114 153 195
Airplane	1	162	162	162	162	162
	2	71 173	71 173	71 173	71 173	71 173
	3	69 127 183	69 127 183	69 127 183	69 127 183	69 127 183
	4	67 106 145 185	67 106 145 185	67 106 145 185	66 105 144 185	65 104 143 185
	5	60 89 123 155 187	60 90 123 156 187	60 89 124 156 187	60 90 124 156 188	61 92 126 157 188
Baboon	1	109	109	109	109	109
	2	78 143	78 143	78 143	78 143	78 143
	3	46 99 152	45 98 152	45 98 152	45 99 152	46 100 153
	4	36 75 115 160	35 75 115 160	33 73 114 159	35 75 116 160	36 78 118 162
	5	33 69 104 138 173	33 69 104 138 172	33 68 104 138 172	32 68 104 138 172	34 71 106 140 174
Butterfly	1	123	123	123	123	123
	2	115 176	115 176	115 176	115 176	115 176
	3	88 137 190	90 141 193	107 161 210	101 154 205	88 137 190
	4	74 120 172 219	74 120 172 219	74 120 172 219	74 120 171 219	74 120 171 219
	5	72 113 145 178 220	73 115 146 178 220	73 115 146 178 220	73 115 146 179 219	72 112 144 177 220
Ostrich	1	126	126	126	126	126
	2	120 180	120 180	120 180	120 180	120 180
	3	76 125 184	76 125 184	76 125 184	76 125 184	76 125 184
	4	67 113 157 200	65 112 156 200	52 99 144 199	56 102 146 194	73 119 161 202
	5	39 82 123 164 204	30 76 120 162 203	30 76 120 163 204	30 78 121 164 204	36 82 124 165 204
Starfish	1	155	155	155	155	155
	2	91 170	91 170	91 170	91 170	91 170
	3	75 130 183	75 130 183	75 130 183	75 130 184	75 130 183
	4	68 116 164 206	67 115 163 205	68 116 164 206	67 115 162 205	66 113 161 204
	5	56 93 132 171 209	56 94 133 171 209	56 93 132 171 210	56 93 132 171 209	56 94 133 171 209

Tablo 2' nin devamı

Flower	1	136	136	136	136	136
	2	117 175	117 175	117 175	117 175	117 175
	3	82 133 187	80 131 186	80 131 186	80 131 186	80 131 186
	4	74 119 160 206	74 119 160 206	74 119 160 206	74 118 160 205	74 119 160 206
	5	69 108 142 177 216	70 110 145 180 217	71 112 148 183 219	70 109 143 178 216	69 107 140 176 215
Plane	1	83	83	83	83	83
	2	67 102	66 101	66 101	66 101	66 101
	3	47 84 126	35 73 104	34 72 101	34 72 101	36 75 109
	4	35 74 108 156	35 73 105 159	34 72 102 159	36 72 105 159	34 72 103 157
	5	33 67 97 121 159	34 68 97 121 159	33 67 97 121 159	34 69 96 120 159	34 67 95 119 159
Gırl	1	118	118	118	118	118
	2	115 203	115 203	115 203	115 203	115 203
	3	97 145 203	97 145 203	97 145 203	97 145 203	97 145 203
	4	69 114 161 206	62 107 156 205	53 98 150 204	46 93 145 204	58 103 153 205
	5	37 82 129 171 207	39 83 133 177 208	37 83 134 178 208	39 84 132 175 208	37 81 127 170 207

Tablo 1 ve Tablo 2' deki sonuçlara göre tüm algoritmaların $m = 1$ ve $m = 2$ eşik seviye sayısında hemen hemen eşit sonuç verdikleri görülmektedir. Eşik sayısı $m=3$ olduğunda eşiklenmiş tüm görüntüler için SOS algoritmasının uygunluk değerleri, FA veya PSO algoritmalarının uygunluk değerlerine eşit veya daha yüksek olduğu görülmektedir. Ayrıca bu sonuçlar diğer ABC ve GA algoritmalarının sonuçlarından daha yüksektir. Eşik sayısı $m=4$ olduğu durumda ise Plane ve Gırl görüntüleri dışında, SOS algoritmasının uygunluk değerleri PSO veya FA algoritmalarının uygunluk sonuçlarına eşit veya daha yüksek ve kalan algoritmaların sonuçlarından daha iyi performans gösterdiği görülmüştür. Eşik sayısı $m = 5$ olduğunda Cameraman, Peppers, Ostrich, Flower ve Gırl görüntüleri için, SOS algoritmasının sonuçlarının daha başarılı olduğu sonucuna varılmıştır. Eşik sayısının $m = 3, 4$ ve 5 olduğu tüm durumlar içerisinde, SOS algoritması toplam 33 uygunluk değerinden arasından 25'inde daha iyi sonuç üretmiştir. Bu sonuçlara göre Kapur Entropisi tabanlı SOS algoritmasının çok seviyeli eşikleme problemlerinde başarılı olduğu sonucuna varılabilir.



Şekil 16. Kapur tabanlı algoritmalar ile eşiklenmiş Cameraman görüntüleri



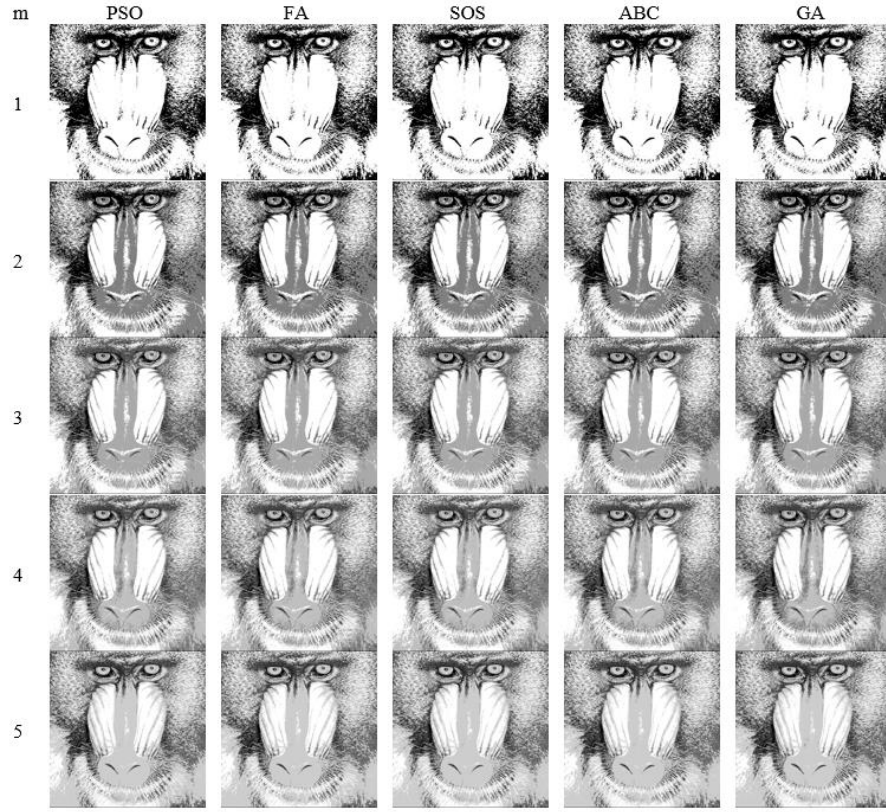
Şekil 17. Kapur tabanlı algoritmalar ile eşiklenmiş Lena görüntüleri



Şekil 18. Kapur tabanlı algoritmalar ile eşiklenmiş Peppers görüntüleri



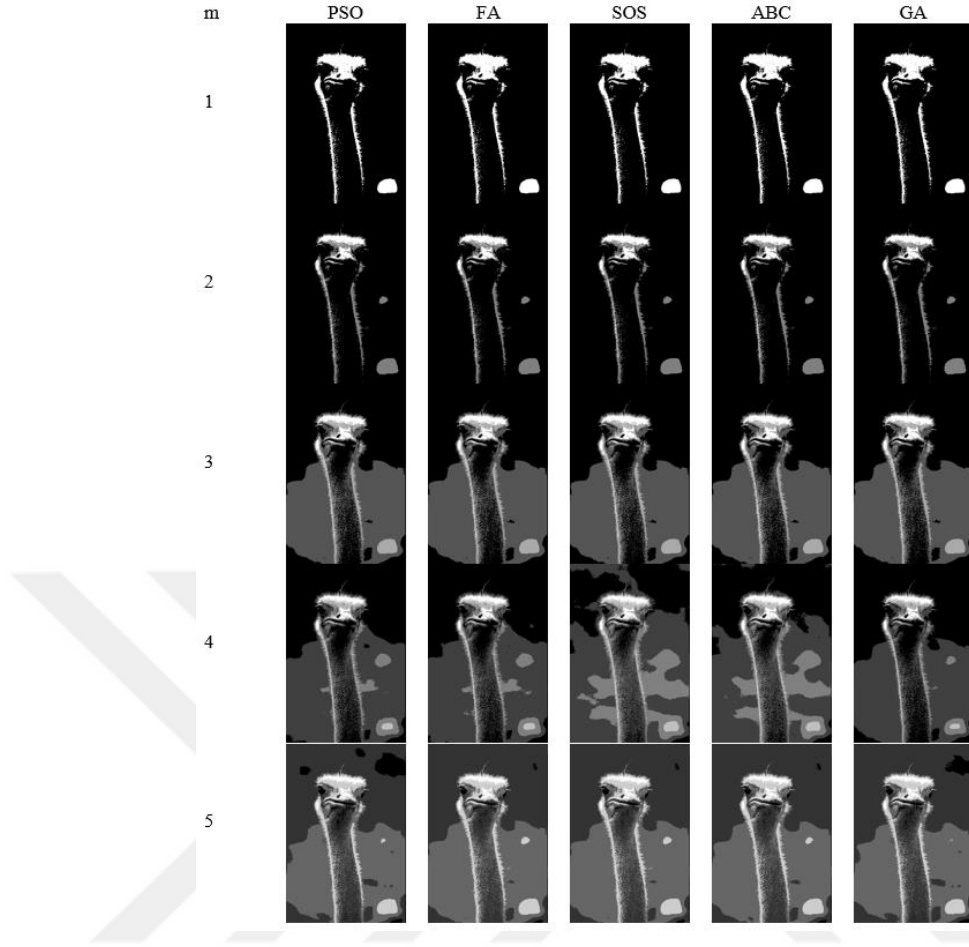
Şekil 19. Kapur tabanlı algoritmalar ile eşiklenmiş Airplane görüntüleri



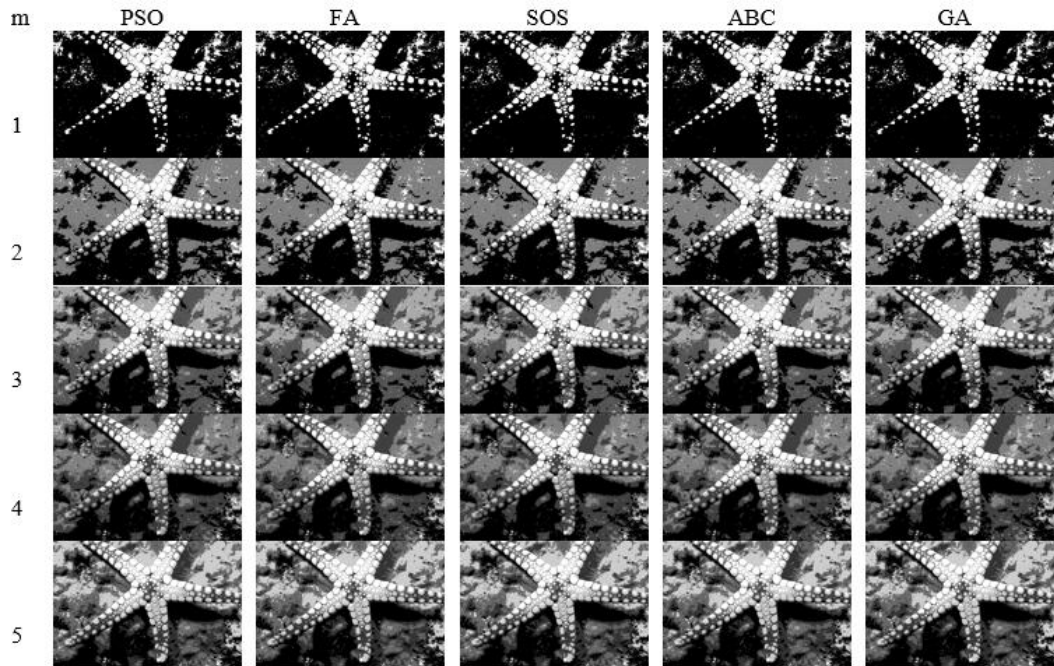
Şekil 20. Kapur tabanlı algoritmalar ile eşiklenmiş Baboon görüntüleri



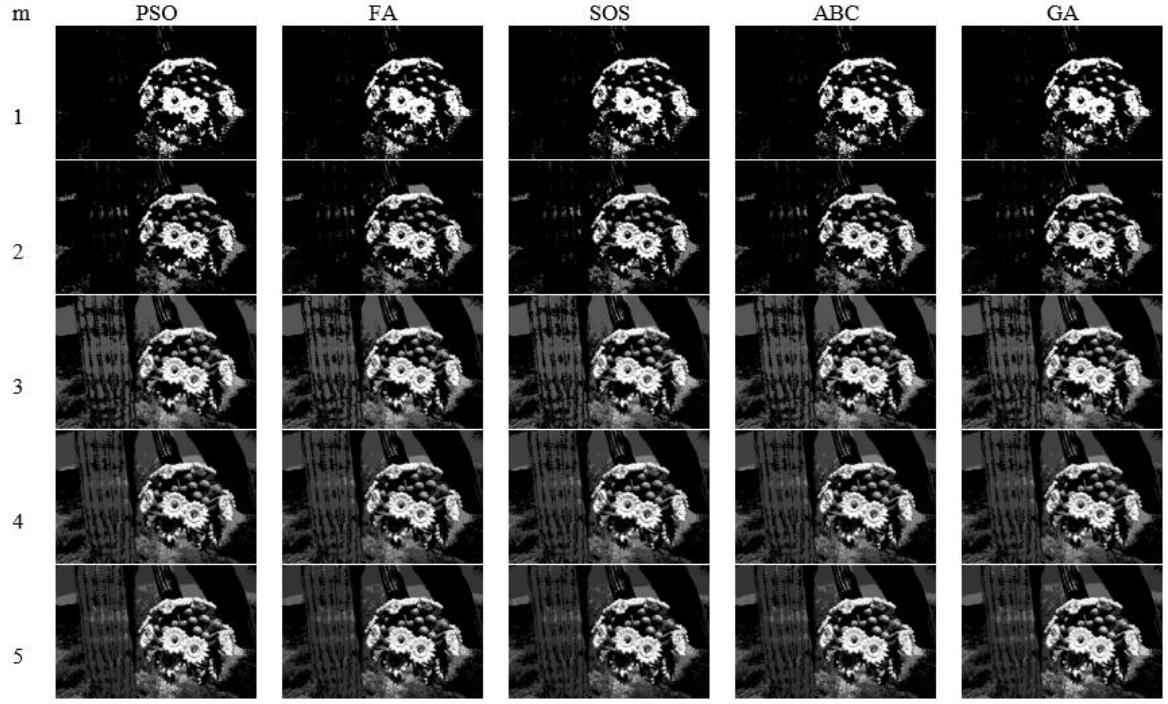
Şekil 21. Kapur tabanlı algoritmalar ile eşiklenmiş Butterfly görüntüleri



