



T.C.
NECMETTİN ERBAKAN
ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



SAVAŞ STRATEJİSİ OPTİMİZASYON
ALGORİTMASIYLA TIBBİ GÖRÜNTÜ
İYİLEŞTİRME

Mehmet BİLGİN

YÜKSEK LİSANS TEZİ

Mekatronik Mühendisliği Anabilim Dalı

Ocak-2024
KONYA
Her Hakkı Saklıdır

TEZ KABUL VE ONAYI

Mehmet Bilgin tarafından hazırlanan “Savaş Stratejisi Optimizasyon Algoritmasıyla Tıbbi Görüntü İyileştirme” adlı tez çalışması .../.../... tarihinde aşağıdaki jüri tarafından oy birliği / oy çokluğu ile Necmettin Erbakan Üniversitesi Fen Bilimleri Enstitüsü Mekatronik Mühendisliği Anabilim Dalı’nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Jüri Üyeleri

Başkan

Prof. Dr. Hüseyin ARIKAN

Danışman

Dr. Yusuf UZUN

Üye

Dr. Onur İNAN

İmza

.....

.....

.....

Fen Bilimleri Enstitüsü Yönetim Kurulu’nun .../.../20.. gün ve sayılı kararıyla onaylanmıştır.

Prof. Dr. Şerife Yurdağül KUMCU
FBE Müdürü

TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

DECLARATION PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

İmza:

Mehmet BİLGİN

Tarih:

ÖZET

YÜKSEK LİSANS TEZİ

SAVAŞ STRATEJİSİ OPTİMİZASYON ALGORİTMASIYLA TIBBİ GÖRÜNTÜ İYİLEŞTİRME

Mehmet BİLGİN

**Necmettin Erbakan Üniversitesi Fen Bilimleri Enstitüsü
Mekatronik Mühendisliği Anabilim Dalı**

Danışman: Dr. Yusuf UZUN

2024, 51 Sayfa

Jüri

Dr. Yusuf UZUN

Prof. Dr. Hüseyin ARIKAN

Dr. Onur İNAN

Tıbbi resimlerde özellikle MR (Manyetik Rezonans) resimlerinde keskinlik değerinin düşük olmasından dolayı resmin kalitesi düşük olabiliyor. Bu da hastalığın teşhisini zorlaştırıyor ve hatta yanlış teşhis konulmasına bile sebep olabiliyor. Bu çalışmada adaptif histogram eşitleme yönteminde gerçek kodlu genetik optimizasyon algoritması ve savaş stratejisi optimizasyon algoritması kullanılarak resimlerin keskinlik değerleri artırılmıştır. Çoklu uygunluk fonksiyonu kullanılmıştır. Görüntünün entropisi, enerjisi, keskinliği, tepe-sinyal-gürültü-oranı, gri seviye eşzamanlı oluşum matrisi ve Sobel kenar özelliklerini çıkarma yöntemleri uygunluk fonksiyonunda kullanılmıştır. Savaş stratejisi algoritması ve gerçek kodlu genetik algoritma elitizm kullanılarak ve kullanılmadan çalıştırılıp hızları ve verdiği sonuçlar kıyaslanmıştır. Bu çalışmada 4 algoritma kıyaslanmıştır. Bunlar: elitizm ile genetik algoritma, elitizm olmadan genetik algoritma, elitizmle savaş stratejisi algoritması ve elitizm olmadan savaş stratejisi algoritması. Bazı resimlerde savaş stratejisi algoritması, gerçek kodlu genetik algoritmaya göre daha iyi sonuçlar vermiştir. Savaş stratejisi algoritmasının bu uygulamada genetik algoritmadan daha hızlı çalıştığı görülmüştür. Elitizm, çoğunlukla sonucu değil ama yakınsama hızını kısaltmıştır.

Anahtar Kelimeler: Genetik Algoritma, Görüntü İyileştirme, Histogram Eşitleme, Optimizasyon Algoritması, Savaş Stratejisi Algoritması, Tıbbi Resimler, Keskinlik Artırma

ABSTRACT

MS THESIS

**MEDICAL IMAGE ENHANCING USING WAR STRATEGY OPTIMIZATION
ALGORITHM**

Mehmet BİLGİN

**THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCE OF
NECMETTİN ERBAKAN UNIVERSITY
THE DEGREE OF MASTER OF SCIENCE
IN MECHATRONICS ENGINEERING**

Advisor: Dr. Yusuf UZUN

2024, 51 Pages

Jury

Dr. Yusuf UZUN

Prof. Dr. Hüseyin ARIKAN

Dr. Onur İNAN

In medical images, especially MRI (Magnetic Resonance) images, the quality of the image may be low due to the low sharpness value. This makes the diagnosis of the disease difficult and may even lead to misdiagnosis. In this study, the sharpness values of the images were increased by using the real-coded genetic optimization algorithm and the war strategy optimization algorithm in the adaptive histogram equalization method. Multiple fitness function was used. Image entropy, energy, sharpness, peak-signal-to-noise-ratio, gray level co-occurrence matrix and Sobel edge feature extraction methods were used in the fitness function. The war strategy algorithm and the real-coded genetic algorithm were run with and without the use of elitism, and their speeds and results were compared. In this study, 4 algorithms were compared. These are: genetic algorithm with elitism, genetic algorithm without elitism, war strategy algorithm with elitism, and war strategy algorithm without elitism. In some images, the battle strategy algorithm gave better results than the actual coded genetic algorithm. The war strategy algorithm has been shown to run faster than the genetic algorithm in this application. Elitism has often shortened the speed of convergence, but not the outcome.

Keywords: Contrast Enhancing, Genetic Algorithm, Histogram Equalization, Image Enhancement, Medical Images, Optimization Algorithm, War Strategy Algorithm

ÖNSÖZ

Tıbbi görüntüleme hastalık ve rahatsızlıkların teşhisinde en önemli araçlardan biridir. Bazı durumlarda yalnızca MR hastanın sorununun belirlenmesine yardımcı olabilir. Bu nedenle araştırmacılar daha iyi görüntüler elde etmek için daha iyi görüntüleme araçları oluşturmanın yeni yollarını bulmaya çalışıyorlar. Ancak tıbbi görüntülerin daha iyi hale getirilmesi için yeni donanımların geliştirilmesi kadar, alınan görüntülerin iyileştirilmesi de bu kısımda önemli bir rol oynamaktadır. Hesaplama süresini azaltmak ve görüntülerin kalitesini artırmak için görüntü iyileştirmede daha iyi ve daha hızlı sonuçlar istenmektedir. Bu makale, tıbbi görüntülerin kontrastını daha hızlı ve daha iyi sonuçlarla artırmak için yeni bir yöntem gösterecektir.

Danışmanım Dr. Yusuf UZUN'a tüm yardımları ve yönlendirmeleri için teşekkür ederim.

Desteklerinden dolayı anne ve babam Zuhâl ve Nazmi BİLGİN'e teşekkür ederim. Ayrıca bu tezdeki matematik konularında bana yardımcı olan ablam Mahi Bilge BİLGİN'e ve bana ilham veren küçük kız kardeşim Yıldız BİLGİN'e de teşekkür etmek isterim.

Mehmet BİLGİN
KONYA-2023



İÇİNDEKİLER

ÖZET.....	iv
ABSTRACT.....	v
ÖNSÖZ.....	vi
İÇİNDEKİLER.....	vii
SYMBOLS AND ABBREVIATIONS.....	ix
1. GİRİŞ.....	1
1.1. Görüntü İyileştirme Yöntemleri.....	1
1.2. Motivasyon.....	3
2. LİTERATÜR TARAMASI.....	5
2.1. Histogram Eşitlemesi.....	5
2.2. Uyarlanabilir Histogram Eşitleme (AHE).....	7
2.3. Kontrast Sınırlı Uyarlanabilir Histogram Eşitleme (CLAHE).....	8
2.4. Parlaklığı Koruyan Çift Histogram Eşitleme (BBHE).....	10
2.5. Seçici Apex Uyarlanabilir Histogram Eşitleme (SLAAHE).....	10
2.6. HDR Kızılötesi Görüntüleme.....	12
2.7. Bulanık Mantık Teknikleri.....	15
2.8. Optimizasyon Algoritmaları.....	16
2.8.1. Giriş.....	16
2.8.2. Optimizasyon algoritmalarına tarihsel bakış.....	16
2.8.3. Optimizasyon sorunlarının türleri.....	18
2.9. Gürültü Azaltma.....	21
3. MATERYAL VE METODOLOJİ.....	22
3.1. Histogram Eşitleme Metodu.....	24
3.2. Uygunluk Fonksiyonu.....	27
3.3. Savaş Stratejisi Optimizasyon Algoritması.....	28
3.3.1. Saldırı Stratejisi.....	29
3.3.2. Rütbe ve Ağırlık Güncelleme.....	29
3.3.3. Savunma Stratejisi.....	30
3.3.4. Zayıf Askerlerin Değiştirilmesi ya da Yer Değişimi.....	30
3.4. Algoritmanın Sözde Kodu.....	30
3.5. Hiperparametreler.....	31
3.6. Savaş Stratejisi Algoritmasında Değiştirilen Yerler.....	31
3.7. Karmaşıklık Matrisi.....	32
4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA.....	34

5. SONUÇ.....	51
6. KAYNAKÇA.....	52
ÖZGEÇMİŞ.....	55



SYMBOLS AND ABBREVIATIONS

Symbols

Simgeleri yazmaya buradan başlayınız ve yazım kılavuzunda belirtildiği şekilde düzenleyiniz. Simgelerin bitiminden sonra, kısaltmalar başlığından önce bir satır boşluk bırakınız.

Abbreviations

Kısaltmaları yazmaya buradan başlayınız ve yazım kılavuzunda belirtildiği şekilde düzenleyiniz.



1. GİRİŞ

Görüntü iyileştirme konusu birçok çalışma alanında önemlidir. Eski ve/veya bozuk görüntülerin iyileştirilmesi, uydu görüntüleri, gözetleme ve güvenlik görüntüleri, tıbbi görüntüler ve su altı görüntülerinin iyileştirilmesi bunlara örnek olarak verilebilir. Görüntü iyileştirme işlemi görüntüde artefakt olmayacak ve bilgi kaybı olmayacak şekilde yapılmalıdır. Görüntü iyileştirme uzaysal alanla sınırlı değildir. Görüntü iyileştirme frekans alanında da yapılabilir. Uzaysal alanda görüntünün pikselleri üzerinde çalışır ancak frekans alanında görüntünün Fourier Dönüşümü alınarak iyileştirme uygulanır. Uzaysal ve frekans alanlarında farklı türde görüntü iyileştirme yöntemleri yapılmaktadır. Bazı örnekler arasında parlaklık/kontrast ayarı, keskinleştirme, gürültü azaltma, görüntü filtreleme, görüntü segmentasyonu, görüntü restorasyonu, renk düzeltme, süper çözünürlük ve HDR görüntüleme yer alır. Bu kategorilerin her birinde, görüntü iyileştirmeyi sağlamak için birden fazla farklı teknik vardır.

1.1. Görüntü İyileştirme Yöntemleri

Bir görüntünün keskinleştirilmesi, farklı yoğunluklardan geçişin daha net hale gelmesi için yapılır. Bu işlemden sonra görüntü daha net bir görünüme kavuşur, bu nedenle bulanık görüntülerde bu yöntem kullanılır. Kenar keskinleştirme yöntemleri arasında Keskinliği Azaltma Maskeleyesi ve görüntünün Dekonvolüsyonu bulunur. Bu yöntemlerin her ikisi de görüntülerin bulanıklığını gidermek için kullanılır.

Gürültü azaltma, bir görüntüde ortam aydınlatması veya kamera donanımı gibi farklı kaynaklardan kaynaklanan gürültüyü azaltmak için uygulanan işlemdir. Farklı gürültü türleri olduğu gibi, bir görüntüde bu gürültüyü azaltmanın da farklı türleri vardır. Tuz ve Biber Gürültüsü, Gauss Gürültüsü, Poisson Gürültüsü ve Benek Gürültüsü bazı örneklerdir. Tuz ve Biber gürültüsü, sanki görüntüye tuz ve karabiber serpmişsiniz gibi görüntü üzerinde beyaz ve siyah noktalardır. Gauss Gürültüsü, yalnızca gerçek piksel değeri yerine piksel değerinin gerçek değeri artı rastgele Gauss fonksiyon değerinin bulunmasıdır.

Poisson gürültüsü görüntünün tespit sürecindeki foton kuantizasyonu ile ilgilidir (Hasinoff, 2014). Olasılık ve istatistikteki Poisson Noktası Süreci şeklinde rastgele gürültü yaratır. SAR (Sentetik Açıklıklı Radar), tıbbi ultrason ve OCT (Optik Koherens Tomografi) gibi tutarlı görüntüleme sistemlerinde benek gürültüsü görülebilir. Benek

gürültüsü, görüntüleme sistemi tarafından üretilen dalgaların girişiminden kaynaklanan, görüntülerde grenli bir yapıdır (Dougherty, 2009).

Görüntü filtreleme 2 alt kategoriye ayrılabilir: Uzaysal alanda görüntü filtreleme ve dönüşüm alanında Görüntü Filtreleme. Uzamsal alanda görüntü, görüntü düzleminde işlenir, yani pikseller üzerinde çalışılır. Ancak bir dönüşüm alanında, önce görüntünün başka bir alana dönüştürülmesi, ardından görüntünün işlenmesi ve ardından işlenmiş görüntünün elde edilmesi için ters dönüşümün yapılması gerekir. Uzamsal filtrelemede, bir filtre maskesi oluşturularak görüntünün her pikselinin çarpımlarının toplamı hesaplanır. Filtre maskesi kayan pencere tekniğindeki gibi hareket ettirilir. Bir dönüşüm alanında filtrelemeyi gerçekleştirmek için, görüntü frekans alanı gibi başka bir alana dönüştürüldükten sonra görüntüye alçak geçiren filtre, bant geçiren filtre veya yüksek geçiren filtre gibi bazı filtreler uygulanır. Daha sonra görüntü, işlenmiş görüntüyü elde etmek için uzaysal alana geri dönüştürülür (Gonzalez & Woods, 2007).

Görüntü segmentasyonu, görüntülerin alt görüntülere bölünmesi için yapılır. Bu yöntem, görüntüyü alt görüntülere bölmek amacıyla görüntüyü bölmek için kullanılır. 2 farklı görüntü segmentasyonu türü vardır. Bunlardan biri, görüntünün kenarlarının birbirinden farklı olması ve görüntünün arka planının ve kenarlarının, kenar algılama, kenar bağlama ve eşikleme yöntemleri kullanılarak arka plandan ayrıştırılabilmesidir. İkinci yol ise benzer özelliklere sahip bölgelerin birlikte segmentlere ayrılmasıdır. Birinci kategori Kenar Tabanlı Segmentasyon, ikincisi ise Bölge Tabanlı Segmentasyon olarak adlandırılmaktadır (Gonzalez & Woods, 2007).

Görüntü restorasyonu, nesnellik açısından görüntü iyileştirmeden farklıdır. Görüntü iyileştirme, uygulamaya bağlı subjektif bir yaklaşım olan görüntünün bazı özelliklerini iyileştirmeye yönelik süreçlerden geçerken, görüntü restorasyonu, görüntünün tıbbi görüntü, uydu görüntüsü veya sanat eseri olmasına bakılmaksızın uygulamalardan bağımsızdır. Görüntü onarımı, bozulan görüntülerin bozulma sürecinin çalışma alanını kapsamaktadır. Bu şekilde bozulmadan önce gerçek görüntüyü elde etmek için bu süreç tersine çevrilir (Gonzalez & Woods, 2007).

RGB (Kırmızı, yeşil, mavi), HSI (Ton, doygunluk, yoğunluk) veya LAB (Parlaklık, a^* , b^*) gibi birden fazla renk uzayı vardır. Bu nedenle renk dönüşümlerinde gri tonlamalı görüntülerde olduğu gibi 1 yerine 3 veya daha fazla değerle ele alınır. Renk dönüşümü, renk uzaylarında bu değerleri değiştirme işlemidir. Renk dönüşümü, spektrumdaki diğer renkleri maskeleyerek görüntüdeki nesnelere tespit etmek için

kullanılabilir. Bu, görüntünün ön planını arka planından ayırmak için kullanılabilir (Gonzalez & Woods, 2007).

1.2. Motivasyon

Bu çalışma, tıbbi görüntülerin uyarlanabilir histogram eşitlemesinin kullanımında 2 farklı optimizasyon algoritmasının karşılaştırılmasını amaçlamaktadır. Görüntü histogramının tanımı, gri tonlamalı bir görüntüdeki piksellerin gri seviye yoğunluklarının frekanslarını gösteren grafikdir. $1024 \times 1024 \times 8$ boyutundaki bir görüntü için maksimum gri seviye yoğunluğu 255, minimum değer ise 0'dır (Pietka, 2000). Histogram eşitleme, görüntünün histogramını düzleştirerek görüntünün dağıtılmış bir histograma sahip olmasını ve dolayısıyla orijinalinden daha yüksek kontrasta sahip olmasını sağlayan bir görüntü iyileştirme yöntemidir (Hummel, 1977). Histogram eşitleme, görüntüleri iyileştirmenin yeni bir yolunu getirdi ancak hızdan yoksundu ve bazen istenmeyen çıktılara neden oluyordu. Uyarlanabilir histogram eşitleme adı verilen yeni ve geliştirilmiş bir histogram eşitleme (Pizer vd., 1987) tarafından önerilmiştir. Bu yöntem daha kısa sürede daha iyi sonuçlar verdi. Ayrıca histogramı eşitlerken kontrastın sınırlandırılması gerektiğini öne sürdü. Bu çalışmada, (Pizer vd., 1987) görüntüdeki gürültünün çoğalmasını önlemek için kontrastın belirli bir değerle sınırlandırılması gerektiği de önerilmiştir. Böylece Kontrast Sınırlı Uyarlanabilir Histogram Eşitleme (CLAHE) doğdu. Görüntü histogramı eşitlenmeden önce yoğunluk değerlerindeki piksel sayısı CLAHE'ye göre kırılmıştır. Kırılan değerler daha sonra eşiği aşmadan 0'dan farklı bir değere sahip olan histogramın tamamına yeniden dağıtılır. (Pizer vd., 1987), çalışmalarında giriş görüntüsünün histogramını alt histogramlara bölünmemiştir. Bu, çıktı görüntüsünün genel parlaklığını değiştirmiştir. Bu sorunun üstesinden gelmek için (Kim, 1997) histogramın 2 alt histograma bölünmesi ve ayrı ayrı eşitlenmesi gerektiği yöntemini önermiştir. Bu yöntem parlaklığın korunması açısından daha iyi sonuçlar vermiştir. Parlaklığın korunması histogram eşitleme açısından büyük bir sorundu; çünkü bu işlem yapay öğeler üretebilir ve görüntüdeki bazı hayati bilgiler kaybolabilirdi. (Kim, 1997), histogramın 2'ye bölünmesi gerektiğini öne sürmüştür. Bu konuyla ilgili daha sonraki çalışmalar (Acharya & Kumar, 2021), bir görüntünün histogramının 4 alt histograma bölünebileceğini ortaya koymuştur. Bu çalışmada da giriş görüntülerinin histogramları 4 alt histograma bölünmüştür. Mevcut çalışmalarda alt histogramlar oluşturulurken histogramın bölünme noktalarının belirlenmesi meta-sezgisel optimizasyon algoritmaları kullanılarak yapılmaktadır. (Acharya & Kumar, 2021), çalışmalarında

optimizasyonda Yönlendirilmiş Arama yöntemi kullanmışlardır. Ancak çalışmamızda Genetik Algoritma ile Savaş Stratejisi Algoritması arasında bir karşılaştırma yapılmıştır. Bu algoritmaların en büyük farklarından biri genetik algortmada popülasyonun yalnızca 2 üyesinin çocuk üyelerin oluşturulmasını etkilemesidir. Ancak savaş stratejisi algoritmasında popülasyonun en iyi ve ikinci en iyileri bir sonraki nesildeki tüm üyelerin oluşumunu etkiler. Bu yaklaşımın avantajları algoritmanın daha hızlı yakınsamasını ve daha iyi sonuçlar bulmasını sağlar. Ayrıca bu çalışma, bir popülasyondaki en iyi üyenin değerlerini genetik algortmadan daha iyi bir şekilde kullanmaktadır.

Farklı yöntemler, histogram eşitlemeyi kullanarak görüntü kontrastını artırır. (Sarkar & Mandal, 2023), (Acharya & Kumar, 2021) ve (Shanmugavadivu & Balasubramanian, 2014) gibi çalışmalarda histogram eşitlemeli meta-sezgisel optimizasyon algoritmaları kullanılmaktadır. Daha büyük boyutlara sahip görüntülerde veya daha büyük sayıdaki görüntülerde yaşanan sorun, hesaplama gücü ve bu görüntüleri iyileştirmek için gereken süredir. Bu çalışmada kullanılan yöntem, görüntü histogramındaki ayrışma noktalarını daha hızlı bulmaktadır. Bu nedenle daha az hesaplama gücü kullanır ve bu da daha kısa sürede daha fazla görüntüyü geliştirme olanağı sağlar. Bu çalışmada savaş stratejisi optimizasyon algoritması kullanılmıştır. Genetik algoritma ile karşılaştırıldığında hız açısından daha iyi sonuçlar vermektedir. En uygun ayrıştırma noktaları daha kısa sürede bulunur. Her iki algoritma da nispeten aynı çıktıları vermiştir. Dolayısıyla bu algoritmalar arasındaki ayırt edici özellik, optimal değerleri bulmak için harcanan zamandır.

