

# **ELEKTROFOREZ GÖRÜNTÜ İŞLEME ARAYÜZÜ VE YAZILIMI GELİŞTİRME**

**2019  
YÜKSEK LİSANS TEZİ  
BİYOMEDİKAL MÜHENDİSLİĞİ**

**Kerim ÖZKARA**

**ELEKTROFOREZ GÖRÜNTÜ İŞLEME ARAYÜZÜ VE YAZILIMI  
GELİŞTİRME**

**Kerim ÖZKARA**

**Karabük Üniversitesi**

**Fen Bilimleri Enstitüsü**

**Biyomedikal Mühendisliği Anabilim Dalı**

**Yüksek Lisans Tezi**

**Olarak Hazırlanmıştır.**

**KARABÜK**

**Haziran 2019**

Kerim ÖZKARA tarafından hazırlanan “ELEKTROFOREZ GÖRÜNTÜ İŞLEME ARAYÜZÜ VE YAZILIMI GELİŞTİRME” başlıklı bu tezin Yüksek Lisans Tezi olarak uygun olduğunu onaylarım.

Dr. Öğretim Üyesi Ahmet Reşit KAVSAOĞLU  
Tez Danışmanı, Biyomedikal Mühendisliği Anabilim Dalı

Bu çalışma, jürimiz tarafından oy birliği ile Biyomedikal Mühendisliği Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir. 21/06/2019

Ünvanı, Adı SOYADI (Kurumu)

İmzası

Başkan : Doç. Dr. Kemal POLAT (BAİBÜ)

Üye : Doç. Dr. Ali ARSLANTAŞ (KBÜ)

Üye : Dr. Öğr. Üyesi Ahmet Reşit KAVSAOĞLU (KBÜ)

...../...../2019

KBÜ Fen Bilimleri Enstitüsü Yönetim Kurulu, bu tez ile Yüksek Lisans derecesini onamıştır.

Prof. Dr. Filiz ERSÖZ  
Fen Bilimleri Enstitüsü Müdürü V.



*“Bu tezdeki tüm bilgilerin akademik kurallara ve etik ilkelere uygun olarak elde edildiğini ve sunulduğunu; ayrıca bu kuralların ve ilkelerin gerektirdiği şekilde, bu çalışmadan kaynaklanmayan bütün atıfları yaptığımı beyan ederim.”*

Kerim ÖZKARA

## **ÖZET**

**Yüksek Lisans Tezi**

### **ELEKTROFOREZ GÖRÜNTÜ İŞLEME ARAYÜZÜ VE YAZILIMI GELİŞTİRME**

**Kerim ÖZKARA**

**Karabük Üniversitesi**

**Fen Bilimleri Enstitüsü**

**Biyomedikal Mühendisliği Anabilim Dalı**

**Tez Danışmanı:**

**Dr. Öğretim Üyesi Ahmet Reşit KAVSAOĞLU**

**Haziran 2019, 77 sayfa**

Elektroforez işlemi sonucu elde edilen agaroz jel görüntülerinin analizini el ile yapmak zahmetli ve hata eğilimi yüksek olan işlemlerdir. Bu çalışmada, elektroforez sistemini yürüten gömülü sistem için bir yazılım ve arayüz tasarlanarak bu görüntülerinin analizinin otomatik olarak yapılması gerçekleştirilmiştir. Bu tasarım sayesinde uzun süren ölçümler ve hesaplamalar sonucu yapılan analizler gömülü sistem ile saniyeler içerisinde yapılabilmektedir. Böylelikle yürütme işlemi biter bitmez agaroz jel görüntüsünün incelenmesine imkân tanınmakta ve sonuçlara daha hızlı ulaşılması sağlanmaktadır.

Bu sistem ile UV altında görülen jel görüntülerinin analizi, tasarlanan arayüz yazılımı ile gerçekleştirilmiştir. Görüntüler kamera aracılığı ile tasarlanan arayüz yazılımına

aktarılıp OpenCV kütüphanesiyle görüntü işleme teknikleri Raspberry pi 3 B+ kullanan mini jel elektroforez sisteminin içine entegre edilmiştir.

Elektroforez görüntü işleme arayüzü ve yazılımının gömülü sistem ile kullanılmasıyla sistem, standart elektroforez görüntüleme sistemleri ve analiz için kullanılan bilgisayarlara göre daha kompakt bir yapıya dönüşmüştür. Yapılan karşılaştırmalar neticesinde bantların bulunma başarısı morfolojik geri çatma ve K-means yöntemleri kullanılarak artırılmıştır. Kullanıcılara kolaylık sağlaması açısından bu parametreleri bir arayüz sayesinde görüntüye göre ayarlamak mümkün kılınmıştır. Tasarlanan sistem klasik sistemlere göre daha hızlı, daha verimli ve daha doğru sonuçlar almaktadır. Sistem emsallerinden daha düşük maliyete sahiptir.

Bu çalışma Karabük Üniversitesi Bilimsel Araştırma Projeleri Koordinasyon Birimi tarafından desteklenmiştir. Proje numarası: KBÜBAP-17-YL-473

**Anahtar Sözcükler :** Elektroforez, görüntü işleme, gömülü sistem, bilgisayar arayüz yazılımı

**Bilim Kodu** : 925.1.021

## **ABSTRACT**

**M. Sc. Thesis**

# **ELECTROPHORESIS IMAGE PROCESSING INTERFACE AND SOFTWARE DEVELOPMENT**

**Kerim ÖZKARA**

**Karabük University**

**Graduate School of Natural and Applied Sciences**

**Department of Biomedical Engineering**

**Asist. Prof. Dr. Ahmet Reşit KAVSAOĞLU**

**June 2019, 77 pages**

The analysis of agarose gel images obtained by electrophoresis process is laborious and has high error tendency. In this study by designing a software and interface for the embedded system that runs the electrophoresis system, it is performed to analyze these images automatically. Via this design, the analyzes made as a result of long lasting measurements and calculations can be performed in seconds with the embedded system. Thus, as soon as the execution is completed, the agarose gel image can be examined and results are obtained more quickly.

With this system, the analysis of gel images under UV was performed with the designed interface software. The images are transferred to the interface software via the camera and the image processing techniques are integrated into the mini gel electrophoresis system using Raspberry pi 3 B + with OpenCV library.

By using the electrophoresis image processing interface and software with the embedded system, the system has become more compact compared to standard electrophoresis imaging systems and computers used for analysis. As a result of the comparisons, the presence of the bands was increased by morphological reconstruction and K-means methods. For the convenience of users, it is possible to adjust these parameters according to the image by means of an interface. The designed system is faster, more efficient and more accurate than conventional systems. The system has lower cost than its peers.

This thesis is supported by the project code KBÜBAP-17-YL-473 within the scope of scientific research projects (BAP).

**Key Words** : Electrophoresis, image processing, embedded systems, computer interface software

**Science Code** : 925.1.021

## TEŐEKKÜR

Bu tez alıřmasının planlanmasında, arařtırılmasında, yrtlmesinde ve tamamlanmasında tm ilgisini, bilgisini ve desteęini esirgemeyen, deneyimlerinden yararlandığım, ynlendirme ve bilgilendirmeleriyle alıřmamı Őekillendiren sayın hocam Dr. ęr. yesi Ahmet Reřit KAVSAOęLU'na teŐekkrlerimi sunarım.

Maddi, manevi hibir desteklerini esirgemedен yanımda oldukları iin sevgili aileme tm kalbimle teŐekkr ederim.

## İÇİNDEKİLER

	<b><u>Sayfa</u></b>
KABUL.....	iii
ÖZET.....	v
ABSTRACT.....	vii
TEŞEKKÜR.....	ix
İÇİNDEKİLER .....	x
ŞEKİLLER DİZİNİ.....	xii
ÇİZELGELER DİZİNİ .....	xv
SİMGELER VE KISALTMALAR DİZİNİ .....	xvi
BÖLÜM 1 .....	1
GİRİŞ .....	1
1.1. KONUNUN TANIMI VE TARİHÇESİ.....	4
1.2. LİTERATÜR İNCELEMESİ.....	6
BÖLÜM 2 .....	8
GÖMÜLÜ SİSTEMLER .....	8
2.1. GÖMÜLÜ SİSTEM TANIMI.....	8
2.1.1. Gömülü Sistem Kartı.....	8
2.1.2. Raspberry Pi Özellikleri .....	8
2.2. RASPBIAN KURULUMU .....	10
2.2.1 Kamera Kurulumu .....	12
BÖLÜM 3 .....	14
GÖRÜNTÜ İŞLEME ORTAMI.....	14
3.1. GÖRÜNTÜ İŞLEME TANIMI .....	14
3.3. PYTHON.....	15
3.3.1 Python Kurulumu .....	16
3.2. OPENCV .....	17

	<b><u>Sayfa</u></b>
3.2.1. OpenCV Kurulumu.....	19
3.3. Thonny .....	23
<b>BÖLÜM 4 .....</b>	<b>25</b>
<b>YÖNTEM VE TEKNİKLER.....</b>	<b>25</b>
4.1. FİLTRELEME VE GÖRÜNTÜ İYİLEŞTİRME YÖNTEMLERİ .....	25
4.1.1. Eşikleme (Thresholding) .....	25
4.1.2. Histogram Eşitleme (Histogram Equalization).....	27
4.1.4. Morfolojik Geri Çatma (Morphological Reconstruction) .....	31
4.1.5. Gabor Filtresi .....	35
4.1.6. K-Means Kümeleme (K-means Clustering) .....	39
4.2. SONUÇLAR VE TARTIŞMA.....	41
<b>BÖLÜM 5 .....</b>	<b>43</b>
<b>ELEKTROFOREZ GÖRÜNTÜLERİNİN ANALİZİ.....</b>	<b>43</b>
5.1. GÖRÜNTÜLERİN ÖLÇME PROGRAMI İLE ANALİZİ.....	43
5.2. GÖRÜNTÜLERİN YAZILIM İLE ANALİZİ .....	48
5.2.1. Şerit ve Bantların Bulunması.....	48
5.2.2. Bantların Konumlarının Tespiti.....	55
5.2.3. Smile Etkisi Düzeltme .....	59
5.3. FİLTRELENMİŞ GÖRÜNTÜLERDE BANTLARIN BULUNMASI.....	62
5.3.1. Mesafeye Göre BP Hesaplayan Fonksiyon Oluşturma .....	64
5.4. SONUÇLAR VE TARTIŞMA.....	66
<b>BÖLÜM 6 .....</b>	<b>68</b>
<b>ARAYÜZ.....</b>	<b>68</b>
6.1. Arayüz kullanımı.....	68
<b>BÖLÜM 7 .....</b>	<b>72</b>
<b>SONUÇLAR VE ÖNERİLER .....</b>	<b>72</b>
<b>KAYNAKLAR .....</b>	<b>74</b>
<b>ÖZGEÇMİŞ .....</b>	<b>77</b>

## ŞEKİLLER DİZİNİ

### Sayfa

Şekil 1.1.	Sistemin blok şeması.....	2
Şekil 1.2.	Arayüz yazılımının akış şeması.....	3
Şekil 1.3.	Sistemin görünümü .....	3
Şekil 2.1.	Raspberry Pi 3 B+.....	9
Şekil 2.2.	Win32DiskImager programı .....	11
Şekil 2.3.	Raspbian işletim sistemi açılış ekranı .....	11
Şekil 2.4.	Raspbian işletim sistemi.....	12
Şekil 2.5.	Raspi-config menüsü kamerayı aktifleştirme seçeneği .....	12
Şekil 2.6.	Kamera sürücülerini yükleyen komut. ....	13
Şekil 2.7.	V4l uygulamasını aktifleştirme komutu.....	13
Şekil 2.8.	Sisteme bağlı kameraların listesini gösteren komut.....	13
Şekil 2.9.	Kamera bağlı ise terminalde görülen çıktı. ....	13
Şekil 3.1.	Raspbian işletim sistemi terminali .....	16
Şekil 3.2.	Raspbian işletim sistemini güncelleme komutu .....	16
Şekil 3.3.	Python kurulum kodları.....	17
Şekil 3.4.	Raspberry konfigürasyon ekranı .....	19
Şekil 3.5.	OpenCV'nin eski sürümlerini kaldırmak için kullanılan komutlar.....	19
Şekil 3.6.	Raspbian işletim sistemi güncelleme komutları.....	20
Şekil 3.7.	Geliştirici araçların kurulum kodları .....	20
Şekil 3.8.	Gerekli kütüphanelerin kurulmasını sağlayan komut.....	20
Şekil 3.9.	OpenCV dosyalarını klasöre kaydeden komutlar.....	21
Şekil 3.10.	Sıkıştırılmış dosyaları çıkaran komutlar .....	21
Şekil 3.11.	Build klasörü oluşturma komutları.....	21
Şekil 3.12.	OpenCV derleme için dosyaları hazır hale getirme komutları.....	22
Şekil 3.13.	Derleme işleminin başarılı olarak tamamlandığını gösteren bildirim mesajları .....	22
Şekil 3.14.	Derleme işlemini başlatan komut.....	23
Şekil 3.15.	OpenCV kurulumunu başlatma komutları. ....	23
Şekil 3.16.	Python 3 IDE OpenCV kütüphanesinin versiyonunu gösteren komutlar. ....	23

Şekil 4.1.	a) Orijinal jel görüntüsü, b) Eşikleme uygulanmış jel görüntüsü. ....	26
Şekil 4.2.	(a) Orijinal jel görüntüsü, (b) Histogram eşitlemesi uygulanan jel görüntüsü, (c) Orijinal jel görüntüsünün histogram grafiği, (d) Histogram eşitlemesi uygulanan jel görüntüsünün histogram grafiği.....	27
Şekil 4.3.	a) HE uygulanmış jel görüntüsü, b) HE uygulanmış jel görüntüsüne eşikleme uygulanmış görüntü, c) İki görüntünün dikey piksellerinin toplamlarının karşılaştırması, d) İki görüntünün ilk şeritlerinin yatay piksellerin toplamının karşılaştırılması. ....	28
Şekil 4.4.	a) Orijinal jel görüntüsü, b) CLAHE uygulanmış jel görüntüsü, c) Orijinal jel görüntüsü histogram grafiği, d) CLAHE uygulanmış jel görüntüsü histogram grafiği. ....	30
Şekil 4.5.	a) Orijinal jel görüntüsü, b) CLAHE ve eşikleme uygulanmış jel görüntüsü, c) Orijinal görüntünün histogram grafiği, d) CLAHE ve eşikleme uygulanmış jel görüntüsünün histogram grafiği. ....	31
Şekil 4.6.	Bir jel görüntüsünün 3D gösterimi.....	33
Şekil 4.7.	Morfolojik geri çatma sonucu elde edilmiş arka planın 3D gösterimi. ...	33
Şekil 4.8.	Arka plan görüntüsü çıkarılmış görüntünün 3D gösterimi. ....	34
Şekil 4.9.	Mini jel elektroforez sistemi ile çekilmiş jel görüntüsü.....	34
Şekil 4.10.	Morfolojik geri çatma ile elde edilen arka plan görüntüsü. ....	35
Şekil 4.11.	Arka plan görüntüsünden arındırılmış görüntü. ....	35
Şekil 4.12.	Her biri 11.25° açı ile yönlendirilmiş 16 Gabor filtre bankası[33]. ....	36
Şekil 4.13.	Diğer parametreleri değiştirmeden ( $\lambda = 30, \theta = 0, \gamma = 0.25, \Psi = 0$ ) ve sigma değerinin Gabor çekirdeğine etkisi. ....	37
Şekil 4.14.	Diğer parametrelerin değiştirmeden ( $\lambda = 30, \gamma = 0.25, \sigma = 10, \Psi = 0$ ) ve teta değerinin Gabor çekirdeğine etkisi. ....	37
Şekil 4.15.	Diğer parametreleri değiştirmeden ( $\theta = 0, \gamma = 0.25, \sigma = 10, \Psi = 0$ ) ve lambda değerinin Gabor çekirdeğine etkisi.....	37
Şekil 4.16.	Diğer parametreleri değiştirmeden ( $\lambda = 30, \theta = 0^\circ, \sigma = 10, \Psi = 0$ ) ve gama değerinin Gabor çekirdeğine etkisi.....	38
Şekil 4.17.	a) Orijinal görüntü, b) Gabor filtresi uygulanmış görüntü. ....	39
Şekil 4.18.	a) K-Means kümeleme ile elde edilen görüntü, b) Eşikleme yöntemi uygulanmış K-Means kümeleme yapılmış görüntü. ....	41
Şekil 5.1.	MiCam 2.4 ölçüm programı [37]. ....	44
Şekil 5.2.	Ölçüm esnasında yapılan hatalar.....	45
Şekil 5.3.	Yürütülmüş Ladder DNA örneği [38]. ....	46
Şekil 5.4.	Baz çifti- mesafe grafiği ve uydurulan eğrinin $R^2$ değeri.....	47
Şekil 5.5.	Programın akış diyagramı. ....	50

Şekil 5.6. findLane() fonksiyonunun akış diyagramı. ....	50
Şekil 5.7. findBands() fonksiyonunun akış diyagramı. ....	51
Şekil 5.8. calculateBP() fonksiyonunun akış diyagramı. ....	51
Şekil 5.9. Mini Jel elektroforez sistemiyle yürütülmüş Ladder DNA görüntüsü. ...	52
Şekil 5.10. Jel görüntüsündeki piksellerin dikey olarak toplanmasıyla oluşan sinyal. .....	52
Şekil 5. 11. Sinyalin Baseline değerlerini bulan komut .....	52
Şekil 5.12. Orijinal sinyal (mavi), baseline sinyali(turuncu). ....	53
Şekil 5.13. Baseline gürültüsü düzeltilmiş sinyal. ....	53
Şekil 5.14. Düzeltilmiş sinyalin türevi. ....	54
Şekil 5.15. Türevi alınan sinyalin Mean Filter ile filtrelenmiş hali (mavi), tepe noktaları şeritlerin sınırları (turuncu). ....	54
Şekil 5.16. Algoritmanın bulduğu şerit sınırlarının orijinal resimde çizgilerle gösterilmesi. ....	55
Şekil 5.17. Şeritteki piksellerin yatay olarak toplanması sonucu elde edilen sinyal..	55
Şekil 5.18. Filtrelenmiş ve tepe noktaları bulunmuş sinyal. ....	56
Şekil 5.19. Tepe noktalarının bulunduğu piksellerin resim üzerinde gösterimi. ....	56
Şekil 5.20. Baz çiftleri ve Piksellerin ilişkisini gösteren grafik ve uydurulan eğrinin $R^2$ değeri. ....	57
Şekil 5.21. Program tarafından bulunan bantların piksel cinsinden konumlarının gösterimi. ....	59
Şekil 5.22. a) Orijinal görüntü sinyali, b) Smile etkisi düzeltilmiş sinyal, c) Orijinal görüntü, d) Smile etkisi düzeltilmiş görüntü. ....	60
Şekil 5.23. Smile etkisi düzeltilmiş görüntü. ....	61
Şekil 5.24. a) Smile etkisi düzeltilmemiş görüntü, b) Smile etkisi düzeltilmiş görüntü. .....	61
Şekil 5.25. Farklı filtrelerle elde edilen görüntülere bant bulma algoritmasının uygulanması. ....	63
Şekil 5.26. K-Means Kümeleme sonrası algoritma ile bantların bulunması.....	63
Şekil 5.27. Piksel-Baz Çifti grafiği, uydurulan eğri fonksiyonlarının karşılaştırılması a) Polinom fonksiyon, b) Üs fonksiyon, c) Üstel fonksiyon .....	65
Şekil 6.1. Programın ana ekranı. ....	69
Şekil 6.2. Yüklenmiş jel görüntüsü. ....	69
Şekil 6.3. Filtrelenmiş görüntü. ....	70
Şekil 6.4. Piksel değerlerinin jel görüntüsü üzerine yazılması. ....	71

## ÇİZELGELER DİZİNİ

	<b><u>Sayfa</u></b>
Çizelge 2.1. Raspberry Pi Model karşılaştırma çizelgesi.....	9
Çizelge 5.1. Ölçüm işlemi sonucunda baz çiftlerinin aldığı mesafeler ( $\mu\text{m}$ ).....	46
Çizelge 5.2. Fonksiyon ile tahmin edilen değerlerin, gerçek değerler ile karşılaştırılması. ....	48
Çizelge 5.3. Baz çiftlerinin bulunduğu pikselleri gösteren çizelge.....	57
Çizelge 5.4. Tahmin edilen BP sayıları ile gerçek değerler arasındaki farkları gösteren çizelge.....	58
Çizelge 5.5. Algoritma ile bulunan bantlar ile baz çifti sayılarını gösteren çizelge. ....	64
Çizelge 5.6. Oluşturulan fonksiyonlarla hesaplanan BP değerleri ile gerçek BP değerleri arasındaki hataları gösteren çizelge. ....	66

## SİMGELER VE KISALTMALAR DİZİNİ

### SİMGELER

$\alpha$  : Alfa

$\beta$  : Beta

$\lambda$  : Lambda

$\gamma$  : Gama

$\sigma$  : Sigma

$\Psi$  : Psi

$\theta$  : Theta

$y$  : Fonksiyon

## KISALTMALAR

DNA	: Deoksiribo Nükleik Asit
Bp	: Baz çifti
UV	: Ultraviyole
pH	: Power of hydrogen (bir çözeltinin asitlik veya bazlık derecesini tarif eden ölçü birimidir.)
mL	: Mililitre
SyBr	: Güvenli DNA jel leke
Ng	: Nanogram
CCD	: Charge Coupled Device (Şarj Eşleştirmeli Cihaz)
Kb	: Kilobaz
CLAHE	: Contrast Limited Adaptive Histogram Equalization (Kontrast limitli adaptif histogram eşitleme)
BG Clear	: Background Clear (Arka plan temizleme)
3D	: 3 Boyutlu
$\mu\text{m}$	: Mikrometre
BG	: Background (arka plan)
IOT	: Internet of things (Nesnelerin interneti)
RF	: Relative Front (Bağıl ön)

## BÖLÜM 1

### GİRİŞ

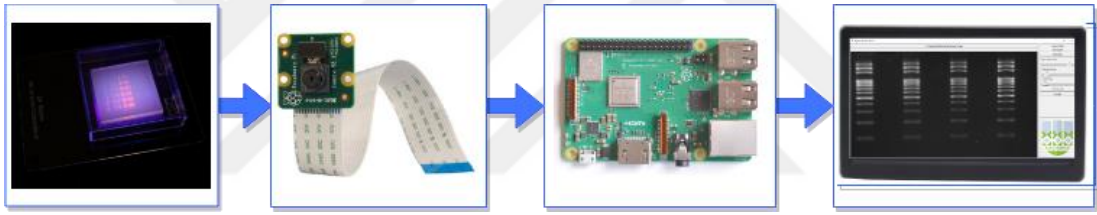
Elektroforez, bir yerden diğere bir yere hareket etme anlamına gelen “phoresis” kelimesinden türetilmiştir. Elektroforez bir elektrik alanının etkisi altında yüklü parçacıkların göç etmesi esasına dayanır. Bu yüklü parçacıklar bakteriler, virüsler, protein molekülleri, nükleik asitler veya sentetik parçacıklar olabilir. Elektroforez yöntemi etkin ayırma gücü, uygulama kolaylığı ve ucuzluğu nedenleri ile tercih edilen, aynı zamanda sürekli geliştirilen bir elektrokinetik yöntemdir [1].