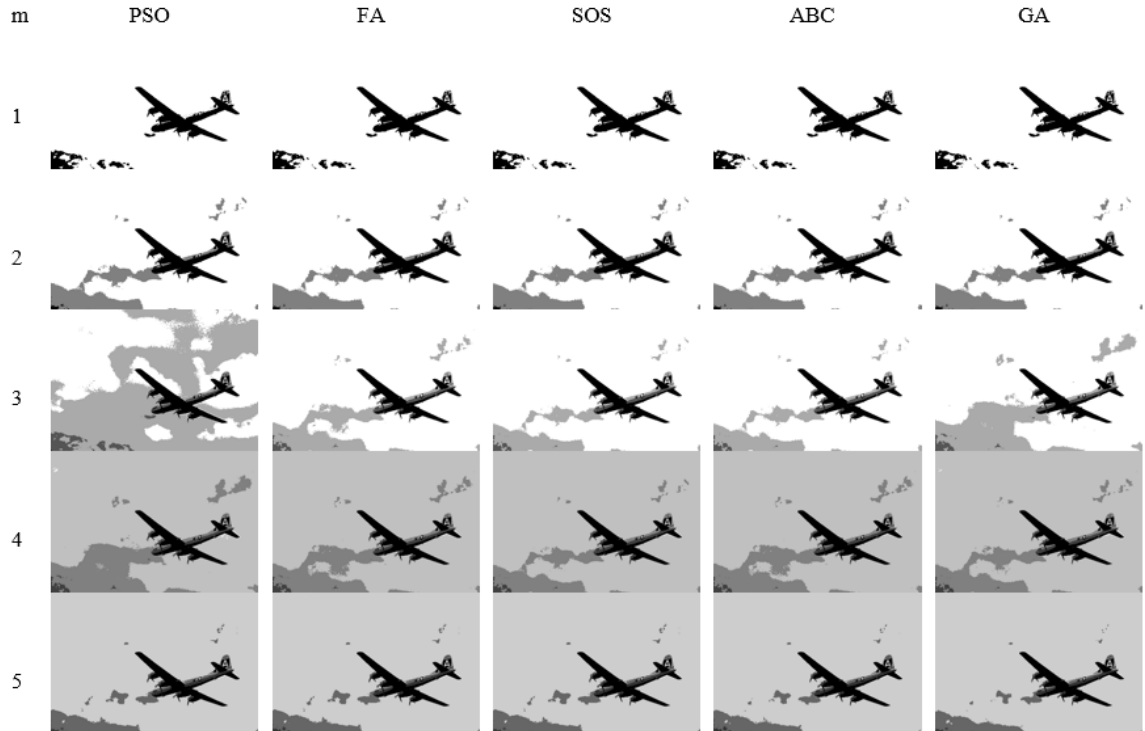
Şekil 22. Kapur tabanlı algoritmalar ile eşiklenmiş Ostrich görüntüleri



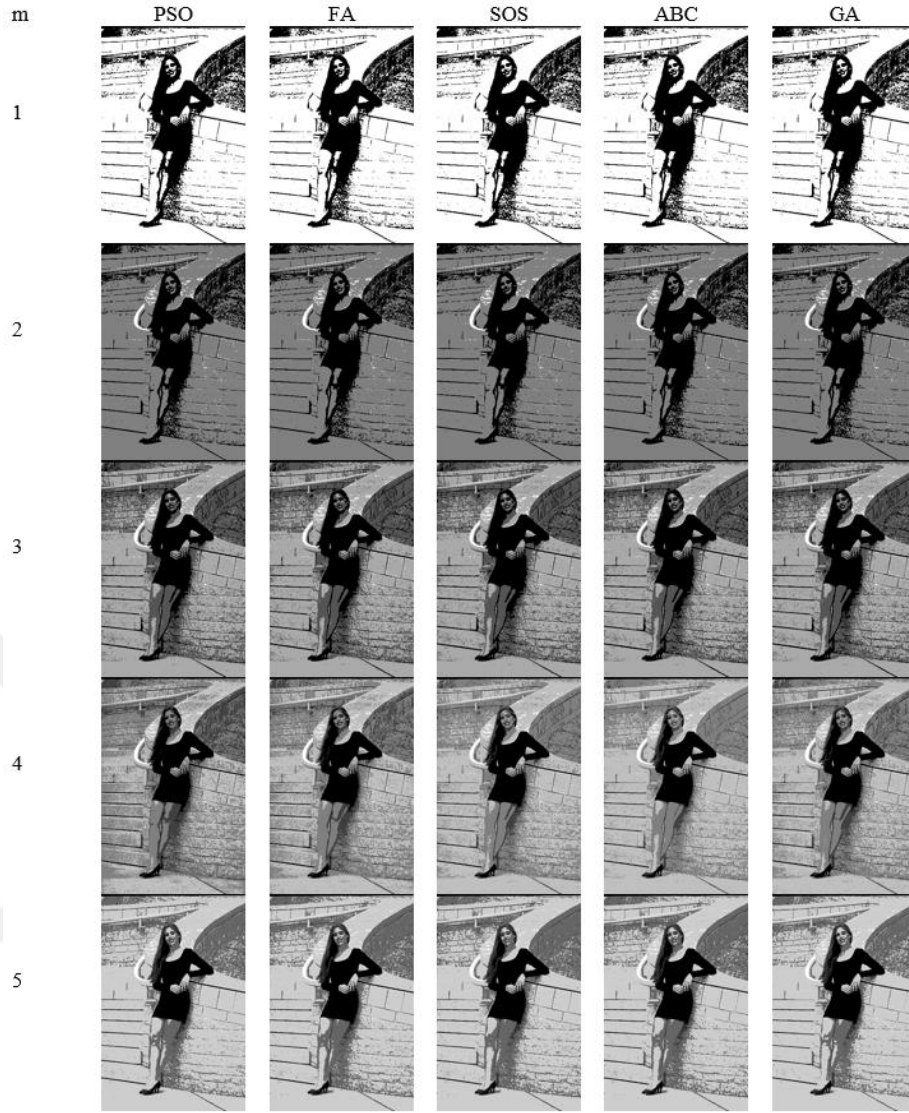
Şekil 23. Kapur tabanlı algoritmalar ile eşiklenmiş Starfish görüntüleri



Şekil 24. Kapur tabanlı algoritmalar ile eşiklenmiş Flower görüntüleri



Şekil 25. Kapur tabanlı algoritmalar ile eşiklenmiş Plane görüntüleri



Şekil 26. Kapur tabanlı algoritmalar ile eşiklenmiş Girl görüntüleri

Kapur Entropisi tabanlı gerçekleştirilmiş algoritmalarından on bir farklı görüntü için elde edilmiş ortalama eşik seviye değerlerine göre eşiklenen sonuç görüntüleri Şekil 16-26 aralığında gösterilmiştir.

Tablo 3' te Otsu yaklaşımı tabanlı gerçekleştirilmiş algoritmalarından görüntüler için elde edilen ortalama amaç fonksiyon değerleri verilmiştir. Tablo 4' te $m=1, 2, 3, 4$ ve 5 eşik seviye değerine kadar elde edilen eşik değerlerinin ortalama değerleri gösterilmiştir. Tablolarda birinci sütun kullanılan görüntüyü, ikinci sütün eşik seviyesi (m) değerini ifade etmektedir.

Tablo 3. Otsu tabanlı algoritmaların ortalama amaç fonksiyon değerleri

Im	m	Amaç fonksiyon				
		PSO	FA	SOS	ABC	GA
Cameraman	1	3245.1514	3245.1514	3245.1514	3245.1514	3245.1514
	2	3609.5601	3609.5601	3609.5601	3609.5601	3609.5470
	3	3683.3513	3683.3483	3683.3513	3683.2696	3683.0622
	4	3739.2670	3739.2608	3739.2649	3738.6383	3739.0971
	5	3770.0570	3770.0536	3770.0157	3768.2473	3769.8053
Lena	1	1601.2302	1601.2302	1601.2302	1601.2302	1601.2302
	2	1961.8170	1961.7755	1961.8170	1961.8170	1961.7575
	3	2128.3077	2128.3077	2128.3077	2128.2544	2128.2806
	4	2191.8700	2191.8700	2191.8700	2191.4438	2191.7416
	5	2217.2793	2217.7494	2217.6232	2216.0920	2217.2841
Peppers	1	2129.3618	2129.3618	2129.3618	2129.3618	2129.3618
	2	2532.3214	2532.3214	2532.3214	2532.3214	2532.3156
	3	2703.5715	2703.5713	2703.5715	2703.5364	2703.5436
	4	2766.4586	2766.4447	2766.4526	2765.7279	2764.9954
	5	2810.8415	2810.8351	2810.8183	2808.7428	2810.1764
Airplane	1	1792.1172	1792.1172	1792.1172	1792.1172	1792.1172
	2	1948.7191	1948.7191	1948.7191	1948.7189	1948.6604
	3	2024.8321	2024.8259	2024.8342	2024.7806	2024.6937
	4	2070.0769	2070.0740	2070.0769	2069.4062	2069.7139
	5	2096.1392	2096.1285	2096.1266	2094.7390	2095.8816
Baboon	1	1220.4946	1220.4946	1220.4946	1220.4946	1220.4946
	2	1549.0831	1549.0831	1549.0831	1549.0831	1549.0831
	3	1639.5322	1639.5322	1639.5322	1639.4833	1639.2085
	4	1693.1954	1693.1954	1693.1915	1692.7582	1692.9406
	5	1719.0504	1719.0335	1718.9793	1717.6416	1718.6250
Butterfly	1	3524.3352	3524.3352	3524.3352	3524.3352	3524.3352
	2	3878.7593	3878.7593	3878.7593	3878.7593	3878.7593
	3	3993.7483	3993.7407	3993.7483	3993.6653	3993.6785
	4	4054.3541	4054.3541	4054.3464	4053.7051	4053.6721
	5	4093.6530	4094.9001	4094.8126	4092.2922	4094.1539
Ostrich	1	758.1093	758.1093	758.1093	758.1093	758.1093
	2	1074.0847	1074.0847	1074.0847	1074.0847	1074.0825
	3	1139.7067	1139.6988	1139.7081	1139.6612	1139.6370
	4	1178.5513	1178.5488	1178.5513	1178.2177	1178.4752
	5	1203.5090	1203.5025	1203.4767	1202.2677	1203.0531
Starfish	1	1987.2285	1987.2285	1987.2285	1987.2285	1987.2285
	2	2551.9745	2551.9745	2551.9745	2551.9745	2551.9316
	3	2784.2272	2784.2272	2784.2272	2784.1470	2784.1458
	4	2869.4308	2869.4297	2869.4243	2868.7587	2869.0765
	5	2916.2721	2916.2713	2916.1948	2913.9051	2915.8551
Flower	1	1285.6988	1285.6988	1285.6988	1285.6988	1285.6988
	2	1823.5081	1823.5081	1823.5081	1823.5081	1823.4972
	3	1978.3569	1978.3526	1978.3569	1978.3155	1978.3281
	4	2049.3689	2049.3583	2049.3689	2048.8357	2049.0964
	5	2087.6606	2087.6587	2087.5689	2085.5904	2087.1175
Plane	1	436.6506	436.6506,	436.6506	436.6506	436.6506
	2	565.8626	565.8626	565.8626	565.8626	565.8536
	3	608.8348	608.8348	608.8348	608.8317	608.8104
	4	627.2286	627.2286	627.2286	627.0761	627.1837
	5	636.9316	636.9313	636.9120	636.3327	636.7630
Girl	1	1624.3823	1624.3823	1624.3823	1624.3823	1624.3823
	2	2111.6090	2111.6055	2111.6090	2111.6090	2111.6034
	3	2216.1875	2216.1875	2216.1875	2216.1428	2216.0897
	4	2269.2491	2269.2473	2269.2496	2268.7985	2268.9041
	5	2300.1857	2300.1892	2300.1547	2298.8695	2299.8187