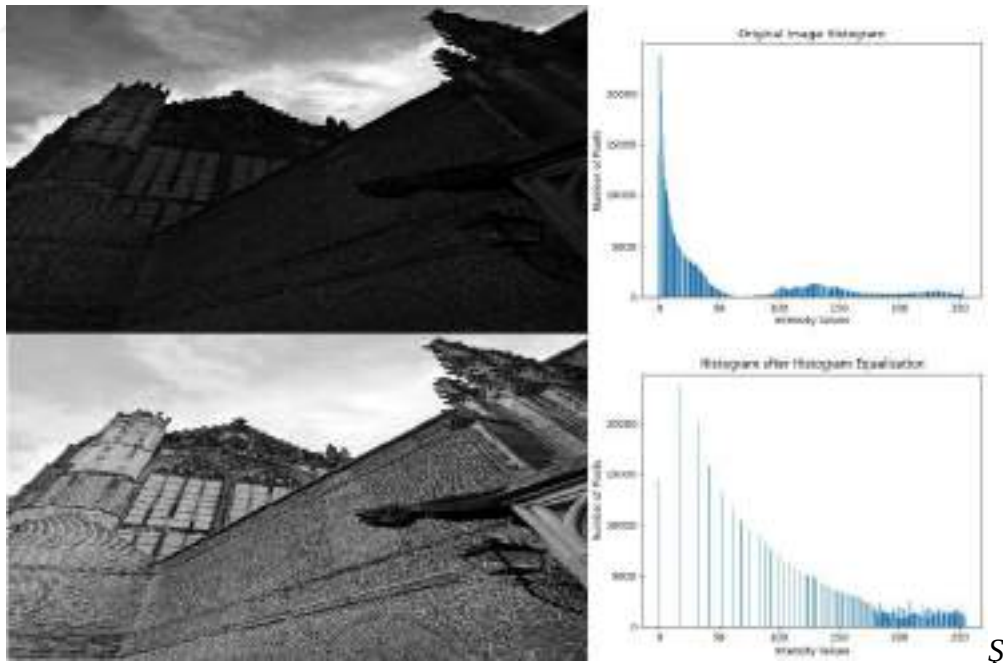
Bu çalışma, savaş stratejisi optimizasyon algoritmasının görüntü kontrastı iyileştirme teknikleriyle çok iyi çalıştığını göstermektedir. Bu şekilde tıbbi, uydu ve HDR görüntüleme gibi sektörlerde görüntülerin sonradan işlenmesi daha hızlı olacak ve daha az güç gerektirecektir. Bu, daha güçlü iyileştirme algoritmalarının kullanılmasına ve dolayısıyla çıktı görüntüsünün kalitesinin artırılmasına yol açabilir.

2. LİTERATÜR TARAMASI

Görüntü iyileştirme, tıbbi uygulamalar, eski ve bozuk görüntü restorasyonu, kalite kontrol uygulamaları, uydu görüntüsü uygulamaları gibi pek çok çalışma alanında önemli bir yere sahiptir. Bu çalışmada uyarlamalı histogram eşitleme tekniği kullanılarak kontrast iyileştirilmesi incelenmiştir. Histogram eşitleme tekniği, özellikle tıbbi görüntüleme sistemlerinde görüntünün iyileştirilmesi söz konusu olduğunda basit ama güçlü bir araçtır. Histogram eşitleme görüntüden daha fazla bilgi çıkarmaz ancak görüntünün sahip olduğu verilerin insan ya da makine tarafından daha iyi algılanabilmesi için görüntüyü netleştirir (Hummel, 1977).

2.1. Histogram Eşitlemesi

Bir histogram bir görüntüdeki yoğunluk değerinin oluşum sayısıdır. 8 bitlik bir görüntü için maksimum yoğunluk değeri 255 ve minimum değer 0'dır. Histogram eşitleme, pikselleri histogramdaki yoğunluk değerlerinin daha geniş bir aralığına yeniden dağıtarak görüntünün kontrastını artırmaya yönelik bir yöntemdir. Bu, orijinal görüntünün iyileştirilmesini sağlar. Bu teknik ilk kez 1977'de tanıtıldı (Hummel, 1977).



ekil 1: Sol üstte orijinal görüntü ve sol altta HE uygulanmış görüntü, sağlarında ise histogramları

Fotoğraf Daniella M.

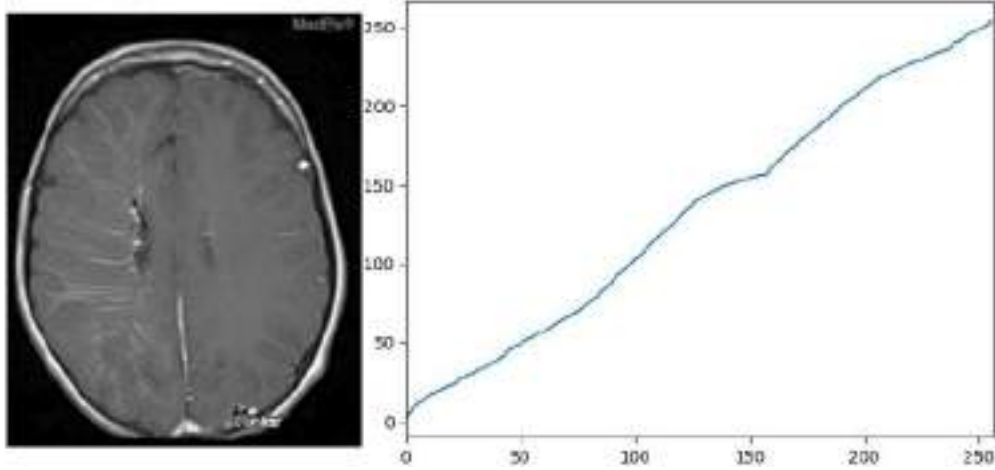
Şekil 1' de sol üst köşedeki orijinal görüntü, sağ üst köşedeki grafik ise o görüntünün histogramıdır. Sol alt köşedeki görüntü, histogram eşitlemeyle geliştirilmiş görüntüdür ve sağ alt köşedeki grafik, bu geliştirilmiş görüntünün histogramıdır. Orijinal histogramın bazı bölgelerde daha yoğun olduğu görülebilmektedir ancak geliştirilmiş görüntünün histogramında yoğunluk değerlerinin spektrum üzerinde iyi dağıldığı açıktır. Orijinal görüntüde gözden kaçan detaylar, geliştirilmiş görüntüde kolaylıkla görülebilmektedir. Histogram eşitleme, Şekil 1' deki görüntü gibi çok düşük veya çok yüksek parlaklığa sahip görüntülere daha uygundur (Pizer vd., 1987). Yoğunluk değerlerinin dağılımı, görüntünün başlangıçta çok düşük veya çok yüksek olan parlaklığına aracılık etmesine neden olur.

Görüntünün ayrıntılarının daha belirgin olması için histogram eşitleme uygulanır. Bir görüntünün histogramını eşitlemek için öncelikle gri tonlamalı görüntünün histogramını elde etmek gerekir. Olasılık Yoğunluk Fonksiyonu (PDF) bir sonraki adımda hesaplanır.

$$P(X_i) = \frac{X_i}{N} \quad (1)$$

X_i yoğunluk değeridir ve N toplam piksel sayısıdır. Daha sonra histogramın Kümülatif Dağılım Fonksiyonunun (CDF) hesaplanması gerekir.

$$C(P_i) = \sum_{k=0}^i (P_k) \quad (2)$$



Şekil 2: Solda beyin MR'si ve sağda onun CDF'si

Burada P histogramın PDF'sidir. Bu, 2' deki gibi bir fonksiyonla sonuçlanacaktır. CDF hesaplandıktan sonra, iyileştirilmiş görüntüyü yeniden oluşturmak için haritalama fonksiyonu uygulanır.

Histogram Eşitleme basit ve hesaplama açısından ucuz bir yöntemdir. Oldukça basit olmasına rağmen, düşük kontrastlı görüntüleri etkili bir şekilde geliştirirken kullanışlıdır. Ancak bu yöntemin en büyük dezavantajı genel görüntünün yanı sıra gürültüyü de arttırmasıdır. Bu, görüntüleri gürültüyle geliştirirken Histogram Eşitlemeyi kötü bir seçim haline getirir. Bu durum, Görüntü Filtreleme ve Görüntü Segmentasyonu gibi gürültü azaltma teknikleriyle birlikte Histogram Eşitleme kullanılarak azaltulabilir. Ancak bu konu, Histogram Eşitleme konusunda Uyarlanabilir Histogram Eşitleme (AHE) ve Kontrast Sınırlı Uyarlanabilir Histogram Eşitleme (CLAHE) gibi farklı yaklaşımlarla ele alınmaktadır (Pizer vd., 1987).

2.2. Uyarlanabilir Histogram Eşitleme (AHE)

Bir görüntüye histogram eşitleme uygulandığında görüntünün bazı kısımları az pozlanmış, bazı kısımları ise aşırı pozlanmış olacaktır. Bunun nedeni görüntünün farklı bölümlerinin farklı dağılımlara sahip olmasıdır. Bu sorunun üstesinden gelmek için Uyarlanabilir Histogram Eşitleme yöntemi önerilmiştir. Bu yöntemde kayan pencere tekniği uygulanmakta ve gridlerin histogramları her piksel için ayrı ayrı eşitlenmektedir. Bu, görüntünün tek bir pikselinin eşlenmesinin elde edilmesi için görüntünün bir bölümünün eşitlenmesi gerektiği anlamına geliyordu. Ancak bu yöntem hesaplama açısından pahalıydı. Çözüm olarak komşuluk yöntemi önerildi (Pizer vd., 1987). Bu yöntemde görüntü örnek noktalara göre ızgaralara bölünür. Bu örnek noktalarla çevrelenen pikseller, bu pikselleri çevreleyen enterpolasyon kullanılarak eşlenir. Kenardaki pikseller özel duruma sahiptirler. Onları çevreleyen iki örnek nokta kullanılarak haritalandırılırlar. Enterpolasyon sırasında yalnızca köşe piksellerine en yakın olan 1 örnek nokta dikkate alınır. Görüntüyü Bağlamsal Bölgeler adı verilen ızgaralara bölerek hesaplama karmaşıklığını basitleştirmek için başka bir yöntem önerilmiştir. Bu yöntemde görüntünün tüm Bağlamsal Bölgeleri eşitlendikten sonra bu bölümler arasında enterpolasyonlar yapılır. Bu yöntemle, farklı konumlardaki kontrast farklılıklarına sahip görüntüler daha az hesaplama gücü gerektirerek daha iyi iyileştirilebilmektedir (Pizer vd., 1987).

Uyarlanabilir Histogram Eşitleme, orijinal Histogram Eşitlemenin geliştirilmiş halidir. Çünkü lokal olarak gözden kaçırılan detaylar AHE ile tutulabilmektedir. Çünkü AHE histogramı lokal olarak eşitler. Bu, AHE'nin görüntünün kontrastını artırırken görüntünün küçük ayrıntılarını korumasına olanak tanır. Orijinal HE yöntemi görüntüyü çok fazla geliştirebilir ve artefaktların oluşmasına neden olabilir. Ancak AHE, yerel

histogram eşitlemeyi kullandığı için görüntünün aşırı geliştirilmesini önler. AHE, eldeki görüntünün içeriğine uyum sağlamaya daha yatkındır. Değişen aydınlatma koşullarına sahip görüntüleri daha iyi geliştirebilir. AHE ayrıca kontrastı artırırken orijinal görüntüye daha yakın sonuçlar yaratır (Gonzalez & Woods, 2007).

Görüntülerin kontrast geliştirme düzeyi Bağlamsal Bölgelerin sayısı artırılarak ayarlanabilir. Görüntüler daha fazla bölüme bölünerek daha az Bağlamsal Bölge Alanına sahip olsaydı, bölgesel ayrıntılar daha fazla geliştirilebilir. Görüntünün Bağlamsal Bölge sayısı bire yaklaştıkça sonuçlar normal histogram eşitlemeye daha benzer olur.

Görüntülerin kontrast geliştirme düzeyi Bağlamsal Bölgelerin sayısı artırılarak ayarlanabilir. Görüntüler daha fazla bölüme bölünerek daha az Bağlamsal Bölge Alanına sahip olsaydı, bölgesel ayrıntılar daha fazla geliştirilebilir. Görüntünün Bağlamsal Bölge sayısı 1'e yaklaştıkça sonuçlar normal histogram eşitlemeye daha benzer olur.

2.3. Kontrast Sınırlı Uyarlanabilir Histogram Eşitleme (CLAHE)

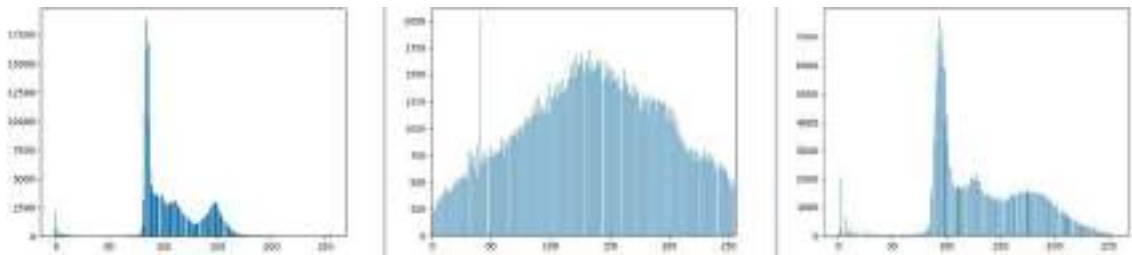
Tıbbi görüntülerde gürültü, iyonize radyasyon görüntüleme için fotonların rastgele doğası, görüntüleme cihazlarındaki elektronik bileşenler, hasta hareketi ve düşük radyasyon dozu vb. gibi farklı nedenlerden dolayı ortaya çıkmaktadır (Mahesh, 2013). Kontrast Sınırlı Uyarlanabilir Histogram Eşitleme tekniği, görüntüdeki gürültünün artmasını önlemek için geliştirilmiştir. Görüntülerdeki nispeten tek tip bölgeler, histogramda aynı yoğunluk değerleri civarındadır. Bu, bazı değerlerin histogramın bazı noktalarında zirveye ulaşmasını sağlar. Bu sorun çözülmeden histogram eşitlenirse artefaktlar ortaya çıkacak ve görüntü iyileşmeyecek ancak bozulacaktır. Şekil 3' te sadece AHE uygulandığında gürültünün de arttığı görünmektedir. Bu nedenle istenmeyen çıktılar oluşacaktır. Ancak AHE ile Şekil 3' teki üçüncü görüntüdeki gibi kontrast sınırlaması uygulandığında sonuçlar çok daha iyi olmuştur (Pizer vd., 1987).



Şekil 3: Sırasıyla orijinal görüntü, AHE uygulanmış görüntü ve CLAHE uygulanmış görüntü

Sonuçlar arasındaki fark Şekil 4' teki görüntülerin histogramında görülmektedir. CLAHE uygulanan görüntünün histogramı, AHE uygulanan görüntünün histogramına göre orijinal görüntüye daha benzerdir. Görüntüler iyileştirilirken giriş ve çıkış görüntülerinin benzer olması istenir. Bu aynı zamanda histogramları için de geçerlidir (Pizer ve diğerleri, 1987). Bu, genel görüntünün değişmediğini, dolayısıyla bilgi kaybının minimum düzeyde olduğunu ve hiçbir yapıt (artefact) oluşmadığını gösterir. Bunu kontrol altında tutmak için görüntü iyileştirme algoritmalarının çoğu, bu çalışmada da kullanıldığı gibi uygunluk fonksiyonlarındaki entropi değişimini kullanır.

Eşleme fonksiyonunun giriş histogramından çıkış histogramına eğimi, kontrast iyileştirmesi olarak düşünülebilir. Daha yüksek eğimler daha fazla iyileştirme anlamına gelirken, 1 eğimi hiçbir iyileştirme olmadığı anlamına gelir. Bu bağlamda kontrast iyileştirme, haritalama fonksiyonunun (CDF) eğimini sınırlayan faktördür. Eşleme işlevi kümülatif işlevle orantılıdır. Kümülatif fonksiyonun türevi hesaplanırsa histogramın kendisini veririrdi. Bu dikkate alındığında kontrast iyileştirmede kümülatif fonksiyonun eğiminin histogramın yüksekliğine bağlı olduğu söylenebilir. Histogramın kırılmasının kontrast artışının sınırlandırılmasıyla sonuçlanmasının nedeni budur (Pizer vd., 1987).



Şekil 4: Sırasıyla orijinal görüntünün histogramı, AHE uygulanmış histogram ve CLAHE uygulanmış histogram

2.4. Parlaklığı Koruyan Çift Histogram Eşitleme (BBHE)

Bir görüntünün histogram eşitlemesi, görüntünün parlaklığını değiştirebilir ve bu da artefaktların oluşması gibi farklı sorunlara yol açabilir. Histogramın dağılımına bağlı olarak görüntünün parlaklığı artabilir veya azalabilir. Bir görüntünün kontrastı düşükse ve görüntünün histogramı ortada yoğunlaşıyorsa, piksellerin daha geniş bir alana yeniden dağıtılması nedeniyle görüntünün parlaklığı artacaktır. Ancak görüntünün kontrastı yüksekse ve görüntünün histogramı iyi bir dağılıma sahipse görüntünün parlaklığı azalacaktır (Gonzalez & Woods, 2007). Bu sorunun üstesinden gelmek için Parlaklığı Koruyan Bi Histogram Eşitleme tekniği önerildi. Bu yöntemde histogramın tamamını eşitlemek yerine görüntü yoğunluklarının ortalaması kullanılarak orijinal görüntüden 2 alt görüntü oluşturulur. Bu 2 alt görüntünün histogramları ayrı ayrı eşitlenir. Bu yöntem, orijinal görüntünün genel parlaklığını koruyarak histogram eşitleme yönteminin, parlaklığı korumanın çok önemli bir rol oynadığı tüketici elektroniği ürünlerine uygulanmasına olanak tanır (Dutta vd., 2021; Kim, 1997).

2.5. Seçici Apex Uyarlanabilir Histogram Eşitleme (SLAAHE)

Görüntü iyileştirmede histogram eşitleme yöntemlerinin farklı türleri vardır. Örneğin (Sarkar & Mandal, 2023) eşitleme öncesinde görüntü ızgaralarının boyutunun ve kontrastın sınırlandırılması gerekip gerekmediğinin belirlenmesi için bir yöntem önermişler ve eğer gerekiyorsa En Az Kare Ortalama algoritması (Least Mean Square Error) kullanarak kontrast sınır seviyesinin ne olması gerektiğini belirlemektedirler. (Sarkar & Mandal, 2023), ultrason görüntüleri üzerinde çalışmışlar ve alt görüntülerin optimum ızgara boyutunu tahmin etmek için Evrensel Görüntü Kalitesi İndeksini (UIQI) kullanmışlar. Ortalama kare hatası gibi geleneksel yöntemleri kullanmak yerine, iyileştirilmiş görüntülerin kalitesini değerlendirmek için geleneksel yöntemlerden önemli ölçüde daha iyi olan UIQI'yi kullanıyorlar. UIQI, bir görüntünün korelasyonunun, kontrastının ve parlaklığının bozulması olan 3 unsurdan oluşur. İyileştirilmiş görüntünün kalitesinin değerlendirilmesi açısından bu çalışmada kullanılan yöntemle benzerlik göstermektedir. Çünkü çok fonksiyonlu bir değerlendirme sürecini kullanmışlardır. (Sarkar & Mandal, 2023)'ın kullandıkları yöntem 3 adımdan oluşmaktadır. İlk adımda görüntünün alt görüntülere ayrıştırılması için en uygun pencere boyutu bulunur. İkinci aşamada, optimum kontrast sınır seviyesi aranır ve seçilen bölümlenmiş görüntülerin histogramlarına CLAHE uygulanır. CLAHE işlemi tüm alt görüntülere uygulanmaz çünkü alt görüntülerin bazıları iyi bir yoğunluk frekans dağılımına sahip olabilir. Bu çalışmayı ilgili diğer çalışmalardan farklı kılan

özelliklerden biri de budur. (Sarkar ve Mandal, 2023), H ve W görüntünün yüksekliği ve genişliği olmak üzere 5×5 'ten $(H/4 \times W/4)$ 'e kadar değişen boyutlardaki alt görüntülere CLAHE uygulamaktadır.

Her alt görüntünün kırpma sınırı, En Küçük Ortalama Kareler (LMS) algoritması uygulanarak bulunur. Bu algoritmada girdi olarak görüntü histogramının dağılım fonksiyonu kullanılır ve çıktı olarak maksimum hata e seçilir. Başlangıç ağırlığı, kümülatif dağılım fonksiyonundan haritalama fonksiyonuna kadar korelasyon ve çapraz korelasyon matrisleri kullanılarak hesaplanır. Ağırlıkların fonksiyonunu güncellemek için En Dik İniş (Steepest Descent) yöntemi uygulanır. Gradyan iniş yöntemleri her zaman optimum gradyanla hatanın azalma yönünde hareket eder. Bu nedenle olasılık yoğunluk fonksiyonu şu şekilde türetilir:

$$\bar{P}_i(k) = P(x) - \bar{P}_{i-1} \quad (3)$$

Yeni keşfedilen olasılık yoğunluk fonksiyonu kullanılarak alt görüntülerin her biri için kırpma sınırı çıkarılabilir.

Alt görüntüye CLAHE uygulanıp uygulanmayacağına karar vermek için ilk adımda alt görüntünün histogramı hesaplanır. Daha sonra maksimum derecesi 7 olan çeşitli sayıda polinom eğrileri çizilir. Maksimum polinom derecesi 7 seçilir çünkü derecesi 7'den yüksek olan polinomlar daha az ayarlanmış R-kare değerleri (Adjusted R-Square values) sağlar. Düzeltilmiş R-kare değeri, tahmin modeline eklenen ek bağımsız değişkenlerin yardımcı olup olmadığını kontrol etmek için kullanılır. Bu yöntemle, yardımcı olmayan değişkenler tahmin modellerinde hesaba katılmaz. Polinom eğrilerinden histograma en iyi uyum, ayarlanmış R-kare değerleri kullanılarak seçilmiştir. Yerel maksimum noktalarını bulmak için polinom eğrisinin birinci ve ikinci dereceden türevleri hesaplanmıştır. Önceki adımda bulunan kırpma sınırı X eksenini çizmek için kullanılmıştır. Yerel maksimumların her iki tarafındaki yerel minimumlar bulunmuştur ve aralarındaki mesafe hesaplanmıştır. Yerel maksimumların her iki tarafındaki yerel maksimumlar ile yerel minimumlar arasındaki mesafelerden biri, yerel minimumlar arasındaki mesafelerin iki katından fazla ise, o alt görüntüye CLAHE uygulanmıştır. Aksi takdirde mevcut alt görüntünün iyi bir histogram dağılımına sahip olduğu düşünülmüş ve CLAHE uygulanmamıştır. Bu, hesaplama süresinden ve gücünden tasarruf edilmesine yardımcı olmuştur (Sarkar & Mandal, 2023).

Bu çalışma çoğunlukla ultrason görüntülerini ele alıyor ve onlarla yeterli derecede iyi sonuçlar yaratmış gibi görünüyor. Global Histogram Eşitleme (GHE), CLAHE, Medyan Ortalama Tabanlı Alt Görüntü Kırılmış Histogram Eşitleme

(MMSICHE) ve Minimum Ortalama Parlaklık Hatası Bi-Histogram Eşitleme (MMBEBHE) ile karşılaştırıldığında; önerilen yöntem (Sarkar & Mandal, 2023), UIQI, Mutlak Ortalama Parlaklık Hatası (AMBE), Çok Ölçekli Yapısal Benzerlik İndeksi Ölçümü (MSSSIM), Entropi ve Ortalama Karekök Hata (RMSE) metriklerinde daha iyi sonuçlara sahiptir.