DNA jel elektroforezi, farklı boyutlarda DNA parçalarını ayıran moleküler biyoloji tekniğidir. DNA jel elektroforezinin uygulamaları arasında DNA parmak izi (genetik tanı), DNA'nın boyut tahmini için DNA ayırımı yer alır. DNA bantlama kalıplarının elektroforetik görüntülerden doğru şekilde yorumlanması zahmetli olabilir ve elle sorgulanan çok sayıda bant olduğundan hata eğilimli olabilmektedir [2].

DNA ve protein elektroforez jellerinin analizi, tür çeşitliliğinin belirlenmesi için gittikçe önem kazanmaktadır. Farklı moleküler ağırlıkların DNA fraksiyonları veya proteinleri, moleküler ağırlığa karşılık gelen pozisyonlara sahip bir dizi bantlara elektroforez kullanılarak ayrılır. Jellerin görüntü analizi, bant pozisyonunun ve numuneler arasındaki yoğunluğun manuel olarak karşılaştırılmasının özelliğinin çoğunu ortadan kaldırır. Bu işlemleri bir algoritma ile gömülü sistemde kompakt bir halde yaptırılarak bilgisayarın kapladığı alan ve harcadığı enerji miktarından tasarruf etmek mümkündür. İşleminde ilk adım, görüntüye bazı filtreler uygulanarak görüntüdeki parazitlerden kurtulmak gerekliliğidir.

Arka plan sisi, tahmin edilerek resimden çıkarılır. Bantlar daha sonra tespit edilir ve her bandın konumları ve yoğunlukları belirlenir. Şeritlerin birinde (Ladder DNA olan şerit) bilinen moleküler ağırlıkların bir dizi protein veya nükleotidini dâhil ederek, her numunedeki bantların bilinmeyen moleküler ağırlıkları, konumlarına dayanarak tahmin edilebilir. Bu uygulamada görüntü analiz tekniklerinin kullanımı örnekler arasındaki farklılıkları nispeten objektif olarak tanımlamanın hızlı ve ucuz yöntemidir [3].

Bu tez kapsamında Gömülü Sistem tabanlı Mini Jel Elektroforez [4] sisteminde yürütülmüş, SyBr safe green ile renklendirilmiş DNA'nın görüntüsünün, gömülü sistemde analizinin en hızlı ve en hassas şekilde, hangi yöntemler kullanarak, en az hata ile tespit edileceği araştırılıp, bu minvalde bir arayüz ve yazılım tasarlanmıştır. Şekil 1.1.'de sistemin blok şeması gösterilmektedir.

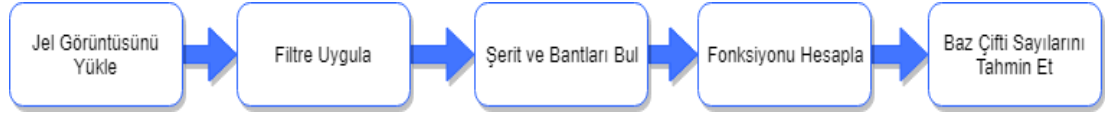


Şekil 1.1. Sistemin blok şeması.

UV transillüminatör altında elde edilen Agaroz jel görüntüleri gömülü sisteme bir kamera vasıtası ile kaliteli bir şekilde aktarılmaktadır. Görüntü alma yöntemi yaygın olarak etidyum bromide ile renklendirilmiş DNA'nın UV ışık altında bir dijital kamera ile görüntüsünün alınmasından oluşmaktadır. Sonra bu görüntü bilgisayar ortamına aktarılıp açısal bozukluğu varsa düzeltilip, görüntü analizi yaptırılmak üzere bir ölçüm programına aktarılmaktadır. Bu programda bantların yürüdükleri mesafeleri belli bir referans baz alan sanal bir cetvel ile ölçülebilmektedir. Bu yöntemde kamera açısını sabitleyerek bozulma önlenirse de ışık yansımaları ya da gönderilen ışığın yüzeyde ya da ortamda patlamasıyla oluşan kontrast bozunmalarını önlemek pek mümkün olmamaktadır.

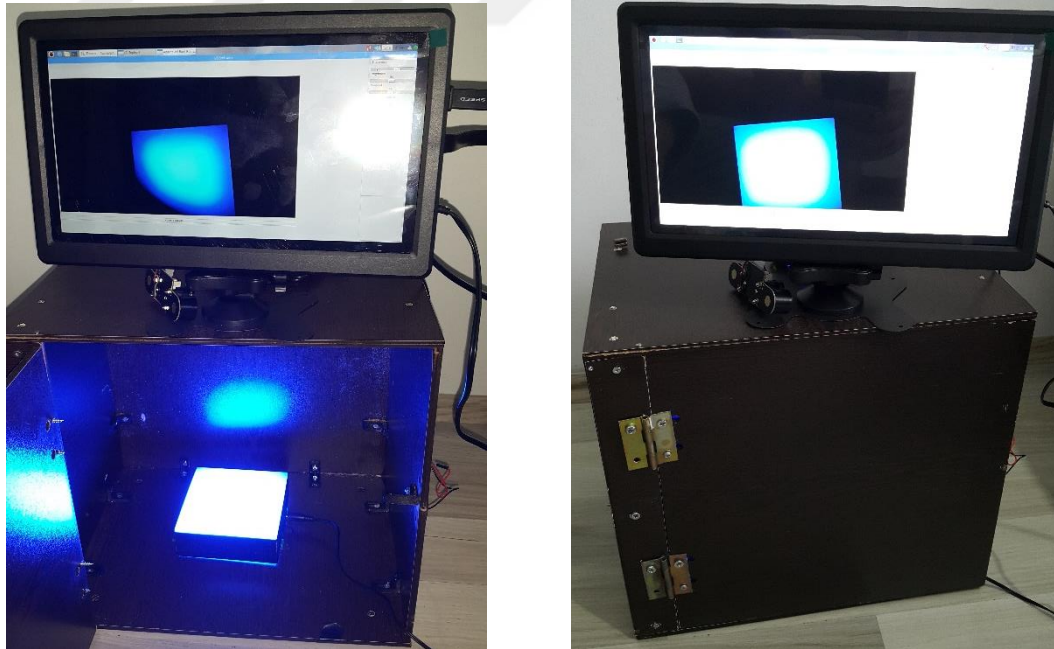
Şekil 1.2.'de bu çalışmada gerçekleştirilen arayüz yazılımının akış şeması verilmektedir. Yükleme işleminden sonra görüntülerdeki gürültünün azaltılması için

morfolojik geri çatma (BG Clear=Background Clear), histogram eşitleme (HE), kontrast limitli adaptif histogram eşitleme (CLAHE), Gabor filtresi, K-means gibi çeşitli filtre ve yöntemler uygulandıktan sonra, şeritlerin ve bantların görüntü işleme algoritmasıyla tespitleri hızlı bir şekilde ve kolaylıkla sağlanmaktadır.



Şekil 1.2. Arayüz yazılımının akış şeması.

Agaroz jel görüntüsünün gömülü sistem içinde çalışabilen, kullanıcı tarafından görüntü ayarlarının yapılabildiği bir arayüze sahip bir uygulama yazarak görüntünün analizini sağlayan bir sistemin mevcut olması işlem kolaylığı sağlamıştır. Gömülü sistem ile görüntü almada oluşabilecek açısız olarak düzgün olmama, istenmeyen ışık yansımaları, ışığın dağılması ya da patlaması gibi görüntü bozukluklarını giderecek bir yazılım ve arayüz tasarlanmıştır. Sistemin görünümü Şekil 1.3’de verilmiştir.



Şekil 1.3. Sistemin görünümü.

Bu tezin yazımında şu aşamalar izlenmiştir. Bölüm 1’de konunun tanımı ve tarihçesi verildikten sonra literatür incelemesi yapılmıştır. Bölüm 2’de Gömülü sistemin tanımı ve kullanılan gömülü sistem kartı hakkında detaylı bilgiler verilmiştir.

Bölüm 3’de görüntü işlemenin tanımı ve kullanım alanlarından bahsedilmiştir. Hangi programlama dilinin ve hangi kütüphanenin kullanılacağına dair bilgiler verilmiştir. Bölüm 4’te görüntü işlemede kullanılacak yöntem ve tekniklere değinilmiş, uygulanan filtreler hakkında bilgiler verilmiş ve bu filtrelerin jel görüntüleri üzerinde etkileri gösterilmiştir.

Bölüm 5’te Elektroforez görüntülerinin analizi hakkında bilgiler verilmiş ölçüm programı ile yapılan analizler ile algoritma ile yapılan analizler karşılaştırılmıştır. Bölüm 6’da tez çalışması kapsamında tasarlanan program arayüzünün kullanımına dair açıklamalara yer verilmiştir.

Bölüm 7’de ise elde edilen bilgilerle uygulamaların sonuçları değerlendirilmektedir. Ayrıca ileride yapılabilecek çalışmalar hakkında önerilerde bulunmaktadır.

## **1.1. KONUNUN TANIMI VE TARİHÇESİ**

Smith vd. göre elektroforez makro molekülleri büyüklük, elektrik yükü ve diğer fiziksel özellikler temelinde ayıran bir yöntemdir. Elektroforez (electrophoresis) terimi yüklü partiküllerin elektrik akımı etkisi altında hareketini açıklamaktadır. Elektro (electro) elektriğe karşılık gelir, forez (phoresis) ise anlamı taşımak/karşıya geçirmek olan yunanca kökenli bir kelimedir. Kâğıt, selüloz asetat, nişasta jeli ve polikrilamid jeli olmak üzere sıvı desteği olarak bir dizi farklı ortam kullanılmıştır. Net bir pozitif yük taşıyan iyonlar veya parçacıklar, elektrik alanının negatif kutbuna doğru hareket etme eğilimindedir, tersi de doğrudur. Yük-kütle oranına bağlı olarak, belirli bir iyon veya parçacık türünün hareket hızı olarak da tanımlanabilir. Bu fenomen, proteinlerin, nükleik asitlerin ve analitik diğer yüklü moleküler türlerin ayrılmasında ve ayrıca dağınmış hücrelerin veya diğer makroskobik parçacıkların heterojen popülasyonlarının analitik veya fraksiyonlanmasında yaygın olarak uygulanmıştır. Bir diğer tanımı ise iyonların veya yüklü parçacıkların göç etmelerine neden olma işlemi veya böyle bir olguya dayanan herhangi bir teknik olarak tanımlanmaktadır. Jel elektroforezi, bir çeşit bölge elektroforezi olan, destekleyici ortam olarak çoğunlukla poliakrilamid, agaroz veya nişasta gibi konsantrasyon kullanılan bir elektroforez tekniğidir [5].

İletken bir çözeltiye gerilim uygulandığında çözültide bulunan iyonlar sayesinde bir akıntı oluşur. Aynı çözelti kullanılarak koyu kıvamlı bir jel oluşturulur. Bu yüzden jel de iletken bir yapıya sahiptir. Burada jelin amacı çözelti içindeki iyonlar akıp giderken jelin içindeki iyonların daha yoğun yapıya sahip bir ortamda olduklarından dolayı fırlayıp gitmesini önlemek için kullanılır. Jel ortam kuyucuklarında bulunan örneklerin yavaş ilerlemesini sağlayan bir nevi yoğun kıvamlı çözültidir. Elektroforez için gerekli güç, tankın problemlerine bağlı elektrotlara uygulanan voltajdan elde edilmektedir. Moleküllerin elektrik alan etkisiyle jel ortamdaki hareket etme hızı, moleküllerin özelliklerine ve jelin yoğunluğuna bağlı olarak değişmektedir.

Elektroforez sırasında, bir örnek DNA elektrik akımına maruz kalır. Pozitif yüklü iyonlar negatif elektrotlara doğru hareket eder ve negatif yüklü iyonlar pozitif yüklü elektrotlara doğru hareket eder. Farklı iyonlar farklı oranlarda göç ettiklerinden, elektroforez ile etkili bir şekilde ayrılabilirler [5].

Elektroforez ayırma tekniği ilk önce 1807 yılında bilim adamları tarafından kullanılmak üzere geliştirilmiştir. O zamandan beri araştırmacılar, kimyagerler ve teknisyenler, farklı yüklü parçacıkları bir elektrik alanı kullanarak ayırmak için elektroforezi kullanmışlardır. Her ne kadar elektroforez ilk olarak 19. yüzyılın başlarında keşfedilse de 1937 yılına kadar bilimsel uygulamalarda yaygın olarak kullanılmamıştır. Bu ayırma yöntemi İsveçli kimyacı Arne Tiselius tarafından serum proteinleri üzerinde çalışırken geliştirilmiş ve bu çalışmadan dolayı kendisi 1948'de Nobel Ödülü ile ödüllendirilmiştir. Elektroforezin kapillerde gerçekleştirilmesinin avantajları 1980'lerin başlarında Jorgenson ve Lukas'ın çalışmaları ile belirginleşmiştir. 1937'de tanımlanan serbest solüsyon elektroforezi, frontal elektroforez veya "moving boundary" elektroforezidir. Tiselius bir elektrolit solüsyonunda çözünmüş olan proteinleri, protein-elektrolit solüsyonunun bulunduğu "U" şeklindeki kuartz bir borunun içinden elektrik akımı geçirerek ayırmıştır.  $pH$  7.6'da albumin,  $\alpha$ ,  $\beta$  ve  $\gamma$  olarak isimlendirilen 4 serum protein fraksiyonunu saptamış ve bu bantların sınırları arasındaki absorbans değişikliğini optik olarak ölçmüştür. Bu teknik günümüzde de elektroforetik hareketlilik ve protein-protein etkileşimi ile ilgili araştırmalarda kullanılmaktadır; fakat klinik laboratuvarlarda rutin çalışmalar için kullanılmaz, karmaşık bir araç gerektirir, tekniği zordur. En yaygın

elektroforez uygulamaları, serum proteinleri, hemoglobinler ve izoenzimleri içermektedir [6].

## 1.2. LİTERATÜR İNCELEMESİ

Tüm elektroforez tiplerinin temeli  $V = I.R$  ve  $P = I^2R$  elektriksel eşitliklerine dayanır. Aralarındaki başlıca fark destek ortamının tipi ve konumudur. Dikey ya da yatay konumdaki destek ortamı; selüloz, agaroz ya da poliakrilamid vb. jellerden hazırlanabilir. Selüloz, amino asitler ve karbonhidratlar gibi düşük molekül ağırlıklı olan moleküller için, poliakrilamid gibi jeller ise daha büyük olanlar (protein, nükleik asitler) için destek ortamı olarak kullanılır [7].

Kevin M. Bourzac vd. aktardığına göre Etidyum bromide kanserojen bir boya olduğu için bunun yerine daha güvenilir ve gelişmiş boyama özelliklerine sahip SyBr diye bilinen safe green renklendiricisi kullanılarak jel görüntüsü alınmaktadır [8].

Green ve Sambrook'un araştırmalarına göre DNA'nın çoğu rutin elektroforetik analizi agaroz mini jeller kullanılarak çok hızlı bir şekilde gerçekleştirilebilir. Mini jeller, klonlama protokolünde bir sonraki adımın atılmasından önce hızlı bir cevap gerektiğinde özellikle faydalıdır. Kuyular daha küçük ve jeller daha ince olduğundan, görselleştirme için normalden daha az DNA gereklidir. Ayrıca, jeller önceden hazırlanıp hızlıca çalıştırılabildiğinden ve daha az miktarda reaktif gerektirdiğinden, hem zamandan hem de paradan önemli miktarda tasarruflar sağlamaktadır. Pek çok araştırmacı haftanın başında bir jel hazırlar ve bir deney boyunca tekrar tekrar kullanır. Böylece, belirli bir numune seti yüklenebilir, jelin üzerine dökülür ve görselleştirilir. Jel daha sonra örnekleri jelden tamponun içine sokmak suretiyle "silinebilir". Mini jellerin 3 kilo bazdan daha küçük DNA parçalarının analizi için en uygun ebatlardır. Daha büyük parçalar, genel olarak kullanılan daha büyük voltajlar ve jelin nispeten kısa olan boyu nedeniyle zayıf şekilde çözülür. Çeşitli tiplerde minyatür elektroforez tankları ticari olarak üretilmektedir. Her bir jel yuvası, jelin kalınlığına ve tarak dişlerinin genişliğine bağlı olarak 3 – 12 $\mu$ L örnek alır. Genellikle, tercih edilen jel doldurma tamponunda 10 – 100ng DNA bir yuvaya uygulanır. Jele daha sonra 30-60 dakika boyunca jelin boyutlarına göre cm başına 5 – 20V arası gerilim uygulanıp

SyBr Safe green uygun mesafeye geçene kadar çalıştırılır [9]. Buradan yorumlanabileceği üzere 5 cm'lik bir agaroz jele 25 – 100V arası gerilim uygulamak gerekmektedir. Yürütme işlemi tamamlandığında agaroz jelin görüntüsü UV ışığı altında bir CCD kamera ile yakalanabilmektedir.

Bant buldurma işlemini yapan GelAnalyzer [10], Gelj [11], PyElph [12], GELect [2], GelClust [13] gibi birkaç program bulunmaktadır. Bu programların karşılaştırması Apichart Intarapanich ve arkadaşları tarafından yapılmıştır. Bu karşılaştırma sonucunda GELect performansının diğerlerinden daha iyi olmasına karşın şeritlerdeki bantların sayılarının az olduğu görüntülerde şerit tespit performansı oldukça zayıf kalmıştır [2]. Ancak bu programlar gömülü sistem için tasarlanmamışlardır. Windows ve MacOS işletim sistemleri ile çalışmaktadırlar.

## BÖLÜM 2

### GÖMÜLÜ SİSTEMLER

#### 2.1. GÖMÜLÜ SİSTEM TANIMI

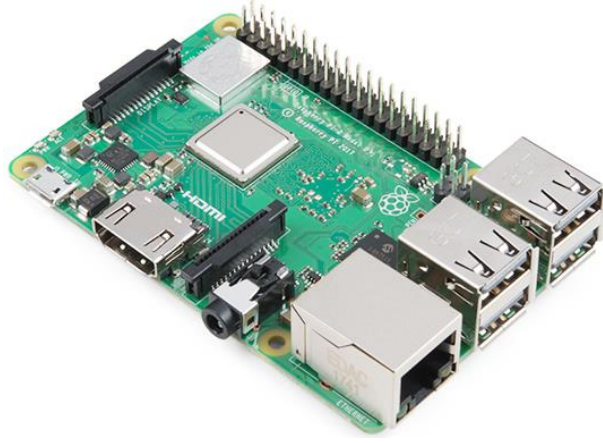
Gömülü sistem tıbbi cihaz, otomobil vb. gibi tüketici araçları için uygulamaya özel geliştirilen cihazlara gömülü bir elektronik alt sistemidir. Gömülü sistemler, CPU'larda veya özel donanım hızlandırıcılarında çalışan yazılımı kullanarak tüm sistem işlevselliğini çeşitli şekillerde içerebilir. Farklı gömülü sistemlerinin programlanması için farklı elektronik kartlar ve bilgisayar desteği gerekebilir. Böyle durumda bazı üreticilerin gömülü sistemlere yazılım yükleme işini dış sağlayıcılardan alınan hazır programlama cihazları ile yaptıkları görülmüştür [14].

##### 2.1.1. Gömülü Sistem Kartı

Raspberry Pi, dünyanın en çok satan üçüncü bilgisayar markasıdır. Raspberry Pi, TV'ye veya ekrana takılabilen kredi kartı boyutunda bir bilgisayardır. Kodlamayı öğrenmek ve elektronik projeler geliştirmek için, e-tablolar, sözcük işleme, internette gezinme gibi masaüstü bilgisayarının yaptığı birçok şey için kullanılabilir. Aynı zamanda projelerin kontrol birimi olarak kullanılabilir. Raspberry Pi, programlama ve dijitalleşmeyi öğrenmek için dünyanın her yerindeki yetişkinler ve çocuklar tarafından kullanılmaktadır [15].

##### 2.1.2. Raspberry Pi Özellikleri

Raspberry pi 3 B+'dan (Şekil 2.1.) önceki versiyonlarında görüntü işleme yapmak, düşük işlemci hızlarından ve küçük RAM miktarından dolayı çok uzun süreler alabilmekteydi. Bu modelin özellikleri görüntü işlemeye daha uygun olduğundan dolayı bu gömülü sistemi tercih edilmiştir [16].



Şekil 2.1. Raspberry Pi 3 B+ [15].

Diğer gömülü sistemlere karşılaştırmak gerekirse:

- Diğer basit gömülü sistemlerin aksine kendi içine gömülü bir işletim sistemi mevcuttur.
- Diğer birçok gömülü sistemi bilgisayara bağlayıp, bilgisayara yüklü IDE vasıtası ile içine program yüklenebiliyorken, Raspberry Pi'ya program yüklemek için sadece klavye, fare ve monitöre ihtiyacınız vardır.
- Bazı gömülü sistemlerden çok daha hızlı bir işlemciye ve işletim sistemine sahiptir.
- Donanımları basit gömülü sistemlerdeki gibi kendi üzerinde bulunan pinler vasıtasıyla kontrol etmek daha kolaydır.
- İşletim sistemi ile gelen Python derleyicileri mevcut olduğu gibi internette çok fazla açık kaynak desteği bulunmaktadır.

Çizelge 2.1'de Raspberry Pi Model karşılaştırma çizelgesi gösterilmektedir.

Çizelge 2.1. Raspberry Pi Model karşılaştırma çizelgesi [15].