Tablo 4. Otsu tabanlı algoritmalarından elde edilen ortalama eşik değerleri

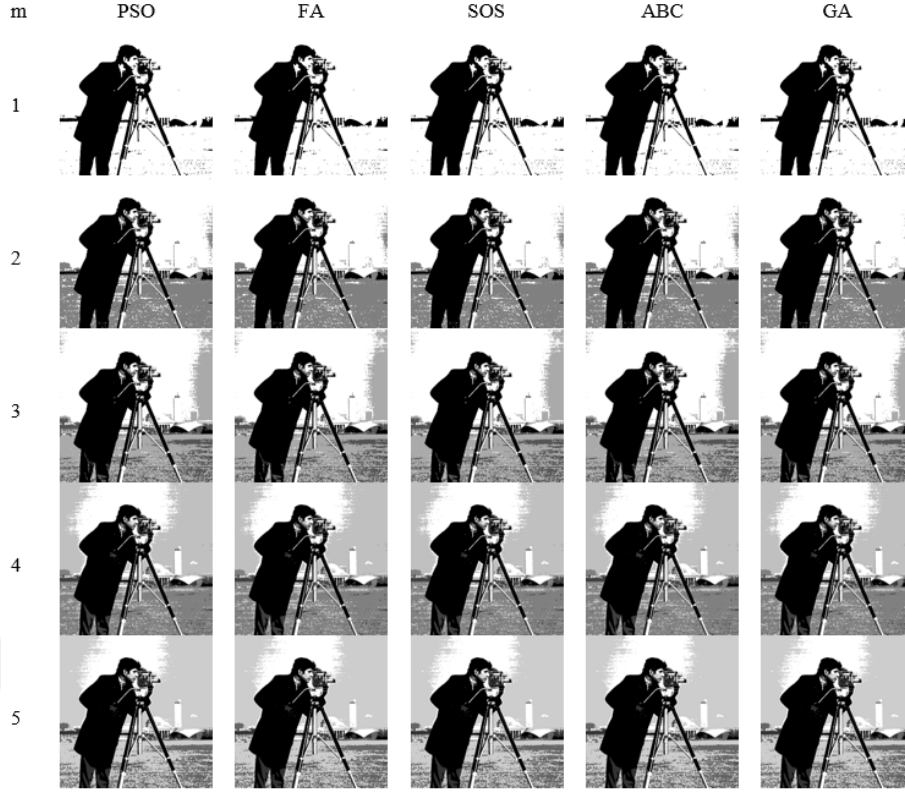
Im	m	Eşik değerleri				
		PSO	FA	SOS	ABC	GA
Cameraman	1	88	88	88	88	88
	2	70 144	70 144	70 144	70 144	70 144
	3	57 116 154	57 116 154	57 116 154	57 116 154	56 115 154
	4	41 94 139 169	41 94 139 169	41 94 139 169	40 93 139 169	41 94 139 169
	5	36 83 122 149 173	36 83 122 149 173	36 83 122 149 173	37 84 123 149 173	36 84 122 149 173
Lena	1	118	118	118	118	118
	2	93 151	93 151	93 151	93 151	93 151
	3	81 127 171	81 127 171	81 127 171	81 127 171	81 127 171
	4	75 114 145 180	75 114 145 180	75 114 145 180	75 114 145 180	75 114 145 180
	5	70 103 131 156 186	73 109 136 160 188	73 108 136 160 188	72 107 134 159 187	72 106 132 157 186
Peppers	1	120	120	120	120	120
	2	68 135	68 135	68 135	68 135	68 135
	3	63 119 166	63 119 166	63 119 166	62 119 166	63 119 166
	4	47 86 126 169	47 86 126 169	47 86 126 169	47 86 126 169	51 91 130 171
	5	43 79 113 146 177	43 79 113 146 177	43 79 113 146 177	43 79 114 146 177	44 80 113 147 178
Airplane	1	153	153	153	153	153
	2	113 173	113 173	113 173	113 173	113 173
	3	93 145 191	93 145 191	93 145 191	93 145 191	92 144 190
	4	84 129 172 203	84 129 172 203	84 129 172 203	85 129 173 203	83 128 171 202
	5	69 107 143 180 205	69 107 143 180 205	69 107 143 180 205	70 107 143 179 205	71 108 143 180 205
Baboon	1	129	129	129	129	129
	2	98 150	98 150	98 150	98 150	98 150
	3	86 125 161	86 125 161	86 125 161	86 125 161	85 124 161
	4	72 106 137 168	72 106 137 168	72 106 137 168	72 106 137 168	72 107 138 168
	5	68 100 126 150 175	68 99 125 150 175	68 100 126 150 175	67 97 123 148 174	67 99 124 149 174
Butterfly	1	135	135	135	135	135
	2	85 156	85 156	85 156	85 156	85 156
	3	82 145 201	82 145 201	82 145 201	82 145 201	82 145 201
	4	78 122 168 208	78 122 168 208	78 122 168 208	78 123 168 208	77 121 167 208
	5	62 92 132 174 210	60 89 129 172 209	60 89 129 172 209	60 89 130 171 210	61 90 130 173 209
Ostrich	1	83	83	83	83	83
	2	75 135	75 135	75 135	75 135	75 135
	3	70 102 150	70 102 150	70 102 150	69 102 150	70 102 150
	4	66 93 126 176	66 93 126 176	66 93 126 176	66 93 127 178	66 94 126 177
	5	56 78 101 131 180	56 78 101 131 180	56 78 101 132 180	56 78 101 131 180	58 81 102 132 181
Starfish	1	116	116	116	116	116
	2	85 157	85 157	85 157	85 157	85 157
	3	69 120 178	69 120 178	69 120 178	69 120 178	69 120 178
	4	60 101 138	60 101 138	60 101 138 187	60 101 138 188	60 101 138
	5	187 52 86 117 150 194	187 52 86 117 150 194	52 86 118 150 194	54 87 118 150 194	187 54 88 118 151 194

Tablo 4'ün devamı

Flower	1	120	120	120	120	120
	2	74 152	74 152	74 152	74 152	74 152
	3	64 107 174	64 107 174	64 107 174	64 107 174	64 107 174
	4	56 88 125 188	56 88 125 188	56 88 125 188	57 88 126 188	57 88 126 188
	5	50 77 103 140 198	50 77 103 140 198	50 77 103 140 198	50 77 104 141 198	52 79 105 142 199
Plane	1	82	82	82	82	82
	2	66 118	66 118	66 118	66 118	66 118
	3	61 109 130	61 109 130	61 109 130	61 109 130	61 109 130
	4	54 96 117 132	54 96 117 132	54 96 117 132	54 96 117 132	55 96 117 132
	5	53 93 112 125 136	53 93 112 125 136	53 93 112 125 136	53 93 111 124 136	53 93 111 124 136
Gir1	1	94	94	94	94	94
	2	67 140	67 140	67 140	67 140	67 140
	3	62 127 162	62 127 162	62 127 162	62 127 162	62 127 162
	4	56 112 144 172	56 112 144 172	56 112 144 172	56 112 143 172	55 111 143 171
	5	46 96 127 153 177	46 96 127 153 177	46 96 127 153 177	48 97 128 153 177	47 96 127 152 176

Tablo 3 ve Tablo 4 incelendiğinde eşik sayısı $m = 1$ ve $m = 2$ olduğunda, Otsu tabanlı tüm algoritmaların amaç fonksiyon ve eşik değeri olarak neredeyse eşit sonuçlara ulaştığı görülmektedir. Eşik sayısı $m = 3$ olduğunda, SOS algoritması PSO veya FA algoritmaları ile eşit veya daha üstün performans göstermiştir. Ayrıca bu performansın ABC ve GA algoritmalarından da daha iyi olduğu görülmektedir. Eşik sayısı $m = 4$ olduğunda, Lena, Airplane, Ostrich, Flower, Plane ve Gir1 görüntüleri için SOS algoritmasının amaç fonksiyon değerleri diğer algoritmaların amaç fonksiyon değerlerinden daha yüksek ya da PSO ve FA sonuçlarına eşit değerler olduğu görülmüştür. Eşik sayısı $m = 5$ seçildiğinde, PSO ve FA algoritmalarının amaç fonksiyon değerleri SOS algoritmasının değerlerinden daha başarılı olduğu görülmektedir. Ancak SOS algoritmasının, ABC ve GA algoritmalarının sonuçlarından daha iyi sonuçlara sahip olduğu da görülmektedir. Sonuç olarak Otsu tabanlı SOS algoritmasının $m=4$ eşik seviyesine kadar eşikleme işlemlerinde başarılı olduğu söylenebilir.

Şekil 27-37 aralığında Otsu tabanlı algoritmalarından elde edilen eşik değerleri ile $m=1, 2, 3, 4$ ve 5 eşik seviyesine kadar eşiklenmiş test görüntüleri gösterilmiştir.



Şekil 27. Otsu tabanlı algoritmalar ile eşiklenmiş Cameraman görüntüleri



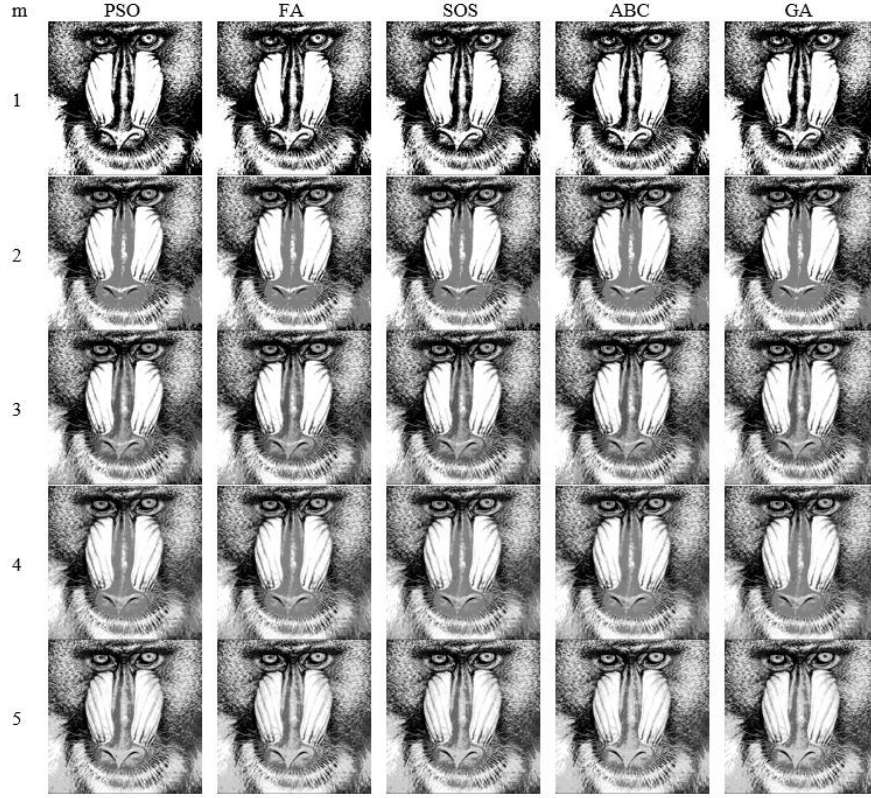
Şekil 28. Otsu tabanlı algoritmalar ile eşiklenmiş Lena görüntüleri



Şekil 29. Otsu tabanlı algoritmalar ile eşiklenmiş Peppers görüntüleri



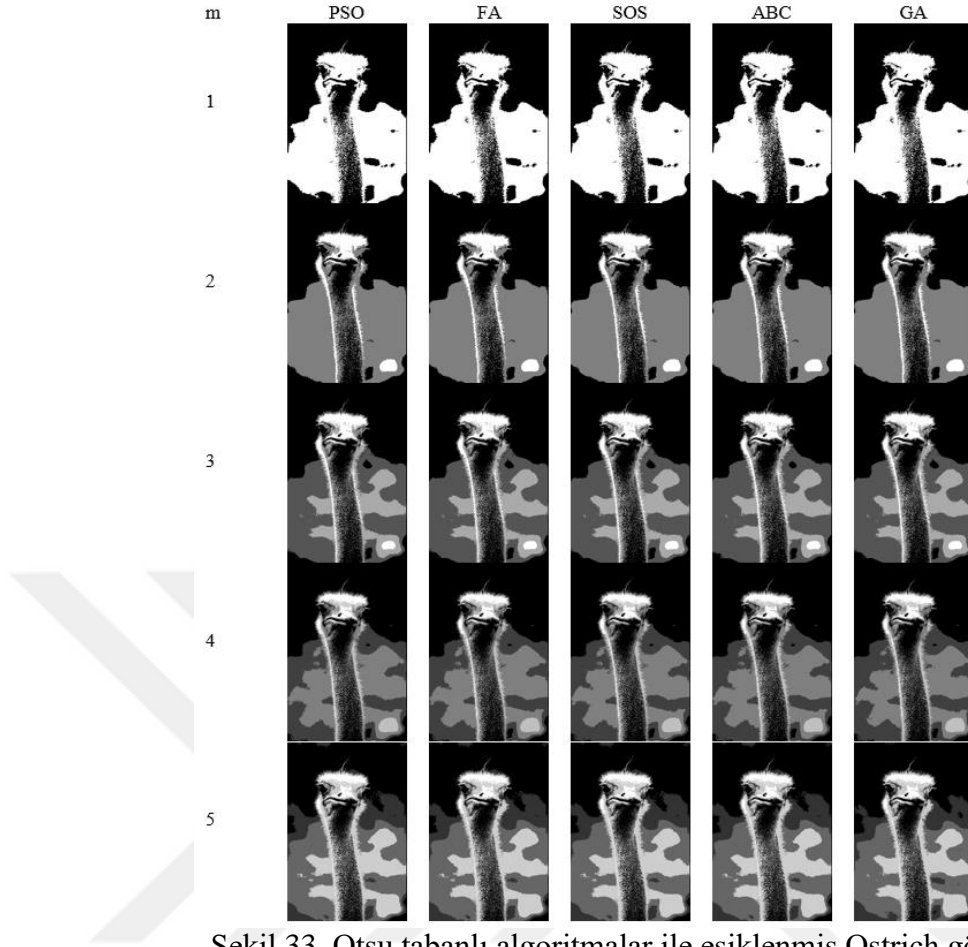
Şekil 30. Otsu tabanlı algoritmalar ile eşiklenmiş Airplane görüntüleri