Bu çalışmadaki yöntemle karşılaştırıldığında, önceki çalışma görüntünün tam histogramının bölümleriyle ilgilenmez ancak tam görüntünün bir alt görüntüsünün histogramıyla ilgilenir. Bu anlamda orijinal uyarlamalı histogram eşitleme yöntemine daha çok benzemektedir. Orijinal görüntüyü veya histogramı, histogram eşitleme konusunda konularına veya içeriklerine göre parçalara bölmek yaygın bir uygulamadır. Bunun nedeni, HE tekniğinin görüntünün tamamına uygulanmasından kaynaklanan global histogram eşitlemenin dezavantajıdır. Bu nedenle görüntünün parlaklığı değişir ve tekdüze (uniform) gürültü, çıktı görüntüsünü daha fazla etkiler.

2.6. HDR Kızılötesi Görüntüleme

Histogram eşitleme yöntemi yalnızca tıbbi görüntülerle sınırlı değildir. Kızılötesi görüntülerde de yaygın olarak kullanılır. Kızılötesi görüntüleme, tespit, keşif ve izlemede kullanılmaktadır (Zhang vd., 2023). Kızılötesi görüntüler, kızılötesi ışığı algılayan kızılötesi kameralar tarafından yakalanır. Bu kameralar 14 bit veri ile oldukça detaylı görüntüler yakalayabilmektedir. Bu mükemmel bir sonuç gibi görünebilir ancak aşılması gereken yeni zorlukları da beraberinde getirmektedir. Çoğu cihaz, bir görüntüyü görüntülemek için 8 bit kullanır. Bu, 14 bitlik bir kızılötesi görüntünün 8 bitlik bir cihazda görüntülenmeye çalışılması durumunda görüntünün saturasyona uğrayacağı anlamına gelir. Önemli bilgi kaybı yaşanacaktır. Bu sorunun üstesinden gelmek için ton haritalama yöntemleri önerilmiştir. Ancak kızılötesi görüntüler üzerinde çalışırken görüntülerin düşük kontrasta, yüksek dinamik aralığa (HDR) ve bulanık kenarlara sahip olduğu dikkate alınmalıdır (Zhang vd., 2023).

Global ve yerel ton eşlemeleri bu konuda kullanılan geleneksel yöntemlerdir. Doğrusal germe (LS) ve histogram eşitleme, HDR kızılötesi görüntüler için küresel haritalama seçenekleriydi. LS yönteminde gri seviyeler için minimum ve maksimum olmak üzere 2 eşik değeri kullanılmış ve daha sonra gri seviye aralığını $[0, 255]$ 'e dönüştürmek için doğrusal haritalama kullanılmıştır. HE aynı zamanda HDR kızılötesi görüntünün gri seviyeli dinamik aralığını 8 bitlik gri seviye değerine ayarlamak için de kullanılır. Ancak bu, ön plandaki görüntülerin ayrıntılarını iyileştirmek yerine, görüntünün arka planında olan görüntünün tekdüze kısımlarını geliştirir. Bu sorunun

üstesinden gelmek için Plato histogram eşitlemesi (PHE) önerilmiştir (Vickers, 1996; Zhang vd., 2023). PHE yöntemi, orijinal istatistik histogramını bir sınır değeri kullanarak doyurur (sature eder). PHE ayrıca histogramı eşitlerken ton eşlemesini de yapar. Eşik sınırı nedeniyle görüntünün arka plan gürültüsü çok az iyileştirilir. Plato eşik sınırı sonsuza yaklaşırken PHE normal HE'ye dönüşecektir. Bu nedenle arka plandaki gürültü etkili bir şekilde bastırılmayacaktır. PHE eşiği çok düşük bir değere ayarlanırsa görüntünün iyileştirilmesinden sonra önemli miktarda bilgi kaybı yaşanacaktır. Bu sorunun üstesinden gelmek için çift plato histogram eşitlemesi (DPHE) önerilmiştir (Song vd., 2008; Zhang vd., 2023). Bu yöntemde 2 eşik sınırının olması gerektiği önerilmiştir. Bir üst ve bir alt sınır, sinyal ve görüntü işleme konularında bir bant geçiren filtre gibidir (Gonzalez & Woods, 2007). Üst sınır, görüntünün arka plan gürültüsünü basturarak iyileştirme adımında gürültünün önlenmesini sağlar. Alt sınır, kızılötesi görüntüdeki loş-küçük hedeflerin güçlendirilmesi için ayarlanır. Loş-küçük hedefler, görüntüdeki düşük sıcaklık veya zayıf ısı yansıtma özelliklerine sahip nesnelere sahiptir. Bu nesnelere, bu nedenle kızılötesi görüntülerde karanlık ve düşük kontrastlı olarak görülürler. DPHE, görüntünün histogramına daha düşük bir sınır koyarak bu sorunun üstesinden gelir. PHE ve DPHE'nin dezavantajı, dönüşüm matrisinin hiçbir ayırım yapılmadan görüntünün tamamı için oluşturulmasıdır. Bu, görüntünün aşırı iyileştirilmesine ve görüntünün arka plan gürültüsünün artmasına neden olabilir.

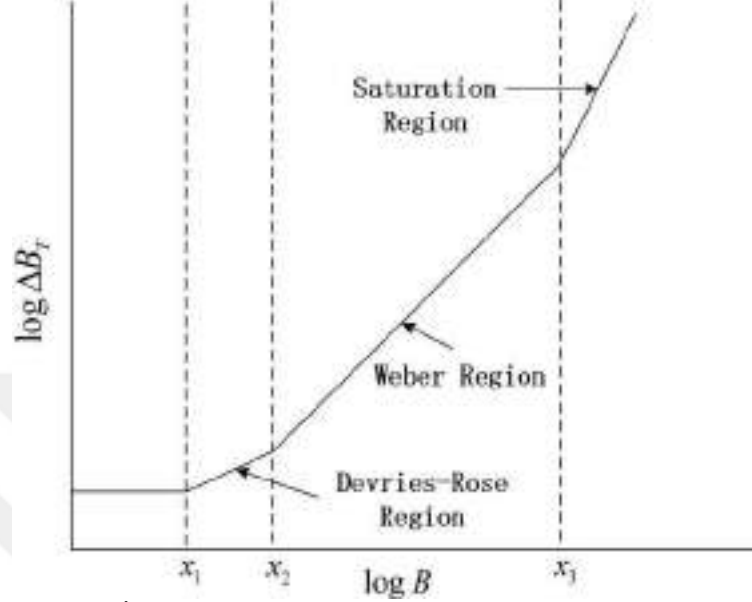
PHE yöntemini uygulamak için histogramın PDF'sinin formül 4 kullanılarak hesaplanması gerekir. Bundan sonra PDF'nin formül Hata: Başvuru kaynağı bulunamadı kullanılarak doyurularak değiştirilmesi gerekir. T , PHE için eşiktir. PDF değeri T 'yi aşarsa değer T olarak değiştirilir. T 'den küçükse aynı kalır.

$$P_M(X_i) = \begin{cases} P(X_k) & \text{for } P(X_k) \leq T \\ T & \text{for } P(X_k) > T \end{cases} \quad (4)$$

Yukarıda belirtildiği gibi DPHE üst ve alt limitler için 2 eşik değeri kullanır. Bununla birlikte, bu yöntemlerin ana dezavantajı, HDR görüntülerinin yoğunluk değerlerini normal görüntüleme cihazlarının 8 bitlik bir değerine tonlamak için birleşik bir dönüşüm işlevi kullanmalarıdır. Bölgesel HE, ister histogram açısından ister konum açısından olsun, (Pizer vd., 1987), (Acharya & Kumar, 2021) veya (Sarkar & Mandal, 2023)'de gösterildiği gibi daha iyi sonuçlar verir. Bu nedenle (Zhang vd., 2023), insanın görme algısını temel alan başka bir yöntem önermiştir.

ΔB_T , insan gözünün hedef ile arka plan arasında algılayabildiği minimum parlaklık farkıdır. Buchsbaum eğrisi, insan gözünün görüntüdeki parlaklık kısımlarını

sınıflandırdığı alanları tanımlar (Buchsbaum, 1980; Davis vd., 2013; Gödrich vd., 2023; Kundu & Pal, 1986; Zhang vd., 2023). 3 bölüme ayrılmıştır: sırasıyla az aydınlatılmış, iyi aydınlatılmış ve aşırı aydınlatılmış bölgeleri gösteren Devries-Rose Bölgesi, Weber Bölgesi ve Doygunluk Bölgesi.



Şekil 5: İnsan gözünün algıladığı 3 bölge

Infrared Physics and Technology

Şekil 5'te bölgeler gösterilmektedir. Görüntü Buchsbaum eğrisine göre 3 parçaya bölünür ve her görüntüye ayrı ayrı PHE uygulanır. Bundan sonra görüntü geliştirilmiş bir görüntü olarak yeniden düzenlenir. Farklı HDR görüntü türleri, Buchsbaum eğrisindeki bölgelerin sabit sınırlarını belirlemeyi zorlaştırır. HDR görüntüsünün arka planının yoğunluk aralığı, farklı sahnelere ve ekipmanlara göre değişir. Bu nedenle konuya daha uyarlanabilir bir yöntemle yaklaşılması gerektiği öne sürülmüştür. (Zhang vd., 2023) doğrusal haritalama sonrasında görüntünün ortalama parlaklığının kullanılması gerektiğini önermiştir.

Yüksek sıcaklık farklılıklarına sahip nesnelerin bulunduğu görüntülerde HDR görüntü histogramının aralığı daha fazladır. Bu, orijinal görüntü histogramında boş yoğunluk değerleri ile sonuçlanır. HE, PHE ve DPHE gibi diğer yöntemlerle karşılaştırıldığında; bu yöntem, daha iyi yerel kontrast ve daha verimli dinamik aralık azaltmayla daha iyi sonuçlar vermiştir.

2.7. Bulanık Mantık Teknikleri

Bulanık teknik ilk olarak 1965 yılında Lütü Aliasker Zade adlı bir matematikçi tarafından ortaya atılmıştır. Bilgisayar mantığı birler ve sıfırlarla ilgilenirken bulanık teknik, birler ve sıfırlar arasındaki sayılarla da ilgilenen, daha insani algıya dayalı bir yaklaşımı benimser. Bulanık teknikte bir girdinin hepsinde yer aldığı ancak üyelik katsayılarına sahip olduğu düşünülen kategoriler bulunmaktadır. Örneğin hava 24°C ise aynı anda ılık, soğuk ve sıcak kategorilerin üyesi olabilir ancak üyelik katsayıları ile eğer-o halde kurallarıyla çalışır. Örneğin hava sıcaksa klimayı açın. Hava soğuksa ısıtıcıyı açın. Ancak girdi tarafında kategoriler olduğu gibi çıktı tarafında da kategoriler var. Yukarıdaki örneklerde bu kategoriler fan hızına göre ayarlanabilir. Fan hızı kategorileri yavaş, orta ve hızlı olabilir. Bu kurallar konunun uzmanı tarafından belirlenmelidir. Bulanık yöntem 2 yaklaşım vardır: Bulanık çıkarım sistemi (FIS) oluşturmak için Mamdani ve Takagi-Sugeno (TSK) yöntemleri. Bu yöntemler kural koyma açısından birbirinden farklılık göstermektedir. Mamdani bulanık çıkarım sistemi, daha önce açıklandığı gibi kuralları, yani eğer-o halde kurallarını belirler. Mamdani FIS'teki her kuralın çıktısı bulanık bir kümedir. Bu bulanık küme, birleşim için maksimum (OR) işlemi veya kesişim işlemleri için minimum (AND) işlemiyle elde edilir. Mamdani FIS çıktısının, ağırlık merkezi yöntemi gibi yöntemler kullanılarak net bir değer elde etmek için durulaştırılması gerekir. Öte yandan Takagi-Sugeno FIS, kuralları temsil etmek için matematiksel veya fonksiyonel bir form kullanır. Kurallar doğrusal veya doğrusal olmayan fonksiyonlar kullanılarak ayarlanır. TSK FIS'in çıktısı net bir sayısal çıktıdır.

Bir görüntünün histogramının yeniden dağıtımı, bu bulanık çıkarım sistemlerinden biri kullanılarak yapılabilir. Bu şekilde görüntünün kontrastı artırılabilir. (Dutta vd., 2021) tam da bunu yapan bir teknik önermiştir. Bu çalışmada Mamdani yaklaşımı kullanılmıştır. Giriş üyelik fonksiyonları üçgen ve Gauss üyelik fonksiyonlarından oluşur ve çıkış üyelik fonksiyonlarının tümü Gauss fonksiyonlarıdır. Bu çalışmada aynı zamanda CLAHE, BBHE ve dualistik alt görüntü histogram eşitleme (DSIHE) gibi geleneksel yöntemlerle karşılaştırıldığında yönteminin etkinliğini kontrol etmek için AMBE, PSNR ve normalleştirilmiş ortalama kare hata (NMSE) ölçümleri kullanılmıştır. Önerilen yöntemin bu metrikler açısından geleneksel yöntemlere göre daha iyi olduğu gösterilmiştir.

2.8. Optimizasyon Algoritmaları

2.8.1. Giriş

Optimizasyon algoritmaları, bir fonksiyonun en iyi değerini bulmak için kullanılan hesaplamalı yöntemlerdir. En iyi değer eldeki fonksiyona bağlıdır: Eğer amaç üretim maliyetini düşürmekse, optimizasyon algoritması fonksiyonun değerini en aza indirmeye çalışmalıdır. Amaç kârı artırmaksa algoritma fonksiyonun değerini maksimuma çıkarmaya çalışmalıdır.

Optimizasyon algoritmaları çoğu mühendislik probleminin temelini oluşturur. Örneğin; bir tasarımda kullanılan malzemeyi en aza indirmede, bir sistemin güç tüketimini en aza indirmede veya bir yapay zeka modelinin doğruluğunu en üst düzeye çıkarmada kullanılır. Sistemin matematiksel modeli oluşturulursa bu problemler optimizasyon algoritmaları kullanılarak çözülebilir. Bir sistemin giriş parametreleri değiştirilerek o sistem için yeni çözümler yaratılır. Oluşturulan yeni çözümleri değerlendirmek için bir metriğe ihtiyaç vardır. Bu değerlendirme uygunluk fonksiyonları kullanılarak yapılır.

Optimizasyon algoritmaları optimum değeri yinelemeli olarak arar. Ancak bir sonraki iterasyonda değerlerin kalitesi aynı kalabilir veya düşebilir. Ancak genel olarak değerlerin kalitesi artar. Optimizasyon algoritmaları optimum değere yakınsama mekaniğine sahiptir. Bu algoritmaların 2 bölümü vardır: Keşfet ve yararlan. Keşfetme aşamasında algoritmalar henüz araştırılmamış yeni çözüm uzaylarını kapsamaya çalışır. Ancak yararlanma aşamasında, algoritmalar mevcut yinelemedeki en iyi değerden yararlanmaya çalışır. Bu sayede arama uzayında gelecek vaat eden bölgeler ayrıntılı bir şekilde aranabilir. Optimizasyon algoritmaları; Sürekli Optimizasyon Algoritmaları, Ayrık Optimizasyon Algoritmaları, Sezgisel ve Metasezgisel Optimizasyon Algoritmaları, Stokastik Optimizasyon Algoritmaları, Deterministik Optimizasyon Algoritmaları, Global Optimizasyon Algoritmaları ve Sıralı Model Tabanlı Optimizasyon Algoritmaları olarak kategorize edilebilir. Bu sınıflandırma çeşitli kriterlere göre yapılmaktadır. Bu nedenle optimizasyon yöntemlerinden bazıları birden fazla kategoriye girmektedir. Örneğin, gradyan iniş yöntemi hem sürekli hem de deterministik optimizasyon algoritmalarının kapsamına girer.

2.8.2. Optimizasyon algoritmalarına tarihsel bakış

Optimizasyon algoritmalarının tarihi Antik Yunan, Antik Mısır ve Babil gibi eski çağlarda başlar. Yunan filozof Pisagor, gerçek dünyadaki her sistemin matematiksel modelinin oluşturulabileceğini savunmuştur. İskenderiyeli Öklid, bir

nokta ile bir dairenin çevresi arasındaki en kısa ve en uzun çizgiyi bulmanın çözümünü bulmuştur. Zenodorus, Dido'nun problemini incelemiştir (Kochenderfer & Wheeler, 2019). Kraliçe Dido, M.Ö. 9. yüzyılda Fenike kralının kızıydı. Kocasının, erkek kardeşi tarafından öldürülmesinin ardından Tunus'a kaçmıştır. Tek bir boğanın derisinin çevreleyebileceği kadar arazi isteyerek yerel lider Yarb'ı kandırmıştır. Yerel lider bunu mütevazı bir istek olarak görmüştür ve onun dileğini yerine getirmiştir. Dido, bir boğanın derisini ince şeritler halinde keserek geniş bir alanı çevrelemiştir. Bu bölge daha sonra Kartaca şehri olarak anılacaktır. Dido, sabit uzunlukta bir eğri tarafından çevrelenebilecek maksimum alanı bulma optimizasyon problemini sezgisel olarak çözmüştür (Bandle, 2017).

17. yüzyılda matematik (Calculus) devrimi optimizasyon konusunu değiştirmiştir. Isaac Newton ve Gottfried Wilhelm Leibniz kalkülüsü geliştirdiler ve kalkülüs fonksiyonların minimum ve maksimumlarını bulmak için kullanılmıştır. Bu şekilde sürekli optimizasyonun temeli oluşturulur.

20. yüzyılda doğrusal programlama, Sovyet matematikçi Leonid Kantorovich ve Amerikalı ekonomist Wassily Leontief tarafından optimizasyonu bir sonraki seviyeye taşımıştır. Doğrusal programlama, yalnızca eşitsizlikler açısından bazı kısıtlamaları olan bir doğrusal ifadenin optimum değerini bulan bir algoritmadır. Doğrusal programlama, işçi sayısı ve üretim için harcanan zamanın doğrusal olması gibi temel varsayımları yapar. Teorik olarak doğrusal programlama, doğrusal olmayan ifadeleri de çözebilir ancak çok fazla hesaplama gücü gerektirir ve bu da onu, doğrusal olmayan problemlerin çözümünde verimsiz kılar (*Linear Programming | Definition & Facts | Britannica*, 2023). Daha sonraki çalışmalarda doğrusal programlamanın çözemediği sorunları çözebilecek doğrusal olmayan programlama geliştirilmiştir. Bu şekilde optimizasyon kullanılarak daha karmaşık problemler çözülmüştür.

Küresel optimizasyon, yerel minimum veya maksimumları bulmak yerine, bir problemin küresel maksimum veya minimumunu bulmak için çalışılır. Tavlama benzetimi ve genetik algoritmalar bu kategorideki örneklerden bazılarıdır.

Metasezgisel algoritmalar, kesin çözümü bulmak için değil, sistem için yeterince iyi olan yaklaşık çözümü bulmak amacıyla geliştirilen algoritmalarlardır. Bu algoritmalar doğadan, insan davranışından veya fizikten ilham alınarak oluşturulmuştur. Evrimden ilham alan genetik algoritma, bal arısı kolonilerinden ilham alan yapay arı kolonisi, Newton'un yerçekimi ve hareket kanunundan ilham alan yerçekimi arama algoritması, büyük patlama ve büyük çöküşten ilham alan büyük patlama-büyük çatlama algoritması

bunlara örnek olarak gösterilebilir (Ahmadi, 2017; Erol & Eksin, 2006; Holland, 1992; Karaboga & Basturk, 2007; Rashedi vd., 2009). Bu algoritmalar çözümlerini yinelemeli olarak bulurlar ve eldeki problemin çok karmaşık olduğu durumlarda ona kesin çözüm bulmak için kullanılırlar. Bu problemlere NP-zor problemler denir. NP-zor, hesaplama karmaşıklığı teorisindeki bir problem kategorisi olan deterministik olmayan polinom-zaman zoru anlamına gelir. Bir problemin NP-zor problem olarak kabul edilebilmesi için en az NP'deki en zor problemler kadar zor olması gerekir. NP problemi, deterministik olmayan bir Turing makinesinin bunu polinom sürede çözebileceği anlamına gelir. Polinom süresi, bir makinenin bir sorunu çözmek için bir dizi işlemi yürütmesi için geçen süre anlamına gelir. Bir algoritmanın NP-zor problemi çözmesi için gereken adım sayısı, N'nin polinom fonksiyonu olmalıdır. N, problemin girdilerinin sayısıdır. Üretilen çözüm, eğer problem NP ise kolaylıkla doğrulanabilir. Örnek olarak modern satranç turnuvaları verilebilir. Oyuncular resmi bir oyundayken ve maç canlı olarak yayımlandığında, izleyen kişiler oyunu güçlü satranç botları üzerinde çalıştırabilir. Bu nedenle insanlar oyunun herhangi bir noktasında bitirme hamlesi olup olmadığını görebilirler. Ancak güçlü satranç robotları olmasaydı hareketi göremezlerdi. Bir çözüm sağlandıktan sonra o çözüm kolayca doğrulanabilir.

Optimizasyon algoritmaları günümüzde yoğun olarak çalışılmaktadır. Bunun nedeni makine öğrenimi, yapay zeka ve veri biliminin sorunlarını çözmek için optimizasyon algoritmalarına ihtiyaç duymasıdır. Optimizasyon algoritmaları özellikle sinir ağlarının eğitiminde kullanılmaktadır. Optimizasyon algoritmalarını yalnızca bu sektörler değil aynı zamanda sağlık, finans, ulaşım ve telekomünikasyon sektörleri de kullanıyor.

2.8.3. Optimizasyon sorunlarının türleri

Dışbükey optimizasyon problemleri, yerel ve küresel çözümler açısından dışbükey olmayan problemlerden farklıdır. Örneğin; problem f' nin verildiğini ve $\Delta f'$ nin de f' nin en dik iniş yönünü gösterdiğini varsayalım. Bu yöne doğru gidildiği takdirde fonksiyonun azaldığı görülecektir. Dışbükey fonksiyonlarda yerel çözümler global çözümlerdir. Ancak dışbükey olmayan problemlerde yerel çözümler küresel çözüm olmayabilir. Bu nedenle bu sorunlara farklı şekilde yaklaşılması gerekir.

Sürekli optimizasyon algoritmaları, değişkenlerin belirli bir aralıkta herhangi bir sayı alabildiği problemleri çözer. Bu değerler genellikle gerçek sayılardır. Öte yandan, ayrık optimizasyon algoritmaları ayrık değerlere sahip problemlerle ilgilenir. Bu ikili sayılar veya tam sayılar olabilir. Sürekli optimizasyon problemlerinin çözümü, ayrık

optimizasyon problemlerine kıyasla nispeten daha kolaydır. Çünkü ayrık problemlerin yalnızca ayrık değerler alması gerekir. Bu da başka bir kısıt olarak düşünülebilir. Bazı ayrık optimizasyon problemlerinin sonlu çözümleri vardır, bazılarının ise sonsuz çözümleri vardır. Sonlu sayıda çözümü olan problemlerde bile her seçeneği deneyip içlerinden en iyisini seçmek mümkün olmayabilir. Çünkü test edilecek çok fazla seçenek olabilir (Kochenderfer & Wheeler, 2019).