Model	Hız (MHz)	RAM	USB Ports	Ethernet	Wi-Fi	Bluetooth
A+	700	512MB	1	No	No	No
B+	700	512MB	4	100Base-T	No	No
2 B	900	1GB	4	100Base-T	No	No
3 B	1200	1GB	4	100Base-T	802.11n	4.1
3 B+	1400	1GB	4	1000Base-T	802.11ac/n	4.2
Zero	1000	512MB	1	No	No	No
Zero W	1000	512MB	1	No	802.11n	4.1
Zero WH	1000	512MB	1	No	802.11n	4.1

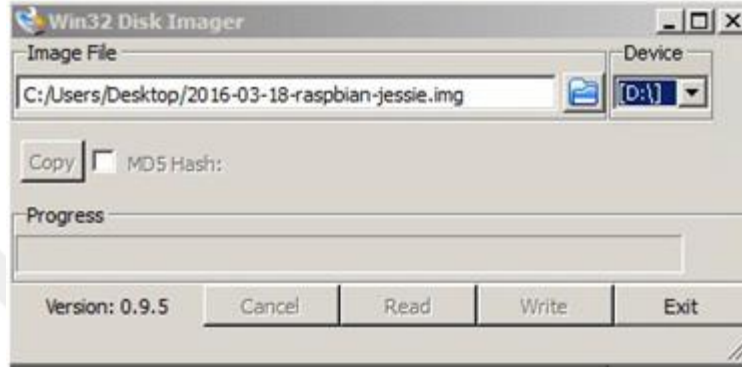
### Raspberry Pi 3 Model B+ Teknik Özellikleri [17]:

- Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz
- 1GB LPDDR2 SDRAM
- 2.4GHz ve 5GHz destekli IEEE 802.11.b/g/n/ac kablosuz ağ bağlantısı, Bluetooth 4.2, BLE destekli
- USB 2.0 üzerinden gigabit Ethernet (maks. hız 300 Mbps)
- Genişletilmiş 40-pin GPIO header bağlantısı
- HDMI çıkışı (tam boy)
- 4 adet USB 2.0 portu
- Kamera bağlantısı için CSI portu
- Ekran bağlantısı için DSI portu
- 4-uçlu (TRRS) 3.5mm stereo ses ve kompozit video çıkışı
- İşletim sistemi ve depolama için mikro SD kart yuvası
- 5V/2.5A DC güç girişi
- Power-over-Ethernet (PoE) desteği (ayrı olarak satılan PoE HAT gerektirir)

## 2.2. RASPBIAN KURULUMU

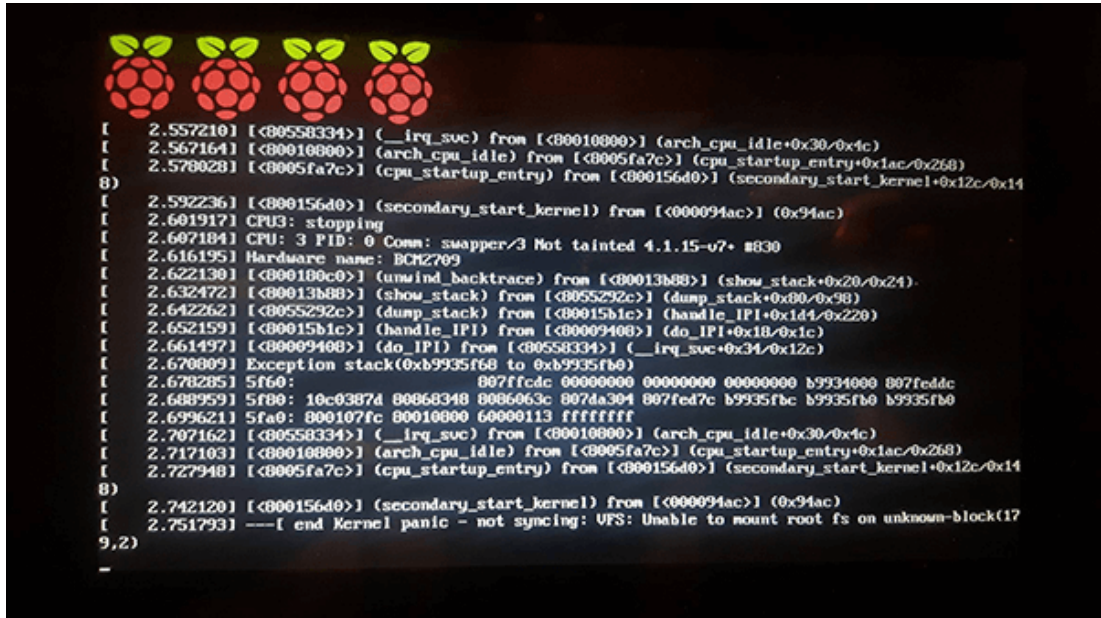
İşletim sisteminin kurulumu için minimum 16 GB kapasiteye sahip SD Kart'a ve Win32DiskImager programlarına ihtiyaç duyulmaktadır. Raspberry Pi (<https://www.raspberrypi.org/downloads/>) internet sitesinden Raspbian işletim sisteminin imajı indirilebilmektedir. İndirilen işletim sistemi imajı Win32diskImager

programı (Şekil 2.2) ile SD karta yüklenebilmektedir. Programda “Device” kısmından SD kartın sürücü harfi seçilip, siteden indirilen imaj, “Image File” bölümünden seçilir ve “Write” butonuna basılarak imaj SD karta yüklenebilmektedir. Yükleme başarılı bir şekilde tamamladıysa karşımıza “Write Successful” yazan mesaj kutusu gelmektedir.



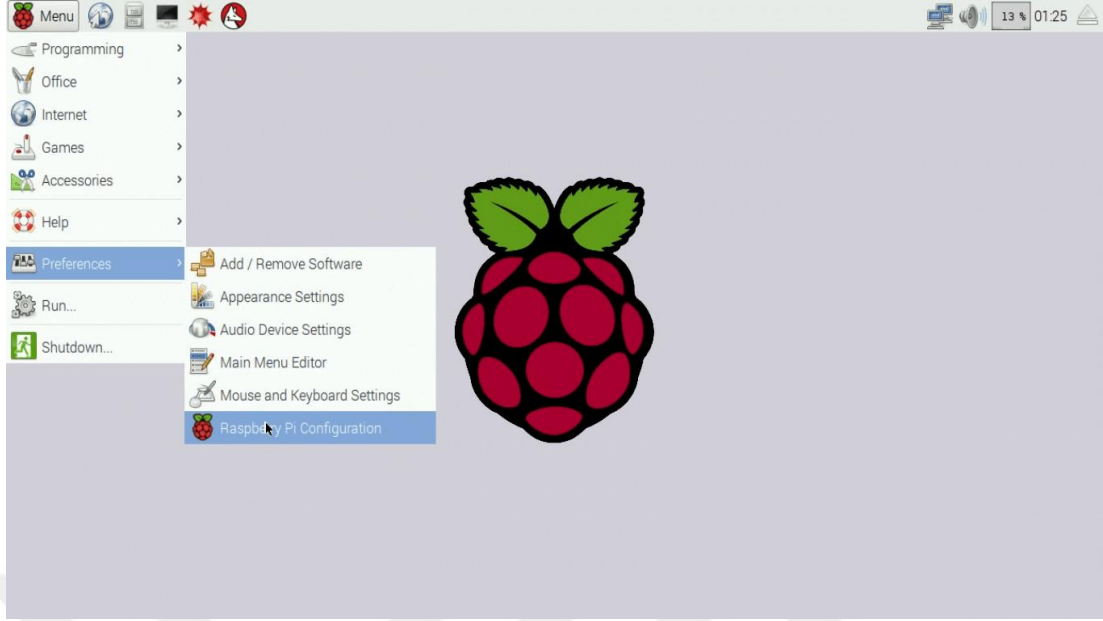
Şekil 2.2. Win32DiskImager programı.

Yükleme işlemi tamamlandıktan sonra SD kartı Raspberry Pi gömülü sistem kartına takılır ve sisteme enerji verildiğinde karşımıza Şekil 2.3’teki boot ekranı gelmektedir.



Şekil 2.3. Raspbian işletim sistemi açılış ekranı.

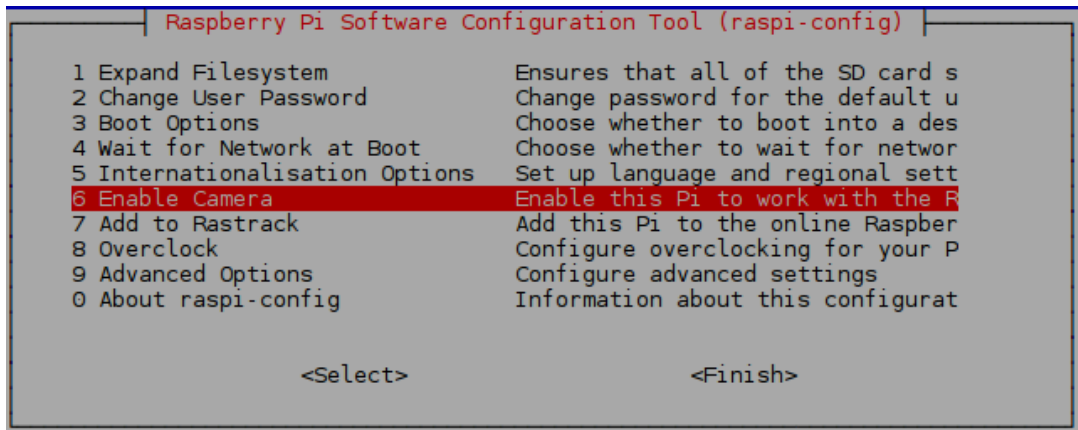
Şekil 2.4’te Raspbian işletim sisteminin masaüstü görünümü görünmektedir.



Şekil 2.4. Raspbian işletim sistemi.

## 2.2.1 Kamera Kurulumu

Kamera kurulumu için ilk olarak “raspi-config” komutuyla Raspberry konfigürasyon menüsünden (Şekil 2.5) kameraya izin verilmesi gerekmektedir.



Şekil 2.5. Raspi-config menüsü, kamerayı aktifleştirme seçeneği.

İzin işleminin tamamlanmasından sonra kamera sürücülerini kurmak için Şekil 2.6’deki komutlar terminale yazılıp çalıştırılmaktadır.

---

```
sudo apt-get -y install libv4l-dev v4l-utils
```

---

Şekil 2.6. Kamera sürücülerini yükleme komutu.

Yüklediğimiz v4l (video4linux) uygulamasını Şekil 2.7’deki komutla aktifleştirmek gerekmektedir.

---

```
sudo modprobe bcm2835-v4l2
```

---

Şekil 2.7. V4l uygulamasını aktifleştirme komutu.

Kamera sisteme bağlı ise Şekil 2.8’deki komutu terminale yazarak Şekil 2.9’daki kamera listesini görüntülemek mümkün olmaktadır.

---

```
v4l2-ctl -list-devices
```

---

Şekil 2.8. Sisteme bağlı kameraların listesini gösterme komutu.

---

```
mmal service 16.1 (platform:bcm2835-v4l2):  
/dev/video0
```

---

Şekil 2.9. Kamera bağlı ise terminalde görülen çıktı.

## BÖLÜM 3

### GÖRÜNTÜ İŞLEME ORTAMI

#### 3.1. GÖRÜNTÜ İŞLEME TANIMI

Görüntü İşleme (image processing), görüntüyü dijital form haline getirmek ve bazı işlemleri gerçekleştirmek için geliştirilmiş, özellikli görüntü elde etmek veya ondan bazı yararlı bilgiler çıkarmak için kullanılan bir yöntem olarak tanımlanmaktadır. Bu yöntemin girdisi video görüntüsü (anlık ya da kayıt) veya fotoğraf gibi bir görüntüden oluşmaktadır. Çıktısı ise görüntünün istenilen ya da dikkat edilmesi gereken özelliklerine ya da bir görüntüye karşılık gelmektedir. Genellikle görüntü işleme sistemi, önceden belirlenmiş sinyal işleme (Signal Processing) yöntemlerini uygularken görüntüleri iki boyutlu sinyaller olarak ele almaktadır [18].

Görüntü işleme temel olarak aşağıdaki üç adımı içerir [18]:

- Görüntünün optik tarayıcı ile veya dijital kameralarla alınması.
- Veri sıkıştırma, görüntü iyileştirme ve uydu fotoğrafları gibi insan gözü ile yorumlaması zor olan lekelenme kalıplarını içeren görüntüyü analiz etme.
- Çıktı, sonuçların görüntü analizine dayalı olarak bir özelliğe ya da görüntüye değiştirilmiş olarak kullanıma hazır hale getirme.

Görüntü işlemenin amacı 5 gruba ayrılmıştır. Bunlar [18]:

- Gørselleştirme – Görünmesi zor nesnelere gözlemlenme
- Görüntü keskinleştirme ve restorasyon – Gürültülü görüntüleri iyileştirme
- Görüntü alımı – İlgi çekici ve yüksek çözünürlüklü görüntü arama
- Desen Tanıma – Bir görüntüdeki çeşitli nesnelere tanımlama.
- Görüntü Tanıma – Bir görüntüdeki nesnelere ayırt etme.

Günümüzde işletmelerin çeşitli yönleriyle kullandıkları görüntü işleme sistemleri hızla büyüyen teknolojiler arasında yer almaktadır. Görüntü işleme, mühendislik ve bilgisayar bilimleri disiplinlerinde de temel araştırma alanını oluşturmaktadır [18].

Görüntü işleme alanı, insanların günlük yaşamlarını kolaylaştırma ve ham verileri nitelikli bilgi haline dönüştürme amacıyla üzerine sıklıkla çalışılan bir araştırma alanı olarak karşımıza çıkmaktadır. Özellikle son yıllarda, görüntü veri tabanlarının büyümesi ve yaygınlaşması ile bu verilerin sınıflandırılması üzerine geliştirilen çalışmaların sayısının da gittikçe arttığı görülmektedir. Günümüzde, görüntü sınıflandırma üzerine pek çok farklı yöntem kullanılmaktadır [19].

Gömülü sistem teknolojilerindeki gelişmeler sayesinde görüntü işleme uygulamaları günümüzde bir gömülü sistem içerisinde gerçek zamanlı olarak yapılabilmeye imkân tanımaktadır. Bu imkân sayesinde büyük bilgisayarlar yerine daha kompakt gömülü sistemler kullanarak, daha az enerji sarfiyatı ile görüntü işleme yöntemleri kotarılabilir. Burada piyasadaki görüntü işleme yazılımlarının paralı lisansları bulunması ve bir gömülü sistemde çalışmayacak gereksinimlere (yüksek performanslı bilgisayar ihtiyacı) sahip olduğu (sitelerindeki bilgilere bakılarak) bilinmektedir. Uygulamaların sadece yalın hallerini satın alarak görüntü işleme yapmak bile yeterince maliyetli olmaktadır. Yüksek gereksinimlere sahip olmaları ve paralı olmaları sebebiyle bu programlar gömülü sistemler için hem uygun değildir, hem de her hangi bir görüntü işleme uygulamasında gereksiz maliyet artışına neden olmaktadır. Bu tez kapsamında Python ve OpenCV kütüphanesinin birlikte kullanılmasının en önemli sebeplerinden biri çok düşük gereksinimlere sahip olmaları bir diğeri ise ücretsiz olmalarıdır. Python uygulaması ve OpenCV kütüphanesinin açık kaynak olmaları sebebiyle çok fazla örneklerinin bulunması da tercih nedeni olmaktadır.

### **3.3. PYTHON**

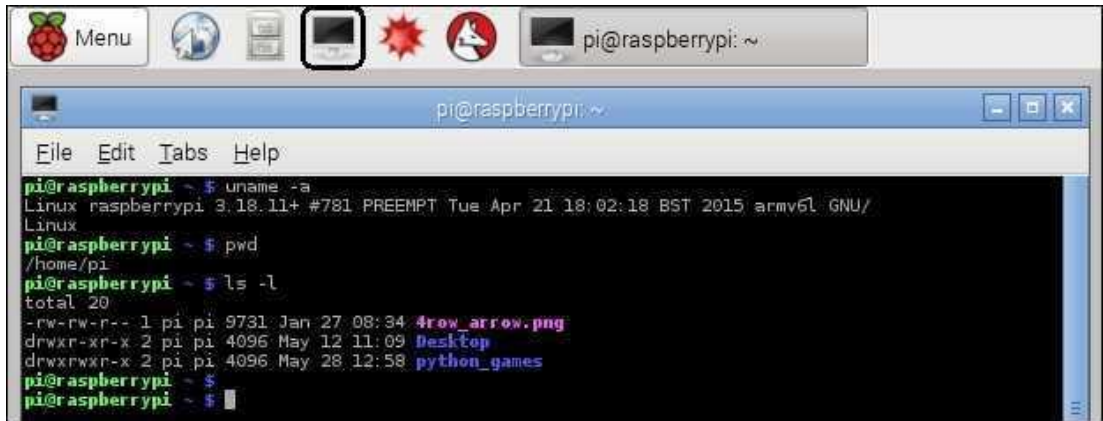
Python, Guido Van Rossum adlı Hollandalı bir programcı tarafından yazılmış bir programlama dilidir. Özellikle yazım kurallarının karışık olmaması, çabuk

öğrenebilme ve az kodla çok iş yapılabilmesine olanak sağladığı için, Python günümüzde çokça tercih edilmektedir [20].

Python ile yazılan bir yazılımı, Python kurulu olan herhangi bir işletim sisteminde kolayca çalıştırabilmek mümkündür. Diğer birçok programlama dillerinde yazılan programları başka işletim sistemlerinde çalıştırabilmek için programı o işletim sisteminde derlemek gerekmektedir. Bu da diğer programlama dillerine olan ihtiyacı en aza indirmektedir. Açık kaynak olmasından dolayı da herhangi bir ücret ödmeden bu programa ulaşmak mümkün olmaktadır. Python kodlarının okunma kolaylığı ve modüler sistemi birden çok yazılımcıyla geliştirilen ekip çalışmalarında büyük kolaylık sağlamaktadır [21].

### 3.3.1 Python Kurulumu

Python 3.5 sürümü Raspbian işletim sisteminde yüklü halde bulunmaktadır. Python 3.7 sürümünü yüklemek için ilk olarak Raspbian terminali (Şekil 3.1) açılır ve Şekil 3.2'deki kod çalıştırılıp işletim sisteminin güncellenmesi gerekmektedir.



Şekil 3.1. Raspbian işletim sistemi terminali.

---

*sudo apt-get update -y*

---

Şekil 3.2. Raspbian işletim sistemini güncelleme komutu.

Python resmi sitesinden (<https://www.python.org/downloads/source/>) Python son sürümü indirilebilmektedir. Şekil 3.3'teki komutlar kullanılarak Python programının son sürümü kurulabilmektedir. Bu komutları indirilen sürüme göre güncellemek gerekmektedir.

```
wget https://www.python.org/ftp/python/3.7.0/Python-3.7.0.tar.xz
tar xf Python-3.7.0.tar.xz
cd Python-3.7.0
./configure
make -j 4
sudo make altinstall
```

Şekil 3.3. Python kurulum kodları.

### 3.2. OPENCV

OpenCV (Açık Kaynak Bilgisayarlı Görme Kitaplığı) açık kaynaklı bir bilgisayarlı görü (Computer Vision) ve makine öğrenme (Machine Learning) yazılım kütüphanesidir. Görüntü işleme ile ilgili yüzlerce temel ve ileri seviye fonksiyonu, optimize edilmiş halleriyle barındırmaktadır. İlk olarak Intel'in Rusya'daki laboratuvarlarında, 1999 yılında geliştirilmeye başlanmıştır [22].

Bilindiği gibi açık kaynak kodlu projeler çok değişik lisanslara sahip olabilmektedir. Lisansın türüne göre kısıtlamalar içerebilmektedirler. OpenCV BSD lisansı altında dağıtılmaktadır. Bu lisans türünde ticari uygulamalardaki kullanımlara dahi hiçbir engel teşkil etmemektedir. Başlangıçta C ile kodlanmaya başlanmış olmasına rağmen, 2.0 versiyonundan itibaren C++ temelli bir yapıya kavuşmuştur. OpenCV 3.0 ile daha modern bir C++ yapısına geçmiştir. C/C++ dışında pek çok dil (Python, Java, Matlab/Octave, C#) ile beraber kullanılabilir [22].

OpenCV, programlama dillerinde olduğu gibi platform ve işletim sistemi konusunda da çok geniş bir yelpazede çalışma imkânı sağlamaktadır. Windows, Linux ve

MacOSx gibi yaygın kullanılan pek çok işletim sistemi üzerinde çalışabilmektedir. Günümüzün öne çıkan alanlarından olan mobil aygıtlar (Android, Blackberry, iPhone) ve embedded geliştirme kartları (Beaglebone, Raspberry Pi), yine OpenCV'nin kullanılabileceği platformlardır [22].

OpenCV kütüphanesi, hem klasik hem de modern bilgisayarlı görü ve makine öğrenme algoritmalarının kapsamlı bir setini içeren 2500'den fazla optimize edilmiş algoritmaya sahiptir. Bu algoritmalar, yüzleri algılamak ve tanımak, nesnelere tanımlamak, videolarda insan hareketlerini sınıflandırmak, kamera hareketlerini izlemek, hareketli nesnelere izlemek, nesnelere 3B modellerini çıkarmak, stereo kameralardan 3B nokta bulutları üretmek, bir görüntü veri tabanından benzer görüntüler bulma, flaş kullanılarak çekilen görüntülerden kırmızı gözleri kaldırma, göz hareketlerini izleme, vb. işlemler için kullanılabilir. OpenCV 47 binden fazla kullanıcı sayısına sahip bir topluluktur ve 18 milyondan fazla tahmini indirme sayısına sahiptir. OpenCV kütüphanesi şirketler, araştırma grupları ve devlet kurumları tarafından yaygın olarak kullanılmaktadır [23].

OpenCV kütüphanesini kullanan Google, Yahoo, Microsoft, Intel, IBM, Sony, Honda, Toyota gibi köklü firmaların yanı sıra, OpenCV'yi yaygın olarak kullanan Applied Minds, VideoSurf ve Zeitera gibi birçok yeni girişim bulunmaktadır. OpenCV'nin kullanımları, İsrail'deki gözetim videosunda izinsiz girişleri tespit etmeye, Çin'deki maden ekipmanlarını izlemeye, robotların Willow Garage'daki nesnelere taşımalarına ve toplamalarına, Avrupa'da yüzme havuzunda boğulma kazalarını tespit etmelerine, İngiltere ve İspanyada interaktif sanat çalışması yapmaya, Türkiye'deki enkaz pistlerini kontrol etmeye, dünyadaki fabrikalardaki ürün etiketlerini incelemekten Japonya'daki hızlı yüz tespiti konusuna kadar geniş bir alana uzanmaktadır [23].

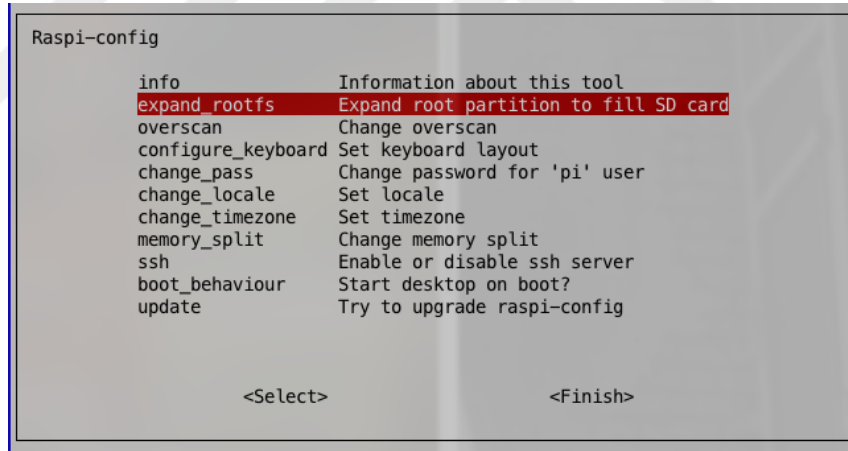
C ++, Python, Java ve MATLAB dilleri grafiksel arayüzlere sahiptir ve Windows, Linux, Android ve Mac OS desteklemektedirler. OpenCV çoğunlukla gerçek zamanlı görüş uygulamalarına dayanmaktadır. OpenCV, C++ dilinde yazılmıştır ve STL kaplarıyla sorunsuz çalışan bir arayüzü bulunmaktadır [23].

### 3.2.1. OpenCV Kurulumu

Python kurulumu tamamlandıktan sonra OpenCV kütüphanesini yüklemek gerekmektedir. Kurulumun yapılabilmesi için SD kartın tamamının kullanılabilmesi için genişletilmesi gerekmektedir. Bunun için terminale Raspi-config komutu yazıp Şekil 3.4'teki menüden "expand\_rootfs" seçeneğini seçip genişletme işlemi tamamlandıktan sonra sistemi yeniden başlatmak gerekmektedir.

Kurulum biraz karmaşık olduğundan dolayı birden fazla kurulum yapıldıysa eski sürümleri kaldırmak için ilk olarak Şekil 3.5'deki komutlar terminale girilerek eski ya da hatalı kurulum dosyalarından kurtulmak mümkün olmaktadır.

Eski sürümleri kaldırma işlemi bittikten sonra Şekil 3.6'daki komutlarla sistem güncellenmesi gerekmektedir. Eğer daha önce güncelleme yapılmadıysa bu işlem uzun sürmektedir.



Şekil 3.4. Raspberry konfigürasyon ekranı.