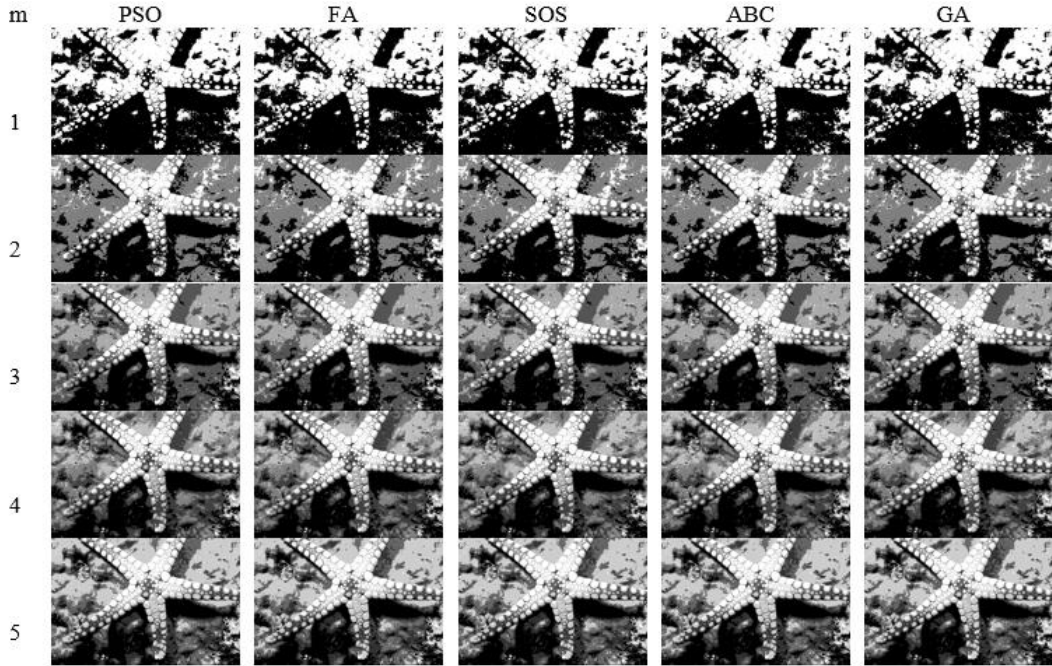
Şekil 31. Otsu tabanlı algoritmalar ile eşiklenmiş Baboon görüntüleri



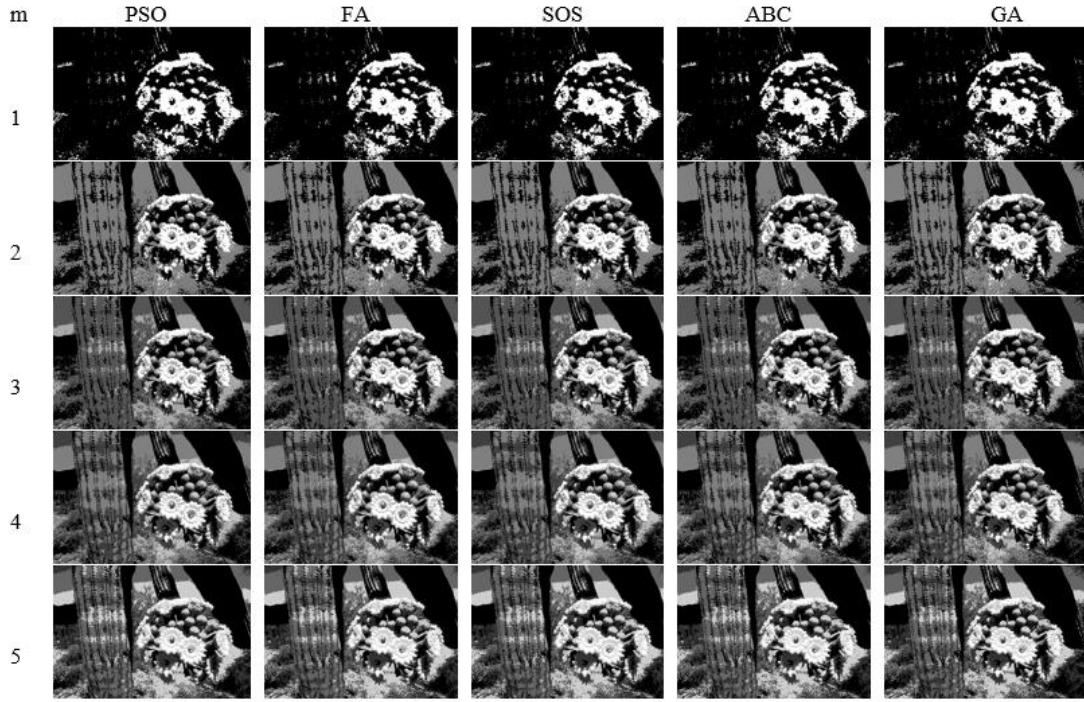
Şekil 32. Otsu tabanlı algoritmalar ile eşiklenmiş Butterfly görüntüleri



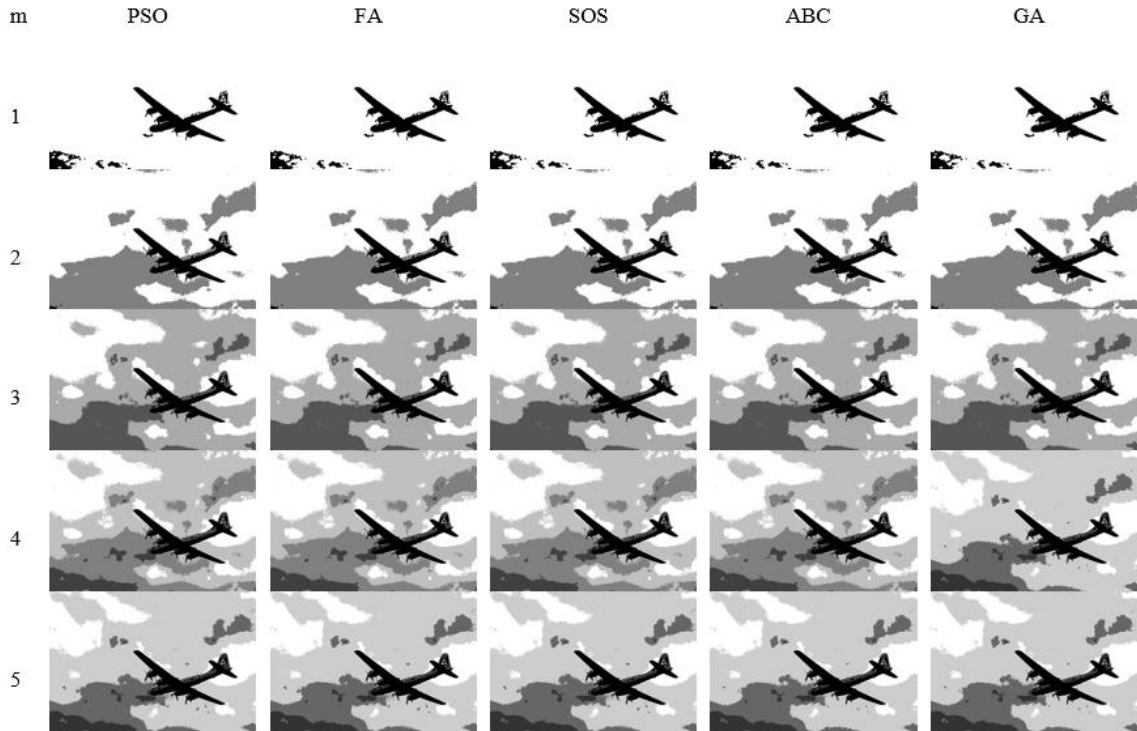
Şekil 33. Otsu tabanlı algoritmalar ile eşiklenmiş Ostrich görüntüleri



Şekil 34. Otsu tabanlı algoritmalar ile eşiklenmiş Starfish görüntüleri



Şekil 35. Otsu tabanlı algoritmalar ile eşiklenmiş Flower görüntüleri



Şekil 36. Otsu tabanlı algoritmalar ile eşiklenmiş Plane görüntüleri



Şekil 37. Otsu tabanlı algoritmalar ile eşiklenmiş Girl görüntüleri

2.7.2. PSNR ve SSIM Değerleri Deneysel Sonuçları

Bölütlenmiş görüntülerde her sınıf, sınıftaki gri seviye değerlerinin ortalamasından hesaplanan bir gri seviye değeri atanarak oluşturulmuştur. Kapur Entropisi tabanlı algoritmalar ile bölütlenmiş görüntüler için hesaplanan ortalama PSNR ve SSIM değerleri sırasıyla Tablo 5 ve Tablo 6'da; Otsu tabanlı bölütlenmiş görüntüler için elde edilen ortalama PSNR ve SSIM değerleri sırasıyla Tablo 7 ve Tablo 8 'de gösterilmiştir. Tablolarda ilk sütunlar görüntüyü, ikinci sütunlar eşik sayısını temsil etmektedir.

Tablo 5. Kapur Entropisi tabanlı algoritmalarından elde edilen ortalama PSNR değerleri

Im	m	PSNR				
		PSO	FA	SOS	ABC	GA
Cameraman	1	12.4282	12.6051	12.4282	12.4282	12.6048
	2	18.2349	18.2309	18.2349	18.2310	18.2233
	3	22.5918	22.2896	22.2572	22.2701	22.4914
	4	26.9163	26.9152	26.9163	26.8968	26.9185
	5	27.7846	27.5441	27.6235	27.5528	27.9257
Lena	1	19.6790	19.6790	19.6790	19.6790	19.6790
	2	22.8152	22.8152	22.8152	22.8152	22.8152
	3	25.9555	25.9555	25.9555	25.9555	25.9394
	4	27.6659	27.6473	27.4876	27.5806	27.8356
	5	28.7840	28.7919	28.7959	28.7657	28.8841
Peppers	1	19.2387	19.2387	19.2387	19.2387	19.2387
	2	22.0925	22.0925	22.0925	22.0925	22.0925
	3	25.0758	25.0758	25.0758	25.0691	25.0648
	4	26.1568	25.8774	25.8024	25.7513	26.0159
	5	27.7707	27.6799	27.6829	27.5955	27.6444
Airplane	1	22.3847	22.3847	22.3847	22.3847	22.3847
	2	23.5612	23.5506	23.5462	23.5462	23.5525
	3	26.5905	26.5900	26.5905	26.5838	26.5841
	4	28.1670	28.1650	28.1670	28.1349	28.1165
	5	29.0242	29.0354	29.0285	29.0575	29.1149
Baboon	1	20.0935	20.0935	20.0935	20.0935	20.0935
	2	23.6537	23.6537	23.6537	23.6537	23.6537
	3	25.1646	25.1280	25.1179	25.1252	25.1561
	4	26.5304	26.5281	26.4541	26.5078	26.6311
	5	28.5474	28.5366	28.5325	28.3981	28.5030
Butterfly	1	19.7296	19.7296	19.7296	19.7296	19.7296
	2	21.5570	21.5572	21.5570	21.5570	21.5563
	3	23.8679	23.7193	22.8217	23.1228	23.8710
	4	26.3734	26.3733	26.3741	26.3468	26.3716
	5	27.3274	27.2701	27.2596	27.2119	27.3400
Ostrich	1	20.0238	20.0238	20.0238	20.0238	20.0238
	2	20.6073	20.6073	20.6073	20.6073	20.6071
	3	26.6639	26.6647	26.6639	26.6654	26.6663
	4	27.1989	27.1843	27.0678	27.0731	27.2478
	5	27.7863	27.5625	27.5580	27.2283	27.5242
Starfish	1	16.8359	16.8359	16.8359	16.8359	16.8359
	2	21.2642	21.2642	21.2642	21.2642	21.2642
	3	24.0990	24.0963	24.0990	24.1049	24.1123
	4	25.5621	25.5859	25.5462	25.5950	25.6815
	5	27.3448	27.3328	27.3399	27.2392	27.3129
Flower	1	18.5385	18.5381	18.5381	18.5381	18.5385
	2	20.0588	20.0588	20.0588	20.0589	20.0607
	3	23.9076	24.1421	24.1421	24.1092	24.1335
	4	25.5347	25.5393	25.5358	25.5480	25.5129
	5	26.7574	26.5716	26.4070	26.6336	26.8638
Plane	1	24.6355	24.6355	24.6355	24.6355	24.6355
	2	27.1156	27.0264	27.0264	27.0264	27.0505
	3	27.6878	27.6809	27.3167	27.3710	28.0087
	4	28.5069	27.7882	27.4014	27.8753	27.8808
	5	30.8269	30.8200	30.8282	30.4859	30.5107

Tablo 5' in devamı

Gir1	1	19.0352	19.0352	19.0352	19.0352	19.0352
	2	19.3561	19.3587	19.3561	19.3552	19.3543
	3	22.7422	22.7422	22.7422	22.7666	22.7450
	4	24.6760	24.9098	25.1693	25.2418	25.0055
	5	27.1428	27.0426	27.0256	26.9616	27.0898

Tablo 6. Kapur Entropisi tabanlı algoritmalarından elde edilen ortalama SSIM değerleri

Im	m	SSIM				
		PSO	FF	SOS	ABC	GA
Cameraman	1	0.5499	0.5533	0.5499	0.5499	0.5533
	2	0.6646	0.6645	0.6646	0.6645	0.6643
	3	0.8015	0.8097	0.8094	0.8094	0.8044
	4	0.8336	0.8339	0.8336	0.8354	0.8335
	5	0.8490	0.8465	0.8491	0.8525	0.8513
Lena	1	0.6427	0.6427	0.6427	0.6427	0.6427
	2	0.6881	0.6881	0.6881	0.6881	0.6881
	3	0.7425	0.7425	0.7425	0.7425	0.7419
	4	0.7765	0.7784	0.7766	0.7763	0.7789
	5	0.7873	0.7873	0.7879	0.7904	0.7899
Peppers	1	0.6268	0.6268	0.6268	0.6268	0.6268
	2	0.6492	0.6492	0.6492	0.6492	0.6492
	3	0.7040	0.7040	0.7040	0.7034	0.7033
	4	0.7223	0.7167	0.7157	0.7145	0.7197
	5	0.7545	0.7531	0.7531	0.7493	0.7515
Airplane	1	0.7832	0.7832	0.7832	0.7832	0.7832
	2	0.7923	0.7920	0.7919	0.7919	0.7920
	3	0.8324	0.8325	0.8324	0.8324	0.8323
	4	0.8615	0.8615	0.8615	0.8613	0.8613
	5	0.8742	0.8742	0.8742	0.8731	0.8749
Baboon	1	0.5937	0.5937	0.5937	0.5937	0.5937
	2	0.7338	0.7338	0.7338	0.7338	0.7338
	3	0.7753	0.7744	0.7741	0.7742	0.7750
	4	0.8051	0.8050	0.8034	0.8048	0.8092
	5	0.8597	0.8594	0.8594	0.8566	0.8583
Butterfly	1	0.6333	0.6333	0.6333	0.6333	0.6333
	2	0.7047	0.7047	0.7047	0.7047	0.7047
	3	0.7724	0.7676	0.7386	0.7483	0.7727
	4	0.8158	0.8158	0.8159	0.8149	0.8158
	5	0.8326	0.8322	0.8321	0.8300	0.8335
Ostrich	1	0.7531	0.7531	0.7531	0.7531	0.7531
	2	0.7720	0.7720	0.7720	0.7720	0.7720
	3	0.8646	0.8647	0.8646	0.8645	0.8647
	4	0.8741	0.8739	0.8729	0.8721	0.8744
	5	0.8825	0.8827	0.8828	0.8767	0.8816
Starfish	1	0.4796	0.4796	0.4796	0.4796	0.4796
	2	0.6133	0.6133	0.6133	0.6133	0.6133
	3	0.7002	0.7001	0.7002	0.7000	0.7005
	4	0.7439	0.7445	0.7434	0.7446	0.7477
	5	0.7924	0.7920	0.7923	0.7907	0.7916