Ayrık optimizasyon, sistemdeki değişkenlerin alabileceği değer türleri anlamında sürekli optimizasyon algoritmalarının tersidir. Sürekli optimizasyonda bir değişkenin alabileceği değer belirli bir aralıkta sonsuzdur. Ancak ayrık optimizasyon; yalnızca tamsayıları, nesne kümelerini, kombinasyonları veya rotaları alır. İyi bilinen gezici satıcı problemi ayrık bir optimizasyon problemidir. Sürekli optimizasyon problemlerinin aksine bu problemlerin çözümleri sonludur.

Kısıtlı optimizasyon, eldeki problemin değişkenler üzerinde bazı kısıtlamalara sahip olduğu anlamına gelir. Basit bir örnekle açıklanabilir:

$$\begin{aligned} & \underset{x}{\text{maximize}} (x_1 + x_2) \\ & \text{subject to } |x| \leq 3 \end{aligned} \quad (5)$$

Problem 5'te, herhangi bir değer olan x , $-\sqrt{3}$ ile $\sqrt{3}$ arasında olmak zorundadır. Bu problemin bu kısıtlama dikkate alınarak çözülmesi gerekmektedir. Çünkü çözümün bu kısıtla mümkün olduğu belirtiliyor. Kısıtlı optimizasyon problemlerini çözmek için Lagrange Çarpanları veya KKT yöntemleri kullanılabilir. Ancak kısıtlı optimizasyon problemleri Ceza ve Bariyer yöntemleri kullanılarak kısıtsız optimizasyon problemlerine dönüştürülebilir. Bu yöntemler amaç fonksiyonuna ceza veya engel terimleri ekler. Bu terimler, çözümün kısıtlamaları ihlal etmesi durumunda çözümün uygunluk değerini azaltır. Bu terimlerin cezası, çözümün orijinal problem için uygun ve optimal bir çözüme yaklaşması için artırılabilir.

Ancak kısıtsız optimizasyon algoritmalarında bu gibi kısıtlamalar yoktur. Kısıtlanmamış bir optimizasyon probleminin değişkenleri, sınırsız herhangi bir gerçek değere sahip olabilir. Bu, kısıtlanmamış optimizasyon algoritmalarıyla çalışmayı kolaylaştırır. Çünkü algoritmaların arama alanını kontrol etmek zordur. Kısıtlı optimizasyon problemlerinin çözümünde Ceza ve Bariyer yöntemlerinin çok popüler olmasının nedeni budur.

2.8.3.1. Hiç, bir ya da birçok hedef

Optimizasyon problemleri bir ağacın birden fazla dalı gibi kategorize edilir. Bunlar aynı zamanda amaç fonksiyonlarının sayısına göre de sınıflandırılabilir. Şu ana

kadar tek bir problemin tek bir amaç fonksiyonuna sahip olduğu varsayılmıştır. Ancak bu üç vakadan sadece biridir. Optimizasyon problemlerinin hiç, bir veya birçok amaç fonksiyonu olabilir.

Bir optimizasyon probleminin amaç fonksiyonu yoksa veya sıfır amaç fonksiyonu varsa; bu, değişkenlerin hiçbirinin optimizasyona ihtiyacı olmadığı anlamına gelir. Çözümün kısıtlama kurallarına uyması için problemin çözülmesi gerekir. Bu kısıtlamalar eşitlik veya eşitsizlik kısıtlamaları olabilir. Sıfır amaç fonksiyonlu optimizasyon problemleri çoğunlukla karar verme ve kaynak tahsisi problemlerinde kullanılmaktadır.

Bir veya tek amaç fonksiyonlu problemlerin, optimize edilmesi gereken tek bir amaç fonksiyonu vardır. Bunlar tek boyutlu problemlerdir. Çözüm kötü bir işlev üretiyorsa, arama yönünü veya arama alanını değiştirmek doğru yönde atılmış bir adım olabilir. Ancak çok amaçlı fonksiyonların boyutsal sorunları vardır, bu da sistemin amaç fonksiyonlarının sayısıdır. Çünkü aynı anda optimizasyona ihtiyaç duyan birden fazla amaç fonksiyonu vardır. Bu işlevler için en iyi çözüm yoktur. Çünkü farklı amaç fonksiyonları genellikle birbirleriyle çatışır. Bu, hedefler arasındaki dengeler hakkında ayrıntılı bilgi verilerek belirsizlik çözülmediği takdirde sorunun birden fazla yanıtının olacağı anlamına gelir. Çözümler arasında seçim yapılırken Pareto Optimal çözümünün uygulanması gerekmektedir. Bu çözümlerde, eğer amaç fonksiyonlarından hiçbiri o çözüme hakim değilse, çözümün Pareto optimal olduğu söylenebilir. Bu çözümler Pareto kümesini oluşturur. Bu Pareto seti, sistem hakkında pek çok şeyi açıklayan amaç fonksiyonlarının değiş tokuşlarını (trade-off) gösterir (Kochenderfer & Wheeler, 2019).

2.8.3.2. Belirsizlik altında optimizasyon

Şu ana kadar analiz edilen problem türleri deterministik optimizasyon problemleriydi. Bu, bir modelin tüm varsayımlarının doğru olduğu ve sistemin herhangi bir çıktısının doğru olarak bilinebileceği anlamına gelir. Ancak gerçek dünyadaki birçok uygulamalı mühendislik problemi için durum böyle değildir. Sayısal hesaplamalarda ölçüm hatalarından, gürültüden veya yaklaşımlardan kaynaklanabilecek belirsizlikler her zaman vardır. Bir sistemi optimize ederken bu belirsizlikler dikkate alınmalıdır. Bu belirsizlikler gerçek sistem hakkındaki bilgi eksikliğinin sonucudur. Bunlara stokastik optimizasyon problemleri denir (Kochenderfer & Wheeler, 2019).

$$\begin{aligned}
& \underset{x}{\text{minimize}} f(x, z) \\
& g_i(x) \leq 0, i=1, \dots, m, \\
& h_j(x) \leq 0, j=1, \dots, n, \\
& z \in Z
\end{aligned} \tag{6}$$

6' da g ve h sistemin kısıtlarını, z ise sistemi etkileyen belirsizliklerden kaynaklanan rastgele değişkenleri temsil etmektedir. Bu sistemde yalnızca x değişkeni üzerinde kontrol mevcuttur.

2.9. Gürültü Azaltma

Gürültü, yapay nesnelerin oluşmasına veya bilgi kaybına yol açan veri deformasyonudur. Görüntünün alınması veya aktarımı sırasında gürültü oluşabilir. Görüntülerde farklı nedenlerden kaynaklanan farklı türde gürültüler vardır. Görüntü gürültüsü iki kategoriye ayrılabilir: Fiziksel gürültü ve donanım gürültüsü. Fiziksel gürültünün kökeni, ışığın küçük ayrı parçacıklardan oluşmasından kaynaklıdır. Donanım kaynaklı gürültü ise görüntüyü oluşturan ve/veya aktaran cihazın elektromekanik parçalarındaki sorunlardan kaynaklıdır (Jullian vd., 2016). Karanlık atış ve foton atış gürültüsü fiziksel gürültüye aittir.

Görüntülerdeki gürültü, ortamın ve nesnenin kötü şekilde aydınlatılmasından, görüntünün kamera sensöründen dijital depolama cihazına aktarımından ve analogdan dijital dönüşürücüden (ADC) kaynaklanabilir.

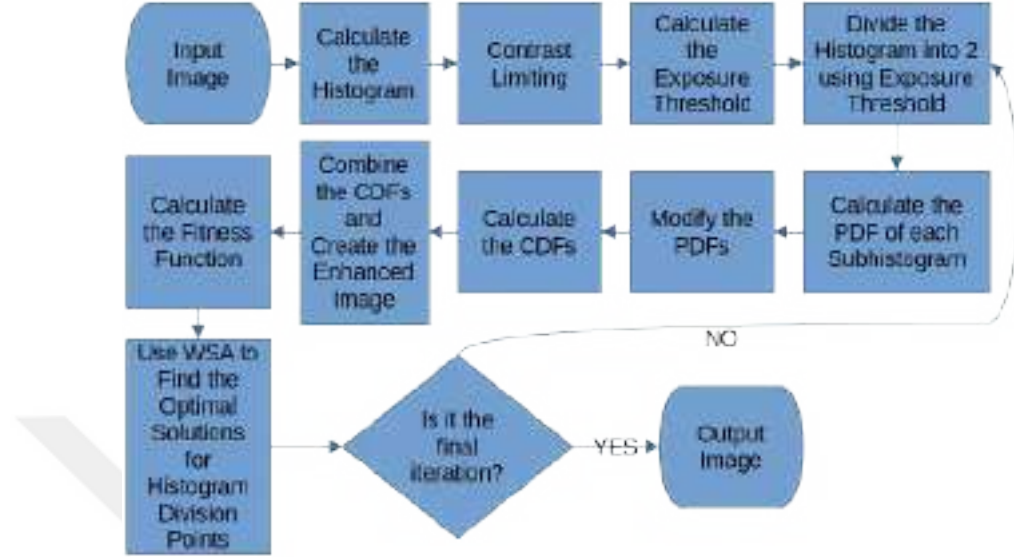
Bu çalışmada kullanılan yöntem kontrast ayarı için histogram eşitlemedir.

3. MATERYAL VE METODOLOJİ

Bu projenin gerçekleştirilebilmesi için Python programlama diline ve röntgen gibi tıbbi görüntülere ihtiyaç duyuldu. OpenCV kütüphanesi görüntüleri okumak ve değiştirmek için kullanıldı. Bu çalışmada Numpy, numba ve matplotlib kütüphanelerinden yararlanılmıştır. Ayrıca kodlarımızı çalıştıran sistem, Linux dağıtımlarından biri olan Fedora 38 işletim sistemine sahip, 12. Nesil Intel® Core™ i7-12700H × 20 işlemcili ve NVIDIA GeForce RTX™ 3060 Laptop GPU kartına sahip bir makinedir. Kodun daha hızlı çalışmasını sağlamak için NVIDIA hesaplama birleşik cihaz mimarisi (CUDA) yazılım katmanı kullanılarak çalıştırıldı. CUDA, GPU'nun komut setine ve paralel hesaplama öğelerine erişime izin verir. CUDA piyasaya sürülmeden önce programcıların grafiksel programlama bilgisine sahip olması gerekiyordu. Ancak CUDA, GPU'nun Python, Fortran, C ve C++ gibi programlama dilleriyle kullanılmasına izin verir. Numba kütüphanesi Python'da CUDA-etkin paralel yürütmenin kullanılmasına izin verir. Ayrıca, kodun bazı bölümleri, tam zamanında (JIT) derleyici kullanılarak daha hızlı çalışmasını sağlamak için yerel kodda derlendi. Normalde, JIT derleyicisi çalışma zamanında kodun "sıcak" kısımlarını seçer ve bunları yerel kodda derler. Süreci hızlandırmak için derlenecek "sıcak" parçalar önceden belirlendi. Bahsedilen bilgisayarda ilk denemeler gerçekleştirildi. Asıl karşılaştırma sonuçları 53 bağımsız bilgisayarda yapıp gerçekleştirilmiştir. Kodlar çalıştırılabilir hale getirilip, Manjaro Linux dağıtımı olan 16 GB RAM'e ve AMD Ryzen 3 işlemcili eşdeğer bilgisayarlarda aynı anda çalıştırılıp sonuçlar csv dosyalarına kaydedilmiştir. Sonrasında csv dosyaları toplanıp grafik halinde bu teze eklenmiştir.

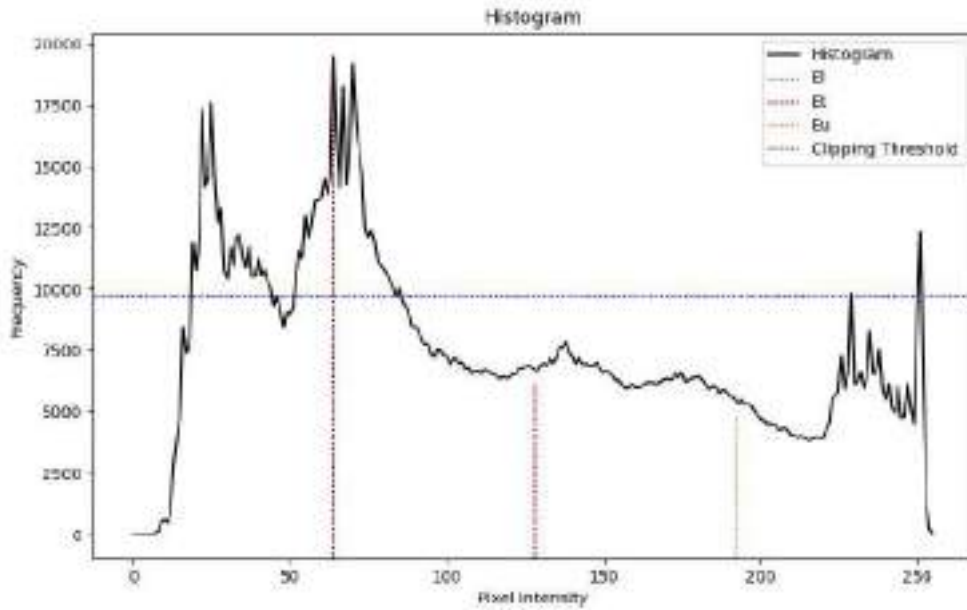
Histogram Eşitleme tekniğini kullanmak için öncelikle kullanılacak görüntülerin histogramları hesaplanmıştır. Histogram, o görüntüde kaç pikselin hangi yoğunlukta olduğunu gösteren bir diyagramdır. Daha sonra histogramın kesme eşiği hesaplandı ve histogram bu kesme eşiğiyle kesildi. Bu kesintinin nedeni gürültüyü azaltmaktır. Maruz kalma eşiği hesaplandı ve bu maruz kalma eşiğiyle histogram iki parçaya bölündü. Arama uzayında optimize edilecek rastgele seçilen değerlere sahip askerler (bireyler) oluşturuldu. Mevcut iki histogram, optimize edilecek değerlerle birlikte tekrar 2 parçaya bölündü. Tüm alt histogramların olasılık yoğunluk fonksiyonları (PDF) hesaplandı. PDF değerleri, toplam PDF değerleri kullanılarak değiştirildi. Kümülatif dağılım fonksiyonu (CDF), değiştirilmiş PDF kullanılarak hesaplandı. Hesaplanan CDF kullanılarak her bir alt histogramın değerleri güncellendi. Geliştirilmiş bir görüntü oluşturmak için 4

histogramın tümü birleştirildi. Bu görüntülerin uygunluk fonksiyonları her bir asker (birey) için hesaplanmıştır. Bunun için çoklu uygunluk fonksiyonu kullanıldı. Bu çalışmada üç uygunluk fonksiyonu algoritmayı eşit derecede etkilemiştir.



Şekil 6: Sürecin akış şeması

Şekil 5' da sürecin akış şeması görünmektedir.



Şekil 7: Histogram bölünmesi

Şekil 7' de histogramın dört alt-histograma nasıl bölüldüğü görünmektedir. Mavi çizgi, histogramın kesileceği kesme eşiğini gösterir.

Bu çalışmada; elitizmsiz genetik algoritma, elitizimli genetik algoritma, elitizimli savaş stratejisi optimizasyon algoritması ve elitizmsiz savaş stratejisi optimizasyon algoritmaları kullanılıp karşılaştırılmıştır. Genetik algoritmalarda turnuva usulü seçim metodu kullanılmıştır.

Toplam 9 farklı tıbbi görüntü kullanılmıştır.

3.1. Histogram Eşitleme Metodu

Kesme eşiği aşağıdaki formüller kullanılarak hesaplanmıştır. Histogramın ortalaması ve medyanı toplanarak ikiye bölünmüştür. Aşağıdaki formüllerde $h(k)$ görüntünün histogramıdır. L histogramda bulunan ve sıfır olmayan maximum gri seviyedir.

$$C_{median} = median(h(k)) \quad (7)$$

$$C_{mean} = mean(h(k)) \quad (8)$$

$$C_{threshold} = \frac{C_{mean} + C_{median}}{2} \quad (9)$$

Histogramı 2 alt histograma bölmek için görüntünün parlaklığı göz önüne alınmıştır. Çünkü görüntüde daha parlak olan alandan daha karanlık olanı ayırt etmek istenmiştir. Bu aşırı iyileştirmeden kaçınmak için yapılmıştır. Pozlama, görüntülerde parlaklıkla bağıntılıdır ve 0 ile 1 arasında bir değere sahip olur. Bir sonraki adımda pozlama eşiği (E_t), formül 10 kullanılarak hesaplanmıştır.

$$Exposure = \frac{1}{L} \frac{\sum_{k=1}^L h(k)k}{\sum_{k=1}^L h(k)} \quad (10)$$

Pozlama, formül 11'deki pozlama eşiğini hesaplamak için kullanılmıştır.

$$E_t = L \times (1 - Exposure) \quad (11)$$

Görüntünün histogramı E_l , E_t ve E_u kullanılarak bölünmüştür. Bu alt histogramlar ayrı ayrı eşitlenmiştir. Yani her alt histogramın PDF' si ve CDF' si de ayrı ayrı hesaplanmıştır. Kullanılan optimizasyon algoritması E_l ve E_u için optimal değerleri bulmaya çalışır çünkü E_t her görüntü için sabittir. Her alt histogram için PDF'leri hesaplamak için 12, 13, 14 ve 15' teki formüller kullanılmıştır.

$$P_{l1} = \frac{h(k)}{N_{l1}} \text{ for } 0 \leq k \leq E_l \quad (12)$$

$$P_{l2} = \frac{h(k)}{N_{l2}} \text{ for } E_l < k \leq E_t \quad (13)$$

$$P_{u1} = \frac{h(k)}{N_{u1}} \text{ for } E_t < k \leq E_u \quad (14)$$

$$P_{u2} = \frac{h(k)}{N_{u2}} \text{ for } E_u < k \leq L \quad (15)$$

Yukarıdaki formüllerde L ana histogramın maksimum gri seviyesidir. N_{l1} , N_{l2} , N_{u1} ve N_{u2} ise her alt histogramdaki piksel sayılarıdır.

(Acharya & Kumar, 2021) alt historamların PDF' lerini değiştirmiştir. Bu değişim, sonuçları iyi yönde etkilediği için bu çalışmada da PDF' ler 16, 17, 18 ve 19' daki formüller kullanılarak değiştirilmiştir.

$$P_{l1m} = \frac{P_{l1}}{\sum_{k=1}^{k=E_l} (P_{l1}(k)) + P_{l1}} \quad (16)$$

$$P_{l2m} = \frac{P_{l2}}{\sum_{k=E_l+1}^{k=E_t} (P_{l2}(k)) + P_{l2}} \quad (17)$$

$$P_{u1m} = \frac{P_{u1}}{\sum_{k=E_t+1}^{k=E_u} (P_{u1}(k)) + P_{u1}} \quad (18)$$

$$P_{u2m} = \frac{P_{u2}}{\sum_{k=E_u+1}^{k=L} (P_{u2}(k)) + P_{u2}} \quad (19)$$

Bir sonraki adımda CDF' ler, 20, 21, 22 ve 23' deki formüller kullanılarak hesaplanmıştır.

$$C_{l1} = \sum_{k=0}^{k=E_l} P_{l1m}(k) \quad (20)$$

$$C_{l2} = \sum_{k=E_l+1}^{k=E_l} P_{l2m}(k) \quad (21)$$

$$C_{u1} = \sum_{k=E_l+1}^{k=E_u} P_{u1m}(k) \quad (22)$$

$$C_{u2} = \sum_{k=E_u+1}^{k=L} P_{u2m}(k) \quad (23)$$

Haritalama fonksiyonu, histogram eşitlemesi işlemi sonrasında gri seviyelerin yeniden dağıtılması için kullanılan fonksiyondur. Tablo 1' de örnek 8-bit bir görüntü için haritalama fonksiyonunun nasıl uygulandığı görülmektedir. Haritalama sütunu, piksel sayısı sütunu ile eşleştirilmiştir. Haritalama sütunu, gri seviyeleri; piksel sayısı sütunu ise piksel sayısını ifade eder. Haritalama sütununda tekrar eden değerler bulunabilir. Örneğin, haritalama sütununda 5 değerine sahip birden fazla satır vardır. Bu, o satırda bulunan piksel sayısı sütunundaki gri seviye değerlerinin hepsinin 5 olması gerektiği anlamına gelir.

Tablo 1: Örnek 8-bit bir görüntünün histogram eşitlemesi

Gri Seviyesi	Piksel Sayısı	PDF	Değiştirilmiş PDF	CDF	Haritalama
0	0	0,000	0,000	0,063	0
1	2	0,118	0,105	0,105	1
2	5	0,294	0,227	0,333	2
3	7	0,412	0,292	0,624	4
4	3	0,176	0,150	0,774	5
5	0	0,000	0,000	0,774	5
6	0	0,000	0,000	0,774	5

Alt histogramlar için haritalama fonksiyonları, 24, 25, 26 ve 27 formülleri kullanılarak hesaplanmıştır:

$$T_{l1} = E_l \times C_{l1} \quad (24)$$

$$T_{l2} = (E_l + 1) + (E_l - (E_l + 1)) \times C_{l2} \quad (25)$$

$$T_{u1} = (E_l + 1) + (E_u - (E_l + 1)) \times C_{u1} \quad (26)$$

$$T_{u2} = (E_u + 1) + (L - (E_u + 1)) \times C_{u2} \quad (27)$$

3.2. Uygunluk Fonksiyonu

Bu çalışmada (Acharya & Kumar, 2021) tarafından önerilen çoklu uygunluk fonksiyonu kullanılmıştır. Başlangıç uygunluk fonksiyonu, her bir askerin (bireysel) geliştirdiği görüntünün kontrastı, entropisi ve enerjisine göre hesaplanır. Kontrast ve enerji formülü, gri seviye eşlik matrisi (GLCM) kullanılarak hesaplanır. GLCM' nin denklemi 28' de gösterilmiştir. Burada bir görüntünün yüksekliği ve genişliği, sırasıyla M ve N olarak verilmiştir.

$$GLCM_{\Delta x, \Delta y}(i, j) = \sum_{x=1}^N \sum_{y=1}^M \begin{cases} 1, & \text{if } I(x, y) = i \text{ and } I(x + \Delta x, y + \Delta y) = j \\ 0, & \text{otherwise} \end{cases} \quad (28)$$

Daha sonra GLCM 29' daki formül kullanılarak normalize edilmiştir.

$$GLCM_{normalized} = \frac{GLCM(i, j)}{\sum_{i=1}^N \sum_{j=1}^M GLCM(i, j)} \quad (29)$$

Bir görüntünün enerjisi 30 kullanılarak hesaplanmıştır.

$$I_{energy} = \sum_{i, j=1}^N GLCM_{normalized}(i, j) \quad (30)$$

Keskinlik aşağıdaki gibi hesaplanmıştır:

$$I_{contrast} = |i - j|^2 \times GLCM_{normalized}(i, j) \quad (31)$$

Shannon Entropisi bir görüntünün bilgi içeriğini gösterir. Entropi değerinin yüksek olması o görüntüde daha fazla bilgi bulunduğu anlamına gelir. Bir görüntünün entropisi 32 kullanılarak hesaplanmıştır.

$$SE = \sum_{k=0}^{L-1} p_k \cdot \log_2(p_k) \quad (32)$$

30, 31 ve 32 kullanılarak uygunluk fonksiyonunun ilk kısmı hesaplanabilir:

$$pf_1 = \log \left(\frac{I_{contrast} \times e^{SE}}{I_{energy}} \right) \quad (33)$$

Uygunluk fonksiyonunun ikinci kısmı iyileştirilmiş görüntünün Sobel görüntüsüne ve Sobel görüntüsünde kaç kenar pikseli bulunduğuyla bağlıdır. Sobel işlemi, iyileştirilmiş görüntünün kenarlarını tespit etmek için kullanılmıştır. Sobel işlemi iki ayrı eksendeki kenarları tespit etmek için iki çekirdek kullanır; biri x-yönü için, diğeri ise y-yönü içindir. Çekirdekler 34 ve 35' te gösterilmiştir.