---

```
sudo apt-get remove libopencv*
```

```
sudo apt-get autoremove
```

---

Şekil 3.5. OpenCV'nin eski sürümlerini kaldırmak için kullanılan komutlar.

---

```
sudo apt-get update  
sudo apt-get upgrade  
sudo rpi-update  
sudo reboot
```

---

Şekil 3.6. Raspbian işletim sistemi güncelleme komutları.

Güncelleme işlemleri tamamlandıktan sonra reboot işlemini yapılmaktadır. Sonrasında geliştirici araçların (developer tools) kurulumunun yapılması gerekmektedir. Bu kurulum için gerekli komutlar Şekil 3.7’de verilmiştir.

---

```
sudo apt-get install build-essential cmake cmake-curses-gui pkg-config
```

---

Şekil 3.7. Geliştirici araçların kurulum kodları.

Geliştirici araçlarının kurulumu bitince OpenCV kullanırken ihtiyaç duyulacak kütüphanelerin kurulumu yapılması gerekmektedir. Bu kütüphanelerin kurulumu için gerekli komutlar Şekil 3.8’de verilmiştir.

---

```
sudo apt-get install |  
libjpeg-dev |  
libtiff5-dev |  
libjasper-dev |  
libpng12-dev |  
libavcodec-dev |  
libavformat-dev |  
libswscale-dev |  
libeigen3-dev |  
libxvidcore-dev |  
libx264-dev |  
libgtk2.0-dev
```

---

Şekil 3.8. Gerekli kütüphanelerin kurulmasını sağlayan komutlar.

Hazırlık işlemleri tamamlandıktan sonra artık OpenCV kurulumuna geçilebilmektedir. Bunun için /home/pi klasörü altına opencv adlı bir klasör açılması gerekmektedir. Bu klasörün içine girip Şekil 3.9'daki komutlar çalıştırılıp OpenCV kurulumu için gerekli dosyalar indirilmektedir.

---

```
cd /home/pi/opencv  
wget https://github.com/opencv/opencv/archive/3.2.0.zip -O opencv_source.zip  
wget https://github.com/opencv/opencv_contrib/archive/3.2.0.zip -O opencv_contrib.zip
```

---

Şekil 3.9. OpenCV dosyalarını klasöre kaydeden komutlar.

İndirilen (\*.zip) uzantılı sıkıştırılmış dosyaları açmak için Şekil 3.10'daki komutlar kullanılmaktadır.

---

```
unzip opencv_source.zip  
unzip opencv_contrib.zip
```

---

Şekil 3.10. Sıkıştırılmış dosyaları çıkaran komutlar.

OpenCV'yi çalıştırılabilir hale getirmek için derleme işlemlerinin yapılması gerekmektedir. Bu işlemler için opencv-3.2.0 klasörü içine girip build klasörü açıp bu klasörün içine derlenmektedir. Şekil 3.11'de build klasörü oluşturma komutları gösterilmektedir.

---

```
cd opencv-3.2.0  
mkdir build  
cd build
```

---

Şekil 3.11. Build klasörü oluşturma komutları.

Build klasörünün içine derlemeyi yapmadan önce Şekil 3.12'deki komutları çalıştırarak derleme için dosyaları hazır hale getirmek gerekmektedir.

---

```
cmake -D CMAKE_BUILD_TYPE=RELEASE |
-D CMAKE_INSTALL_PREFIX=/usr/local |
-D BUILD_WITH_DEBUG_INFO=OFF |
-D BUILD_DOCS=OFF |
-D BUILD_EXAMPLES=OFF |
-D BUILD_TESTS=OFF |
-D BUILD_opencv_ts=OFF |
-D BUILD_PERF_TESTS=OFF |
-D INSTALL_C_EXAMPLES=ON |
-D INSTALL_PYTHON_EXAMPLES=ON |
-D OPENCV_EXTRA_MODULES_PATH=../../opencv_contrib-3.2.0/modules |
-D ENABLE_NEON=ON |
-D WITH_LIBV4L=ON | ../
```

---

Şekil 3.12. OpenCV derleme için dosyaları hazır hale getirme komutları.

Bütün bu işlemler tamamlandıktan sonra kurulum işlemlerinin başarılı olup olmadığını anlamak için kurulum işleminin son satırlarında Şekil 3.13'teki mesajlar görülmektedir.

---

```
.....
-- Configuring done
-- Generating done
-- Build files have been written to: /home/pi/usbmem/opencv/opencv-3.2.0/build
```

---

Şekil 3.13. Derleme işleminin başarılı olarak tamamlandığını gösteren bildirim mesajları.

Derleme işlemini başlatmak için Şekil 3.14'teki komut çalıştırılmaktadır. Bu komutun sonundaki “-j4” parametresi derleme işleminin 4 çekirdekte yapılacağı anlamına gelmektedir. Böylelikle derleme işlemi daha kısa sürede olabilmektedir. İşlemcinin daha rahat çalışabilmesi için bu işlem sırasında işlemciyi soğutmak faydalı olmaktadır.

---

```
make -j4
```

---

Şekil 3.14. Derleme işlemini başlatma komutu.

Derleme işlemi tamamlandığında ekranda “[100% Built target ...]” ifadesi görülmektedir. Kurulumu tamamlamak için son olarak Şekil 3.15’teki komutlar çalıştırılmaktadır.

---

```
sudo make install  
sudo ldconfig
```

---

Şekil 3.15. OpenCV kurulumunu başlatma komutları.

Kurulum işlemi bittikten sonra Python 3 IDE’sini açarak Şekil 3.16’daki komutları yazılarak OpenCV kurulumunun başarılı bir şekilde gerçekleşip gerçekleşmediği anlaşılmaktadır.

---

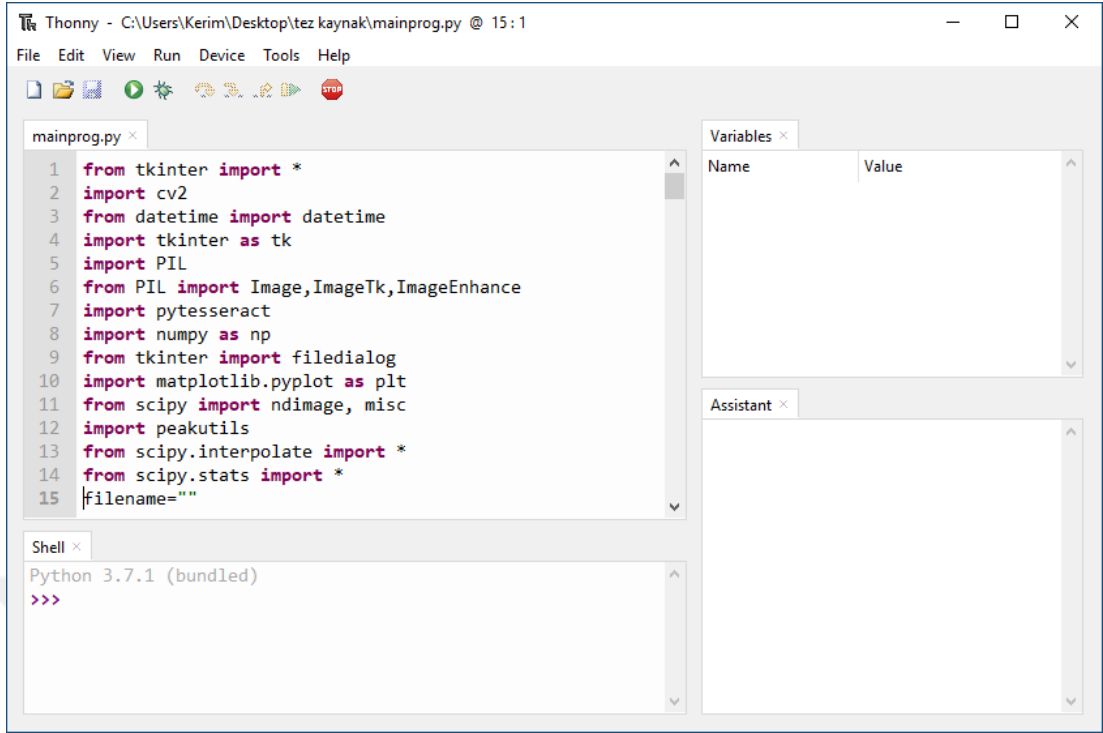
```
>>> import cv2  
>>> print (cv2.__version__)  
3.2.0
```

---

Şekil 3.16. Python 3 IDE OpenCV kütüphanesinin versiyonunu gösteren komutlar.

### 3.3. THONNY

Raspbian işletim sisteminde yüklü olarak gelen Thonny (Şekil 3.17) çok basit bir arayüze sahip ve RAM’i çok yormayan bir yapısı bulunan birçok Python IDE’den (Integrated Development Enviroment, Tümlleşik Geliştirme Ortamı) birisidir. Görüntü işleme uygulamaları üzerinde çalışılacağı için RAM’i çok kullanmayan bu geliştirme ortamı tercih edilmektedir.



Şekil 3.17. Thonny, Python IDE arayüzü

## BÖLÜM 4

### YÖNTEM VE TEKNİKLER

Jel elektroforez deneyleri, dijital görüntüler halinde gösterilebilir, bu nedenle önemli bilgileri çıkarmak için moleküler biyoloji uygulamalarında yaygın olarak kullanılır [24]. Şeritlerin ve bantların konumlarını bulmada kullanılan birçok yöntem mevcuttur. Hemen hemen her yöntem görüntünün iyileştirmesi ile elde edilen sinyalden bilgi çıkarma temeline dayanmaktadır. Görüntü iyileştirmesinde kullanılan birçok yöntem ve teknik mevcuttur. Uygulanan yöntemlerin en basitlerinden biri jel görüntülerindeki parlaklık değerlerini bir eşik değerine göre ayırıp arka alan ile bantlar arasındaki ton farkını açarak bantları tespit etmektir [25].

Bir diğer yöntem ise görüntüye bulanıklaştırma, kenar buldurma, adaptif eşikleme, histogram eşitleme, Gabor filtresi uygulama, morfolojik geri çatma gibi bir takım görüntü filtreleme ve görüntü iyileştirme işlemlerinin birinin ya da birkaçının uygulanmasından sonra bantların konumları belirginleştirilerek tespit edilmesidir [26,32].

#### 4.1. FİLTRELEME VE GÖRÜNTÜ İYİLEŞTİRME YÖNTEMLERİ

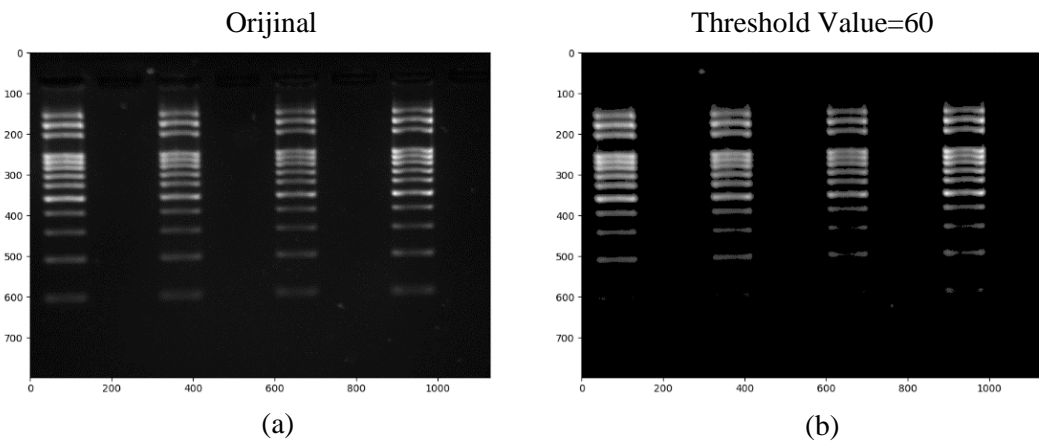
##### 4.1.1. Eşikleme (Thresholding)

Bilindiği üzere gri tonlamalı (grayscale) bir görüntü 0 ile 255 arası parlaklık değeri içeren piksellerden oluşmaktadır. 0 değeri en siyah tonu temsil ederken 255 değeri de en beyaz tonu temsil etmektedir. Jel resimleri siyah-beyaz kontrast içeren resimler olarak dönüştürüldüğünde belli bir değerden büyük olan değerleri beyaz, küçük değerleri ise siyah olarak tanımlayan bir eşik değeri (threshold value) belirlenerek bantları gürültülü arka fondan ayırmaya çalışarak bantları belirginleştirmeye yarayan bir yöntemdir [25].

Eşik değeri ile filtreleme ise 0-255 arası bir eşik değeri (threshold value) seçilir ve bu değerin altında kalan değerler “0” üstünde kalan değerlerde ya olduğu gibi bırakılır ya da “255” değeri verilerek sadece siyah beyaz bir görüntü elde edilebilmektedir. Örneğin bu değeri 60 olarak alırsak, bu değerin altında kalan bütün değerleri 0’a, üstünde olan bütün değerleri de olduğu gibi bırakarak, görüntüyü dönüştürürsek temelde sadece arka planda kalan gürültüsü tamamen silinmiş olan bir görüntü elde edilmektedir. Bu yöntem, bir eşik değeri altındaki gürültülü verilerden kurtulma için kullanılmaktadır [27].

Şekil 4.1 (a)’da mini jel elektroforez sistemiyle çekilmiş bir jel görüntüsünün orijinal hali ile Şekil 4.1 (b)’de 60 eşik değeri ile eşikleme yapılmış hali görülmektedir. Eşikleme yöntemi kontrastı çok geniş olan jel görüntülerinde etkili olarak bantları arka fondaki gürültüden ayırabilirken, bu yöntem kontrast aralığı dar olan jel görüntülerinde çok etkili olamamaktadır.

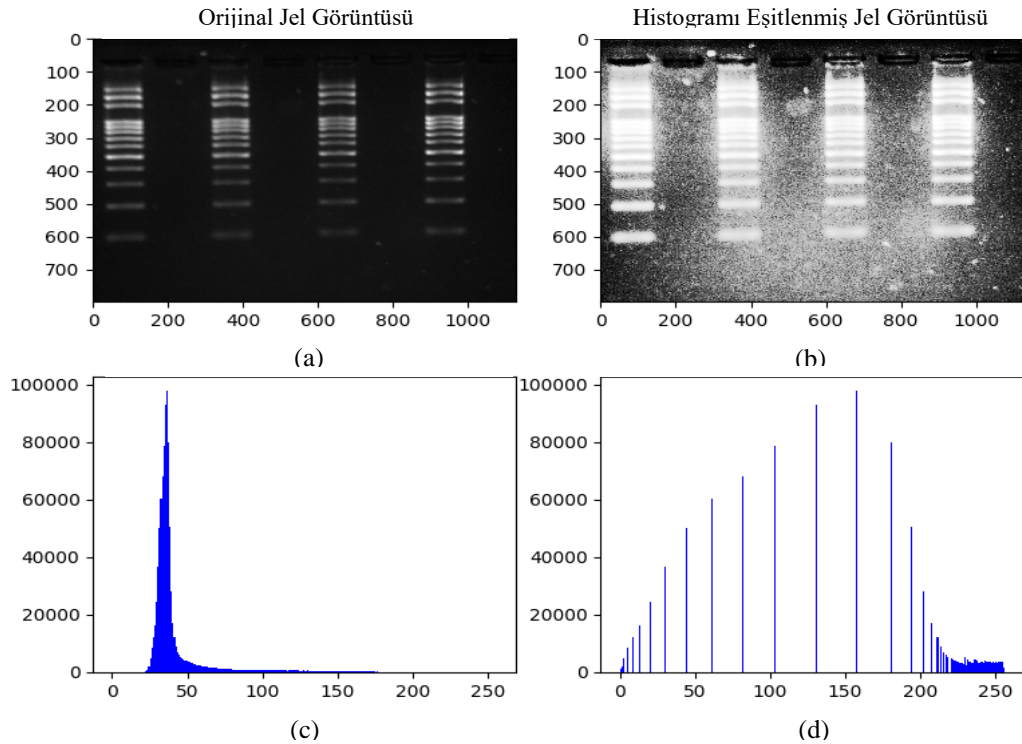
Bu yüzden dolayısıdır ki Şekil 4.1 (b)’deki gibi jel görüntüsündeki çok dağılmış veya yoğunluğu az olan bantların eşik değerinin altında kaldığından kaybolduğu gözlemlenmektedir. Bantların kaybolması oluşturulacak fonksiyonu ve dolayısıyla analiz sonuçlarını etkileyeceğinden eşikleme yöntemi her jel görüntüsü için iyi sonuçlar vermemektedir.



Şekil 4.1. a) Orijinal jel görüntüsü, b) Eşikleme uygulanmış jel görüntüsü.

#### 4.1.2. Histogram Eşitleme (Histogram Equalization)

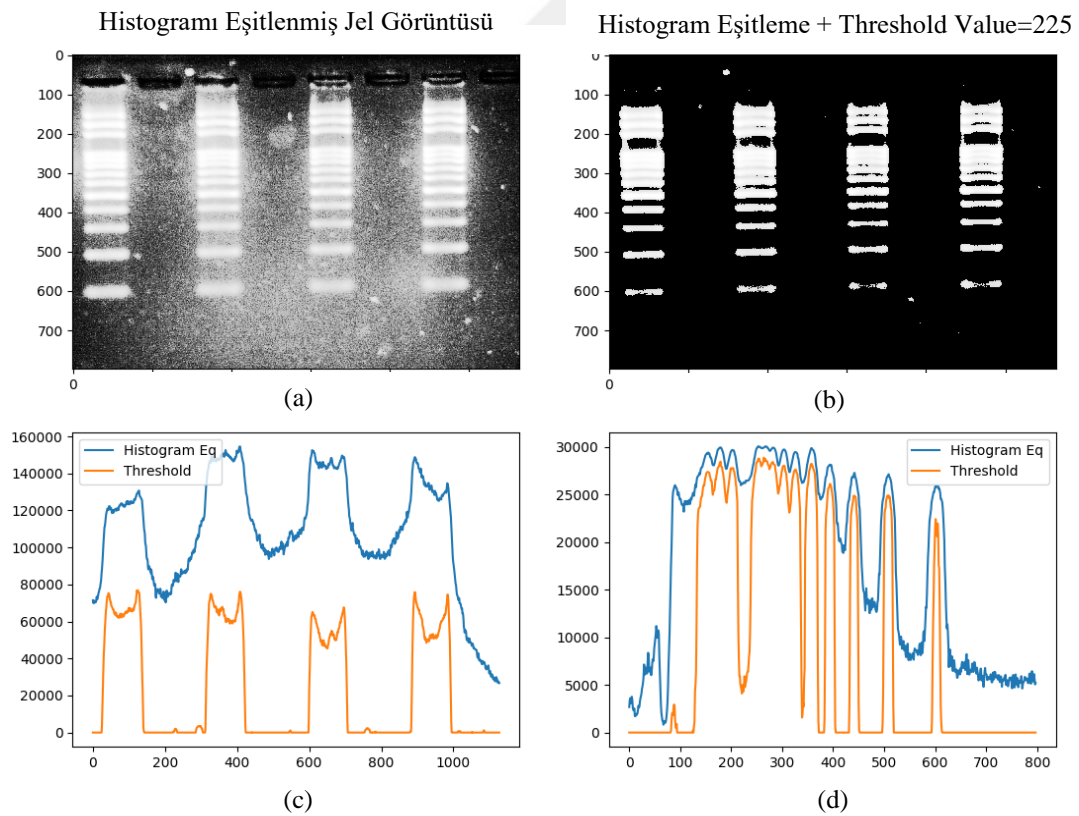
Histogram bir görüntüdeki piksel değerlerinin dağılımının grafiksel olarak gösterimidir. X eksenini piksel değerlerini gösterirken Y eksenini de bu piksel değerlerinin görüntüdeki frekansını göstermektedir. Jpeg gibi tamsayı tabanlı görüntülerde piksel değerleri 0-255 aralığında yayıldığı için X eksenini 256 farklı değer alacaktır [27]. Histogram eşitleme, gri tonlamalı resimlerde oluşan sis (fog) gürültülerini kaldırmada uygulanan yöntemlerden biridir. Bilindiği üzere histogram eşitleme resimde bulunan 256 farklı ton frekans dağılımlarını eşitlemeye çalışarak görüntü iyileştirmesi yapmaktadır [28]. Histogram eşitlemesi tüm görüntüye uygulanmaktadır. Histogramında yığılmaları olan ve içerisinde çok farklı ton bulunan resimlerde etkili bir görüntü iyileştirmesi yaptığı bilinmektedir. Ancak jel görüntülerinde geniş skalada bir renk ton dağılımı mevcut değildir. Bu yüzden kontrast aralığı dar olan resimlerde histogram eşitlemenin gürültüyü azaltmak yerine gürültüyü artırdığı Şekil 4.2'de görülmektedir.



Şekil 4.2. (a) Orijinal jel görüntüsü, (b) Histogram eşitlemesi uygulanan jel görüntüsü, (c) Orijinal jel görüntüsünün histogram grafiği, (d) Histogram eşitlemesi uygulanan jel görüntüsünün histogram grafiği.

Global Histogram eşitlemede Şekil 4.2 (b)'de görüldüğü üzere beyaz tonlar, beyazın yoğun olduğu bölgelerde yığılmalar oluştururken siyah tonlar, siyahların yoğun olduğu bölgelerde yığılmalara neden olabilmektedir. Bu durum ise her ne kadar bantların parlaklığını artırıp, eşikleme yöntemi uygulandığında şeritleri bulunması için piksellerin sütun toplamlarından elde edilen sinyali düzeltse de birbirine yakın olan bantların tepe noktalarının kaybolmasına neden olmaktadır.

Şekil 4.3'teki Histogram eşitlemesi ve eşikleme yöntemi uygulanmış jel görüntüleri işlemler sonucu dikey piksellerin toplamı ile elde edilen görüntü ve ilk şeritlerin yatay piksellerin toplanmasıyla elde edilen sinyallerinin karşılaştırılması gösterilmektedir. Grafiklerde ve görsellerde görüldüğü üzere bu filtre veri kayıplarına (bantların kenar ve tepe noktalarının kaybolmasına) sebebiyet vermektedir.