Tablo 6' nın devamı

Flower	1	0.4910	0.4910	0.4910	0.4910	0.4910
	2	0.5716	0.5716	0.5716	0.5716	0.5717
	3	0.7404	0.7497	0.7497	0.7491	0.7497
	4	0.7875	0.7877	0.7875	0.7880	0.7871
	5	0.8168	0.8120	0.8077	0.8140	0.8195
Plane	1	0.9177	0.9177	0.9177	0.9177	0.9177
	2	0.9204	0.9224	0.9224	0.9224	0.9220
	3	0.9211	0.9242	0.9275	0.9268	0.9214
	4	0.9181	0.9237	0.9270	0.9239	0.9241
	5	0.9078	0.9081	0.9079	0.9091	0.9115
Girl	1	0.5013	0.5013	0.5013	0.5013	0.5013
	2	0.5196	0.5206	0.5196	0.5197	0.5198
	3	0.6680	0.6680	0.6680	0.6676	0.6679
	4	0.7216	0.7198	0.7172	0.7117	0.7184
	5	0.7800	0.7758	0.7750	0.7763	0.7786

Tablo 7. Otsu yöntemi tabanlı algoritmalarından elde edilen ortalama PSNR değerleri

Im	m	PSNR				
		PSO	FA	SOS	ABC	GA
Cameraman	1	20.3593	20.3593	20.3593	20.3593	20.3593
	2	24.4332	24.4332	24.4332	24.4332	24.4330
	3	26.0788	26.0791	26.0788	26.0787	26.0729
	4	27.9434	27.9425	27.9434	27.9206	27.9372
	5	29.4573	29.4578	29.4539	29.3463	29.4421
Lena	1	19.7456	19.7456	19.7456	19.7456	19.7456
	2	22.9640	22.9630	22.9640	22.9640	22.9624
	3	26.0312	26.0312	26.0312	26.0311	26.0310
	4	28.1992	28.1992	28.1992	28.1770	28.1916
	5	29.4948	29.5245	29.5134	29.4217	29.5015
Peppers	1	19.2449	19.2449	19.2449	19.2449	19.2449
	2	22.4392	22.4392	22.4392	22.4392	22.4388
	3	25.1272	25.1273	25.1272	25.1270	25.1264
	4	26.7721	26.7714	26.7720	26.7504	26.7205
	5	28.4785	28.4782	28.4774	28.3766	28.4425
Airplane	1	22.5429	22.5429	22.5429	22.5429	22.5429
	2	25.0051	25.0051	25.0051	25.0050	25.0033
	3	27.0145	27.0144	27.0149	27.0128	27.0096
	4	28.8800	28.8797	28.8802	28.8444	28.8677
	5	30.4844	30.4832	30.4856	30.3844	30.4698
Baboon	1	20.5702	20.5702	20.5702	20.5702	20.5702
	2	24.2972	24.2972	24.2972	24.2972	24.2972
	3	26.3391	26.3391	26.3391	26.3375	26.3314
	4	28.2456	28.2456	28.2454	28.2266	28.2306
	5	29.5870	29.5867	29.5828	29.5084	29.5631
Butterfly	1	19.8601	19.8601	19.8601	19.8601	19.8601
	2	23.1179	23.1179	23.1179	23.1179	23.1179
	3	25.0780	25.0780	25.0780	25.0749	25.0775
	4	26.6239	26.6239	26.6237	26.6053	26.6091
	5	28.0433	28.0944	28.0905	27.9844	28.0599
Ostrich	1	21.0812	21.0812	21.0812	21.0812	21.0812
	2	25.3227	25.3227	25.3227	25.3227	25.3225
	3	27.1487	27.1470	27.1478	27.1500	27.1475
	4	28.7640	28.7638	28.7640	28.7475	28.7583
	5	30.2502	30.2489	30.2458	30.1608	30.2008

Tablo 7' nin devamı

Starfish	1	17.9792	17.9792	17.9792	17.9792	17.9792
	2	21.4039	21.4039	21.4039	21.4039	21.4024
	3	24.3540	24.3540	24.3540	24.3528	24.3523
	4	26.2780	26.2775	26.2768	26.2566	26.2680
	5	27.8619	27.8616	27.8587	27.7573	27.8392
Flower	1	18.6539	18.6539	18.6539	18.6539	18.6539
	2	22.7051	22.7051	22.7051	22.7051	22.7053
	3	25.2614	25.2607	25.2614	25.2588	25.2588
	4	27.2436	27.2421	27.2436	27.2213	27.2297
	5	28.8656	28.8656	28.8602	28.7623	28.8280
Plane	1	24.6383	24.6383	24.6383	24.6383	24.6383
	2	28.3823	28.3823	28.3823	28.3823	28.3811
	3	31.0204	31.0204	31.0204	31.0179	31.0205
	4	32.9525	32.9523	32.9523	32.9301	32.9395
	5	34.4458	34.4467	34.4453	34.3520	34.4398
Girl	1	19.3640	19.3640	19.3640	19.3640	19.3640
	2	23.8906	23.8905	23.8906	23.8906	23.8904
	3	26.0674	26.0674	26.0674	26.0659	26.0658
	4	27.8063	27.8068	27.8053	27.7898	27.7978
	5	29.2822	29.2822	29.2798	29.2023	29.2621

Tablo 8. Otsu yöntemi tabanlı algoritmalarından elde edilen ortalama SSIM değerleri

Im	m	SSIM				
		PSO	FF	SOS	ABC	GA
Cameraman	1	0.7404	0.7404	0.7404	0.7404	0.7404
	2	0.7845	0.7845	0.7845	0.7845	0.7846
	3	0.8275	0.8276	0.8275	0.8278	0.8264
	4	0.8352	0.8352	0.8352	0.8337	0.8349
	5	0.8641	0.8642	0.8639	0.8629	0.8643
Lena	1	0.6602	0.6602	0.6602	0.6602	0.6602
	2	0.6916	0.6909	0.6916	0.6916	0.6902
	3	0.7447	0.7447	0.7447	0.7452	0.7450
	4	0.7834	0.7834	0.7834	0.7835	0.7835
	5	0.8074	0.8069	0.8066	0.8043	0.8065
Peppers	1	0.6273	0.6273	0.6273	0.6273	0.6273
	2	0.6728	0.6728	0.6728	0.6728	0.6727
	3	0.7053	0.7053	0.7053	0.7055	0.7054
	4	0.7323	0.7321	0.7323	0.7313	0.7299
	5	0.7651	0.7651	0.7651	0.7647	0.7642
Airplane	1	0.7838	0.7838	0.7838	0.7838	0.7838
	2	0.8261	0.8261	0.8261	0.8261	0.8261
	3	0.8326	0.8327	0.8327	0.8328	0.8338
	4	0.8469	0.8469	0.8469	0.8466	0.8458
	5	0.8666	0.8666	0.8664	0.8641	0.8656
Baboon	1	0.6184	0.6184	0.6184	0.6184	0.6184
	2	0.7559	0.7559	0.7559	0.7559	0.7559
	3	0.8141	0.8141	0.8141	0.8137	0.8126
	4	0.8585	0.8585	0.8585	0.8581	0.8584
	5	0.8889	0.8886	0.8887	0.8854	0.8873

Tablo 8' in devamı

Butterfly	1	0.6355	0.6355	0.6355	0.6355	0.6355
	2	0.7560	0.7560	0.7560	0.7560	0.7560
	3	0.7862	0.7863	0.7862	0.7863	0.7863
	4	0.8231	0.8231	0.8231	0.8225	0.8221
	5	0.8441	0.8445	0.8444	0.8423	0.8440
Ostrich	1	0.7894	0.7894	0.7894	0.7894	0.7894
	2	0.8461	0.8461	0.8461	0.8461	0.8460
	3	0.8593	0.8592	0.8593	0.8592	0.8592
	4	0.8771	0.8771	0.8771	0.8772	0.8772
	5	0.8899	0.8901	0.8898	0.8885	0.8911
Starfish	1	0.5066	0.5066	0.5066	0.5066	0.5066
	2	0.6215	0.6215	0.6215	0.6215	0.6213
	3	0.7121	0.7121	0.7121	0.7121	0.7120
	4	0.7714	0.7714	0.7713	0.7710	0.7714
	5	0.8177	0.8176	0.8175	0.8155	0.8170
Flower	1	0.5153	0.5153	0.5153	0.5153	0.5153
	2	0.7159	0.7159	0.7159	0.7159	0.7159
	3	0.7888	0.7888	0.7888	0.7891	0.7887
	4	0.8393	0.8393	0.8393	0.8396	0.8398
	5	0.8720	0.8720	0.8720	0.8696	0.8702
Plane	1	0.9192	0.9192	0.9192	0.9192	0.9192
	2	0.9017	0.9017	0.9017	0.9017	0.9015
	3	0.9041	0.9041	0.9041	0.9039	0.9037
	4	0.8994	0.8994	0.8994	0.9009	0.9003
	5	0.9161	0.9161	0.9157	0.9141	0.9153
Girl	1	0.4455	0.4455	0.4455	0.4455	0.4455
	2	0.6469	0.6469	0.6469	0.6469	0.6469
	3	0.7620	0.7620	0.7620	0.7621	0.7623
	4	0.8204	0.8205	0.8202	0.8202	0.8202
	5	0.8567	0.8567	0.8564	0.8559	0.8565

Tablolar incelendiğinde eşik sayısı $m=1$ ve $m=2$ olduğunda tüm algoritmaların birbirine eşit veya çok yakın değerler verdikleri görülmektedir. Eşik sayısı $m=3,4$ ve 5 olduğunda Tablo 5 ve Tablo 6' da SOS algoritmasının bazı sonuçlar dışında diğer algoritmalar kadar iyi PSNR ve SSIM değerine sahip olmadığı anlaşılmaktadır. Bu tablolarda PSO ve GA algoritmalarının daha sıklıkta başarılı sonuç ürettiği görülmektedir. Fakat bu sonuçlar Tablo 7 ve Tablo 8' deki sonuçlarla karşılaştırıldığında Otsu tabanlı SOS algoritmasının, Kapur tabanlı SOS algoritmasından daha başarılı PSNR ve SSIM değerlerine sahip olduğu görülmektedir. Otsu tabanlı algoritmaların PSNR değerlerinde PSO, FA ve SOS algoritmalarının başarılı sonuçları sıklıktadır. Otsu tabanlı algoritmaların SSIM değerlerinde ise PSO algoritmasının daha başarılı olduğu söylenebilir. Ayrıca Tablo 6 ve Tablo 8 ' de eşik sayısı arttıkça SSIM değerlerinin de artış gösterdiği görülmektedir. Bu sonuç eşik sayısı arttıkça görüntünün orijinaline daha çok benzediğini gösterir.

2.7.3. Standart Sapma Değeri Deneysel Sonuçları

Algoritmaların kararlılığını ölçmek için Kapur ve Otsu amaç fonksiyonlarının standart sapma değerleri 30 çalıştırılma için hesaplanmıştır. Düşük standart sapma değeri algoritmanın daha kararlı bir yapıya sahip olduğunu gösterir. Kapur ve Otsu tabanlı algoritmalar için hesaplanan standart sapma değerleri sırasıyla Tablo 9 ve Tablo 10’ da gösterilmiştir.

Tablo 9. Kapur amaç fonksiyonuna dayalı algoritmaların standart sapma değerleri

Im	m	Std				
		PSO	FA	SOS	ABC	GA
Cameraman	1	0.07120	0.00126	0.00000	0.00000	0.06559
	2	0.00000	0.00000	0.00000	0.00000	0.00023
	3	0.03442	0.01085	0.00000	0.00130	0.04046
	4	0.00000	0.00028	0.00028	0.00572	0.00109
	5	0.04602	0.01778	0.00859	0.02833	0.04168
Lena	1	0.00000	0.00000	0.00000	0.00000	0.00003
	2	0.00000	0.00000	0.00000	0.00000	0.00013
	3	0.00000	0.00000	0.00000	0.00050	0.00183
	4	0.01034	0.00524	0.00239	0.00482	0.00739
	5	0.00004	0.00022	0.00239	0.01661	0.02192
Peppers	1	0.00000	0.00003	0.00000	0.00000	0.00000
	2	0.00000	0.00000	0.00000	0.00000	0.00010
	3	0.00001	0.00001	0.00000	0.00024	0.00079
	4	0.02040	0.00043	0.00022	0.00863	0.02208
	5	0.04381	0.00287	0.00090	0.02243	0.02580
Airplane	1	0.00000	0.00031	0.00000	0.00000	0.00000
	2	0.00363	0.00052	0.00000	0.00000	0.00001
	3	0.00000	0.00055	0.00000	0.00071	0.00060
	4	0.00000	0.00059	0.00000	0.00400	0.00129
	5	0.00051	0.00077	0.00184	0.01492	0.00425
Baboon	1	0.00000	0.00018	0.00000	0.00000	0.00000
	2	0.00000	0.00000	0.00000	0.00000	0.00003
	3	0.00143	0.00077	0.00000	0.00059	0.00204
	4	0.00193	0.00204	0.00100	0.00318	0.00714
	5	0.00011	0.00097	0.00388	0.01055	0.00927
Butterfly	1	0.00000	0.00000	0.00000	0.00000	0.00000
	2	0.00000	0.00001	0.00000	0.00000	0.00001
	3	0.00344	0.00336	0.00260	0.00274	0.00542
	4	0.00000	0.00005	0.00000	0.00434	0.00172
	5	0.02564	0.00000	0.00624	0.02712	0.03581
Ostrich	1	0.00000	0.00069	0.00000	0.00000	0.00002
	2	0.00000	0.00000	0.00000	0.00000	0.00063
	3	0.00000	0.00000	0.00000	0.00070	0.00038
	4	0.00592	0.00420	0.00478	0.00691	0.00538
	5	0.11356	0.00552	0.00416	0.03146	0.12952
Starfish	1	0.00000	0.00169	0.00000	0.00000	0.00006
	2	0.00000	0.00002	0.00000	0.00000	0.00009
	3	0.00003	0.00003	0.00000	0.00045	0.00077
	4	0.00029	0.00059	0.00018	0.00302	0.00406
	5	0.00081	0.00059	0.00192	0.01391	0.00550