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (34)$$

$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (35)$$

Sobel resimleri bu çekirdeklerin, iyileştirilmiş resim üzerinde konvolüsyonu sonucu elde edilmiştir.

$$I_{Sobel-x} = (I * G_x)(x) \quad (36)$$

$$I_{Sobel-y} = (I * G_y)(y) \quad (37)$$

36 ve 37' deki formüller tek bir Sobel görüntüsü elde etmek için kullanılmıştır. Bu, (Szeliski, 2022)' nun bahsettiği doğrusal karıştırma işlemi kullanılarak gerçekleştirilmiştir.

x ve y eksenlerinin uygunluk fonksiyonunu eşit şekilde etkilemesi için α , 0,5 olarak seçilmiştir. Bunlar kullanılarak uygunluk fonksiyonunun ikinci kısmı yazılabilir. 38' de n_{edges} , görüntüdeki kenarların sayısını temsil eder. w ve h ise görüntünün enini ve yüksekliğini temsil eder.

$$pf\ 2 = \frac{\log \left(\log \left(\sum_{i,j=0} (0.5 \cdot I_{Sobel-x} + 0.5 \cdot I_{Sobel-y}) \right) \right) \times n_{edges}}{w \times h} \quad (38)$$

Uygunluk fonksiyonunun üçüncü kısmı en yüksek sinyalin gürültüye oranı (PSNR) kullanılarak hesaplanmıştır. PSNR bir görüntünün ne kadar gürültüye sahip olduğuyla ilgili bilgi verir. Bu değer ne kadar düşükse, görüntü o kadar gürültülidir. PSNR, formül 39' da ortalama karekök hatası (MSE) kullanılarak hesaplanmıştır.

$$pf\ 3 = 10 \cdot \log \left(\frac{255^2}{MSE} \right) \quad (39)$$

$$MSE = \frac{1}{w \times h} \sum_{i=1}^w \sum_{j=1}^h (I_o(i,j) - I_e(i,j))^2 \quad (40)$$

Bu üç alt uygunluk fonksiyonu 0,33 ile çarpılarak toplanır ve böylece ana uygunluk fonksiyonu elde edilir (Acharya & Kumar, 2021).

$$pf = 0.33 \times pf\ 1 + 0.33 \times pf\ 2 + 0.33 \times pf\ 3 \quad (41)$$

3.3. Savaş Stratejisi Optimizasyon Algoritması

Savaş stratejisi optimizasyon algoritmasında bir kral, bir komutan ve askerler bulunmaktadır. En yüksek uygunluk değerine sahip olan asker kral, ikinci en iyiye sahip olan ise komutan olur. Optimizasyonun herhangi bir yinelemesinde, tüm askerlerin kral

veya komutan olma olasılığı aynıdır. Askerler kralın ve komutanın konumuna göre hareket ederler. İki taktik vardır: saldırı ve savunma. Saldırı, algoritmada istisarı uygularken, savunma keşif gerçekleştirir. Her yinelemede uygulanacak strateji rastgele seçildi. Her askerin bir rütbesi ve ağırlığı vardır ve bunlar o askerin başarısına göre değişir. Sıralamalar her yinelemede değişir. Ancak ilk yinelemede her askerin rütbesi ve ağırlığı aynıdır. Asker eskisinden daha iyi bir pozisyon bulursa askerin rütbesi yükselecektir.

Zayıf askerlerle başa çıkmak için Savaş Stratejisi optimizasyonunun 2 yöntemi vardır. Bunlar rastgele bir askerle değiştirmek veya askerin yerini değiştirmektir (Ayyarao vd., 2022). Normalde WSO algoritması, yeni konumun öncekinden daha kötü olması durumunda algoritmanın konumunu değiştirmez. Bu çalışmada algoritma bu elitizmle ve bu elitizm olmadan çalıştırılmıştır. Yani WSO algoritması her yinelemede konumlar ne olursa olsun değiştirilerek kullanıldı. Ayrıca bu GA'ya da yapıldı. GA algoritmaları elitizmle ve elitizm olmadan çalıştırıldı. Karşılaştırmalar sonuç bölümünde gösterildi.

Histogram eşitlemesinde optimize edilmesi gereken değerler, savaş stratejisi optimizasyon algoritması ve gerçek kodlu genetik algoritma kullanılarak optimize edilmiştir.

3.3.1. Saldırı Stratejisi

İki strateji var. Saldırı stratejisi, düşmana saldırı başlatmak için kralın ve ordu komutanının konumunu kullanır. Bu, askerlerin yeni konumlarının kral ve komutanın konumu kullanılarak hesaplandığı anlamına geliyor. Her askerin konumu formül 42 kullanılarak güncellenir.

$$X_i(t+1) = X_i(t) + 2 \times \rho \times (C - K) + rand \times W_i \times (K - X_i(t)) \quad (42)$$

Burada; C komutan, K kral, $X_i(t+1)$ güncellenmiş konum, $X_i(t)$ önceki konum ve W_i ağırlıktır. ρ , 0 ile 1 arasında seçilmesi gereken bir hiperparametredir. $rand$, 0 ile 1 arasında seçilen ve algoritmanın keşif ve kullanım aşamasına yardımcı olan rastgele bir sayıdır (Ayyarao vd., 2022).

3.3.2. Rütbe ve Ağırlık Güncelleme

Yeni uygunluk fonksiyonu bir öncekinden daha iyiyse, bir askerin ağırlığı ve rütbesi her yinelemede değişir. Asker mevcut pozisyonundan daha iyi bir pozisyon bulamazsa hareket etmeyecek ve mevcut pozisyonunda kalacaktır. Bu aynı zamanda askerin rütbesinin değişmeyeceği anlamına da geliyor. Asker daha iyi bir yer bulursa

askerin rütbesi 1 artacaktır (Ayyarao vd., 2022). Rütbe değiştiği için askerin ağırlığının da 43 kullanılarak güncellenmesi gerekir.

$$W_i = W_i \times \left(1 - \frac{R_i}{\text{Max_iter}}\right)^\alpha \quad (43)$$

3.3.3. Savunma Stratejisi

Savunma stratejisinde rütbe ve ağırlık güncellemesi aynıdır. Ancak askerlerin konumları, kralın konumu ve rastgele bir askerin konumu kullanılarak güncellenir. Rastgele bir asker kullanıldığından savunma stratejisinde keşif aşaması daha baskındır. Ağırlık parametresinin büyük olması askerlerin daha geniş adımlar attığı anlamına gelir (Ayyarao vd., 2022). Askerlerin yeni pozisyonları formül 44 kullanılarak bulunur. .

$$X_i(t+1) = X_i(t) + 2 \times \rho \times (K - X_{rand}(t)) + rand \times W_i \times (C - X_i(t)) \quad (44)$$

3.3.4. Zayıf Askerlerin Değiştirilmesi ya da Yer Değişimi

Zayıf askerler en kötü uygunluk fonksiyonlarına sahip olanlardır. Bunları değiştirmenin iki yolu vardır. İlki basitçe rastgele bir askerle değiştirmektir:

$$X_w(t+1) = Lb + rand \times (Ub - Lb) \quad (45)$$

45' te, Lb ve Ub sırasıyla alt sınır ve üst sınır anlamına gelir.

İkinci yol ise ordunun medyanını alıp askerleri oraya yerleştirmektir:

$$X_w(t+1) = -(1 - randn) \times (X_w(t) - \text{median}(X)) + K \quad (46)$$

İkinci yol algoritmanın daha hızlı yakınsamasına sebep olur.

3.4. Algoritmanın Sözde Kodu

Görüntüyü oku

Görüntüyü gri seviyeye çevir

Görüntünün boyutlarını al

Görüntünün histogramını hesapla

Histogramın kesme eşiğini hesapla

Histogramı kesme eşiğiyle kırp

Görüntünün pozlama eşiğini hesapla

popSize = 50, iterationNo = 50 ro_r = 0.5 alpha = 0.5, discard_the_weak = (int)

popSize/10

for (int i = 0; i < popSize; i++):

El ve Eu, 0 <= El <= Et ve Et < Eu <= L sınırlarında olacak şekilde rastgele

El ve Eu değerlerine sahip askerler oluştur

```

for (int i = 0; i < iterationNo; i++):
    for (int j = 0; j < popSize; j++):
        Orijinal resmin histogramını El ve Eu asker değerleri ve sabit Et
        kullanılarak 4 alt histograma böl
        Her bir alt histograma HE uygula
        Her bir görüntü için çoklu uygunluk fonksiyonunu hesapla ve askerlere
        ata
        Uygunluk fonksiyonlarına göre Kralı, Komutanı ve en zayıf askerleri seçmek
        için sırala
        Ro'yu 0 ile 1 arasında rastgele bir sayı ile başlat
        Eğer ro < ro_r:
            Saldırı stratejisini uygula
        Değilse:
            Savunma stratejisini uygula
        Askerlerin rütbelerini ve ağırlıklarını güncelle
        Ordunun medyanını hesapla
        for(int j = 0; j < discard_the_weak; j++):
            Ordunun medyanını kullanarak yeni askerler oluştur
            En zayıf askerleri yenileriyle değiştir

```

3.5. Hiperparametreler

Genetik algortmada kullanılan hiperparametreler şunlardır: Popülasyon sayısı 50, iterasyon sayısı 200, çarpazlama oranı 0,8, mutasyon oranı 0,01, beta 0,5.

Savaş stratejisi optimizasyon algoritmasında kullanılan hiperparametreler ise şunlardır: popülasyon sayısı 50, iterasyon sayısı 200, ro 0,5, alfa 0,5, zayıf asker değişim oranı 0,1.

3.6. Savaş Stratejisi Algoritmasında Değiştirilen Yerler

Savaş stratejisi optimizasyon algoritmasında; iterasyon sayısı arttıkça askerler, kral ve komutanın pozisyonuna yaklaşırlar. Bu da iterasyonun başlarında arama uzayının daralmasına ve algoritmanın local optimada takılı kalmasına yol açabiliyor. Bundan dolayı her askerin pozisyonu güncellendiğinde, güncellenen askerin ordudaki diğer askerlerden herhangi birine eşit olup olmadığı kontrol edildi ve eğer eşit olduysa o askerin eski yerini değiştirmemesi sağlandı.

WSO'nun diğer bir özelliği ise bir askerin yeni bulunan konumu eski konumundan kötüyse, yer değiştirmemesidir. Bu çalışmada bir askerin yeni pozisyonu

eski pozisyonundan kötüye yine de oraya hareket etmesi sağlanmıştır ve rütbesi 1 azaltılıp ağırlığı hesaplanmıştır. Geleneksel WSO tekniğiyle kıyaslanmıştır. Uygulanan bu tekniğe bu çalışmada elitizmsiz WSO denmiştir ve geleneksel WSO tekniğine ise bu çalışmada elitizimli WSO şekilde hitap edilmiştir.

3.7. Karmaşıklık Matrisi

Karmaşıklık matrisi algoritmaların hem zamansal hem de uzaysal karmaşıklıklarını ölçmek için kullanılır. Burada en yaygın kullanılan metrik ise Big O notasyonudur. Kullanılan 4 algoritmanın (Elitizimli GA, elitizmsiz GA, elitizimli WSO ve elitizmsiz WSO) zamansal ve uzaysal karmaşıklık matrisinin adımlarının karmaşıklık matrisleri aşağıdaki gibidir.

Tablo 2: Zamansal karmaşıklık matrisi

Adım	Zaman Karmaşıklığı (Elitizimli GA)	Zaman Karmaşıklığı (Elitizmsiz GA)	Zaman Karmaşıklığı (Elitizimli WSO)	Zaman Karmaşıklığı (Elitizmsiz WSO)
Başlangıç	O(1)	O(1)	O(1)	O(1)
Popülasyon oluşturma	O(n ²)	O(n ²)	O(n ²)	O(n ²)
Uygunluk değerlendirme	O(n ²)	O(n ²)	O(n ²)	O(n ²)
Ebeveyn seçimi	O(n)	O(n)	-	-
Strateji seçimi	-	-	O(1)	O(1)
Saldırı/Savunma stratejisi	-	-	O(n)	O(n)
Çaprazlama	O(n)	O(n)	-	-
Mutasyon	O(n)	O(n)	-	-
Zayıfları değiştirme	-	-	O(n ²)	O(n ²)
Yeni popülasyon oluşturma	O(n)	O(n)	O(n)	O(n)
Elitizm	O(n)	-	O(n)	-
Durma koşulu	O(1)	O(1)	O(1)	O(1)

Tablo 2’de algoritmaların zamansal karmaşıklık matrisi görülüyor. Tablo 3’te ise algoritmaların uzaysal karmaşıklık matrisi görülüyor. Bu karmaşıklık matrislerinde Big O notasyonu kullanılmıştır. Bu matrislerde gösterilen bazı adımlar sadece genetik algoritmaya bazı adımlar ise sadece savaş stratejisi optimizasyon algoritmasına mahsustur. Bundan dolayı, eğer bir adım o algoritmaya ait değilse tire (-) konularak ona ait olmadığı belirtilmiştir.

Tablo 3: Uzaysal karmaşıklık matrisi

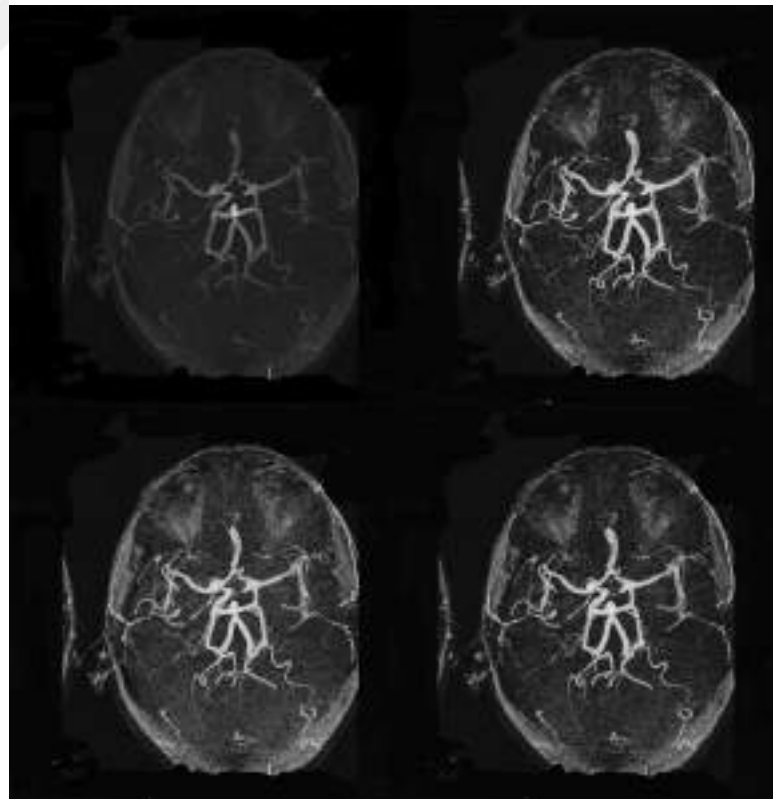
Adım	Uzay Karmaşıklığı (Elitizimli GA)	Uzay Karmaşıklığı (Elitizmsiz GA)	Uzay Karmaşıklığı (Elitizimli WSO)	Uzay Karmaşıklığı (Elitizmsiz WSO)
Başlangıç	$O(n)$	$O(n)$	$O(n)$	$O(n)$
Popülasyon oluşturma	$O(n)$	$O(n)$	$O(n)$	$O(n)$
Uygunluk değerlendirme	$O(1)$	$O(1)$	$O(1)$	$O(1)$
Ebeveyn seçimi	$O(1)$	$O(1)$	-	-
Strateji seçimi	-	-	$O(1)$	$O(1)$
Saldırı/Savunma stratejisi	-	-	$O(n)$	$O(n)$
Çaprazlama	$O(1)$	$O(1)$	-	-
Mutasyon	$O(1)$	$O(1)$	-	-
Zayıfları değiştirme	-	-	$O(1)$	$O(1)$
Yeni popülasyon oluşturma	$O(n)$	$O(n)$	$O(n)$	$O(n)$
Elitizm	$O(1)$	-	$O(1)$	-
Durma koşulu	$O(1)$	$O(1)$	$O(1)$	$O(1)$

4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

Tıbbi görüntülerin tüm vakalarında WSO'nun GA'ya göre daha hızlı sonuç verdiği görüldü. Uygunluk fonksiyonu anlamında aynı sonuçları veya yakın sonuçları verdiler. Ancak hız açısından WSO bu iş için daha iyi bir seçim olduğu sonuçlanmıştır.

Tablo 4: Şekil 8 için sonuçlar

	Elitizm ile Genetik Algoritma	Elitizmsiz Genetik Algoritma	Elitizm ile Savaş Stratejisi Algoritması	Elitizmsiz Savaş Stratejisi Algoritması
Geçen Süre (s):	610,875	601,072	448,367	214,716
İterasyon Başına Geçen Süre (s/it):	3,054	3,005	2,242	1,074
Yakınsama (İterasyon No):	25	200	5	11
En İyi Üyenin El ve Eu' su:	El: 26, Eu: 240	El: 21, Eu: 228	El: 26, Eu: 250	El: 26, Eu: 250
En İyi Uygunluk Fonksiyonu:	19,922	19,924	19,924	19,924

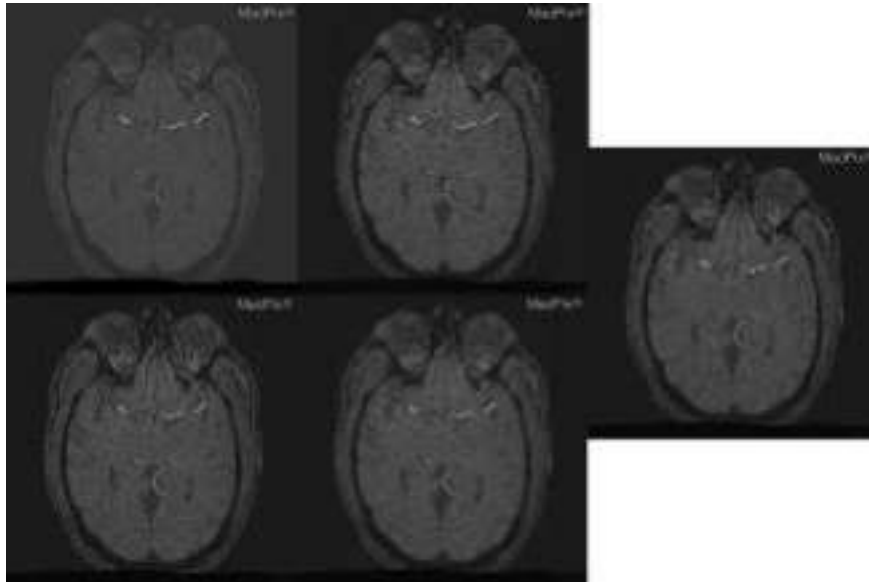


Şekil 8: Orijinal sol üstte, elitizmi GA tarafından geliştirilmiş sağ üstte elitizm, sol altta elitizm olmadan GA tarafından geliştirilmiş, sağ altta elitizmi ve elitizmsiz WSO tarafından geliştirilmiş

Tablo 4’de 200 iterasyon gerçekleştirildi. Bu algoritmaların arasından en hızlısının WSO olduğu görüldü. Ancak WSO ile karşılaştırıldığında ileriki iterasyonlarda elitizmle yakınladı. Tablo 4’de elitizimli WSO’nun en hızlı yakınladığı, elitizmsiz GA’nın ise en yavaş yakınladığı görülmektedir. Bu tablodan elitizmin algoritmaların daha hızlı yakınlamasına yardımcı olduğu sonucu çıkarılabilir. Algoritmaların bulduğu sonuçlar birbirine yakın gibi görünse de WSO, hız açısından GA’dan daha iyi bir seçimdir. Şekil 9’deki iyileştirilmiş görüntüler nispeten birbirine yakın görünmektedir.