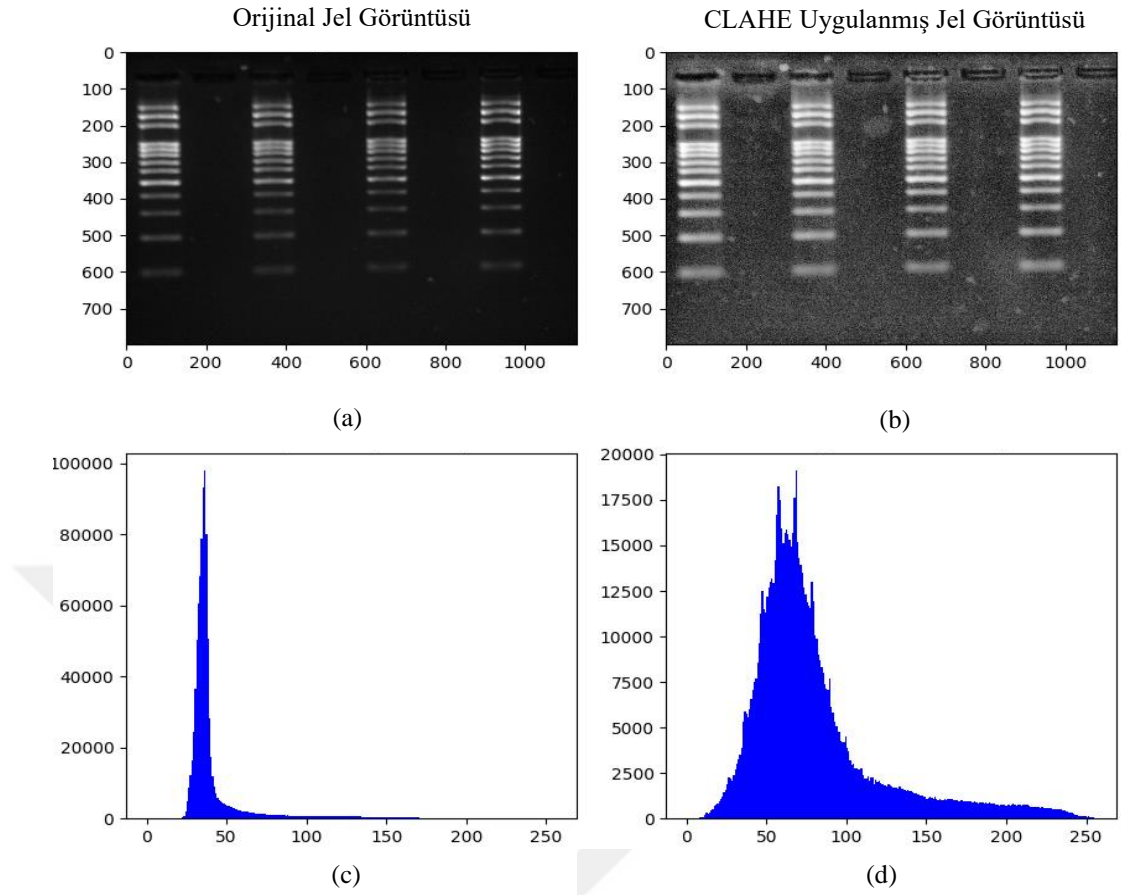
Şekil 4.3. a) HE uygulanmış jel görüntüsü, b) HE uygulanmış jel görüntüsüne eşikleme uygulanmış görüntü, c) İki görüntünün dikey piksellerinin toplamlarının karşılaştırması, d) İki görüntünün ilk şeritlerinin yatay piksellerin toplamının karşılaştırılması.

### 4.1.3. Kontrast Sınırlamalı Adaptif Histogram Eşitleme (Contrast Limited Adaptive Histogram Equalization)

CLAHE (kontrast sınırlamalı adaptif histogram eşitleme) ise görüntünün tamamına uygulamak yerine görüntüye bölgesel olarak uygulanmaktadır. Her bölgeyi kendi içinde eşitlemeye çalışarak, parlak bölgeler ile koyu bölgelerin jel görüntüsünün geneline olan etkisini azaltmak için bölgeye indirgeyerek parlak bölgedeki küçük ton farkları ile koyu bölgelerdeki küçük ton farkları, bulunduğu bölgeye göre dağıtılarak daha dengeli eşitlemeyi sağlamaktadır [27].

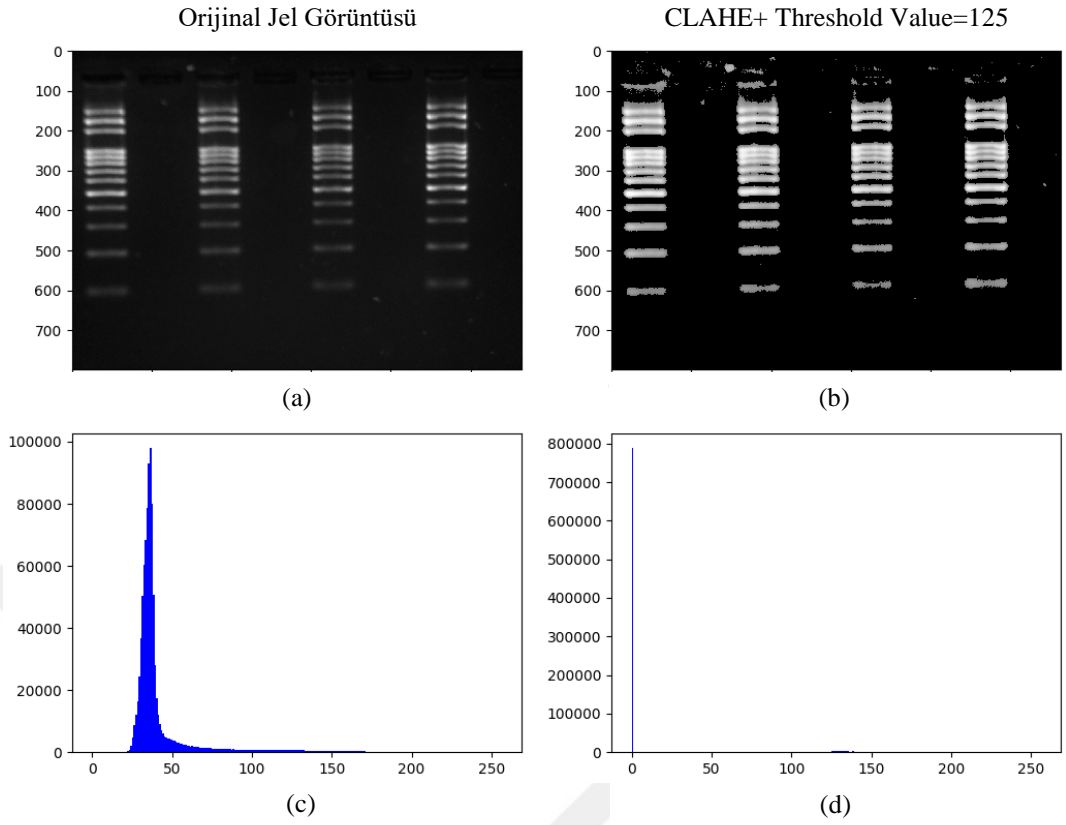
Adaptif histogram eşitlemesi görüntüyü kiremit adı verilen küçük bloklara bölmektedir. (tileSize, OpenCV’de varsayılan olarak 8x8’lik bir matristir, ihtiyaca göre matrisin boyutları değiştirilebilir). Daha sonra bu bloklara histogram eşitleme uygulanır. Bu sayede histogram gürültü olmadığı sürece küçük bir bölge ile sınırlı kalmaktadır. Eğer gürültü oluştuysa engellemek için kontrast sınırlaması uygulamak gerekmektedir (clipLimit, kontrast limitini ayarlama parametresi).

Şekil 4.4’de orijinal görüntü ile CLAHE uygulanmış bir jel görüntüsü verilmiştir. Görüntüde görüleceği üzere siyah alanlar çok fazla beyaz alanlara karışmamış, beyazın yoğun olduğu bölgelerde ise bantların arasındaki siyah bölgeler tamamen kaybolmamış global histogram eşitlemeye göre daha belirgin hatlara sahip bir dağılım oluşturulmuştur.



Şekil 4.4. a) Orijinal jel görüntüsü, b) CLAHE uygulanmış jel görüntüsü, c) Orijinal jel görüntüsü histogram grafiği, d) CLAHE uygulanmış jel görüntüsü histogram grafiği.

Elde edilen görüntünün histogramına bakarak eşik değeri 125 olarak alınmış bir eşikleme yöntemi uygulanırsa Şekil 4.5 b'deki görüntü elde edilmektedir. Görüntüden anlaşılacağı üzere bantlar ilk duruma göre daha belirgin gözükmemektedir. Ancak kuyuların etrafına bakıldığında hala bir miktar gürültü barındırdığı görülmektedir.



Şekil 4.5. a) Orijinal jel görüntüsü, b) CLAHE ve eşikleme uygulanmış jel görüntüsü, c) Orijinal görüntünün histogram grafiği, d) CLAHE ve eşikleme uygulanmış jel görüntüsünün histogram grafiği.

#### 4.1.4. Morfolojik Geri Çatma (Morphological Reconstruction)

Morfoloji (biçimbilim), en genel anlamıyla biyolojide hayvanlar ve bitkilerin biçim ve yapıları ile ilgili olan bir bilim dalıdır. Aynı kelime, matematiksel morfoloji bağlamında, görüntüdeki sınırlar, iskeletler ve dışbükey zarf gibi bölgesel şekillerin tanım ve temsilinde görüntü bileşenlerini elde edebilmek için kullanılacak teknik ya da teknikleri niteler. Söz konusu morfolojik teknikler, budama (ayıklama) inceltme ve morfolojik filtreleme gibi ön veya son işlemlerde kullanılabilir [28].

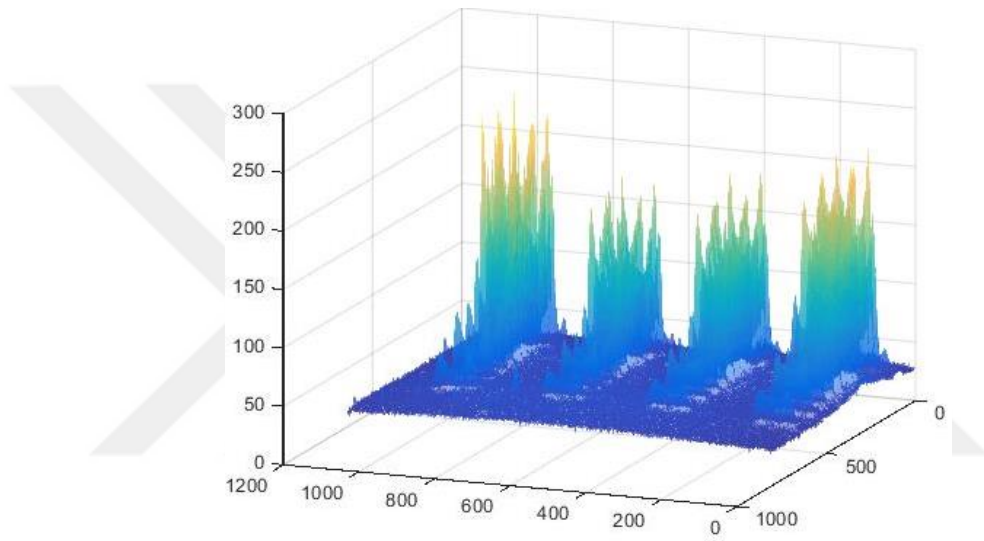
Birçok görüntü işleme ve analiz uygulamasında verilen bir  $f$  görüntüsü  $\Psi$  bir operatörü vasıtasıyla bir  $g$  görüntüsüne dönüştürülür ( $g = \Psi(f)$ ).  $\Psi$  görüntünün önemli özelliklerini korurken görüntüdeki gürültüyü silmek gibi özel bir görevi yerine getirmek için tasarlanmıştır [27].

Bu işlem, jel görüntüsünün farklı bölümlerindeki arka plan seviyelerini tahmin eder ve bunları orijinal görüntüden çıkarır. “Opening” adı verilen uzamsal bir filtreleme tekniği kullanılmıştır. Opening, genişletme (veya dilatasyon) ve aşındırma (veya erozyon) olmak üzere iki temel morfolojik operatörden oluşmaktadır. Genişletme/aşındırma işlemi jel görüntüsündeki her pikseli kendisinin ve en yakın sekiz komşusununki değerinin maksimum/minimum gri değeri ile değiştirmektedir. Morfolojik operatörlerin ayrıntılarına girmeden, genel olarak, bir resmi  $n$  kez genişletmek ve ardından genişletilmiş resmi  $n$  kez küçültmek, alanları  $(2n + 1)$  pikselden daha az olan nesnelere, yani, açılış hareketleri gibi etkili bir şekilde alçak geçirgen bir filtre gibi filtrelemektedir. Görüntüden kaldırılacak nesnelere boyutu, görüntüye uygulanan genişletme ve aşındırma işlemlerinin sayısı ile belirlenmektedir [30].

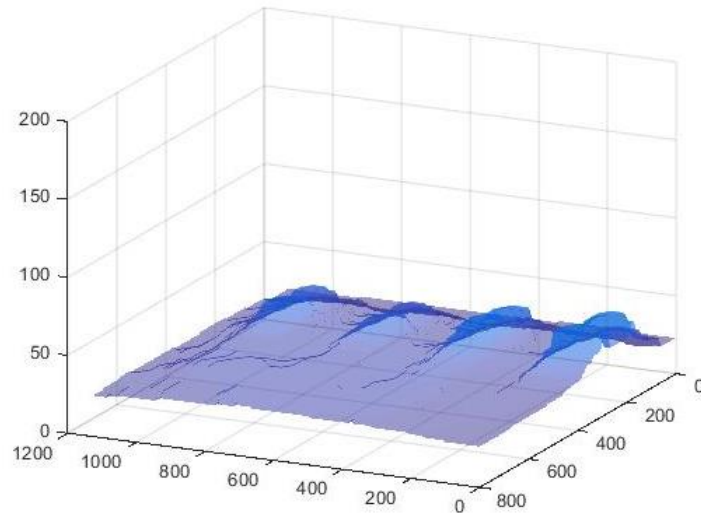
Objeler görüntüden çıkarıldığında, ortaya çıkan görüntüde geride bırakılanların arka plan düzeyi olduğu kabul edilmektedir. Jeldeki arka plan seviyeleri bölgeden bölgeye yumuşak bir şekilde değiştiğinden, mevcut uygulama jel görüntüsündeki koyu renkli nesnelere çıkaracak kadar büyük ayarlanmıştır.  $N$  değeri, görüntüdeki en büyük karanlık alanın boyutuna bağlı olarak (genellikle sayı veya dokunma veya örtüşen noktaların bulunduğu alana karşılık gelir) 20 ila 40 piksel arasında değişir.  $N$  değerinin olması gerekenden fazla uygulanması, az olmasından daha iyi olduğu için (az olması görüntüdeki orijinal nokta sinyallerinin ortadan kaldırılması veya azaltılmasıyla sonuçlanabilir), bu değer çoğu jel görüntü için genellikle yaklaşık 40 piksel olarak ayarlanmaktadır [30].

“Opening” yöntemi erozyon ve genişletme yöntemlerinin artarda tekrarlarıyla elde edilmektedir. Bir görüntüye önce erozyon işlemi uygulayıp ardından genişletme yöntemi uygulanırsa beyaz alanlar önce küçülür, küçük beyaz alanlar yok olur, sonra ise küçülen beyaz alanlar tekrar genişletilerek eski haline getirilir. Böylelikle küçük parazitlerden yok edilmektedir. Bu işlemi önce  $n$  defa erozyon uygulayıp sonra  $n$  defa genişletme uygulayarak görüntüdeki beyaz görüntünün tamamından kurtulup sadece arka alanın bulunduğu siyah alanın kalması sağlanmaktadır. Bu siyah alan görüntünün arkasındaki, sis ya da gürültülü alan olarak adlandırılmaktadır [31].

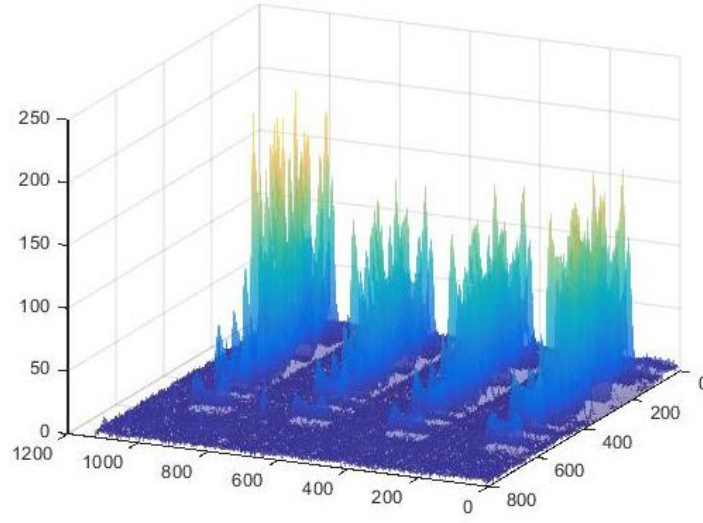
Elde ettiğimiz arka planı orijinal resimden çıkartarak görüntü gürültüden arındırılmış olmaktadır. Elde edilen verinin görüntünün ilk haline göre gerçek değerlerine daha yakın değerler içerdiği söylenebilir. Şekil 4.6’da bir jel görüntüsünün 3 boyuta dönüştürülmüş görünümü mevcuttur. Görüntünün fonuna baktığımızda bir ışık dalgasının var olduğunu görülmektedir. Bu yöntem uygulanarak Şekil 4.7’deki arka alan gürültüsü elde edilmektedir. Orijinal resimden fonu çıkardığımız zaman ise Şekil 4.8’deki görüntüde görüldüğü üzere zemini neredeyse tamamen düz görüntüye ulaşmak mümkün olmaktadır [31].



Şekil 4.6. Bir jel görüntüsünün 3D gösterimi.

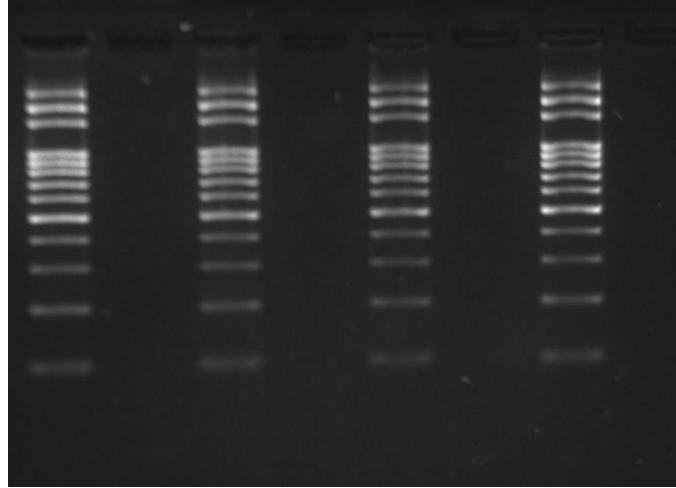


Şekil 4.7. Morfolojik geri çatma sonucu elde edilmiş arka planın 3D gösterimi.

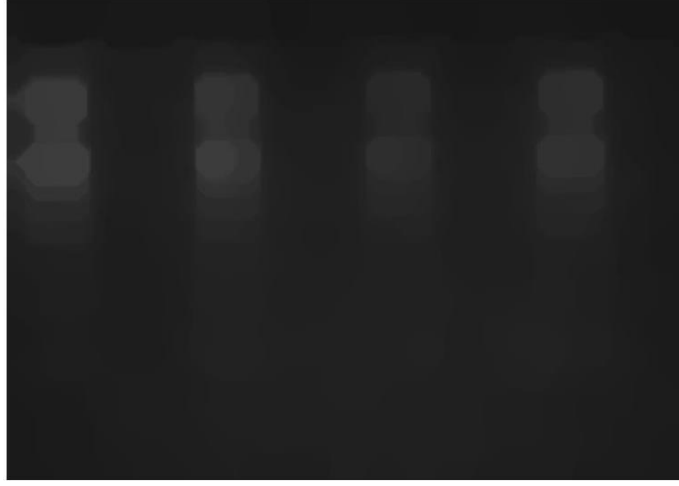


Şekil 4.8. Arka plan gürültüsü çıkarılmış görüntünün 3D gösterimi.

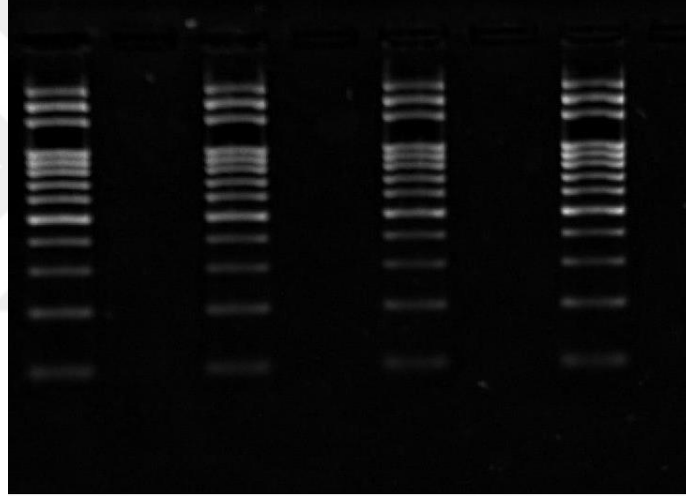
Şekil 4.9’da mini jel elektroforez ile çekilmiş görüntü, Şekil 4.10 bu görüntünün morfolojik geri çatma yöntemi ile elde edilen arka plan gürültüsü ve Şekil 4.11’de morfolojik geri çatma işlemi sonucu elde edilen ve orijinal görüntüden çıkarılmış, arka plan gürültüsünden arındırılmış görüntü görülmektedir.



Şekil 4.9. Mini jel elektroforez sistemi ile çekilmiş jel görüntüsü.



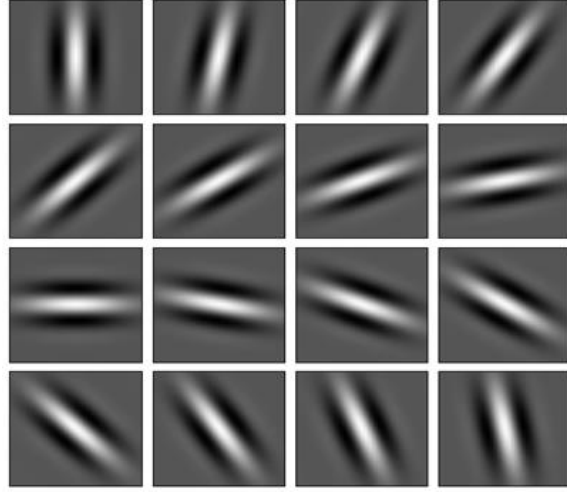
Şekil 4.10. Morfolojik geri çatma ile elde edilen arka plan gürültüsü.



Şekil 4.11. Arka plan gürültüsünden arındırılmış görüntü.

#### 4.1.5. Gabor Filtresi

Adını Dennis Gabor'dan almış bir doğrusal filtredir. Kenar bulma, doku analizi, özellik çıkarma gibi görüntü işleme uygulamalarında kullanılmaktadır [32,33]. Görüntüyü Şekil 4.12'deki filtreler ile tarayıp görüntü üzerindeki filtre açılımlarına karşılık gelen kenarları bularak görüntüdeki nesnelere belirginleştirir.



Şekil 4.12. Her biri  $11.25^\circ$  açı ile yönlendirilmiş 16 Gabor filtre bankası [34].

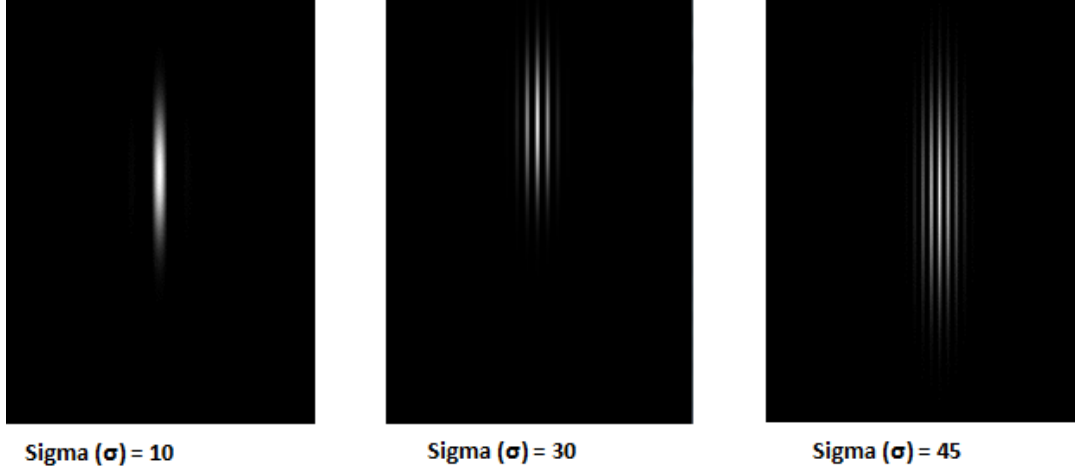
Gabor filtresinin çekirdeği “`cv2.getGaborKernel(ksize, sigma, theta, lambda, gamma, psi, ktype)`” kodundaki parametrelerle şekillenmektedir. Bu parametre değerlerinin Gabor çekirdeğine etkileri şöyle açıklanabilir:

**ksize:** Gabor çekirdeğinin büyüklüğüdür. Eğer  $ksize = (a, b)$  ise, o zaman Gabor çekirdeği boyutunda  $axb$  piksellere sahip olunmaktadır. Diğer birçok evrişim çekirdeğinde olduğu gibi, ksize tercihe bağlıdır ve çekirdek genellikle kare şeklinde olmaktadır.