Tablo 9'un devamı

Flower	1	0.00000	0.00000	0.00000	0.00000	0.00035
	2	0.00029	0.00030	0.00000	0.00009	0.00042
	3	0.01542	0.00020	0.00000	0.00075	0.00812
	4	0.00000	0.00038	0.00069	0.00339	0.00301
	5	0.00276	0.00230	0.00169	0.01035	0.00767
Plane	1	0.00000	0.00484	0.00000	0.00000	0.00000
	2	0.01215	0.00000	0.00000	0.00000	0.00641
	3	0.07162	0.02082	0.00000	0.00340	0.05037
	4	0.09124	0.05829	0.00649	0.02037	0.01160
	5	0.01765	0.00817	0.00178	0.04453	0.04363
Girl	1	0.00004	0.00002	0.00000	0.00000	0.00004
	2	0.00000	0.00188	0.00000	0.00012	0.00110
	3	0.00000	0.00000	0.00000	0.00081	0.00028
	4	0.00140	0.00098	0.00078	0.00584	0.00364
	5	0.03501	0.00209	0.00068	0.02114	0.02918

Tablo 10. Otsu amaç fonksiyonuna dayalı algoritmaların standart sapma değerleri

Im	m	Std				
		PSO	FA	SOS	ABC	GA
Cameraman	1	0.00000	0.00000	0.00000	0.00000	0.00000
	2	0.00000	0.00000	0.00000	0.00000	0.00000
	3	0.00000	0.00000	0.00000	0.06579	0.42091
	4	0.00000	0.02005	0.00000	0.38047	0.33855
	5	0.00931	0.01120	0.13524	0.91146	0.29073
Lena	1	0.00000	0.00000	0.00000	0.00000	0.00000
	2	0.00000	0.00000	0.00000	0.00000	0.09990
	3	0.00000	0.00000	0.00000	0.05850	0.21440
	4	0.00000	0.00057	0.00007	0.31695	0.16431
	5	0.68017	0.01162	0.14568	0.93913	0.72055
Peppers	1	0.00000	0.00000	0.00000	0.00000	0.00000
	2	0.00000	0.00000	0.00000	0.00000	0.06833
	3	0.00000	0.00000	0.00000	0.18141	0.06845
	4	0.00000	0.02177	0.04454	0.54054	1.68477
	5	0.00000	0.00000	0.10717	1.01875	0.32652
Airplane	1	0.00000	0.00000	0.00000	0.00000	0.00000
	2	0.00000	0.00850	0.00000	0.00141	0.05081
	3	0.00000	0.00929	0.00000	0.06770	0.13344
	4	0.00096	0.04867	0.00000	0.38167	0.89083
	5	0.00860	0.05816	0.09390	0.70485	0.23723
Baboon	1	0.00000	0.00000	0.00000	0.00000	0.00000
	2	0.00000	0.00000	0.00000	0.00000	0.00569
	3	0.00000	0.00000	0.00000	0.06949	0.19255
	4	0.00000	0.01009	0.01563	0.28944	0.34388
	5	0.01982	0.02467	0.14021	0.70447	0.62921
Butterfly	1	0.00000	0.00000	0.00000	0.00000	0.00000
	2	0.00000	0.03058	0.00000	0.00000	0.03902
	3	0.00000	0.13710	0.00000	0.12307	0.14966
	4	0.00000	0.00444	0.00000	0.46907	0.26330
	5	0.00439	0.02197	0.03107	1.56414	0.58211

Tablo 10 'un devamı

Ostrich	1	0.00000	0.00000	0.00000	0.00000	0.00000
	2	0.00000	0.00000	0.00000	0.00000	0.01008
	3	0.00246	0.01204	0.00000	0.04030	0.08687
	4	0.00110	0.02031	0.00080	0.26530	0.41428
	5	0.00010	0.02583	0.12565	0.73895	0.45084
Starfish	1	0.00000	0.00000	0.00000	0.00000	0.00000
	2	0.00000	0.00000	0.00000	0.00000	0.02984
	3	0.00000	0.00000	0.00000	0.07046	0.06902
	4	0.00557	0.02375	0.00294	0.54109	0.26511
	5	0.00135	0.02619	0.18347	1.09814	0.66760
Flower	1	0.00000	0.00000	0.00000	0.00000	0.00000
	2	0.00000	0.00000	0.00000	0.00000	0.01693
	3	0.00000	0.00808	0.00000	0.09901	0.09407
	4	0.00000	0.01682	0.00707	0.44650	0.23248
	5	0.00121	0.02529	0.12788	0.88794	0.73223
Plane	1	0.00000	0.00000	0.00000	0.00000	0.00000
	2	0.00000	0.00000	0.00000	0.00000	0.00812
	3	0.00000	0.00000	0.00000	0.04033	0.04477
	4	0.00000	0.00150	0.00000	0.12811	0.09581
	5	0.00000	0.02514	0.03117	0.30241	0.23101
Girl	1	0.00000	0.00000	0.00000	0.00000	0.00000
	2	0.00000	0.00000	0.00000	0.00000	0.00417
	3	0.00000	0.00000	0.00000	0.08724	0.15137
	4	0.00559	0.00858	0.00000	0.36526	0.32731
	5	0.00777	0.01219	0.10809	0.78063	0.64695

Tablolar incelendiğinde $m=2$ eşik seviyesine kadar algoritmaların standart sapma değerlerinin birbirlerine yakın ve düşük olduğu görülmektedir. Eşik sayısı $m=3, 4$ ve 5 olduğunda algoritmaların standart sapma değerlerinde artış görülmektedir. Tablo 9 incelendiğinde $m=3, 4$ ve 5 eşik seviyeleri için toplam 33 standart sapma değerinden 25'inde SOS algoritmasının en düşük standart sapma değerlerine sahip olduğu ve Kapur tabanlı algoritmalarından en kararlı yapıya sahip algoritma olduğu söylenebilir. Tablo 9 için Kapur tabanlı algoritmaların kararlılıkları sırasıyla $SOS>PSO>FA>ABC>GA$ şeklindedir. Tablo 10 incelendiğinde Otsu tabanlı algoritmaların kararlılıkları $PSO>SOS>FA>ABC>GA$ şeklinde sıralanabilir.

2.7.4. CPU İşlem Süresi Deneysel Sonuçları

Otsu ve Kapur tabanlı PSO, FA, SOS, ABC ve GA algoritmaları için minimum CPU işlem süresi Tablo 11' de gösterilmiştir. Tabloda, $m=1, 2, 3, 4$ ve 5 eşik seviyesine kadar farklı tabanlarda gerçekleştirilmiş tüm algoritmalar içinde en az hesaplama süresine sahip

olan algoritmanın PSO olduğu görülmektedir. SOS algoritması diğer algoritmalara göre daha fazla hesaplama süresine sahiptir. Ayrıca, Otsu tabanlı SOS algoritmasının Kapur tabanlı SOS algoritmasından yakınsama hızının daha düşük olduğu sonucuna da varılmıştır.

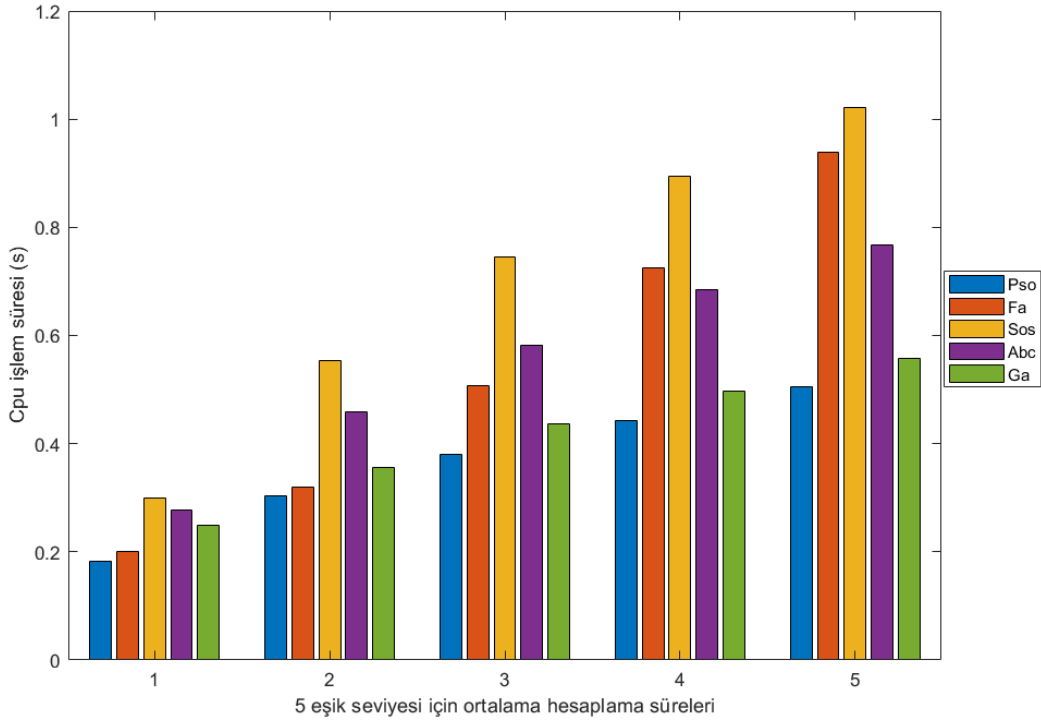
Tablo 11. Farklı tabanlarda gerçekleştirilmiş algoritmaların minimum (s) CPU işlem süresi

Im	m	Kapur Entropisi					Otsu Yöntemi				
		PSO	FA	SOS	ABC	GA	PSO	FA	SOS	ABC	GA
Cameraman	1	0.181	0.199	0.314	0.282	0.252	0.170	0.199	0.261	0.246	0.237
	2	0.304	0.315	0.565	0.472	0.370	0.274	0.294	0.455	0.395	0.338
	3	0.375	0.506	0.758	0.592	0.444	0.337	0.478	0.623	0.507	0.400
	4	0.445	0.693	0.905	0.696	0.509	0.394	0.645	0.787	0.588	0.459
	5	0.514	0.962	1.031	0.783	0.572	0.447	0.886	0.915	0.690	0.511
Lena	1	0.184	0.203	0.298	0.282	0.251	0.171	0.196	0.253	0.250	0.228
	2	0.304	0.307	0.551	0.454	0.366	0.280	0.304	0.463	0.395	0.334
	3	0.379	0.511	0.738	0.592	0.446	0.338	0.496	0.645	0.496	0.391
	4	0.454	0.720	0.893	0.684	0.502	0.405	0.676	0.797	0.613	0.452
	5	0.501	0.930	1.032	0.761	0.552	0.466	0.870	0.941	0.711	0.497
Peppers	1	0.183	0.207	0.296	0.277	0.258	0.172	0.203	0.262	0.258	0.237
	2	0.307	0.317	0.557	0.453	0.363	0.285	0.310	0.442	0.407	0.332
	3	0.378	0.499	0.748	0.567	0.432	0.341	0.492	0.648	0.522	0.405
	4	0.448	0.721	0.912	0.698	0.485	0.399	0.653	0.802	0.609	0.431
	5	0.497	0.970	1.022	0.769	0.566	0.473	0.867	0.967	0.696	0.499
Airplane	1	0.186	0.201	0.295	0.284	0.253	0.173	0.197	0.253	0.256	0.236
	2	0.296	0.318	0.553	0.464	0.371	0.285	0.302	0.466	0.400	0.332
	3	0.385	0.503	0.743	0.587	0.437	0.336	0.498	0.631	0.514	0.411
	4	0.442	0.718	0.903	0.695	0.507	0.408	0.707	0.808	0.639	0.459
	5	0.507	0.917	1.025	0.754	0.556	0.461	0.891	0.945	0.695	0.510
Baboon	1	0.187	0.204	0.306	0.279	0.249	0.175	0.198	0.263	0.248	0.239
	2	0.306	0.360	0.555	0.459	0.374	0.282	0.299	0.461	0.399	0.334
	3	0.382	0.506	0.747	0.561	0.444	0.346	0.476	0.640	0.511	0.401
	4	0.445	0.755	0.899	0.674	0.495	0.401	0.690	0.791	0.608	0.458
	5	0.495	0.923	1.058	0.766	0.553	0.461	0.884	0.961	0.709	0.505
Butterfly	1	0.181	0.194	0.303	0.272	0.241	0.171	0.190	0.255	0.247	0.224
	2	0.303	0.305	0.557	0.457	0.349	0.272	0.300	0.442	0.397	0.316
	3	0.379	0.521	0.746	0.588	0.424	0.340	0.482	0.631	0.509	0.381
	4	0.434	0.720	0.882	0.678	0.492	0.397	0.688	0.791	0.603	0.423
	5	0.509	0.925	1.012	0.757	0.554	0.453	0.890	0.914	0.655	0.484
Ostrich	1	0.185	0.198	0.303	0.276	0.249	0.178	0.204	0.255	0.264	0.228
	2	0.308	0.314	0.564	0.458	0.354	0.285	0.301	0.453	0.435	0.316
	3	0.383	0.518	0.756	0.609	0.440	0.358	0.495	0.636	0.503	0.378
	4	0.446	0.767	0.921	0.695	0.499	0.396	0.699	0.785	0.593	0.434
	5	0.529	0.988	1.040	0.786	0.565	0.452	0.912	0.938	0.675	0.490
Starfish	1	0.176	0.194	0.295	0.274	0.247	0.171	0.194	0.256	0.249	0.230
	2	0.300	0.312	0.560	0.460	0.348	0.266	0.299	0.457	0.404	0.318
	3	0.382	0.521	0.754	0.594	0.433	0.335	0.479	0.634	0.511	0.387
	4	0.448	0.730	0.895	0.669	0.483	0.399	0.682	0.790	0.622	0.442
	5	0.517	0.982	1.037	0.777	0.564	0.462	0.882	0.926	0.686	0.481