Tablo 5: Şekil 9 için sonuçlar

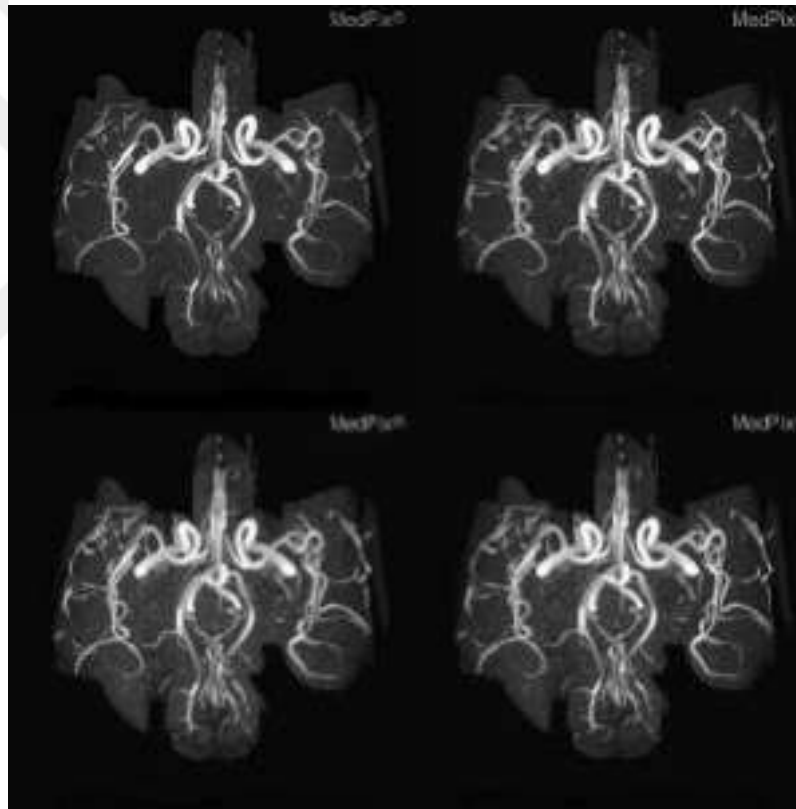
	Elitizm ile Genetik Algoritma	Elitizmsiz Genetik Algoritma	Elitizm ile Savaş Stratejisi Algoritması	Elitizmsiz Savaş Stratejisi Algoritması
Geçen Süre (s):	575,115	567,885	404,569	410,120
İterasyon Başına Geçen Süre (s/it):	2,876	2,839	2,023	2.051
Yakınlama (İterasyon No):	4	200	15	18
En İyi Üyenin El ve Eu’ su:	El: 68, Eu: 144	El: 70, Eu: 141	El: 69, Eu 175	El: 71, Eu 144
En İyi Uygunluk Fonksiyonu:	20,007	20,010	20,018	20,021



Şekil 9: Sol üstte orijinal, üst-ortada elitizm ile GA tarafından geliştirilmiş, sol altta elitizm olmadan GA tarafından geliştirilmiş, alt-ortada elitizm ile WSO tarafından geliştirilmiş ve sağda elitizm olmadan WSO tarafından geliştirilmiş.

Tablo 6: Şekil 10 için sonuçlar

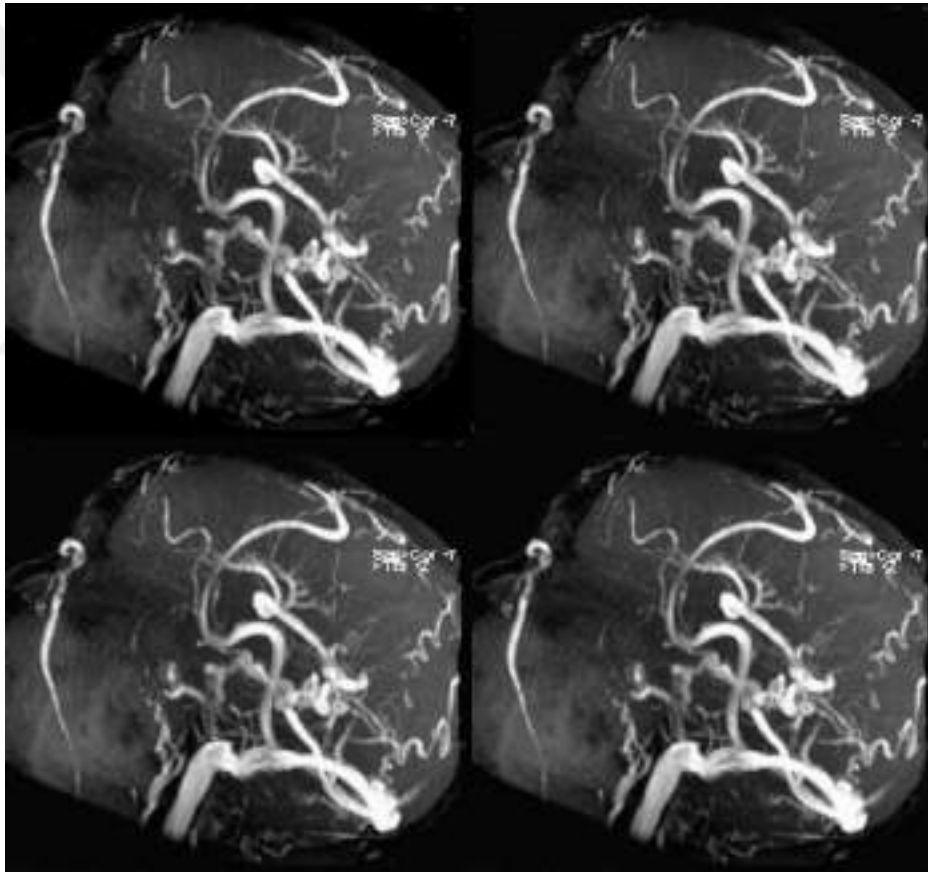
	Elitizm ile Genetik Algoritma	Elitizmsiz Genetik Algoritma	Elitizm ile Savaş Stratejisi Algoritması	Elitizmsiz Savaş Stratejisi Algoritması
Geçen Süre (s):	586,256	578,064	449,203	342,811
İterasyon Başına Geçen Süre (s/it):	2,931	2,890	2,246	1,714
Yakınsama (İterasyon No):	0	47	31	49
En İyi Üyenin El ve Eu' su:	El: 50, Eu: 246	El: 38, Eu: 212	El: 43, Eu 238	El: 43, Eu 238
En İyi Uygunluk Fonksiyonu:	19,591	19,602	19,640	19,640



Şekil 10: Orijinal sol üstte, elitizm ile GA tarafından geliştirilmiş sağ üstte, sol altta elitizm olmadan GA tarafından geliştirilmiş, sağ altta elitizimli ve elitizmsiz WSO tarafından geliştirilmiş görüntüler

Tablo 7: Şekil 11 için sonuçlar

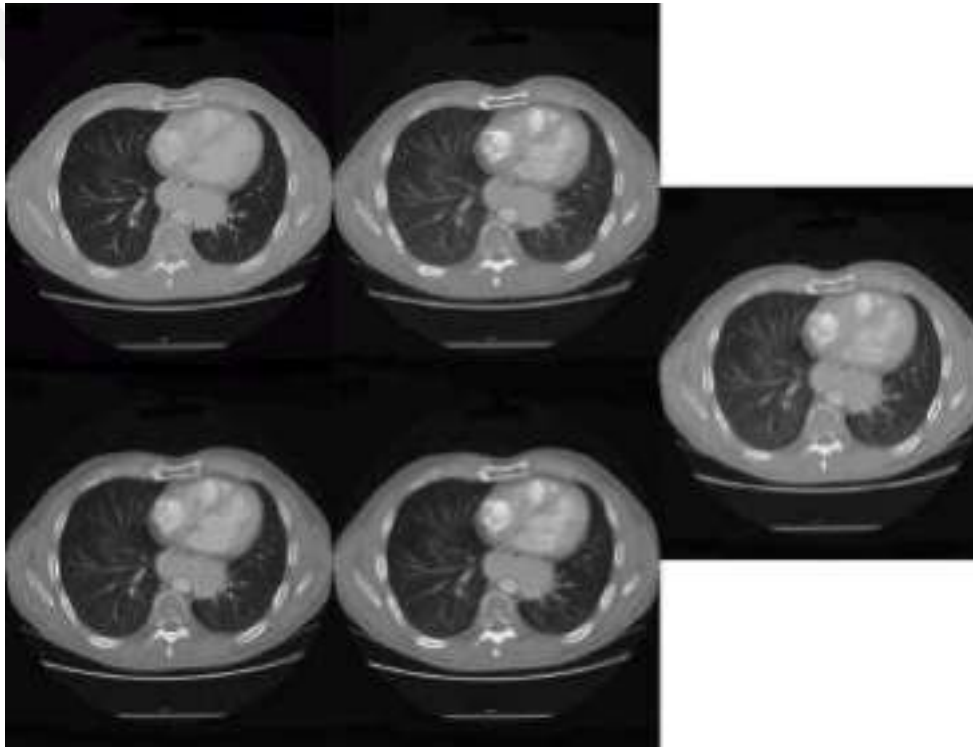
	Elitizm ile Genetik Algoritma	Elitizmsiz Genetik Algoritma	Elitizm ile Savaş Stratejisi Algoritması	Elitizmsiz Savaş Stratejisi Algoritması
Geçen Süre (s):	721,181	701,681	555,230	550,921
İterasyon Başına Geçen Süre (s/it):	3,606	3,508	2,776	2,755
Yakınsama (İterasyon No):	37	200	33	62
En İyi Üyenin El ve Eu' su:	El: 105, Eu: 211	El: 105, Eu: 202	El: 105, Eu: 211	El: 105, Eu: 233
En İyi Uygunluk Fonksiyonu:	19,889	19,848	19,889	19,893



Şekil 11: Sol üstte orijinal, sağ üstte elitizmlı GA ve elitizmlı WSO tarafından geliştirilmiş, sol altta elitizmsiz GA tarafından geliştirilmiş, sağ altta elitizmsiz WSO tarafından geliştirilmiş

Tablo 8: Şekil 12 için sonuçlar

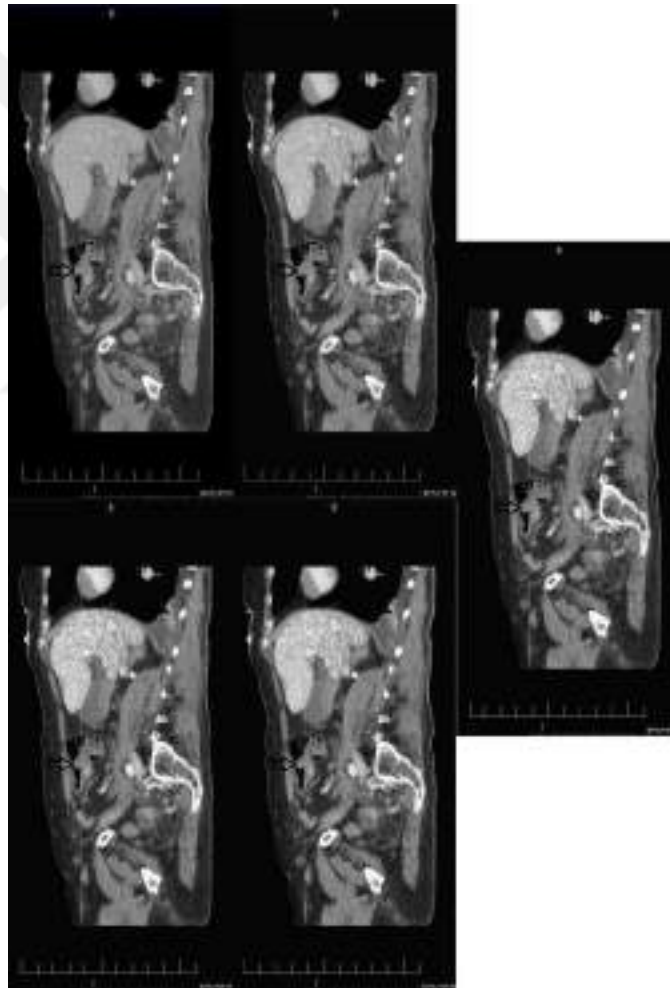
	Elitizm ile Genetik Algoritma	Elitizmsiz Genetik Algoritma	Elitizm ile Savaş Stratejisi Algoritması	Elitizmsiz Savaş Stratejisi Algoritması
Geçen Süre (s):	529,704	526,492	482,372	270,498
İterasyon Başına Geçen Süre (s/it):	2,649	2,632	2,412	1,352
Yakınsama (İterasyon No):	20	200	24	71
En İyi Üyenin El ve Eu' su:	El: 95, Eu: 252	El: 58, Eu: 207	El: 43, Eu: 246	El: 158, Eu: 254
En İyi Uygunluk Fonksiyonu:	19,420	19,406	19,447	19,413



Şekil 12: Sol üstte orijinal, üst-ortada elitizm ile GA tarafından geliştirilmiş, sol altta elitizm olmadan GA tarafından geliştirilmiş, alt-ortada elitizm ile WSO tarafından geliştirilmiş ve sağda elitizm olmadan WSO tarafından geliştirilmiş

Tablo 9: Şekil 10 için sonuçlar

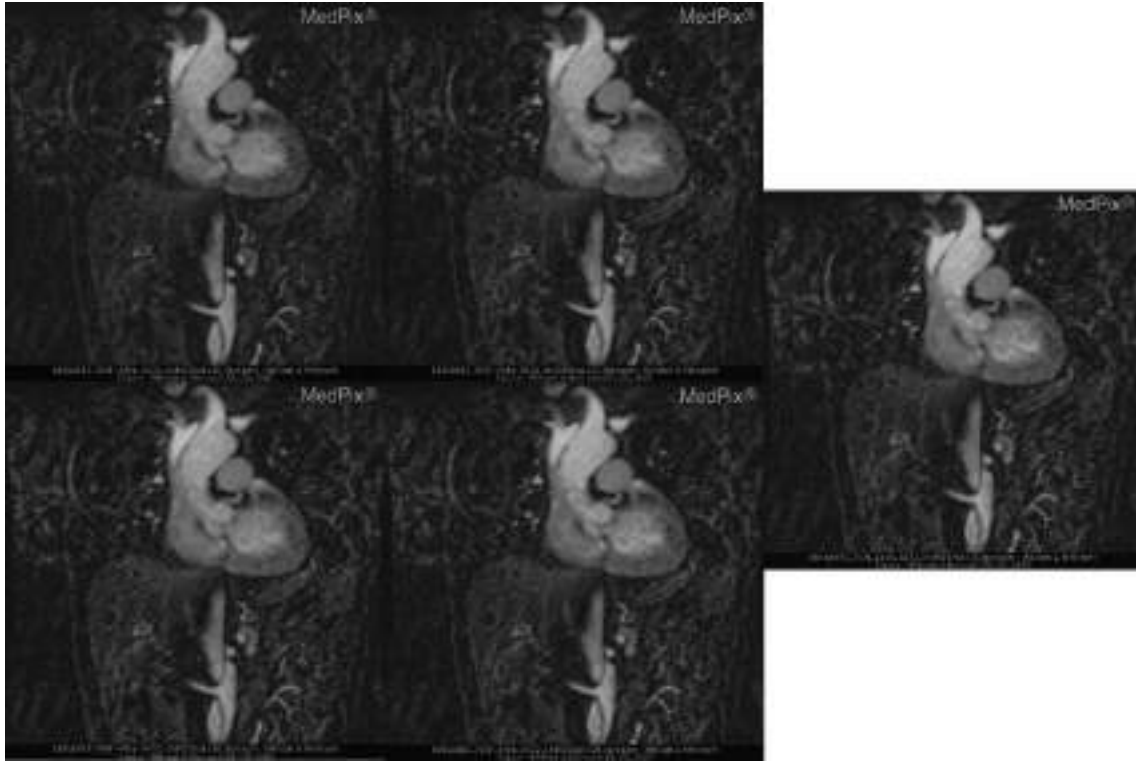
	Elitizm ile Genetik Algoritma	Elitizmsiz Genetik Algoritma	Elitizm ile Savaş Stratejisi Algoritması	Elitizmsiz Savaş Stratejisi Algoritması
Geçen Süre (s):	1290,360	1273,740	1107,400	421,950
İterasyon Başına Geçen Süre (s/it):	6,450	6,370	5,540	2,110
Yakınsama (İterasyon No):	10	200	94	76
En İyi Üyenin El ve Eu' su:	El: 70, Eu: 216	El: 8, Eu: 252	El: 123, Eu: 250	El: 8, Eu: 249
En İyi Uygunluk Fonksiyonu:	20,758	20,749	20,793	20,807



Şekil 13: Sol üstte orijinal, üst-ortada elitizm ile GA tarafından geliştirilmiş, sol altta elitizm olmadan GA tarafından geliştirilmiş, alt-ortada elitizm ile WSO tarafından geliştirilmiş ve sağda elitizm olmadan WSO tarafından geliştirilmiş

Tablo 10: Şekil 12 için sonuçlar

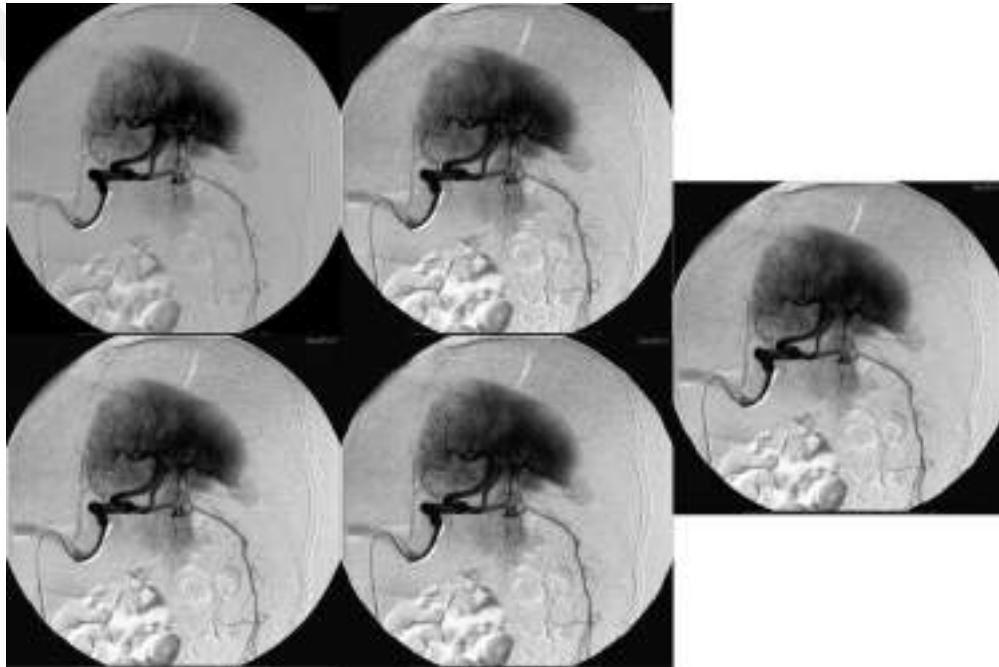
	Elitizm ile Genetik Algoritma	Elitizmsiz Genetik Algoritma	Elitizm ile Savaş Stratejisi Algoritması	Elitizmsiz Savaş Stratejisi Algoritması
Geçen Süre (s):	517,364	505,413	462,644	328,444
İterasyon Başına Geçen Süre (s/it):	2,587	2,527	2,313	1,642
Yakınsama (İterasyon No):	0	200	57	166
En İyi Üyenin El ve Eu' su:	El: 95, Eu: 161	El: 23, Eu: 158	El: 108, Eu: 174	El: 118, Eu: 128
En İyi Uygunluk Fonksiyonu:	19,954	19,975	19,975	19,993



Şekil 14: Sol üstte orijinal, üst-ortada elitizm ile GA tarafından geliştirilmiş, sol altta elitizm olmadan GA tarafından geliştirilmiş, alt-ortada elitizm ile WSO tarafından geliştirilmiş ve sağda elitizm olmadan WSO tarafından geliştirilmiş

Tablo 11: Şekil 15 için sonuçlar

	Elitizm ile Genetik Algoritma	Elitizmsiz Genetik Algoritma	Elitizm ile Savaş Stratejisi Algoritması	Elitizmsiz Savaş Stratejisi Algoritması
Geçen Süre (s):	2123,567	2077,241	1575,963	1551,675
İterasyon Başına Geçen Süre (s/it):	10,618	10,386	7,880	7,758
Yakınsama (İterasyon No):	8	0	60	0
En İyi Üyenin El ve Eu' su:	El: 50, Eu: 155	El: 143, Eu: 68	El: 131, Eu 148	El: 38, Eu 150
En İyi Uygunluk Fonksiyonu:	21,117	21,093	21,128	21,127

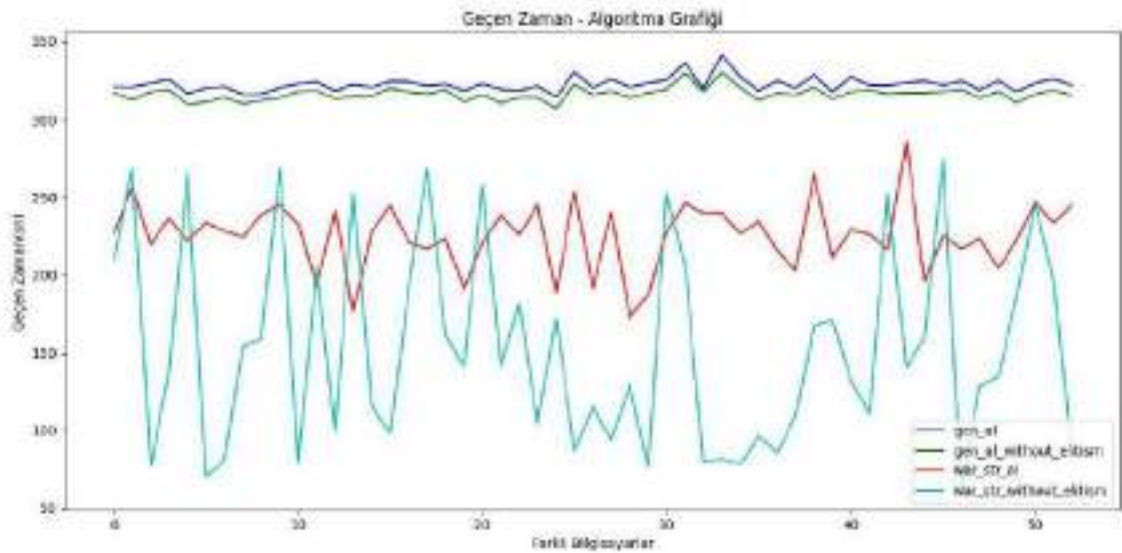


Şekil 15: Sol üstte orijinal, üst-ortada elitizm ile GA tarafından geliştirilmiş, sol altta elitizm olmadan GA tarafından geliştirilmiş, alt-ortada elitizm ile WSO tarafından geliştirilmiş ve sağda elitizm olmadan WSO tarafından geliştirilmiş

Tablo 4' ten Tablo 11' e kadar elitizimli/ elitizmsiz genetik algoritma ile savaş stratejisi optimizasyon algoritmaları arasındaki fark görülebilir. Bu tablolar ve resimler tek bir bilgisayar kullanılarak ortaya çıkarılmıştır. Tüm durumlarda, WSO daha hızlı yakınsadı ve GA' dan daha az zaman harcayarak GA ile aynı veya daha iyi sonucu buldu. Bunun arkasındaki ana nedenlerden biri, WSO'daki üyelerin sıralanmasına gerek

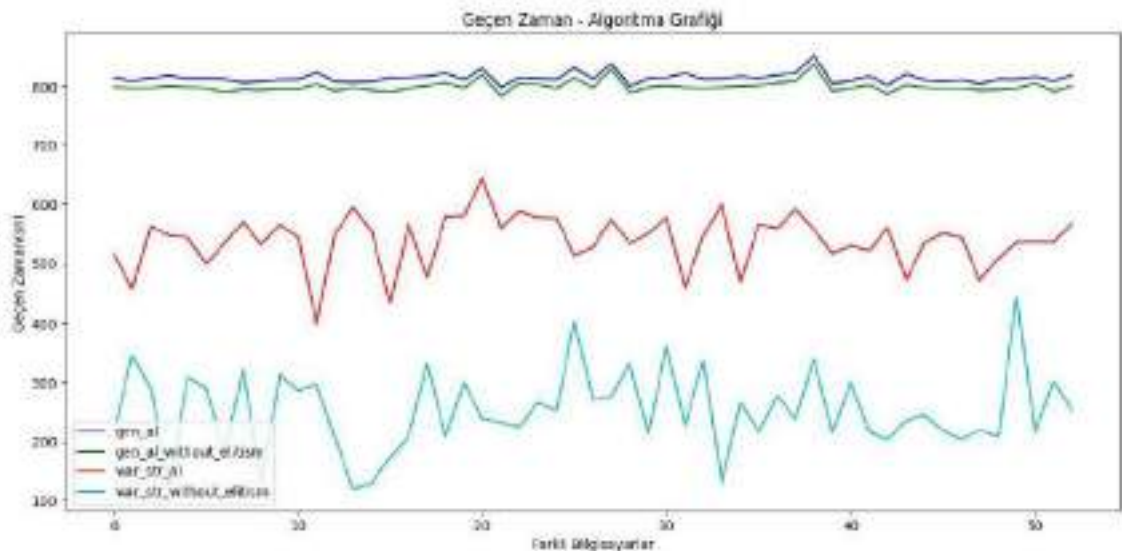
olmamasıdır. En iyinin, ikinci en iyinin ve en kötü üyelerin %10'unun bulunması WSO için yeterlidir. Bu farkın algoritmaların hızı üzerinde büyük etkisi vardır.