**Sigma:** Gabor filtresinde kullanılan Gauss fonksiyonunun standart sapmasıdır. Şekil 4.13’de sigma değerlerinin değişmesiyle elde edilen Gabor filtreleri görülmektedir.

**Teta:** Gabor fonksiyonunun normalin paralel çizgisine açısal olarak yönlendirmesidir. Şekil 4.14’de teta değerinin değişmesi ile Gabor filtresinde oluşan açısal değişiklikler gösterilmektedir. Teta haricindeki parametrelerin sabit olması koşuluyla açı değeri arttıkça Gabor filtresi saat yönünde dönmektedir.

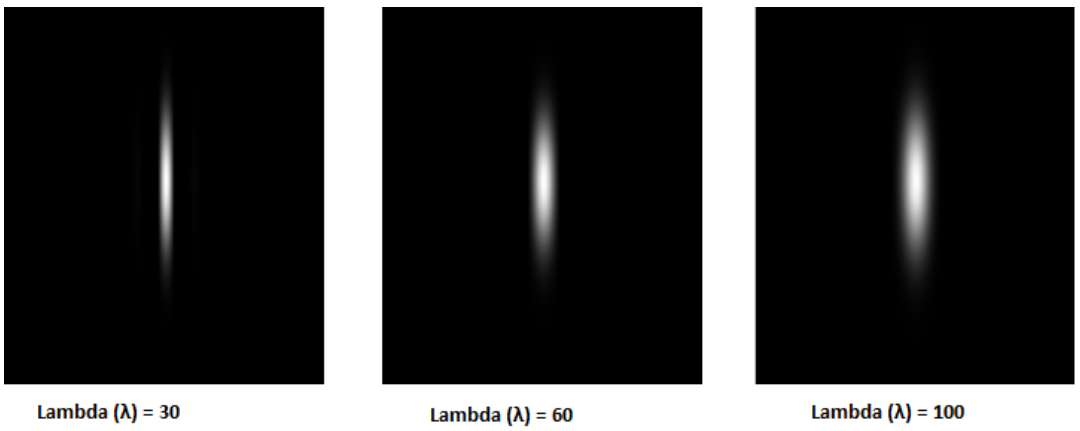
**Lambda:** Gabor filtresindeki sinüzoidal faktörün dalga boyudur. Şekil 4.15’te lambda parametresinin değerlerindeki değişimin (diğer parametreleri sabit tutmak kaydıyla) Gabor filtresine etkileri gösterilmektedir. Lambda değeri arttıkça Gabor filtresi kalınlaşmaktadır.



Şekil 4.13. Diğer parametreleri değiştirmeden ( $\lambda = 30, \theta = 0, \gamma = 0.25, \Psi = 0$ ) ve sigma değerinin Gabor çekirdeğine etkisi [34].

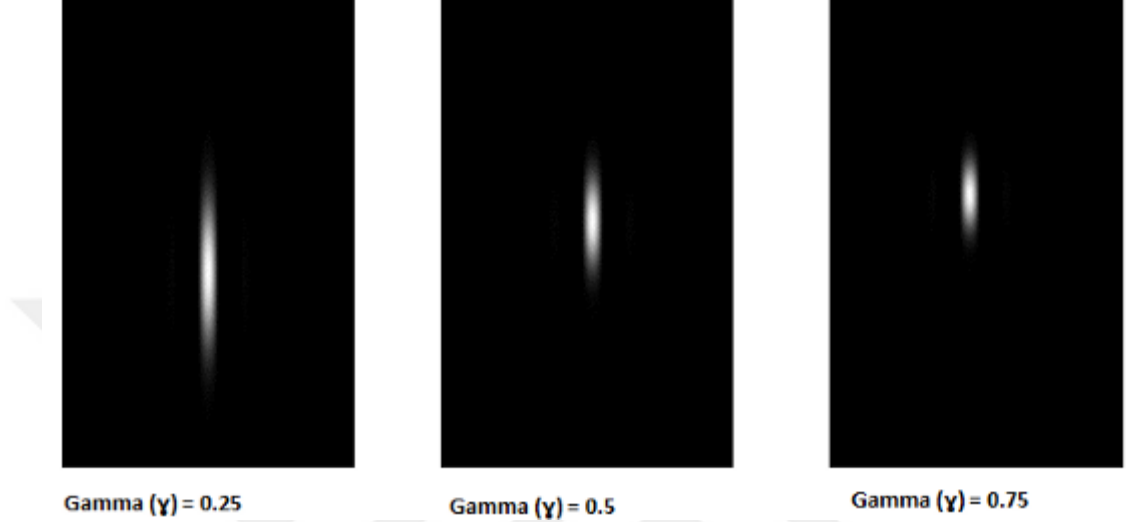


Şekil 4.14. Diğer parametrelerin değiştirmeden ( $\lambda = 30, \gamma = 0.25, \sigma = 10, \Psi = 0$ ) ve teta değerinin Gabor çekirdeğine etkisi [34].



Şekil 4.15. Diğer parametreleri değiştirmeden ( $\theta = 0, \gamma = 0.25, \sigma = 10, \Psi = 0$ ) ve lambda değerinin Gabor çekirdeğine etkisi [34].

**Gama:** uzaysal en boy oranıdır. Gama parametresinin değerlerindeki değişimlerin gabor filtresi üzerindeki etkileri Şekil 4.16’da gösterilmektedir. Diğer parametreler sabit tutulması kaydıyla gamma parametresinin değeri azaldıkça gabor filtresinin boyu uzamaktadır.

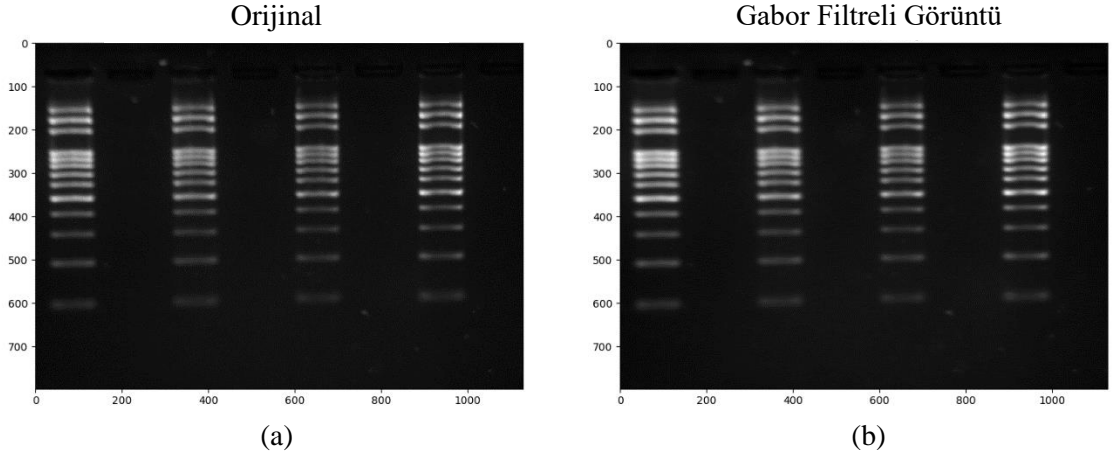


Şekil 4.16. Diğer parametreleri değiştirmeden ( $\lambda = 30, \theta = 0^\circ, \sigma = 10, \psi = 0$ ) ve gama değerinin Gabor çekirdeğine etkisi [34].

**Psi:** Faz kaymasıdır.

**Ktype:** Gabor çekirdeğindeki her pikselin alabileceği değerlerin türünü ve aralığını gösterir [32,34].

Şekil 4.17 (b)’de Gabor çekirdeğinde  $teta = 90^\circ$  alınarak jel görüntüsüne uygulanmıştır. Çünkü bantların görüntüsü yatay bir deseni andırmaktadır.



Şekil 4.17. a) Orijinal görüntü, b) Gabor filtresi uygulanmış görüntü.

#### 4.1.6. K-Means Kümeleme (K-means Clustering)

K-means en eski kümeleme algoritmalarından biridir. 1967 yılında J.B. MacQueen tarafından geliştirilmiştir. K-means kümeleme algoritması veri madenciliği dünyasında en çok kullanılan algoritmaların başında yer almaktadır. Kümeleme algoritmaları ile sınıflandırma algoritmaları arasında bir takım farklılıklar bulunmaktadır. Kümeleme algoritmaları otomatik olarak verileri daha küçük kümelere ya da alt kümelere ayırmaya yarayan algoritmalarlardır. Algoritma istatistiksel olarak benzer nitelikteki kayıtları aynı gruba sokmaktadır. Bir elemanın yalnızca bir kümeye ait olmasına izin verilir. Küme merkezi kümeyi temsil eden değer olarak tanımlanmaktadır [35,36].

Algoritmanın isminde yer alan “k” harfi, aslında küme sayısını belirtir: Algoritma, hata hesaplamada yaygın olarak kullanılan karesel hata fonksiyonunu en aza indirgeyecek “k” küme sayısını da aramaktadır. Verilen “n” sayıdaki veri seti “k” tane kümeye bu hata fonksiyonunu en aza indirgeyecek şekilde yerleştirilmektedir. Bu nedenle küme benzerliği kümedeki değerlerin ortalamaya yakınlıkları ile ölçülmektedir. Bu da kümenin ağırlık merkezi olmaktadır. Kümenin merkezinde yer alan değer kümenin temsilci değeridir ve medoid olarak adlandırılmaktadır [36].

Burada en önemli iki amaç şudur:

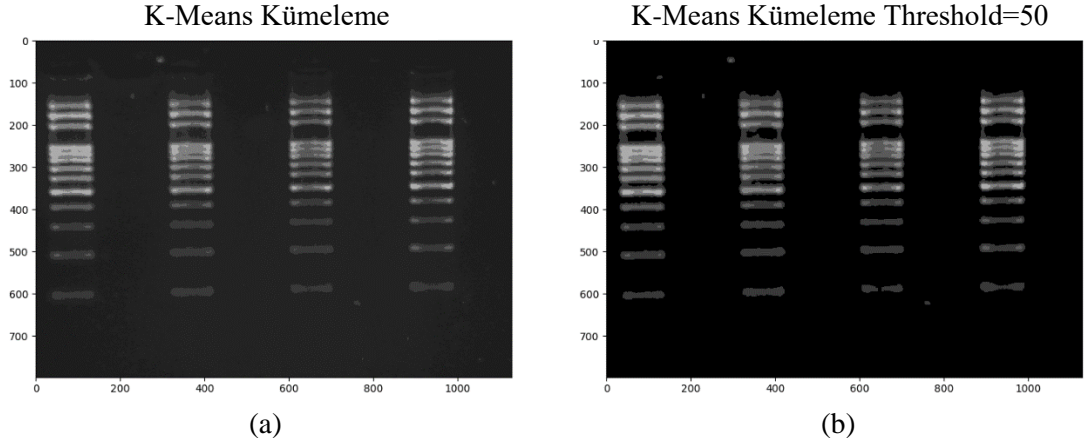
- Küme içindeki değerler birbirlerine en çok benzemeli,
- Kümeler birbirine mümkün olduğunca benzememeli

Bu istekleri gerçekleştirmek için algoritma tarafında sırasıyla şu adımlar gerçekleştirilmelidir:

- Sınıf merkezlerinin belirlenmesi
- Örneklerin mesafelere göre sınıflandırılması
- Yapılan sınıflandırma sonrasında yeni merkezlerin belirlenmesi
- İstenilen hale gelinceye kadar 2. ve 3. adımların algoritmik olarak tekrarlanması [35].

K-means'ın görüntü işlemedeki görevi ise bir görüntüdeki histogram dağılımlarını belli tonlar arasında sınırlandırmak böylelikle histogram tonları az bir ton grubuna indirgenerek istenilen verinin olduğu ton skalası ile istenmeyen verinin olduğu ton skalası birbirinden daha iyi ayrışmaktadır. Sonrasında istenen veriyi alıp, istenmeyen veriyi de görüntüden çıkarmak kolaylaşmaktadır.

Şekil 4.18 (a)'da K-means kümeleme yöntemiyle elde edilmiş görüntü ve (b)'de ise bu görüntüye eşikleme yöntemi uygulanmış görüntü gösterilmektedir. Görüldüğü üzere K-means kümeleme yöntemiyle bantlar arka plandan başarılı bir şekilde ayrıştırılmıştır. Bu görüntü ile şeritleri ve bantları bulmak daha kolay olmaktadır.



Şekil 4.18. a) K-Means kümeleme ile elde edilen görüntü, b) Eşikleme yöntemi uygulanmış K-Means kümeleme yapılmış görüntü.

## 4.2. SONUÇLAR VE TARTIŞMA

Şimdiye kadarki yöntemlerin uygulanmasındaki amaç bantları gözle ayırt edilebilecek kadar en belirgin hale getirmektir. Yaptığımız filtrelemeler ve yöntemler karşılaştırıldığında:

- Eşikleme Yöntemi: Histograma bakılarak belirlenen bir eşik değeri altında kalan verileri arka alan olarak belirleyip görüntüden çıkarma işleminin fazla parlak olmayan bantların kaybolmasına neden olabileceği görülmektedir.
- Histogram Eşitleme: Görüntüye histogram eşitleme yöntemi uygulandığında jel görüntüsündeki ağırlık siyah olduğundan dolayı görüntünün her yerine beyaz tonları dağıtmaktadır. Bu durum beyaz tona en yakın olan yerlerde aşırı parlamalara sebebiyet vermektedir. Bu sonuca eşikleme yöntemi uygulandığında birbirine yakın olan bantların arasındaki bölgelerdeki beyazların artması sonucu bantların birleştiği tepe noktalarının bozulduğu gözlemlenmektedir.
- CLAHE: Bu yöntem uygulanan görüntüde siyah beyaz dağılımının HE yöntemine göre daha dengeli olduğu görülmektedir. Her ne kadar dengeli dağılım gösterse de arka alandaki görüntüde parazitler oluşturmakta ve kuyucukların olduğu yerlerde parlamalar oluşturmaktadır. Eşikleme yöntemi uygulandığında kuyucukların olduğu yerde gürültüler oluşturmaktadır.

- Morfolojik Geriçatma: Arkaplan gürültüsünü silmede etkili yöntemlerden biri olduğu görülmektedir. Bu yöntemi kullanıp sonrasında başka yöntemler uygulanarak bantlar daha belirgin hale getirilebilmektedir.
- Gabor Filtresi: Desenleri daha belirgin hale getirebilen bu filtreleme yöntemi ile Gabor çekirdeği iyi ayarlanırsa etkili sonuçlar elde edilebilmektedir.
- K-Means: Jel görüntüsündeki tonlamalar arasında kümeleme yaparak 255 adet olan ton küme sayısını azaltarak arkaplan ile bantların arasındaki renk farklarını ayırmaktadır. Bu yönteme eşikleme uygulanan görüntüdeki bantların daha belirgin olup kuyucukların etrafında oluşan parazitlerin de kaybolduğu diğer yöntemlere göre parlaklığı az olan bantların gözle görülebilen daha belirgin hatlara sahip olduğu rahatça anlaşılabilir.

Etkili sonuçlar veren yöntemler bu tez çalışmasında geliştirilen arayüz tasarımına eklenmiştir. Bu şekilde, kullanıcılara jel görüntülerine bu yöntemleri uygulayarak bantların bulunmasında kolaylaştırma olanağı sağlanmıştır.

## BÖLÜM 5

### ELEKTROFOREZ GÖRÜNTÜLERİNİN ANALİZİ

#### 5.1. GÖRÜNTÜLERİN ÖLÇME PROGRAMI İLE ANALİZİ

Elektriksel alan etkisiyle göçme sonucu ağır baz çiftleri ile hafif baz çiftleri arasında mesafe açılmaktadır. Başlangıç noktasından ilerleyen baz çiftlerinden ağır olanları fazla yol alamazlarken hafif olan baz çiftleri daha uzak mesafeleri aynı süre içerisinde kat etmektedirler [5].

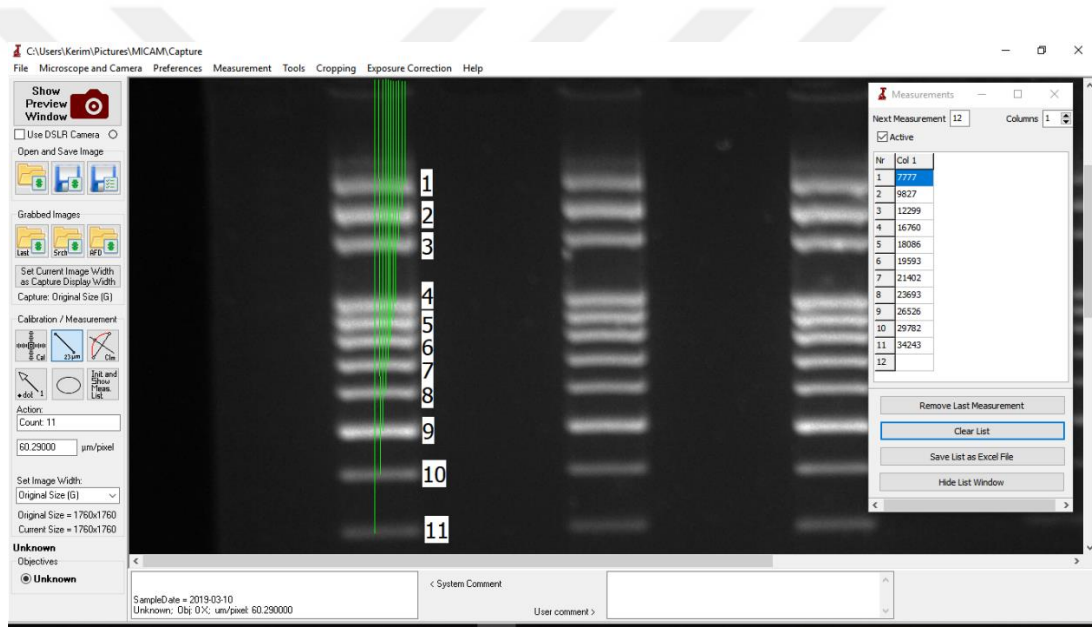
Ladder DNA'yı üreten firmadan alınan bilgi sayfasından DNA örneğinde hangi bandında kaç baz çifti bulunduğunu öğrenilip, baz çifti sayısı ile bantların yürüdüğü mesafe arasında bir fonksiyon oluşturulmaktadır. Oluşturulan bu fonksiyon sayesinde diğer DNA örneklerindeki hareket eden baz çiftlerinin kat ettikleri mesafeler bu fonksiyon ile hesaplanarak barındırdıkları baz çift sayıları tahmin edilmektedir.

Bu işlemin doğruluğu ölçülen baz çift mesafelerinin gerçeğe yakınlığı ve oluşturulan formülün belirtme katsayısı ( $R^2$ ) değeriyle açıklanabilmektedir. Belirtme katsayısı olarak bilinen  $R^2$  değeri, değişkenleri bağımlı-bağımsız değişken olarak düşünüldüğünde bağımlı değişkendeki toplam değişimin yüzde kaçının bağımsız değişken tarafından açıklanabildiğini belirtir.  $R^2$  değeri 0 ile 1 arasında değişmektedir.  $R^2$  değerinin 1'e yaklaşması, bağımlı değişkendeki değişimin büyük bir bölümünün bağımsız değişken tarafından açıklandığını göstermektedir [37]. Bu değer 1'e ne kadar yakın ise formül gerçek değerlere o kadar yakın değerleri tahmin ettiği söylenebilmektedir.

Bunun için ise formülü hesaplamadan önce yapılan ölçümlerin hassasiyeti ve doğruluğu çok önemlidir. Bir görüntü üzerinde cetvel ile yapılan ölçüm ile elde edilen

değer milimetre hassasiyetinde olabilirken, bir program ile yapılan ölçümlerde mikrometre ve nanometre seviyelerine kadar inilebilmektedir.

Şekil 5.1'de MiCam ald ölçüm programı ile bir agaroz jel ölçümü gösterilmiştir. Klasik yöntemde cetvel ile yapılan ölçümler milimetre hassasiyetinde olmaktadır. Bu program ile mikrometre seviyesinde bantların yürüdüğü mesafelerin ölçümü yapılmıştır. Ölçüm hassasiyetinden dolayı bu program, cetvel ile yapılan ölçme işleminden daha doğru sonuçlar elde ettiği ve daha az uğraştırıcı olduğu söylenebilmektedir. Bu tez kapsamındaki amaç iş yükünü azaltmak olduğu için cetvel ile ölçme işleminin kıyaslanmasına gerek duyulmamıştır.



Şekil 5.1. MiCam 2.4 ölçüm programı [38].

MiCam programıyla ölçme, cetvelle ölçme işleminden daha hassas olmasına rağmen yine de bazı hatalarla karşılaşılabilir. Şekil 5.2'de ölçüm sırasında yapılabilecek bazı hatalar gösterilmektedir. Bu hatalar, ölçüm yapanın gözüyle seçilemeyecek boyutlardaki görüntünün olması, fareyi iyi kontrol edememesi, görüntülerdeki bulanıklık, bandın tepe noktasının tam olarak görülebilmesi, tepe noktası tam olarak görülebilse bile; fare türü, resmin piksel boyutları vb. birçok etkenden dolayı tepe noktasının seçilememesi gibi farklılıklardan kaynaklanabilmektedir.

Şekil 5.2’de referans çizgilerinin başlangıç noktalarının aynı seviyede alınmadığı ve bitiş noktalarının da tam olarak her bant için bandın aynı noktasından alınmadığı görülmektedir. Ölçüm hatalarından diğer biri de yine Şekil 5.2’de görüleceği üzere fare kullanarak doğrusal ölçme yapmanın da o kadar kolay olmamasıdır. Tüm bu hatalar toplanınca yapılan ölçmelerin gerçek değerlerinden gittikçe uzaklaştığı çıkarımı yapılabilmektedir.



Şekil 5.2. Ölçüm esnasında yapılan hatalar.

Şekil 5.3’de üretici firmadan alınan Ladder DNA bilgileri gösterilmektedir. Bu bilgiler ile MiCam programı ile yapılan ölçmeler, Çizelge 5.1’de gösterilmektedir. Yapılan ölçümlere göre 3000 baz çifti bulunan 1. bant uygulanan gerilim neticesinde agaroz jel içerisinde 7777 mikrometre, 100 baz çifti bulunan 13. bant ise 48639 mikrometre mesafe yürütülmüştür.