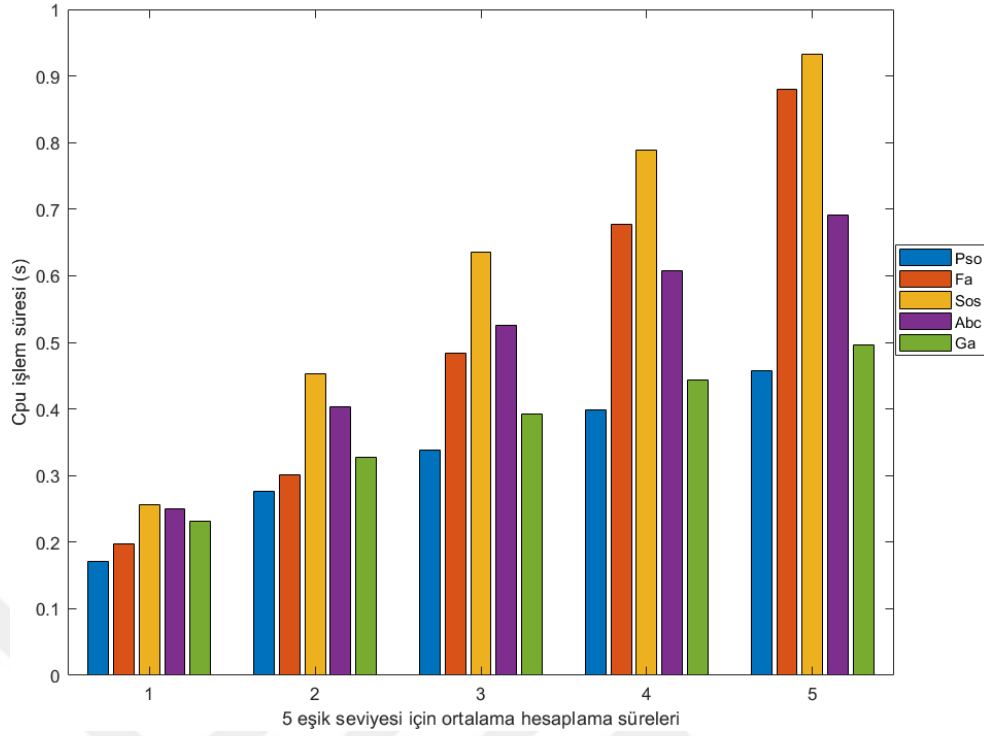
Tablo 11' in devamı

Flower	1	0.188	0.207	0.301	0.274	0.251	0.170	0.196	0.257	0.250	0.234
	2	0.306	0.329	0.540	0.456	0.361	0.277	0.299	0.454	0.409	0.330
	3	0.379	0.489	0.761	0.563	0.438	0.339	0.466	0.639	0.515	0.398
	4	0.438	0.723	0.890	0.686	0.499	0.399	0.659	0.766	0.602	0.444
	5	0.500	0.927	1.015	0.788	0.548	0.455	0.872	0.914	0.693	0.495
Plane	1	0.181	0.198	0.296	0.269	0.238	0.165	0.197	0.258	0.241	0.233
	2	0.296	0.315	0.525	0.444	0.291	0.272	0.304	0.453	0.391	0.333
	3	0.374	0.521	0.709	0.569	0.431	0.336	0.481	0.639	0.694	0.396
	4	0.437	0.705	0.858	0.672	0.498	0.404	0.688	0.774	0.598	0.446
	5	0.489	0.879	0.973	0.739	0.545	0.450	0.861	0.914	0.695	0.497
Gırl	1	0.186	0.202	0.295	0.281	0.253	0.164	0.194	0.249	0.239	0.226
	2	0.298	0.315	0.567	0.466	0.364	0.263	0.301	0.443	0.398	0.315
	3	0.379	0.489	0.734	0.584	0.441	0.323	0.483	0.630	0.504	0.375
	4	0.435	0.720	0.873	0.686	0.502	0.382	0.660	0.787	0.606	0.436
	5	0.506	0.927	1.001	0.755	0.554	0.445	0.858	0.923	0.695	0.492

Tablo 11 'den elde edilen sonuçlara göre 11 farklı görüntü kullanılarak m=1, 2, 3, 4 ve 5 eşik seviyelerine göre ayrı ayrı elde edilen ortalama Cpu işlem süreleri Kapur ve Otsu tabanlı algoritmalar için sırasıyla Şekil 38 ve Şekil 39' da gösterilmiştir.



Şekil 38. Kapur tabanlı algoritmaların ortalama Cpu işlem süresi



Şekil 39. Otsu tabanlı algoritmaların ortalama Cpu işlem süresi

2.7.5. Klasik Yöntemler ile Karşılaştırma

Bu bölümde klasik eşikleme yöntemleri ile eşikleme işleminde en az hesaplama süresine sahip PSO algoritması; amaç fonksiyon, eşik değerleri ve Cpu işlem süresi metrikleri bakımından karşılaştırılmışlardır. Klasik yöntemlerde eşik sayısı arttıkça hesaplama yükü çok arttığından tablolarda sadece $m=4$ eşik seviyesine kadar Cpu işlem süresi hesaplanabilmiştir. Diğer görüntüler için de benzer sonuçlar elde edildiğinden çalışmada kullanılan görüntülerden sadece ilk 4 görüntü için elde edilen sonuçlara yer verilmiştir. Kapur Entropisi ile Kapur tabanlı PSO algoritması eşikleme yöntemlerinin üç farklı metriğe dayalı karşılaştırması Tablo 12' de, Otsu eşikleme yöntemi ile Otsu tabanlı PSO algoritması karşılaştırması Tablo 13' te gösterilmiştir. Tablolarda, ilk sütunlar görüntüyü ve seçilen eşik sayısını, üçüncü ve beşinci sütunlar sırasıyla seçilen eşikleme yöntemi tabanlı PSO algoritmasından 30 çalıştırılma sonucu elde edilen ortalama amaç fonksiyon değerini ve eşik değerlerini, yedinci sütun ise PSO algoritmasının minimum hesaplama süresini göstermektedir. Geriye kalan sütunlar ise sırasıyla klasik eşikleme yönteminden elde edilen amaç fonksiyon değeri, eşik değerleri ve hesaplama sürelerini göstermektedir.

Tablo 12. Kapur Entropisi ve Kapur tabanlı PSO algoritması karşılaştırması

Im	m	Amaç Fonk.		Eşik Değerleri		Cpu Süresi	
		PSO- Kapur	Kapur	PSO-Kapur	Kapur	PSO-Kapur	Kapur
Cameraman	1	8.7868	8.7868	196	196	0.181	0.074
	2	12.2865	12.2865	125 196	125 196	0.304	0.855
	3	15.3744	15.3943	54 111 196	44 102 196	0.375	224.7
	4	18.5567	18.5567	42 96 145 198	42 96 145 198	0.445	48198.0
	5	21.2809	--	34 78 118 161 204	--	0.514	--
Lena	1	8.9413	8.9413	123	123	0.184	0.077
	2	12.3466	12.3466	97 164	97 164	0.304	0.831
	3	15.3183	15.3183	82 126 175	82 126 175	0.379	210.8
	4	18.0016	18.0121	70 107 146 183	64 97 137 179	0.454	45238.0
	5	20.6095	--	63 94 128 163 194	--	0.501	--
Peppers	1	9.1423	9.1423	117	117	0.183	0.105
	2	12.6346	12.6346	75 147	75 147	0.307	0.805
	3	15.6887	15.6887	61 113 165	61 113 165	0.378	203.0
	4	18.5216	18.5392	55 99 141 186	42 77 114 154	0.448	50916.0
	5	21.2730	--	42 77 113 153 193	--	0.497	--
Airplane	1	8.6191	8.6191	162	162	0.186	0.0993
	2	12.2111	12.2115	71 173	71 173	0.296	0.8420
	3	15.5039	15.5039	69 127 183	69 127 183	0.385	210.18
	4	18.3120	18.3120	67 106 145 185	67 106 145 185	0.442	56545.0
	5	20.9087	--	60 89 123 155 187	--	0.507	--

Tablo 13. Otsu yöntemi ve Otsu tabanlı PSO algoritması karşılaştırması

Im	m	Amaç Fonk.		Eşik Değerleri		Cpu Süresi	
		PSO- Otsu	Otsu	PSO-Otsu	Otsu	PSO-Otsu	Otsu
Cameraman	1	3245.1514	3245.1514	88	88	0.170	0.060
	2	3609.5601	3609.5601	70 144	70 144	0.274	0.257
	3	3683.3513	3683.3513	57 116 154	57 116 154	0.337	22.36
	4	3739.2670	3739.2670	41 94 139 169	41 94 139 169	0.394	12022.0
	5	3770.0570	--	36 83 122 149 173	--	0.447	--
Lena	1	1601.2302	1601.2302	118	118	0.171	0.078
	2	1961.8170	1961.8170	93 151	93 151	0.280	0.296
	3	2128.3077	2128.3077	81 127 171	81 127 171	0.338	35.51
	4	2191.8700	2191.8700	75 114 145 180	75 114 145 180	0.405	11250.0
	5	2217.2793	--	70 103 131 156 186	--	0.466	--
Peppers	1	2129.3618	2129.3618	120	120	0.172	0.077
	2	2532.3214	2532.3214	68 135	68 135	0.285	0.331
	3	2703.5715	2703.5715	63 119 166	63 119 166	0.341	21.62
	4	2766.4586	2766.4586	47 86 126 169	47 86 126 169	0.399	12474.0
	5	2810.8415	--	43 79 113 146 177	--	0.473	--
Airplane	1	1792.1172	1792.1172	153	153	0.173	0.080
	2	1948.7191	1948.7191	113 173	113 173	0.285	0.331
	3	2024.8321	2024.8342	93 145 191	93 145 191	0.336	50.20
	4	2070.0769	2070.0771	84 129 172 203	84 129 172 203	0.408	13875.0
	5	2096.1392	--	69 107 143 180 205	--	0.461	--

Tablolar incelendiğinde görüntü için tek eşik seviyesi hesaplanırken klasik yöntemlerin meta-sezgisel tabanlı yöntemden daha az hesaplama süresine sahip olduğu görülmektedir. Ancak eşik sayısı genişletildiğinde hesaplama süresinin klasik yöntemlerde çok arttığı gözlenmektedir. Eşik sayısı $m=4$ 'ten fazla seçildiği durumda hesaplama süresi kabul edilebilir bir sürede değildir. meta-sezgisel tabanlı yöntem ise 1 saniyeden daha az bir süre içerisinde $m=5$ eşik seviyesine kadar hesaplama yapabilmektedir.



3. SONUÇLAR VE TARTIŞMA

Yapılan tez çalışmasında görüntülerin çok seviyeli eşiklenmesine ilişkin literatürde var olan meta-sezgisel tabanlı yöntemler ile 'Berkeley Veri Seti' nden alınan görüntülerin eşikleme işlemi gerçekleştirilmiş ayrıca bölütlemeye kullanılmamış yeni bir meta- sezgisel algoritma olan SOS algoritmasının eşikleme problemlerinde başarımı diğer algoritmalar ile karşılaştırılarak değerlendirilmiştir.

Tez kapsamında, sırasıyla PSO, FA, ABC, GA ve SOS algoritmaları ile Kapur Entropisi ve Otsu eşikleme yöntemlerinin kombinasyonu ile eşikleme işlemleri gerçekleştirilmiştir.

Yapılan çalışmalarda meta-sezgisel algoritmalar kullanılarak gerçekleştirilen eşikleme işlemlerinin hesaplama süresini kısaltmada oldukça başarılı oldukları gözlenmiştir. Elde edilen deneysel sonuçlara göre tüm algoritmaların eşikleme işlemlerinde başarılı sonuçlara sahip olduğu görülmüştür. Ancak PSO, FA ve SOS algoritmalarının amaç fonksiyonu, PSNR ve SSIM ölçüm metriklerinde daha sıklıkta başarılı sonuçlara ulaştığı gözlenmiştir. Ayrıca, bu metrikler bakımından SOS algoritmasının birçok görüntüde PSO ve FA ile aynı veya daha üstün sonuçlara sahip olduğu da görülmüştür. Kapur tabanlı algoritmalarından en kararlı algoritmanın SOS, Otsu tabanlı algoritmalarından PSO olduğu da elde edilen sonuçlar arasındadır. Ayrıca Kapur tabanlı SOS algoritmasının Otsu tabanlıdan daha başarılı eşikleme işlemi gerçekleştirdiği sonucuna varılmıştır. Kapur Entropisi ve Otsu yöntemleri tabanlı algoritmalarından, her iki yöntem için de en kısa işlem süresine sahip algoritmanın PSO algoritması olduğu görülmüştür. Bu sebeple eşikleme işlemlerinde PSO ve SOS algoritmalarının başarısı diğer yöntemlere nazaran ön plana çıkmaktadır.

4. ÖNERİLER

Elde edilen sonuçlara göre meta-sezgisel yaklaşımlarla bölütleme işlemi daha az zaman almakta ve daha iyi sonuçlar vermektedir. Bu bakımdan meta-sezgisel yaklaşımlar bölütleme problemleri için bilgisayarla görme alanında kullanılabilir.

Meta-sezgisel yaklaşımlara her geçen gün bir yenisi eklenmektedir. Önerilen yeni yaklaşımlar bölütleme problemleri için kullanılabilir.

Çalışmada kullanılan meta-sezgisel yaklaşımlar geliştirilerek, hibrit yöntemlerle veya klasik yöntemlerle birleştirilip daha iyi sonuçlar elde edilebilir.

Mevcut SOS algoritması geliştirilerek daha kısa sürede çok seviyeli eşikleme işlemi gerçekleştirebilen daha efektif bir algoritma elde edilebilir.

SOS algoritması ile renkli görüntülerin bölütlemesi gerçekleştirilebilir.

SOS algoritması ile 5' ten fazla eşik seviyesi ile görüntü eşikleme işlemi gerçekleştirilebilir.