Algoritmaların hızları ve her iterasyondaki uygunluk değerleri aşağıdaki grafiklerde görülebilir. Bundan önceki görüntüler ve grafikler sadece tek bir bilgisayardan çıkmış sonuçlardır. Aşağıdaki grafiklerdeki sonuçlar ise 53 bağımsız bilgisayarda aynı anda çalıştırılan algoritmaların sonuçlarıdır.



Şekil 16: Birinci görüntü için algoritma hızları

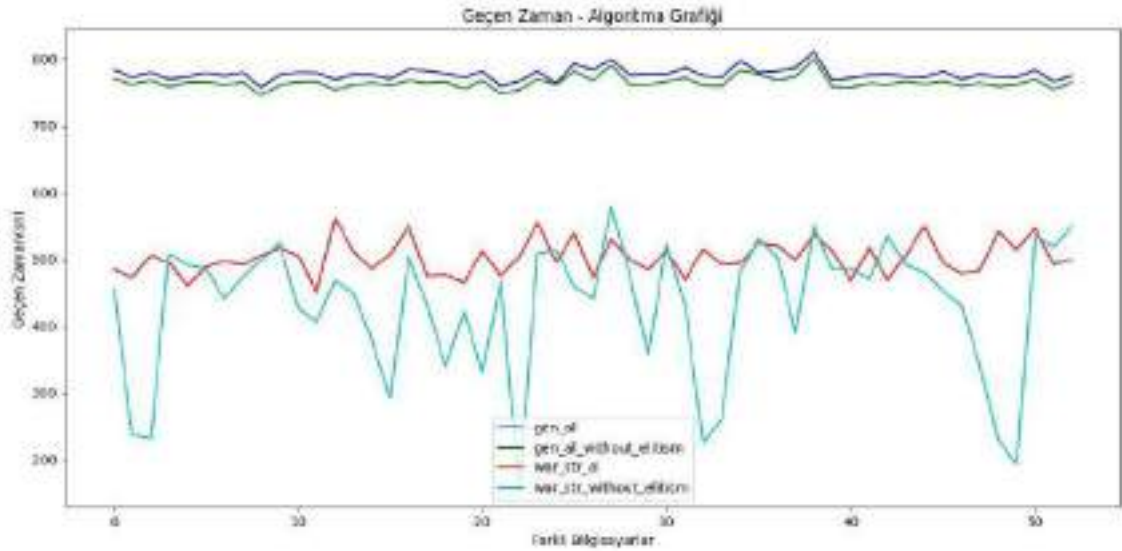
Şekil 16'da y ekseninde geçen zaman saniye olarak verilmiştir ve x ekseninde ise bilgisayar sayıları verilmiştir.



Şekil 17: İkinci görüntü için algoritma hızları

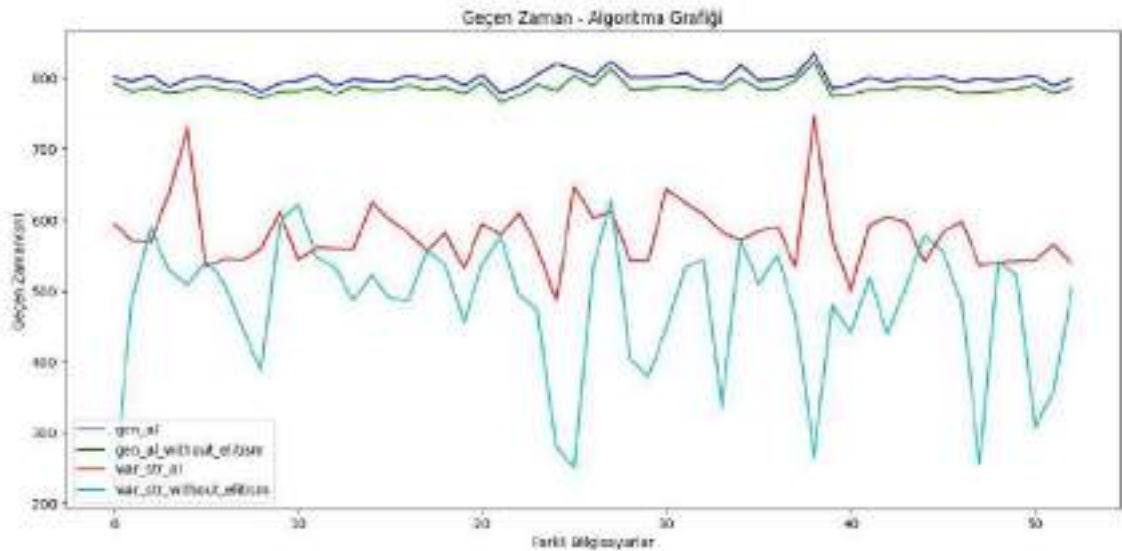
Şekil 17'de ise ikinci görüntünün algoritma sonuçları görünüyor. Elitizmsiz savaş stratejisi algoritmasının çok daha hızlı çalıştığı görülmektedir.

Bu resimlerde mavi çizgi elitizimli genetik algoritma, yeşil çizgi elitizmsiz genetik algoritma, kırmızı çizgi elitizimli savaş stratejisi algoritması ve turkuaz çizgi ise elitizmsiz genetik algoritmanın sonuçlarını vermektedir.



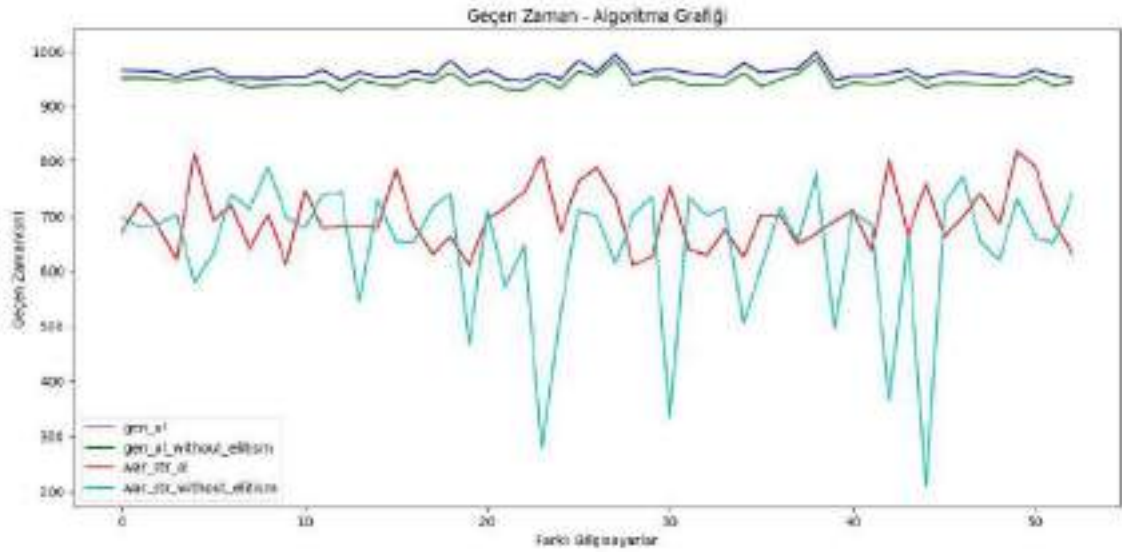
Şekil 18: Üçüncü resim için algoritma hızları

Şekil 18'de üçüncü görüntü için algoritma hız sonuçları görülmüyor. Burada da bu çalışmada önerilen algoritma daha hızlı çalışmıştır.



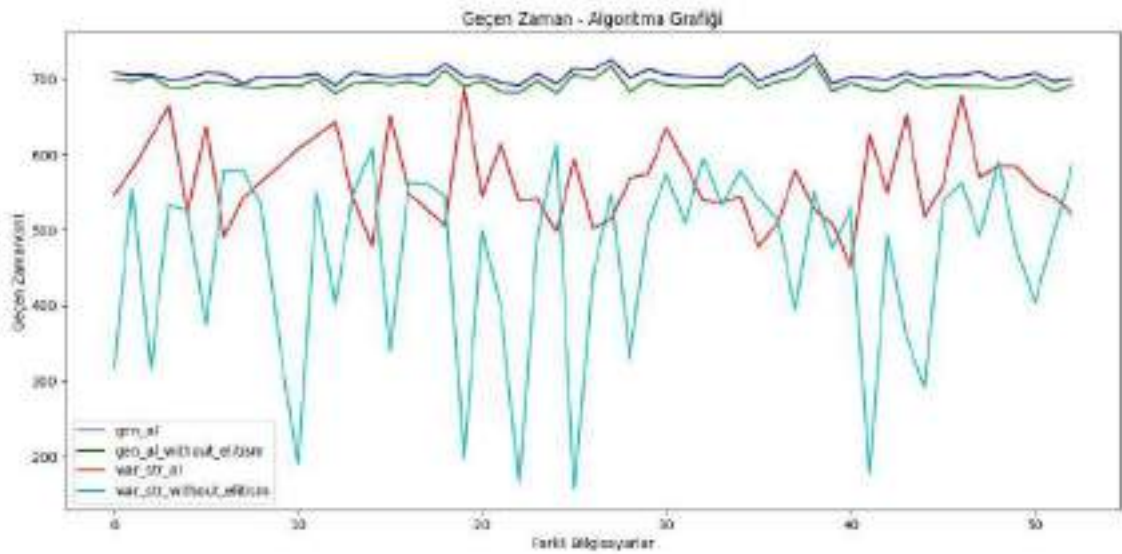
Şekil 19: Dördüncü görüntü için algoritma hızları

Şekil 19'da dördüncü görüntü için algoritma sonuçları görülmektedir. Burada önerilen algoritma bazı bilgisayarlarda dalgalı sonuçlar vermiş olsa da yine de genetik algoritmalarından çok daha hızlı çalıştığı görülmektedir.



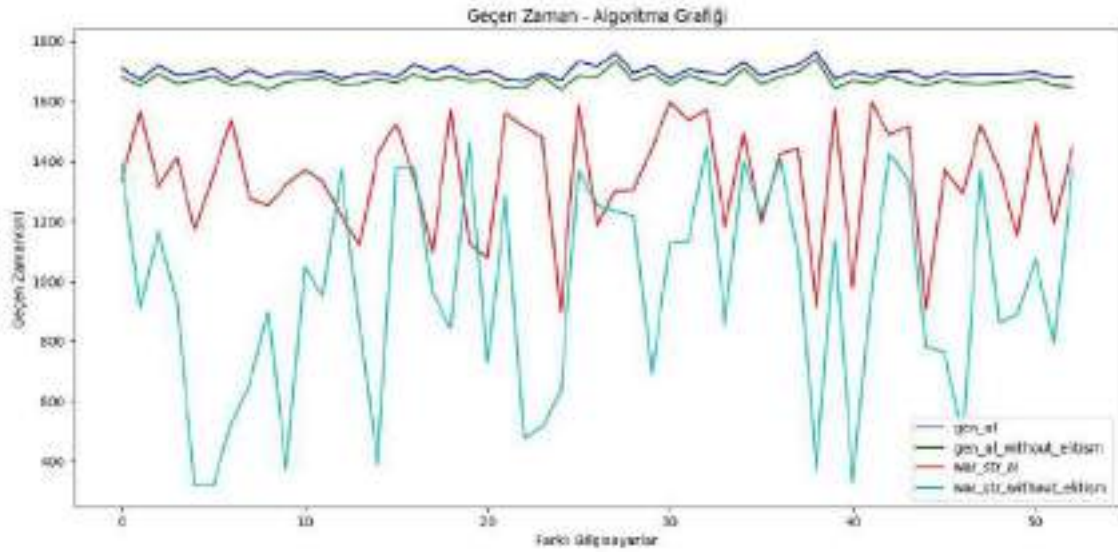
Şekil 20: Beşinci görüntü için algoritma hızları

Şekil 20’de beşinci görüntü için 4 farklı algoritmanın farklı bilgisayarlarda tamamlanma hızları verilmiştir.



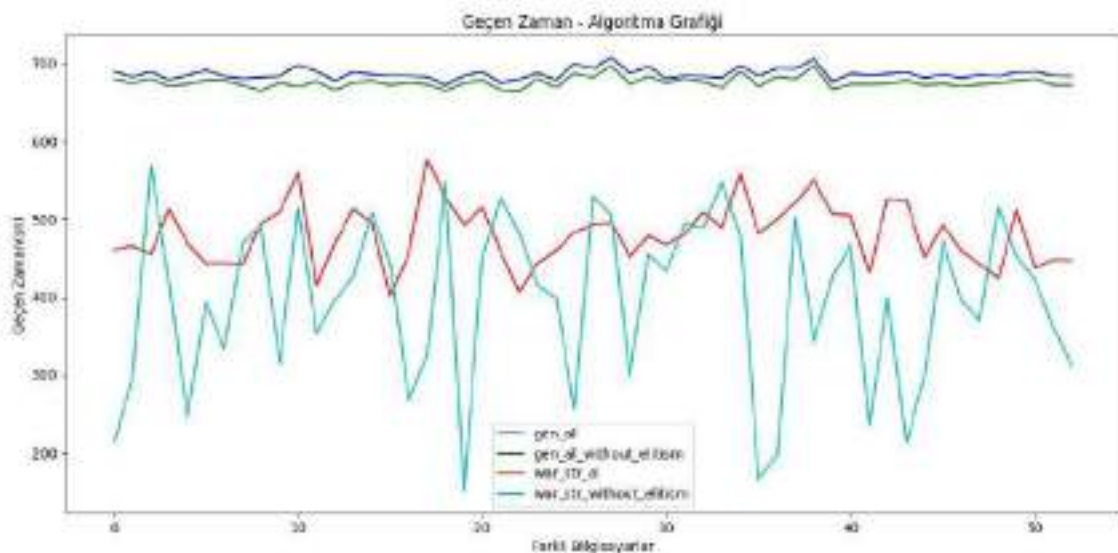
Şekil 21: Altıncı görüntü için algoritma hızları

Şekil 21’de altıncı görüntü için 4 farklı algoritmanın 53 farklı bilgisayarda bitme hızları verilmiştir. Burada önerilen yöntem yine çok dalgalı olmasına rağmen hız açısından çok daha iyi sonuçlar vermiştir. Farklı görüntülerde algoritmaların farklı hızda tamamlanması çok doğaldır. Çünkü bazı resimlerin boyutu ve dolayısıyla piksel sayısı fazla iken, bazılarının ise boyutu çok büyük ve piksel sayısı fazladır. Piksel sayısı arttıkça o görüntüyü işlemek için gereken zaman da artacaktır. Bunun hesabı zamansal karmaşıklık matrisinde daha rahat görebilirsiniz. Hangi adımların girişteki eleman sayısına daha bağlı olduğu Tablo 2’de görülmektedir.



Şekil 22: Yedinci görüntü için algoritma çalışma hızları

Yedinci görüntü için algoritma çalışma hızları Şekil 22’de görülmektedir.



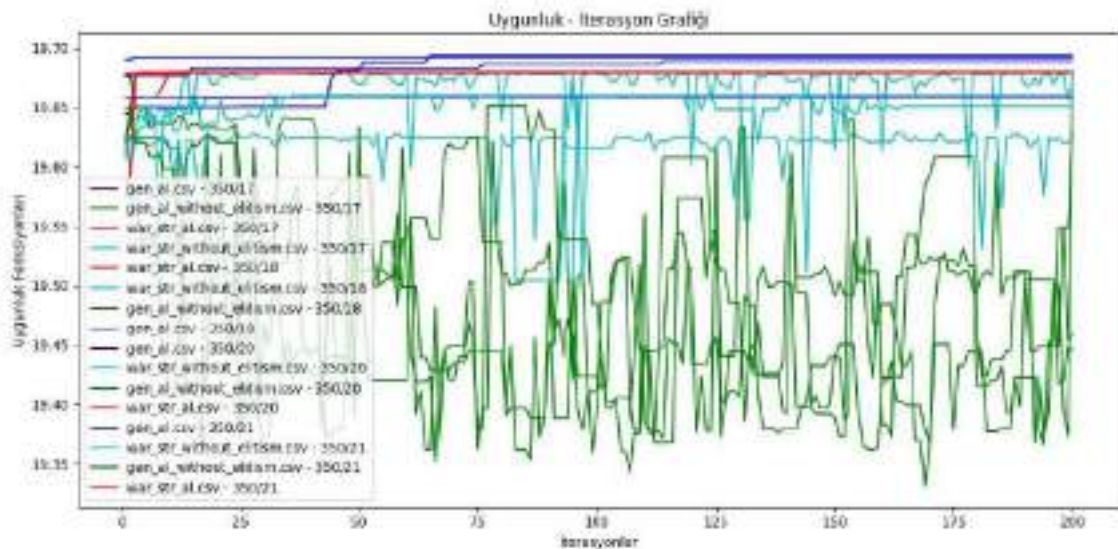
Şekil 23: Sekizinci görüntü için farklı algoritma hızları

Şekil 23’te ise sekizinci görüntü için algoritma tamamlanma süreleri görülüyor. Birden fazla bağımsız bilgisayarda aşağı yukarı aynı performansı göstermeleri bu çalışmanın öne sunduğu algoritmanın etkinliğini görmek için büyük bir avantajdır. Bu grafiklerde de görüleceği üzere algoritmaların bitirme zamanları farklı görüntü ve birden fazla bağımsız bilgisayarlar için aynı sonuçları veriyordur. Genetik algoritma ve elitizimli genetik algoritma iterasyonları tamamlamakta en zayıf kalırken en hızlı yöntemler savaş stratejisi optimizasyon algoritmaları görünüyor



Şekil 24: Dokuzuncu görüntü için farklı algoritma hızları

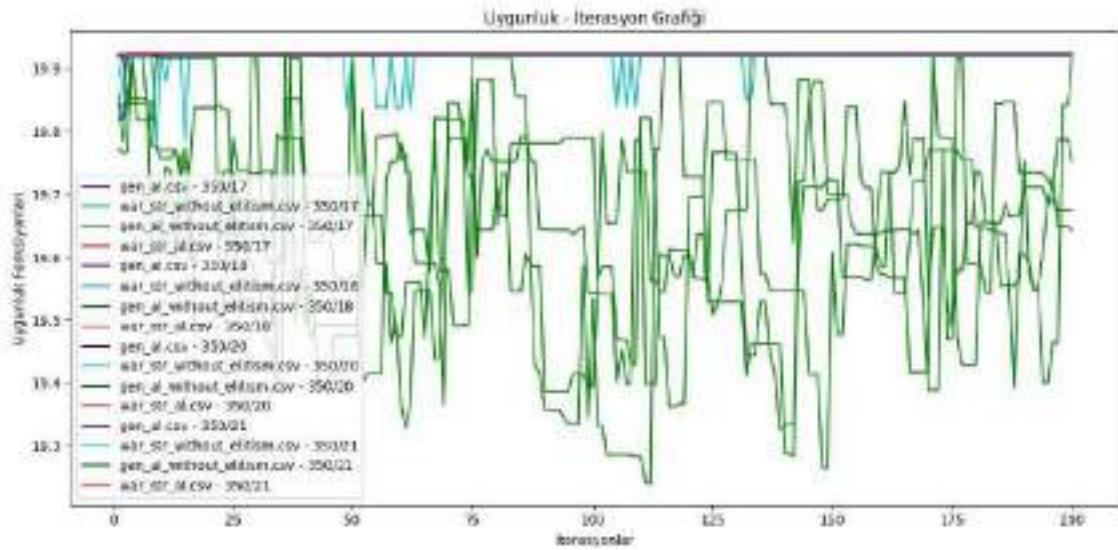
Şekil 24'te dokuzuncu görüntü için algoritma tamamlanma zamanları verilmiştir. Bu görüntü en büyük boyulu görüntülerden biridir. Burada da görüleceği üzere görüntünün boyutu yani algoritmaya giren girdi sayısı arttıkça zamansal karmaşıklığından dolayı savaş stratejileri algoritmaları genetik algoritmalara yaklaşmaktadır.