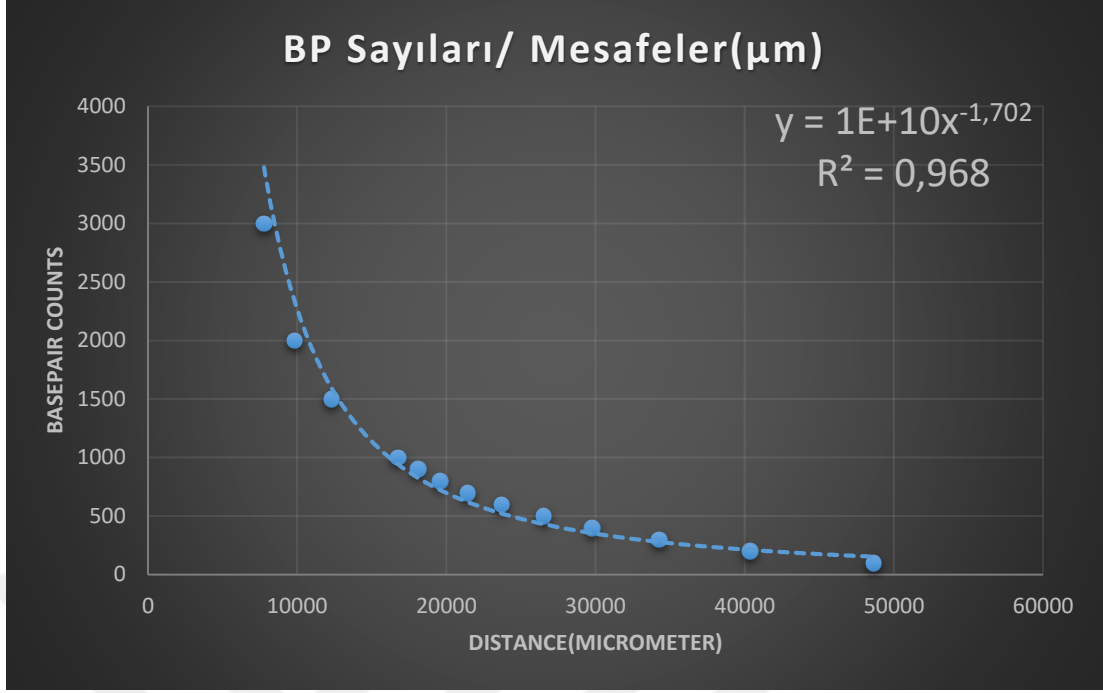
Şekil 5.3. Yürütülmüş Ladder DNA örneği [39].

Çizelge 5.1. Ölçüm işlemi sonucunda baz çiftlerinin aldığı mesafeler (µm)

Band No	Mesafe (µm)	Baz Çift Sayıları
1	7777	3000
2	9827	2000
3	12299	1500
4	16760	1000
5	18086	900
6	19593	800
7	21402	700
8	23693	600
9	26526	500
10	29782	400
11	34243	300
12	40331	200
13	48639	100

Şekil 5.4’de yürünen yola bağlı baz çifti miktarını excel programında grafik kısmında eğri uydurma yöntemi ile oluşturulmuş 0,968 R<sup>2</sup> değerine sahip üs fonksiyon Eşitlik 5.1 gösterilmiştir.

$$y = 10 \cdot 10^9 \cdot x^{-1,702} \quad (5.1)$$



Şekil 5 4. Baz çifti- mesafe grafiği ve uydurulan eğrinin R<sup>2</sup> değeri.

Çizelge 5.2’de görüldüğü üzere bu fonksiyona ladder DNA’daki baz çiftlerinin yürüdüğü mesafeler girilip hesaplandığında gerçeğe en yakın bulunan değer 100 baz çiftinin bulunduğu bant olan 13. bandın 105 baz çifti bulundurduğunu gözlemlenmiştir. Diğer bantların mesafeleri fonksiyona girildiğinde hesaplanan değerler gerçek değerlerine, 13. Bandın gerçek değerine yaklaştığından daha fazla yaklaşmamıştır. Baz çiftlerinin konumları formüle uygulandığında tahmin edilen değerler ile gerçek değerleri arasındaki farklar Çizelge 5.2 gösterilmiştir.

Çizelge 5.2’ye bakıldığında gerçek değerler ile hesaplanan değerler arasındaki farklar %5-%40 arasında değişken değerler aldığı görülmektedir. Ayrıca uydurulan bu fonksiyon ile yürüdüğü mesafeye göre hesaplanan baz çifti sayısının bant başına ortalama %30 hata ile tahmin edilebileceği söylenebilmektedir.

Çizelge 5.2. Fonksiyon ile tahmin edilen değerlerin, gerçek değerler ile karşılaştırılması.

Gerçek BP	Tahmin Edilen BP	Fark	Hata %
3000	2537	463	15,43
2000	1603	397	19,85
1500	1094	406	27,07
1000	646	354	35,4
900	568	332	36,89
800	495	305	38,13
700	426	274	39,14
600	358	242	40,33
500	296	204	40,8
400	243	157	39,25
300	191	109	36,33
200	145	55	27,5
100	105	-5	5
Ortalama Hata %=			<b>30,9</b>

## 5.2. GÖRÜNTÜLERİN YAZILIM İLE ANALİZİ

### 5.2.1. Şerit ve Bantların Bulunması

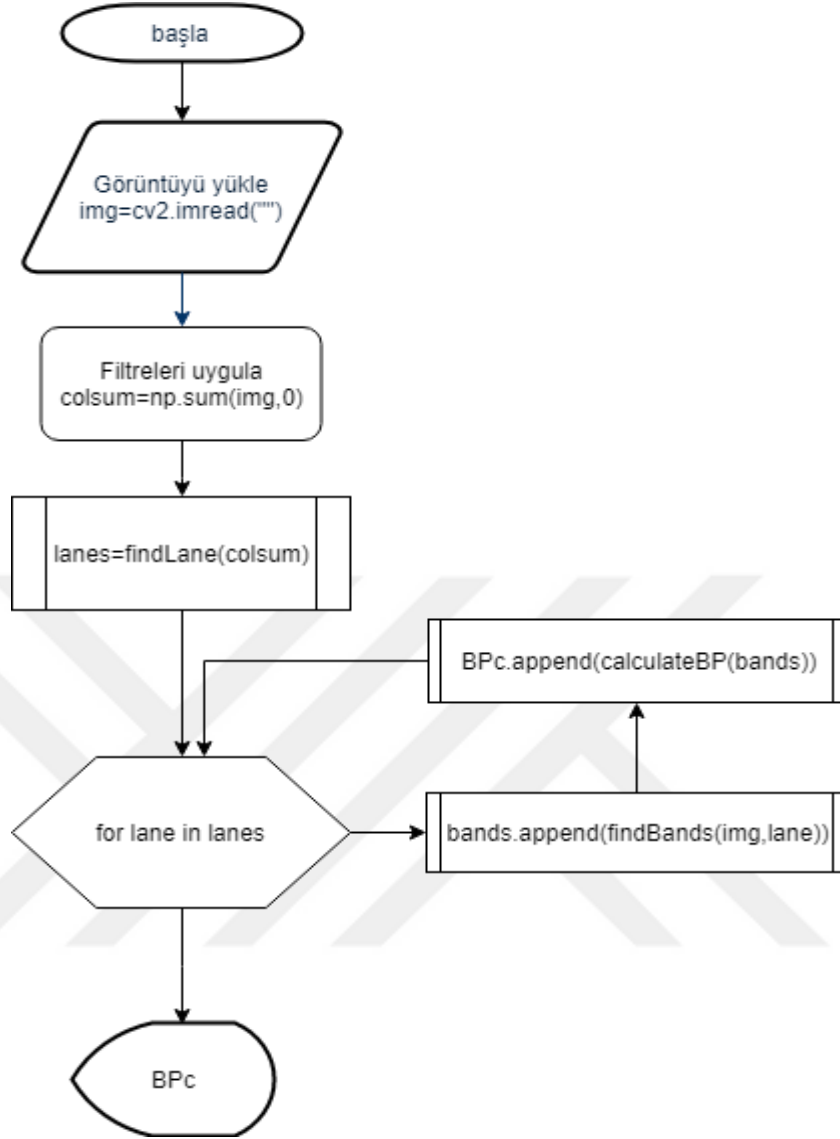
Görüntünün arka alanında ışıktaki patlamalardan ve yansımalarından kaynaklı parazitler ve dalgalanmalar olduğu için ilk önce arka plan gürültüleri, morfolojik erozyon ve morfolojik genişletme komutları ile 10 defa erozyon, 10 defa genişletme yaparak elde edilmektedir. Lokal histogram eşitleme yaparak çok belirgin olmayan bantlar belirgin hale getirilip bir eşik değerin altındaki değerler sıfırlanarak arka alandaki siyah zemin temizlenmektedir. Yapılan filtreleme işlemlerinden sonra görüntü üzerindeki bantların tespiti için yazılım ile görüntüyü sinyale çevirip analiz etmek daha kolay hale gelmektedir.

Görüntüde bantların konumlarını tespit edebilmek için ilk önce şeritlerin yerlerini tespit etmek gerekmektedir. Ardından görüntü şeritlere ayrılıp her şeride ait bantların konumları teker teker bulunup yürüdükleri piksel mesafeleri ile bir fonksiyon oluşturulmaktadır. Bu fonksiyon sayesinde DNA örneklerinin yürüdükleri mesafelere göre en yakın baz çifti sayıları tespit edilmektedir.

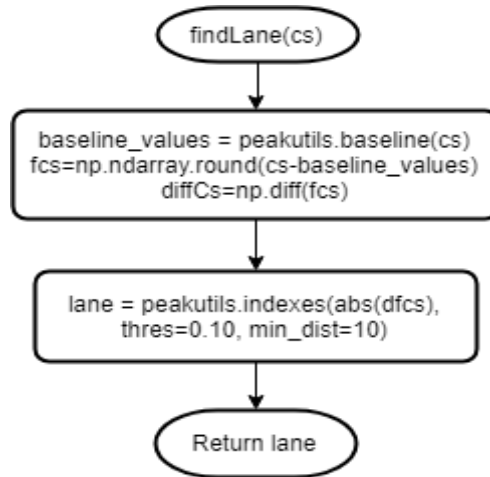
Şekil 5.5’deki akış diyagramında görüldüğü üzere önce pikseller dikey olarak toplanır. Ardından Şekil 5.6’teki akış diyagramı gösterilen findLane() fonksiyonu ile sinyalin baseline’ı düzeltilir ve şeritlerin kenarları tespit edilir.

Şekil 5.7’de gösterilen findBands() fonksiyonu ile görüntüde şeritlerin bulunduğu bölgelerdeki pikseller yatay olarak toplanarak elde edilen sinyalden bantların bulunduğu tepe noktalarının indisleri kat edilen piksel mesafesi olarak “bands” dizisine aktarılır.

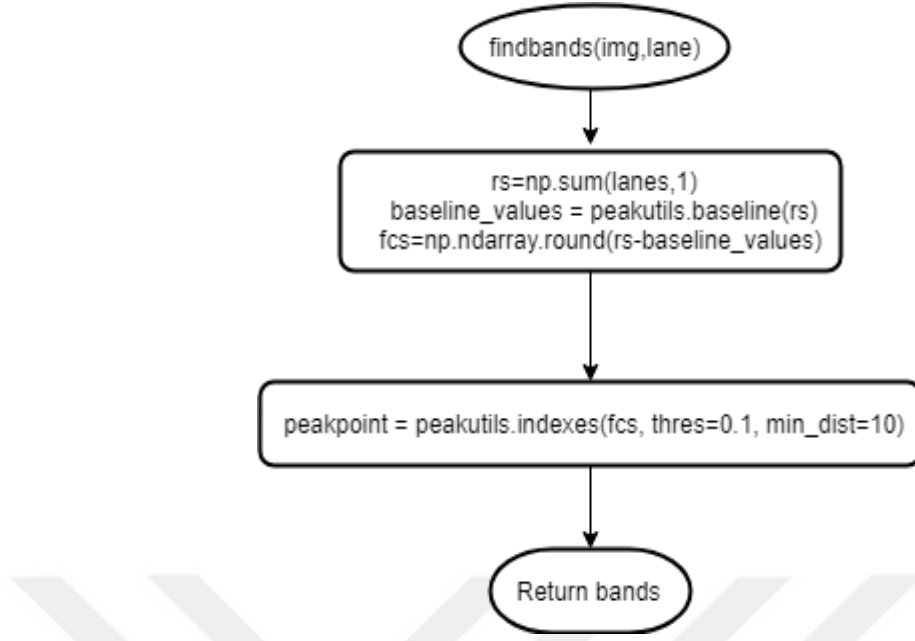
Bu dizideki değerler ile Ladder DNA’daki baz çiftleri arasında bir fonksiyon oluşturmak için calculateBP() fonksiyonu çağırılır. Bu fonksiyona girilen piksel değerleri ile baz çifti sayıları tahmin edilen fonksiyon oluşturulup bu değerlerin bulunmasında kullanılır. Bu fonksiyonun akış diyagramı ise Şekil 5.8’de verilmiştir.



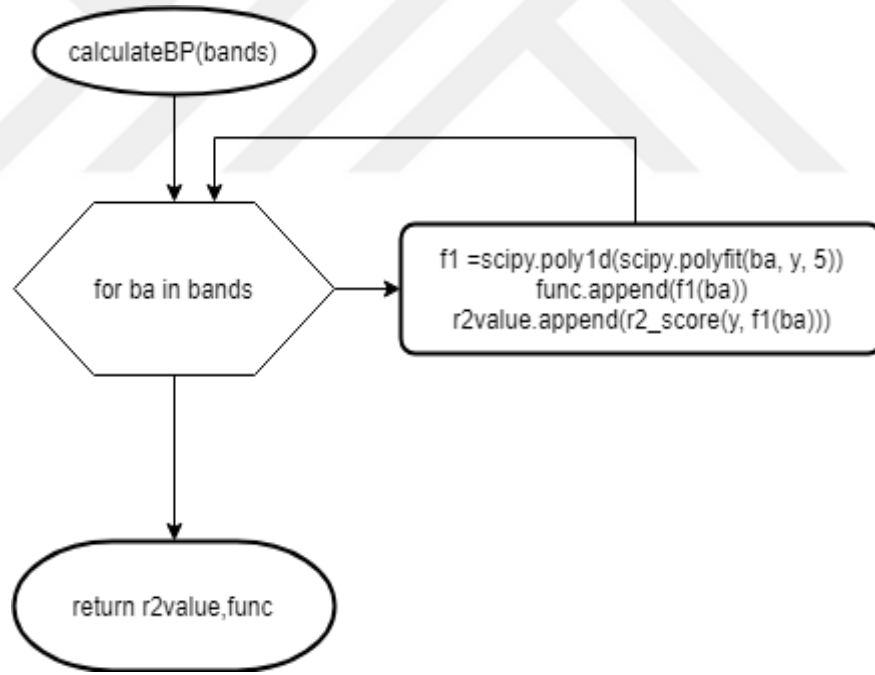
Şekil 5.5. Programın akış diyagramı.



Şekil 5.6. findLane() fonksiyonunun akış diyagramı.

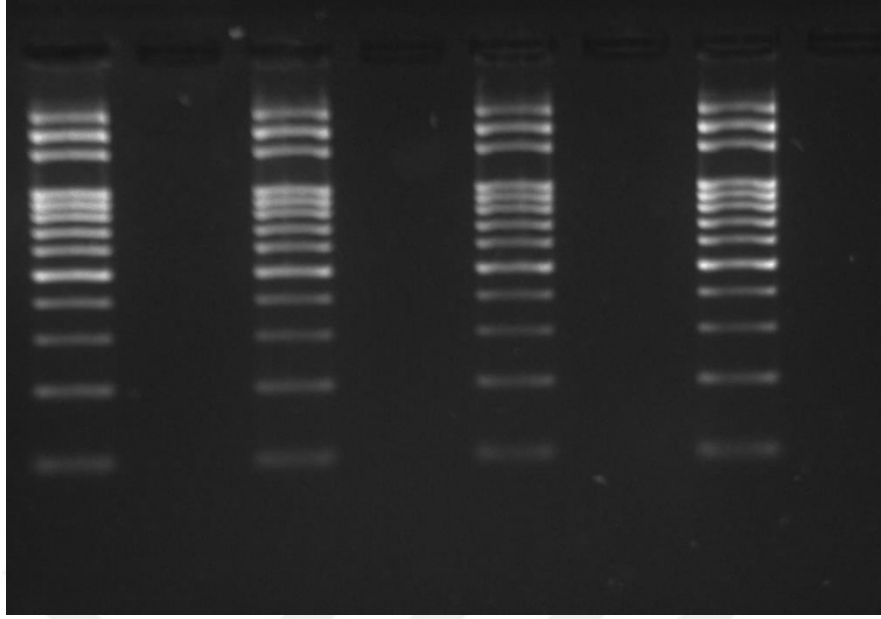


Şekil 5.7. findBands() fonksiyonunun akış diyagramı.

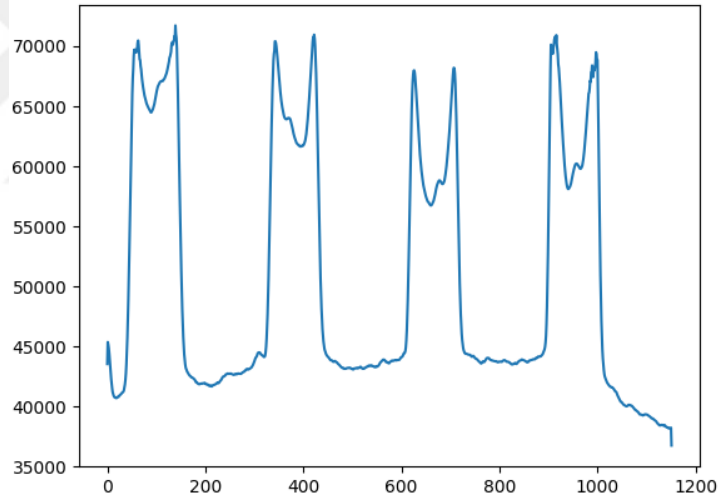


Şekil 5.8. calculateBP() fonksiyonunun akış diyagramı.

Şekil 5.9'deki görüntü matrisinin piksellerinin dikey olarak toplanmasıyla Şekil 5.10'daki sinyal elde edilmektedir. Sinyalden anlaşılacağı üzere sinyalde arka alandaki ışık parlamalarından ve kontrast farklarından dolayı oluşan sinyal bozulmaları mevcuttur.



Şekil 5.9. Mini Jel elektroforez sistemiyle yürütölmüş Ladder DNA görüntüsü.



Şekil 5.10. Jel görüntüsündeki piksellerin dikey olarak toplanmasıyla oluşun sinyal.

Şekil 5.11’de, Şekil 5.10’daki sinyalin baseline değeri bulduran komut verilmiştir.

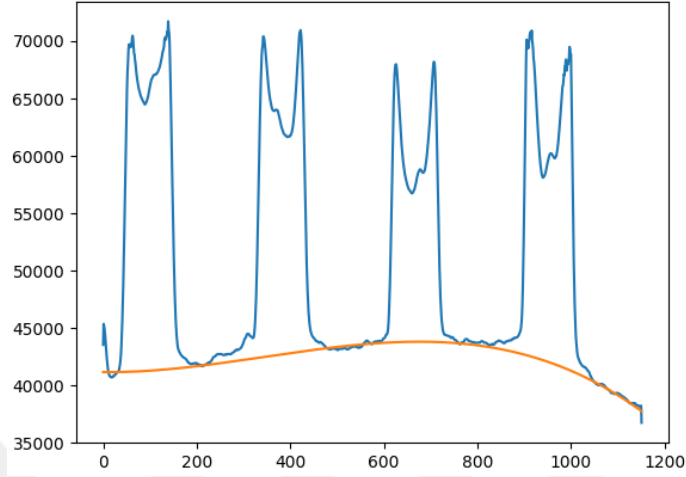
---

```
baseline_values = peakutils.baseline(cs)
```

---

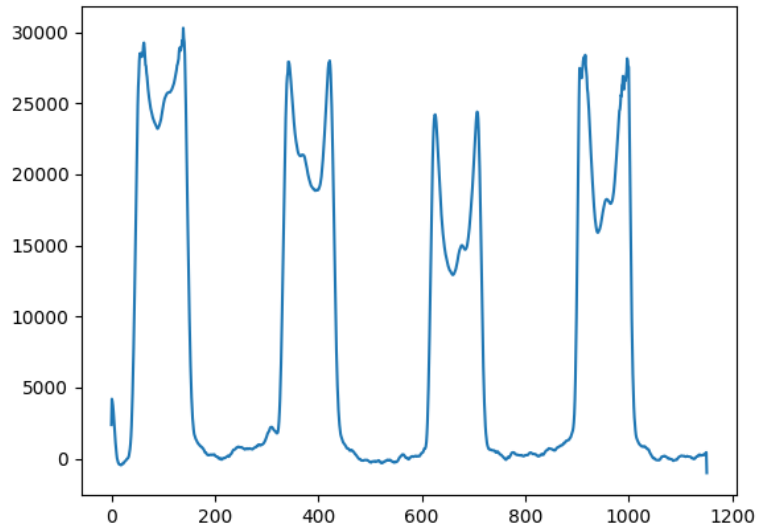
Şekil 5. 11. Sinyalin baseline değeri bulan komut

Şekil 5.12’de ise, Şekil 5.11’de ki komut ile tespit edilen baseline gürültüsü gösterilmektedir.



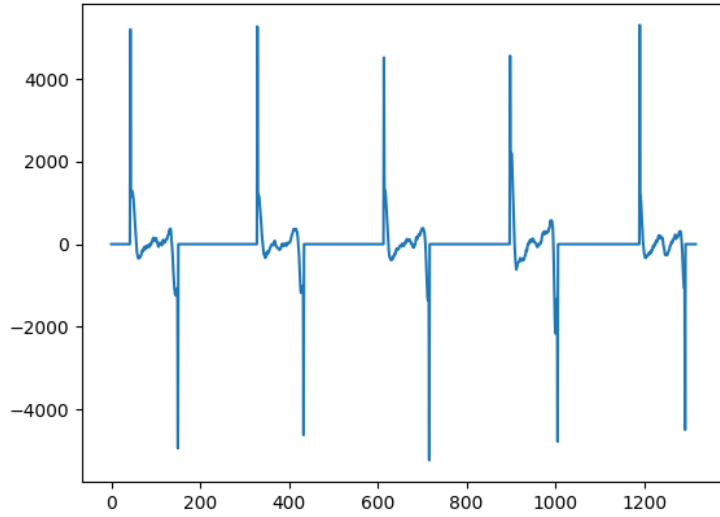
Şekil 5.12. Orijinal sinyal (mavi), baseline sinyali(turuncu).

Bu bozulmaları, gürültülü sinyalden baseline sinyalini çıkararak Şekil 5.13’deki düzeltilmiş sinyal elde edilmektedir.