5. KAYNAKLAR

1. Li H., Zhang J., Lu Q., Zhou J., Zheng X. ve Tang Z., An Improved Quantum-Inspired Genetic Algorithm for Image Multilevel Thresholding Segmentation, Math. Probl. Eng., 2014 (2014) 1–12.
2. Zhiwei Y., Zhengbing H., Huamin W. ve Hongwei C., Automatic threshold selection based on artificial bee colony algorithm, 3rd Int. Work. Intell. Syst. Appl., May 2011, China.
3. He L. ve Huang S., Modified firefly algorithm based multilevel thresholding for color image segmentation, Neurocomputing, 240 (2017) 152–174.
4. Omran M., Particle Swarm Optimization Methods for Pattern Recognition and Image Processing, Philosophiae Doctor, University of Pretoria, Pretoria, 2004.
5. Pare S., Bhandari A. K., Kumar A., Singh G. K., ve Khare S., Satellite image segmentation based on different objective functions using genetic algorithm: A comparative study, Int. Conf. Digit. Signal Process. DSP, Sept. 2015, Singapore, 730–734.
6. Cao L., Bao P., ve Shi Z., The strongest schema learning GA and its application to multilevel thresholding, Image Vis. Comput., 26 (2008) 716–724.
7. Li Y., Jiao L., Shang R., ve Stolkin R., Dynamic-context cooperative quantum-behaved particle swarm optimization based on multilevel thresholding applied to medical image segmentation, Inf. Sci. (Ny), 294 (2015) 408–422.
8. Ridler T. ve Calvard S., Picture thresholding using an iterative selection method, IEEE Transactions on Systems, Man and Cybernetics, Aug. (1978) 630 - 632.
9. Leung C.K. ve Lam F.K. ,Performance analysis for a class of iterative image thresholding algorithms, Pattern Recognition, 29 (1996) 1523-1530.
10. Trussell H. J., Comments on "Picture Thresholding Using an Iterative Selection Method", IEEE Transactions on Systems, Man and Cybernetics, May (1979) 311–311.
11. Yanni, MK ve Horne, E, A new approach to dynamic thresholding, EUSIPCO'94 9th Eur. Conf. Sig. Process, 1994, 34–44.
12. Sezgin M. ve Sankur B., Survey over image thresholding techniques and quantitative performance evaluation, Journal of Electronic Imaging, January 2004, 146–165.
13. Gonzalez R. C. ve Woods R. E., Digital image processing, Ziya Telatar, Üçüncü Baskıdan Çeviri, Palme Yayıncılık, Ankara, 2014.

14. Glasbey C. A., ve Horgan G. W., Image analysis for the biological sciences, Edinburgh, February 1994.
15. Hassanzadeh T. ve Kanan H. R., Fuzzy FA: A modified firefly algorithm, Appl. Artif. Intell., 28 (2014) 47–65.
16. Hammouche K., Diaf M. ve Siarry P., A multilevel automatic thresholding method based on a genetic algorithm for a fast image segmentation, Comput. Vis. Image Underst., 109, 2 (2008) 163–175.
17. Horng M. H. ve Liou R. J., Multilevel minimum cross entropy threshold selection based on the firefly algorithm, Expert Syst. Appl., 38, 12 (2011) 14805–14811.
18. Otsu N., A threshold selection method from gray-level histograms, IEEE Transactions on Systems, Man and Cybernetics ,(1979) 62–66.
19. Sahoo P.K., Kapur J.N. ve Wong A.K.C., A new method for gray-level picture thresholding using the entropy of the histogram, Computer Vision Graphics Image Processing, (1985) 273–285.
20. Tsai W.H., Moment-preserving thresholding: A new approach, Computer Vision, Graphics, and Image Processing, 29 (1985) 377-393.
21. Luca A. D. ve Termini S, A definition of a nonprobabilistic entropy in the setting of fuzzy sets theory, Information and control, 20, 4 (1972) 301-312.
22. Kullback S., Information theory and statistics, Courier Corporation, United State of America, 1997.
23. Sahoo, P., Wilkins, C., ve Yeager, J., Threshold selection using Renyi's entropy. Pattern recognition, 30, 1 (1997) 71-84.
24. Suresh S. ve Lal S., An efficient cuckoo search algorithm based multilevel thresholding for segmentation of satellite images using different objective functions, Expert Syst. Appl., 58 (2016)184–209.
25. Akay B., A study on particle swarm optimization and artificial bee colony algorithms for multilevel thresholding, Appl. Soft Comput. J.,13, 6 (2013) 3066–3091.
26. Bhandari A. K., Singh V. K., Kumar A. ve Singh G. K., Cuckoo search algorithm and wind driven optimization based study of satellite image segmentation for multilevel thresholding using Kapur's entropy, Expert Syst. Appl., 41, 7 (2014) 3538–3560.
27. Sathya P. D. ve Kayalvizhi R. Optimal multilevel thresholding using bacterial foraging algorithm, Expert Systems with Applications, 38, 12 (2011) 15549-15564.

28. Yang X., Nature-inspired metaheuristic algorithms, Second Edition, Luniver Press, United Kingdom, 2010.
29. Kennedy J. ve Eberhart R.C., Particle swarm optimization, IEEE International Conference on Neural Networks, 1995 , 4, 1942–1948.
30. Yang, X. S., Firefly algorithms for multimodal optimization, In International symposium on stochastic algorithms, October 2009, Berlin, 169-178.
31. Karaboga D., An idea based on honey bee swarm for numerical optimization, Technical report-tr06, Erciyes university, engineering faculty, computer engineering department, 200, 2005.
32. Holland J., Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence, The MIT Press, London, England, 1992.
33. Cheng M.-Y. ve Prayogo D., Symbiotic Organisms Search: A new metaheuristic optimization algorithm, Comput. Struct., 139 (2014) 98–112.
34. Bratton, D. ve Kennedy, J., Defining a standard for particle swarm optimization. In 2007 IEEE swarm intelligence symposium, April 2007, USA, 120-127.
35. Lee C. Y., Leou J. J. ve Hsiao H. H., Saliency-directed color image segmentation using modified particle swarm optimization, Signal Processing, 92, 1 (2012) 1–18.
36. Karaboğa D., Yapay Zeka Optimizasyon Algoritmaları, 5. basım, Nobel Yayıncılık, Ankara, 2017.
37. Osuna-Enciso V., Cuevas E. ve Sossa H., A comparison of nature inspired algorithms for multi-threshold image segmentation, Expert Syst. Appl., 40, 4 (2013) 1213–1219.
38. Abdel-Khalek S., Ben Ishak A., Omer O. A., ve Obada A. S. F., A two-dimensional image segmentation method based on genetic algorithm and entropy, Optik (Stuttg), 131 (2017) 414–422.
39. Yin P.Y. ve Chen, L. H., A fast iterative scheme for multilevel thresholding methods. Signal processing, 60, 3 (1997) 305-313.
40. Yin, P.Y., A fast scheme for optimal thresholding using genetic algorithms, Signal processing, 72, 2 (1999) 85-95.
41. Jinsong, C., Hongqi, W. ve Xiaokuan, Z., Entropic thresholding method using genetic algorithm, In IEEE 1999 International Geoscience and Remote Sensing Symposium IGARSS'99, 1999, Hamburg, Germany, 2, 1247-1249.

42. Liao, P. S., Chen, T. S., ve Chung, P. C., A fast algorithm for multilevel thresholding, J. Inf. Sci. Eng.,17, 5 (2001) 713-727.
43. Kim, B. G., Shim, J. I., ve Park, D. J., Fast image segmentation based on multi resolution analysis and wavelets, Pattern Recognition Letters, 24, 16 (2003) 2995-3006.
44. Yang, Z. H., Pu, Z. B. ve Qi, Z. Q, Relative entropy multilevel thresholding method based on genetic optimization, In International Conference on Neural Networks and Signal Processing, 2003, China, 1, 583-586.
45. Feng, D., Wenkang, S., Liangzhou, C., Yong, D., ve Zhenfu, Z., Infrared image segmentation with 2-D maximum entropy method based on particle swarm optimization (PSO), Pattern Recognition Letters, 26, 5 (2005) 597-603.
46. Yin P. Y., Multilevel minimum cross entropy threshold selection based on particle swarm optimization, Applied mathematics and computation, 184, 2 (2007) 503-513.
47. Yen J.C., Chang F.J. ve Chang S., A new criterion for automatic multilevel thresholding, IEEE Trans. Image Process, IP-4 (1995) 370–378.
48. Maitra M. ve Chatterjee A., A hybrid cooperative-comprehensive learning based PSO algorithm for image segmentation using multilevel thresholding, Expert Syst. Appl., 34, 2 (2008) 1341–1350.
49. Li L. ve Li D., Fuzzy entropy image segmentation based on particle swarm optimization, Prog. Nat. Sci.,18, 9 (2008) 1167–1171.
50. Gao H., Xu W., Sun J., ve Tang Y., Multilevel thresholding for image segmentation through an improved quantum-behaved particle swarm algorithm, IEEE Trans. Instrum. Meas., 59, 4 (2010) 934–946.
51. Horng M. H., A multilevel image thresholding using the honey bee mating optimization, Appl. Math. Comput., 215, 9 (2010) 3302–3310.
52. Chander A., Chatterjee A. ve Siarry P., A new social and momentum component adaptive PSO algorithm for image segmentation, Expert Syst. Appl., 38, 5 (2011) 4998–5004.
53. Horng M. H., Multilevel thresholding selection based on the artificial bee colony algorithm for image segmentation, Expert Syst. Appl., vol. 38, no. 11, (2011), 13785–13791.
54. Tang K., Yuan X., Sun T., Yang J. ve Gao S., An improved scheme for minimum cross entropy threshold selection based on genetic algorithm, Knowledge-Based Syst., 24, 8 (2011) 1131–1138.

55. Ghamisi P., Couceiro M. S., Benediktsson J. A., ve Ferreira N. M. F., An efficient method for segmentation of images based on fractional calculus and natural selection, Expert Syst. Appl., 39, 16 (2012) 12407–12417.
56. Manikandan S., Ramar K., Iruthayarajan M. W. ve Srinivasagan K. G., “Multilevel thresholding for segmentation of medical brain images using real coded genetic algorithm,” Meas. J. Int. Meas. Confed., 47, 1 (2014) 558–568.
57. Bhandari A. K., Kumar A. ve Singh G. K., Modified artificial bee colony based computationally efficient multilevel thresholding for satellite image segmentation using Kapur’s, Otsu and Tsallis functions, Expert Syst. Appl., 42, 3 (2015) 1573–1601.
58. Liu Y., Mu C., Kou W. ve Liu J., Modified particle swarm optimization-based multilevel thresholding for image segmentation, Soft Comput., 19, 5 (2015) 1311–1327.
59. Kurban T., Civicioglu P., Kurban R., ve Besdok E., Comparison of evolutionary and swarm based computational techniques for multilevel color image thresholding, Appl. Soft Comput. J., 23 (2014) 128–143.
60. Huang L., Fang Y., Zuo X., ve Yu X., Automatic Change Detection Method of Multitemporal Remote Sensing Images Based on 2D-Otsu Algorithm Improved by Firefly Algorithm, J. Sensors, 2015 (2015) 1–8.
61. Bhandari A. K., Kumar A., ve Singh G. K., Tsallis entropy based multilevel thresholding for colored satellite image segmentation using evolutionary algorithms, Expert Syst. Appl., 42, 22 (2015) 8707–8730.
62. Chen K., Zhou Y., Zhang Z., Dai M., Chao Y., ve Shi J., Multilevel image segmentation based on an improved firefly algorithm, Mathematical Problems in Engineering, 2016 (2016) 12.
63. Pare S., Kumar A., Bajaj V. ve Singh G. K., A multilevel color image segmentation technique based on cuckoo search algorithm and energy curve, Appl. Soft Comput. J., 47 (2016) 76–102.
64. Pare S., Kumar A., Bajaj V. ve Singh G. K., An efficient method for multilevel color image thresholding using cuckoo search algorithm based on minimum cross entropy, Appl. Soft Comput. J., 61 (2017) 570–592.
65. Khairuzzaman A. K. M. ve Chaudhury S., Multilevel thresholding using grey wolf optimizer for image segmentation, Expert Syst. Appl., 86 (2017) 64–76.
66. El Aziz M. A., Ewees A. A. ve Hassanien A. E., Whale Optimization Algorithm and Moth-Flame Optimization for multilevel thresholding image segmentation, Expert Syst. Appl., 83 (2017) 242–256.

67. Pare S., Bhandari A. K., Kumar A. ve Singh G. K., A new technique for multilevel color image thresholding based on modified fuzzy entropy and Lévy flight firefly algorithm, Comput. Electr. Eng., 70 (2018) 476–495.
68. Martin D., Fowlkes C., Tal D., ve Malik J., A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, University of California Berkeley, California, 2001.
69. Bouaziz A., Draa A. ve Chikhi S., Artificial bees for multilevel thresholding of iris images, Swarm Evol. Comput., 21 (2015) 32–40.
70. Haupt R. L. ve Haupt S. E, Practical genetic algorithms, second ed., New Jersey, John Wiley & Sons, 2004.
71. Wang Z., Bovik A. C., Sheikh H. R. ve Simoncelli E. P., Image quality assessment: from error visibility to structural similarity, IEEE transactions on image processing, 13, 4 (2004) 600-612.

URL-1, <http://sipi.usc.edu/database/>. 1 Mayıs 2019.

ÖZGEÇMİŞ

Büşranur KÜÇÜKUĞURLU 1992 yılında Erzurum’da doğdu. İlköğretimi Rize’ de ortaöğretimi Erzurum’ da tamamladı. 2010 yılında Erzurum Mehmet Akif Ersoy Anadolu Lisesi’nden mezun oldu. 2010 yılında Erciyes Üniversitesi Bilgisayar Mühendisliği Bölümü’nde lisans eğitime başladı ve 2015 yılında mezun oldu. 2016 yılında Gümüşhane Üniversitesi Yazılım Mühendisliği Bölümü’nde Araştırma Görevlisi olarak çalışmaya başladı. 2017 yılında Karadeniz Teknik Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı’nda yüksek lisans eğitime başladı. Halen, Gümüşhane Üniversitesi’ nde araştırma görevlisi olarak çalışmaktadır. İyi derecede İngilizce bilmektedir.