Şekil 25: Birinci görüntü için algoritmaların uygunluk değerleri

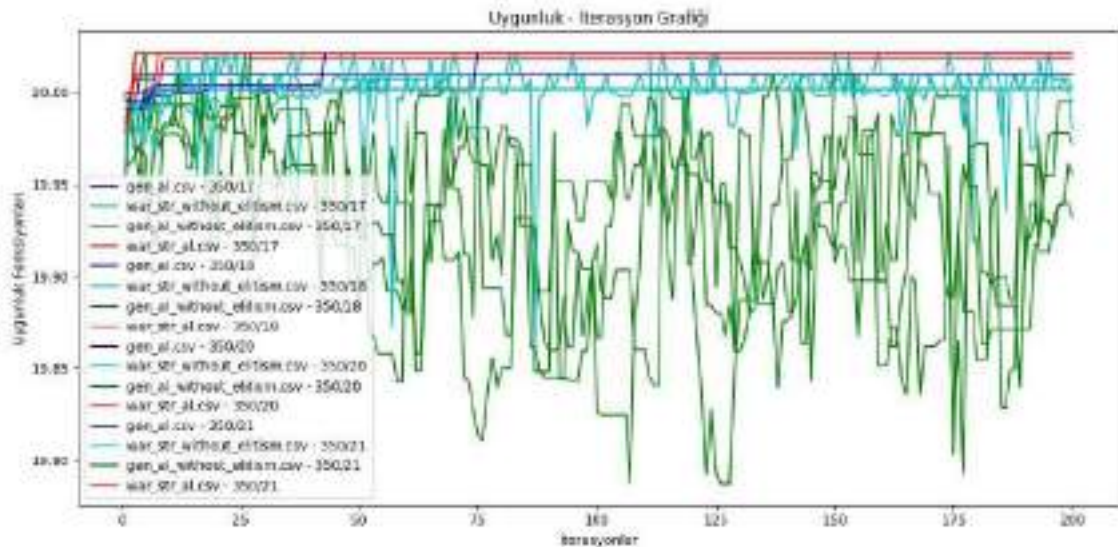
Şekil 25'te birinci görüntü için her iterasyondaki en iyi uygunluk değerleri grafik halinde verilmiştir. Mavi çizgiler elitizimli genetik algoritma, yeşil çizgiler elitizimsiz genetik algoritma, kırmızılar elitizimli savaş stratejisi algoritması ve turkuaz ise

elitizmsiz savaş stratejisi algoritmasıdır. 350/17 veya 350/21 gibi sayılar ise bilgisayarların numaralarını temsil etmektedir.



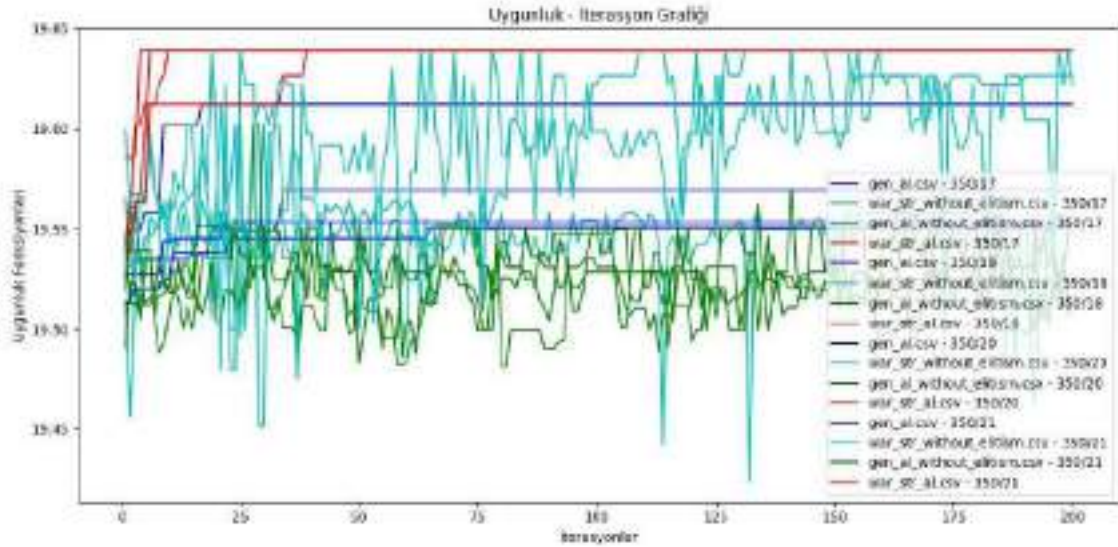
Şekil 26: İkinci görüntü için algoritmaların uygunluk değerleri

Şekil 26'da ikinci görüntü için her iterasyondaki uygunluk değerleri verilmiştir.



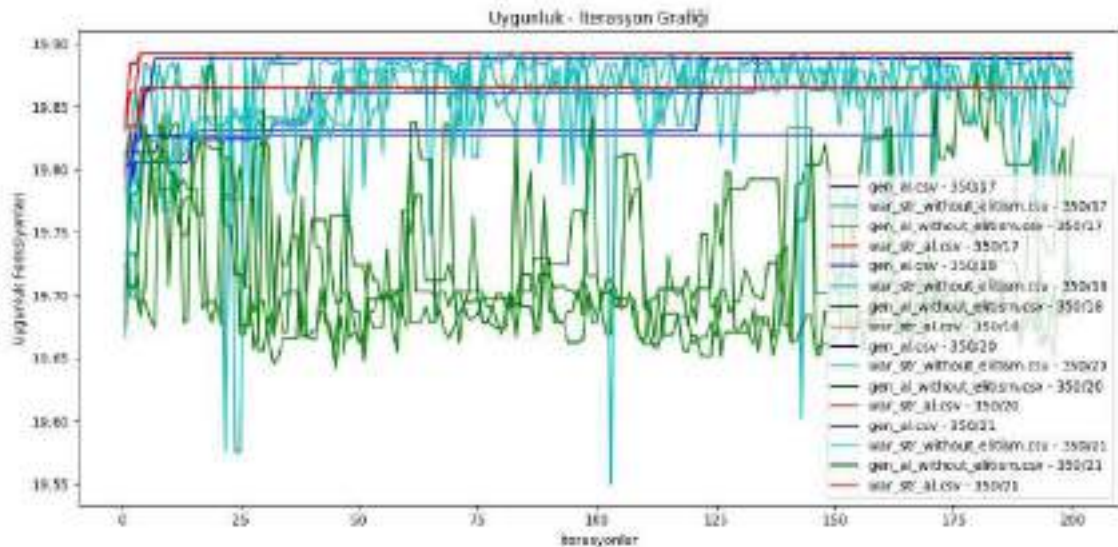
Şekil 27: Üçüncü görüntü için algoritmaların uygunluk değerleri

Şekil 27'de üçüncü görüntü için algoritmaların her iterasyondaki uygunluk değerleri görünüyor. Burada mavi ve kırmızı ile gösterilen algoritmaların elitizmsiz savaş stratejisi olduğu gayet açık bir şekilde görünüyor. Çünkü diğer iki algoritmanın en iyisi sonucu sürekli dalgalı halde değişirken mavi ve kırmızı temsili algoritmaların hiçbir dalgalı hareket yapmadığı görülüyor. Elitizmsiz olan savaş stratejisi ve genetik algoritmalar, aslında Şekil 27'deki gibi dalgalı olmasına rağmen diğer algoritmalarla ya aynı sonucu bulmuş ya da benzer sonuç bulmuşlardır.



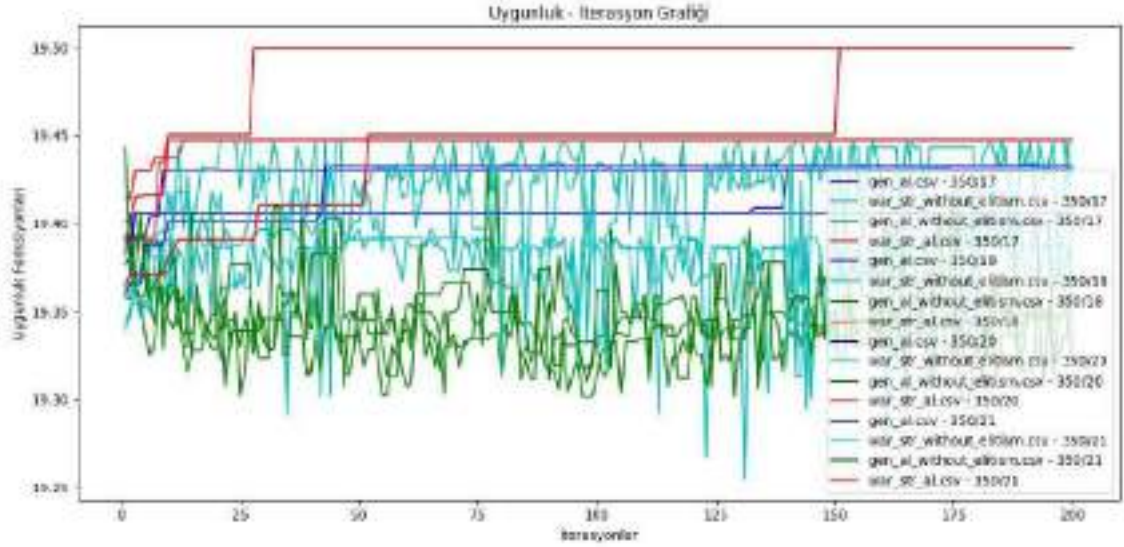
Şekil 28: Dördüncü görüntü için algoritmaların uygunluk değerleri

Şekil 28’de dördüncü görüntü için algoritmaların her iterasyondaki en iyi değerleri verilmiştir. Burada görüldüğü üzere en iyi değerleri elitizimli ve elitizmsiz savaş stratejisi algoritmaları almıştır.



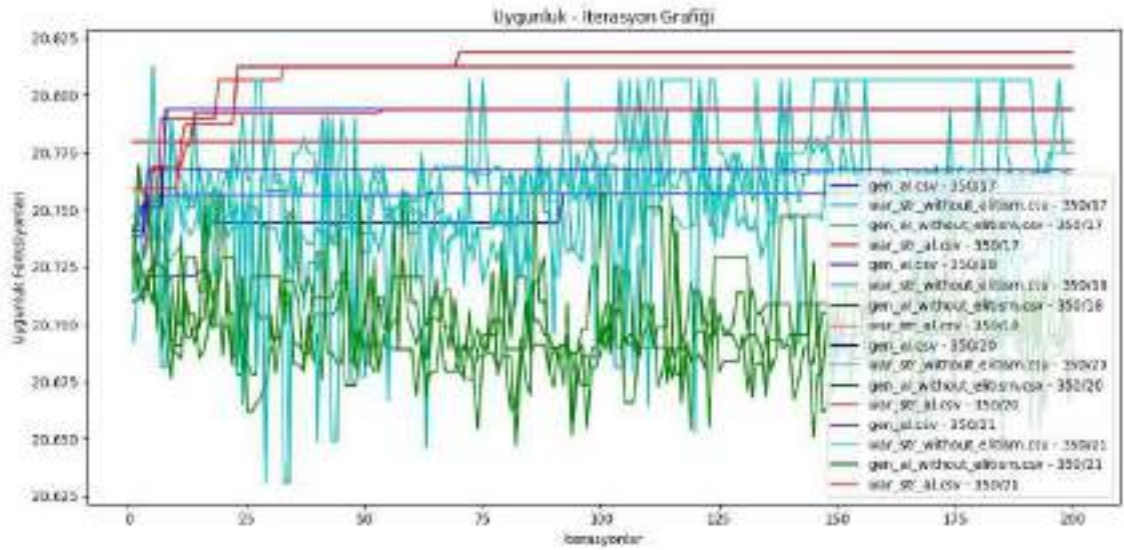
Şekil 29: Beşinci görüntü için algoritmaların uygunluk değerleri

Şekil 29’da beşinci görüntü için farklı bilgisayarlarda da çalıştırılan farklı algoritmaların her iterasyondaki uygunluk değerleri görünüyor. Buradan çıkarılabilecek sonuçlardan bir tanesi elitizmsiz savaş stratejisi algoritmasının genetik algoritmaların çoğuyla aynı ya da çoğundan daha iyi sonuç verdiği görülmektedir. Bundan önceki Şekil 20’yi de göz önünde bulundurursak, önerilen algoritmanın bu resim için istenilen şekilde ve diğer algoritmalara göre daha hızlı çalıştığı söylenebilir.



Şekil 30: Altıncı görüntü için algoritmaların uygunluk değerleri

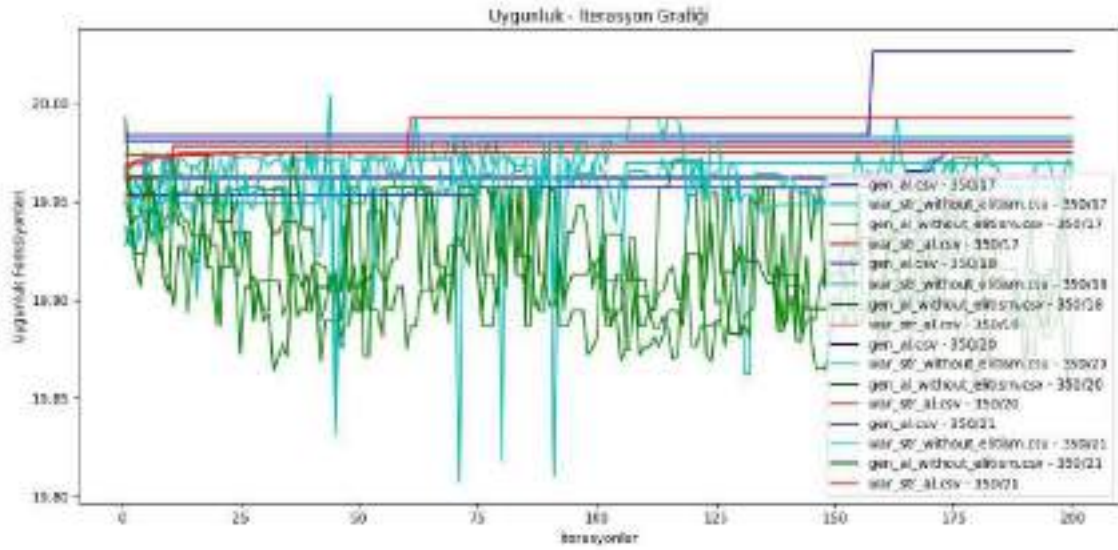
Şekil 29'da altıncı görüntü için farklı bilgisayarlarda çalıştırılan farklı algoritmalar görülebilir.



Şekil 31: Yedinci görüntü için algoritmaların uygunluk değerleri

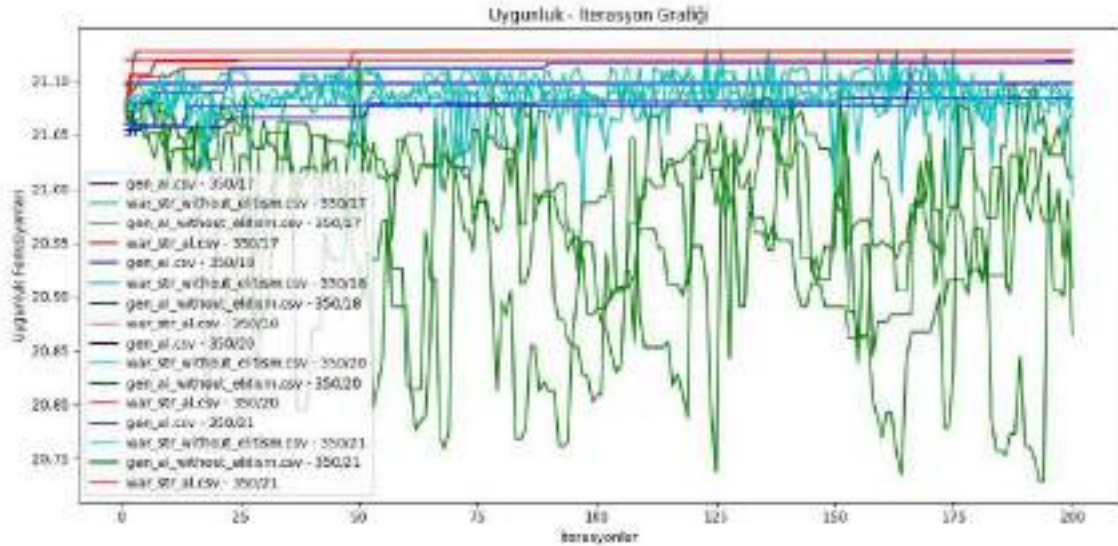
Şekil 31'de yedinci görüntü için en iyi uygunluk değerleri her algoritmanın her iterasyonu için gösterilmiştir. Burada elitizm kullanılmayan algoritmaların daha fazla dalga yaptıkları görülmüştür. Ama her iterasyonda elitizmsiz savaş stratejisi optimizasyon algoritmasının verdiği değerler, yüksek uygunluk değerlerine daha fazla yakınsama göstermiştir. Normal yani elitizimli savaş stratejisi optimizasyon algoritmasının 200 iterasyon sonucunda bulduğu sonuç elitizimli genetik algoritmanınkinden çok daha fazladır. Burada elitizimli savaş stratejisi algoritmasının, elitizimli genetik algoritmadan daha iyi olduğu görülmüştür. Elitizmsiz savaş stratejisi

algoritmasının ise elitizmsiz savaş stratejisi optimizasyon algoritmasından iyi olduğu görülmüştür.



Şekil 32: Sekizinci görüntü için algoritma uygunluk değerleri

Şekil 32’de sekizinci görüntü için her karşılaştırılan algoritmaların her iterasyondaki en iyi değerleri alınmış ve verilmiştir.



Şekil 33: Dokuzuncu görüntü için algoritma uygunluk değerleri

Şekil 33’te dokuzuncu görüntü için algoritma uygunluk değerleri verilmiştir.

5. SONUÇ

Bu tez, 8 bit tıbbi görüntülerin Uyarlanabilir Histogram Eşitlemesi için WSO ve GA algoritmalarının hızlarını karşılaştırmayı amaçladı. Algoritmaların bulunduğu değerlerle histogram 4 parçaya bölünerek oluşturulan histogramların her birinde CLAHE kullanılmıştır. Numpy, opencv, skimage, matplotlib ve Sobel kütüphanelerini içeren Python programlama dili kullanıldı. Bu programları çalıştırmak için CUDA hızlandırması kullanıldı. Kullanılan GPU NVIDIA RTX 3060'tı. WSO'nun orijinal makalesinde elitizm diye bir şey olmamasına rağmen hem GA'da hem de WSO'da elitizm kullanıldı (Ayyarao ve diğerleri, 2022). Bazı görüntü türleri için elitizmi algoritmalar daha iyi sonuçlar verdi. Algoritmayı daha iyi hale getirmek için kullanılan hiperparametrelerden bazıları değiştirilebilir. Örneğin maksimum iterasyon sayısı 200 olarak ayarlandı. Bu 100 de olabilir ve algoritmalar yine de iyi sonuçlar verebilirdi. Yukarıdaki tabloların çoğunda görüldüğü gibi algoritmalar 100 iterasyon sayısına ulaşmadan yakınsamıştır.

6. KAYNAKÇA

- Acharya, U. K., & Kumar, S. (2021a). Directed searching optimized mean-exposure based sub-image histogram equalization for grayscale image enhancement. *Multimedia Tools and Applications*, *80*(16), 24005-24025. <https://doi.org/10.1007/s11042-021-10855-7>
- Acharya, U. K., & Kumar, S. (2021b). Genetic algorithm based adaptive histogram equalization (GAAHE) technique for medical image enhancement. *Optik*, *230*, 166273. <https://doi.org/10.1016/j.ijleo.2021.166273>
- Acharya, U. K., & Kumar, S. (2023). Image sub-division and quadruple clipped adaptive histogram equalization (ISQCAHE) for low exposure image enhancement. *Multidimensional Systems and Signal Processing*, *34*(1), 25-45. <https://doi.org/10.1007/s11045-022-00853-9>
- Ahmadi, S.-A. (2017). Human behavior-based optimization: A novel metaheuristic approach to solve complex optimization problems. *Neural Computing and Applications*, *28*(1), 233-244. <https://doi.org/10.1007/s00521-016-2334-4>
- Ayyarao, T. S. L. V., Ramakrishna, N. S. S., Elavarasan, R. M., Polumahanthi, N., Rambabu, M., Saini, G., Khan, B., & Alatas, B. (2022). War Strategy Optimization Algorithm: A New Effective Metaheuristic Algorithm for Global Optimization. *IEEE Access*, *10*, 25073-25105. <https://doi.org/10.1109/ACCESS.2022.3153493>
- Bandle, C. (2017). Dido's Problem and Its Impact on Modern Mathematics. *Notices of the American Mathematical Society*, *64*(09), 980-984. <https://doi.org/10.1090/noti1576>
- Buchsbaum, G. (1980). An analytical derivation of visual nonlinearity. *IEEE Transactions on Bio-Medical Engineering*, *27*(5), 237-242. <https://doi.org/10.1109/TBME.1980.326628>
- Davis, N., Pittaluga, F., & Panetta, K. (2013). Facial recognition using human visual system algorithms for robotic and UAV platforms. *2013 IEEE Conference on*

Technologies for Practical Robot Applications (TePRA), 1-5.

<https://doi.org/10.1109/TePRA.2013.6556371>

- Dougherty, G. (2009). *Digital Image Processing for Medical Applications*. Cambridge University Press.
- Dutta, M. K., Kaur, M., & Sarkar, R. K. (2021). Image Quality Improvement of Old and Distorted Artworks Using Fuzzy Logic Technique. *Optik*, 249, 168252.
<https://doi.org/10.1016/j.ijleo.2021.168252>
- Erol, O. K., & Eksin, I. (2006). A new optimization method: Big Bang–Big Crunch. *Advances in Engineering Software*, 37(2), 106-111.
<https://doi.org/10.1016/j.advengsoft.2005.04.005>
- Gonzalez, R. C., & Woods, R. E. (2007). *Digital Image Processing*.
- Gödrich, A., König, D., Eilertsen, G., & Teutsch, M. (2023). Joint tone mapping and denoising of thermal infrared images via multi-scale Retinex and multi-task learning. *Infrared Technology and Applications XLIX*, 12534, 275-291.
<https://doi.org/10.1117/12.2663745>
- Hasinoff, S. W. (2014). Photon, Poisson Noise. İçinde K. Ikeuchi (Ed.), *Computer Vision: A Reference Guide* (ss. 608-610). Springer US.
https://doi.org/10.1007/978-0-387-31439-6_482
- Holland, J. H. (1992). *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. MIT Press.
- Hummel, R. (1977). IMAGE ENHANCEMENT BY HISTOGRAM TRANSFORMATION. *Comput Graphics Image Process*, 6(2), 184-185.
[https://doi.org/10.1016/s0146-664x\(77\)80011-7](https://doi.org/10.1016/s0146-664x(77)80011-7)
- Julliand, T., Nozick, V., & Talbot, H. (2016). Image Noise and Digital Image Forensics. İçinde Y.-Q. Shi, H. J. Kim, F. Pérez-González, & I. Echizen (Ed.), *Digital-Forensics and Watermarking* (ss. 3-17). Springer International Publishing.
https://doi.org/10.1007/978-3-319-31960-5_1
- M., Daniella, (2022, September 29). [a building with a tower]. Unsplash.
<https://unsplash.com/photos/3MWqczV7Bo4>

Britannica, T. Editors of Encyclopaedia (2023, August 21). linear programming. Encyclopedia Britannica. <https://www.britannica.com/science/linear-programming-mathematics>

Medical Image Database <https://medpix.nlm.nih.gov>