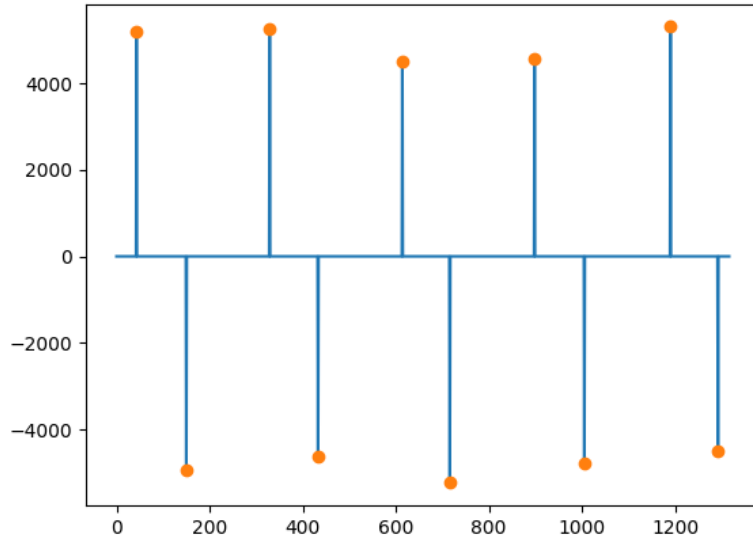
Şekil 5.13. Baseline gürültüsü düzeltilmiş sinyal.

Sinyaldeki büyük değişimlerin olduğu yerler görüntüdeki şeritlerin kenarlarını temsil etmektedir. Kenar noktalarının düzgün bir şekilde tespit edilebilmesi için sinyalin türevi alınarak Şekil 5.14’deki sinyal elde edilmektedir.



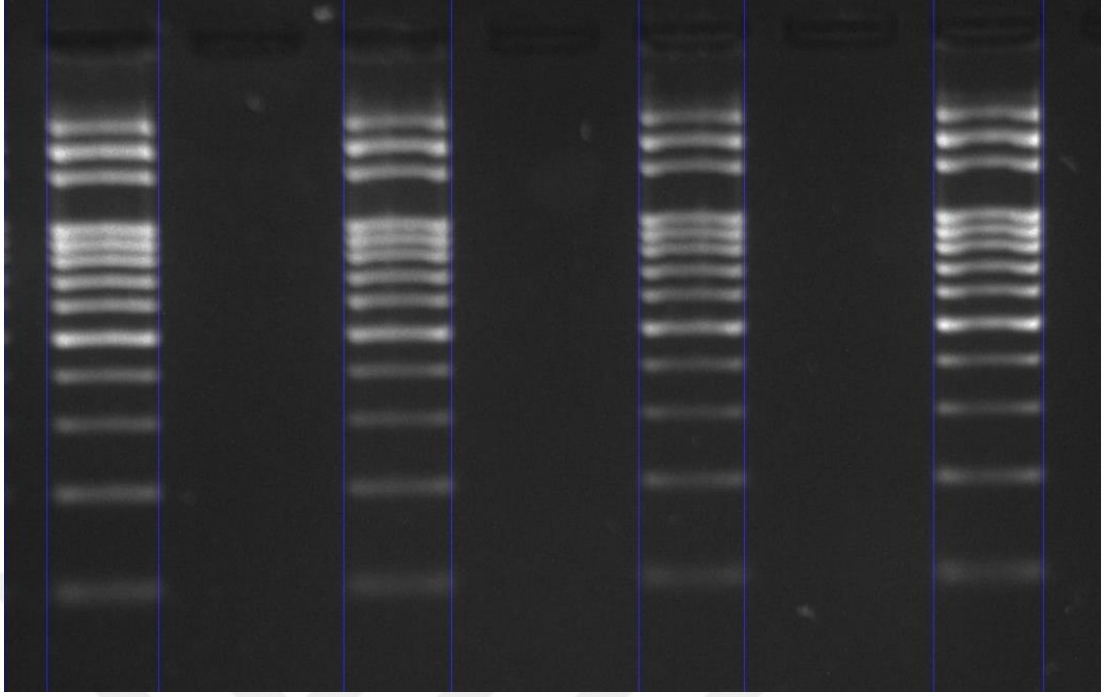
Şekil 5.14. Düzeltilmiş sinyalin türevi.

Sinyalin türevine pozitif ortalama ve negatif ortalama filtreleri uygulanarak sinyaldeki küçük parazitlerden kurtulmak mümkün olmaktadır. Ardından elde edilen sinyalin tepe noktaları bulunmaktadır. Şekil 5.15’de gösterilen tepe noktaları şeritlerin kenar noktalarının konumlarını ifade etmektedir.



Şekil 5.15. Türevi alınan sinyalin Mean Filter ile filtrelenmiş hali (mavi), tepe noktaları şeritlerin sınırları (turuncu).

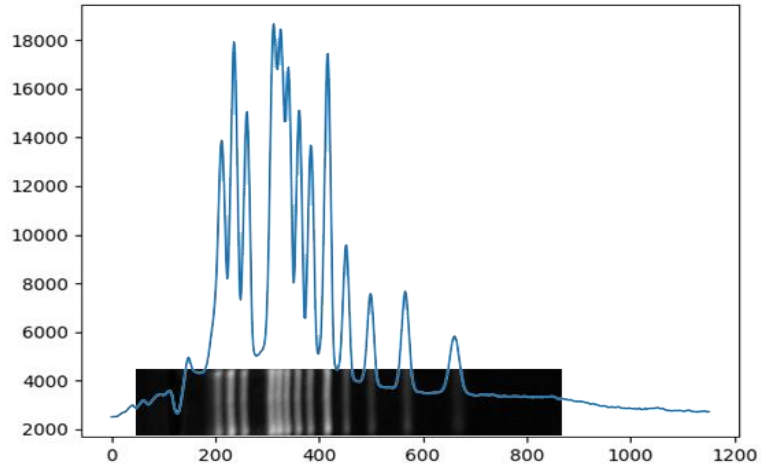
Tespit edilen bu noktalarda görüntüye çizgi çizerek Şekil 5.16’deki görüntü elde edilmekte ve şeritlerin kenarları işaretlenmektedir.



Şekil 5.16. Algoritmanın bulduğu şerit sınırlarının orijinal resimde çizgilerle gösterilmesi.

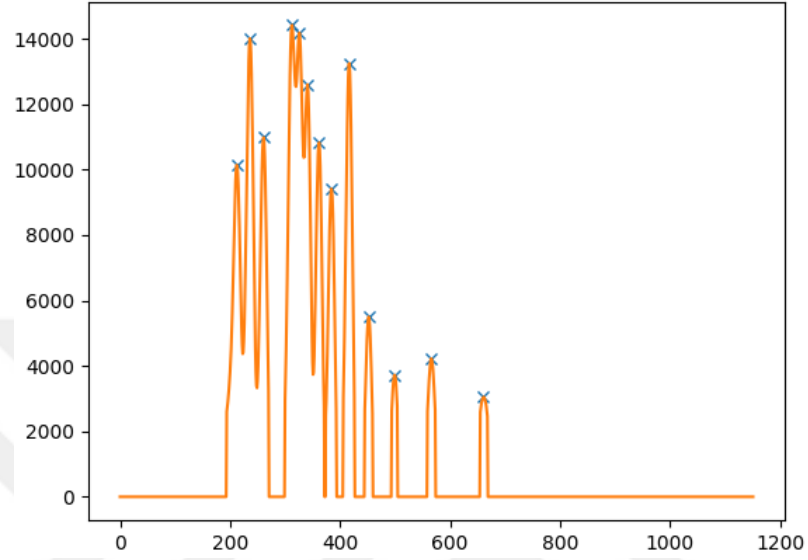
### 5.2.2. Bantların Konumlarının Tespiti

Şeritteki pikseller yatay olarak toplanarak Şekil 5.17'deki sinyal elde edilmektedir. Buradaki sinyali inceleyerek tepe noktalarının tespiti ile bantların konumlarını piksel olarak belirlemek mümkün olmaktadır.



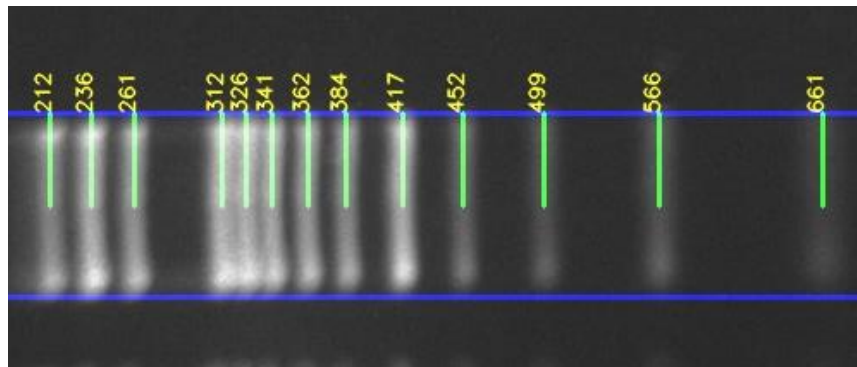
Şekil 5.17. Şeritteki piksellerin yatay olarak toplanması sonucu elde edilen sinyal.

Şekil 5.18'deki sinyalin de baseline sinyali bulunur, gürültülü sinyalden çıkartılarak düzeltilmiş sinyal elde edilmektedir. Sonra bu elde edilen sinyalin tepe noktaları bulunarak bantların yürütüldüğü mesafelerin uzaklıkları piksel cinsinden ifade edilmektedir.



Şekil 5.18. Filtrelenmiş ve tepe noktaları bulunmuş sinyal.

Şekil 5.19'de bantların tepe noktalarının bulunduğu piksel değerleri resmin üzerine yazılarak bu değerleri kullanıcının görmesi sağlanmaktadır. Elde edilen değerlere baktığımızda 3000 bp içeren 1. bant 212 piksellik mesafe yürürken, 100 bp içeren 13. bant ise 661 piksellik mesafe ilerlemiştir.



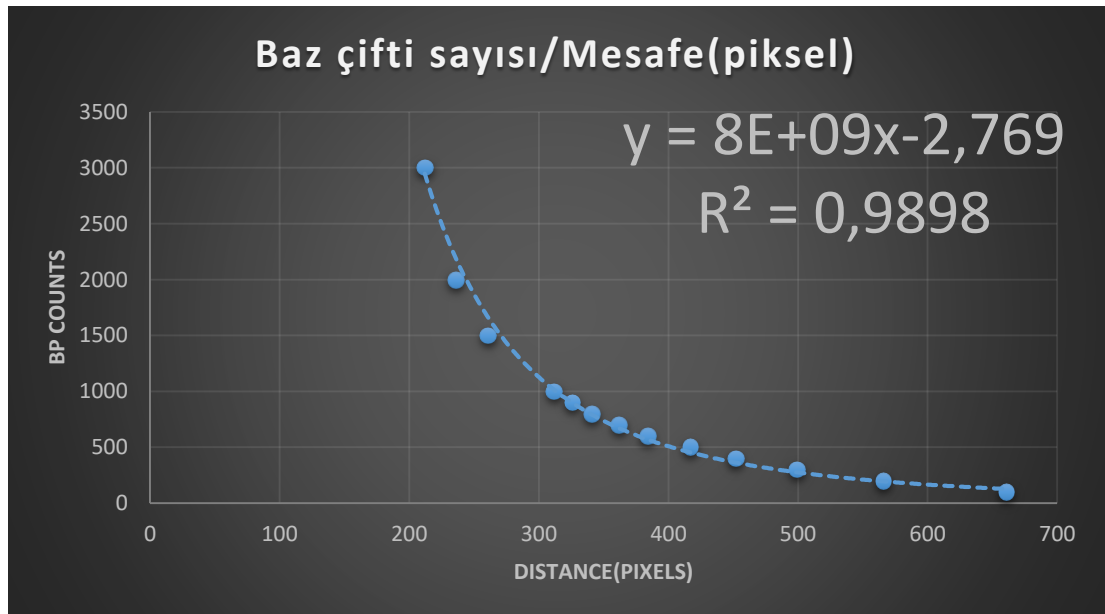
Şekil 5.19. Tepe noktalarının bulunduğu piksellerin resim üzerinde gösterimi.

Bulunan piksel türündeki mesafeler Çizelge 5.3'te gösterilmiştir. Piksel mesafeleri ile baz çifti sayıları arasındaki bağıntıyı sağlayan üs fonksiyon, Eşitlik 5.2'de verilmiştir. Şekil5.20'da Baz çiftleri ve piksellerin ilişkisini gösteren grafik ve uydurulan eğrinin R<sup>2</sup> değeri gösterilmektedir.

$$y = 8.10^9 x^{-2,769} \quad (5.2)$$

Çizelge 5.3. Baz çiftlerinin bulunduğu pikselleri gösteren çizelge.

Bant No	Bant Konumu (Piksel)	Gerçek BP
1	212	3000
2	236	2000
3	261	1500
4	312	1000
5	326	900
6	341	800
7	362	700
8	384	600
9	417	500
10	452	400
11	499	300
12	566	200
13	661	100



Şekil 5.20. Baz çiftleri ve Piksellerin ilişkisini gösteren grafik ve uydurulan eğrinin R<sup>2</sup> değeri.

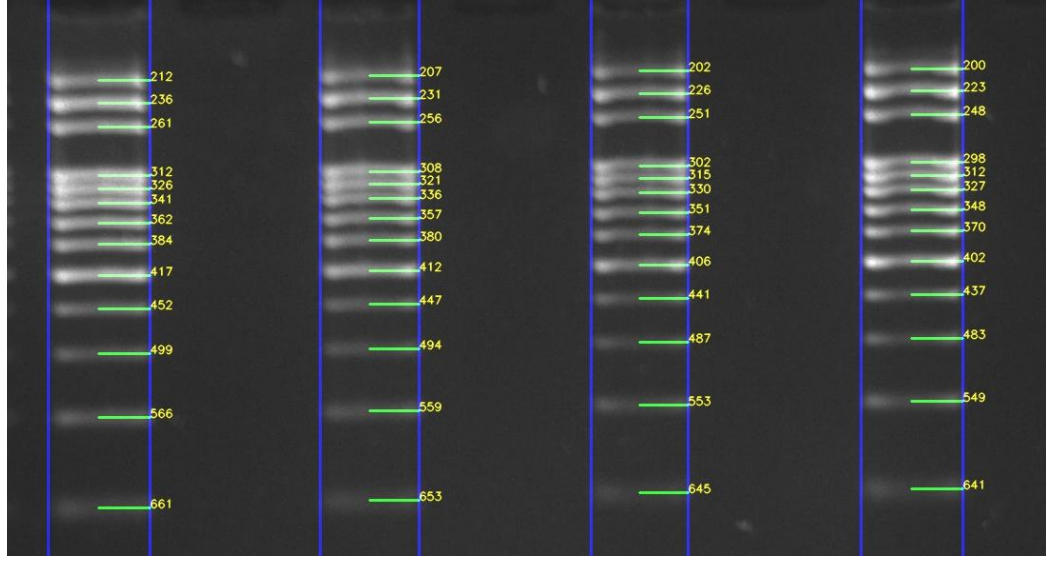
Şekil 5.19’de uydurulan bu fonksiyonun  $R^2$  değeri 0,9898 gibi 1’e çok yakın bir değer olduğu görülmektedir. Sadece bu  $R^2$  değeri bile program ile yapılan ölçüme göre gerçek değerine daha yakın olduğunu söylemek için yeterli olmaktadır. Yine de  $R^2$  değerleri yakın olmasına rağmen aradaki farkın boyutunu göstermek için bulunan piksel değerlerini uydurulan fonksiyona uyguladığımızda Çizelge 5.4’teki hesaplanan değerlere ulaşılmaktadır.

Çizelge 5.4’teki değerler yorumlanacak olursa hata oranlarının %0,7-%24 arasında değişken değerler aldığı görülmekte ve yürüdüğü piksel mesafesine göre BP sayısı hesaplanacak bandın gerçek değeri ile tahmin edilen değer arasındaki hatanın ortalama olarak %7,6 civarında olabileceği görülmektedir. Bu değerleri kıyasladığımızda bant başına düşen hata oranının azaldığını net bir şekilde görülmektedir.

Çizelge 5.4. Tahmin edilen BP sayıları ile gerçek değerler arasındaki farkları gösteren çizelge.

Bant	Piksel	Gerçek	Tahmin Edilen	Fark	Hata %
1	212	3000	2894	106	3,53
2	236	2000	2150	-150	7,5
3	261	1500	1627	-127	8,47
4	312	1000	993	7	0,7
5	326	900	879	21	2,33
6	341	800	776	24	3
7	362	700	658	42	6
8	384	600	559	41	6,83
9	417	500	445	55	11
10	452	400	356	44	11
11	499	300	270	30	10
12	566	200	191	9	4,5
13	661	100	124	-24	24
				<b>Ortalama Hata %=</b>	<b>7,6</b>

Bant bulmadaki işlemler bütün şeritlere uygulanmak suretiyle Şekil 5.21’deki görüntü elde edilmektedir. Görüntüdeki bütün bantların piksel olarak konumları tespit edilen tepe noktaları işaretlenerek gösterilmiştir.

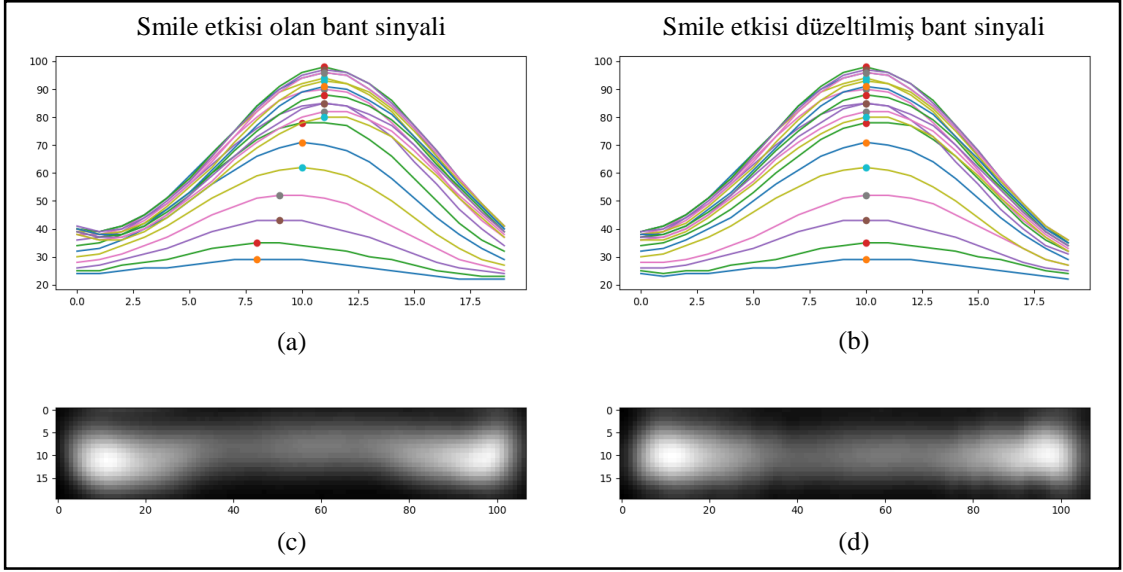


Şekil 5.21. Program tarafından bulunan bantların piksel cinsinden konumlarının gösterimi.

### 5.2.3. Smile Etkisi Düzeltme

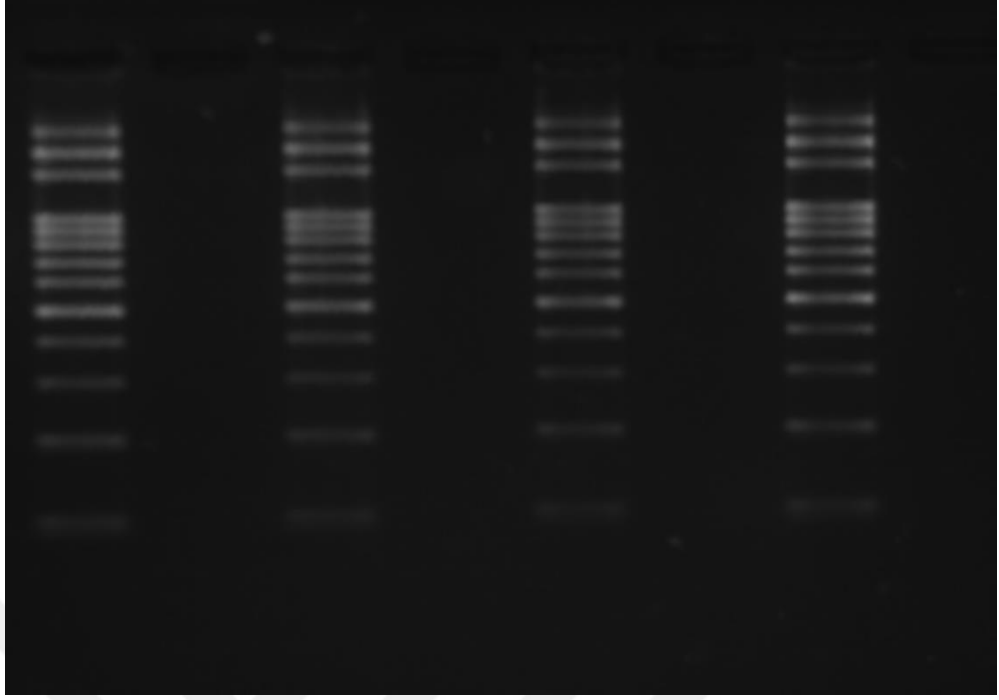
Smiling effects olarak da bilinen gülümseme etkisi olarak adlandırılan, elektroforez sırasında jele uygulanan voltajın büyüklüğü, jeldeki hava boşlukları, jel kuyucuklarının düzgün olmaması gibi varyasyonlardan kaynaklanan geometrik çarpıklıklara denilmektedir [3].

Bu etkiyi düzeltebilmek için bantların konumlarını tespit ettikten sonra bantlardaki her sütunun bir sinyal olarak incelediğimizde Şekil 5.22 (a)'daki sinyaller elde edilmektedir. Her sütunun tepe noktasını bulup, tepe noktalarını aynı hizaya sokmak için her sütunu hizalanması gereken noktaya göre öteleldiği zaman, Şekil 5.22 (b)'deki düzeltilmiş sinyali ve dolayısıyla düzeltilmiş görüntüyü elde etmek mümkün olmaktadır [2].

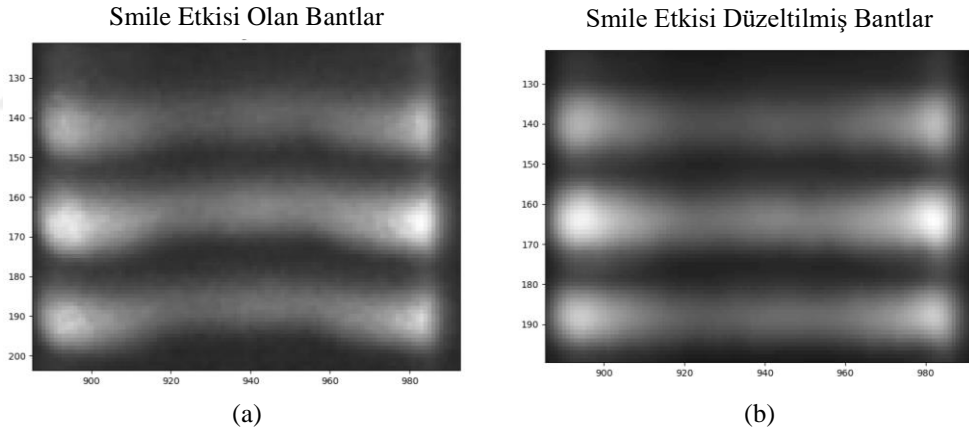


Şekil 5.22. a) Orijinal görüntü sinyali, b) Smile etkisi düzeltilmiş sinyal, c) Orijinal görüntü, d) Smile etkisi düzeltilmiş görüntü.

Şekil 5.22 (a)'da her dikey satırın parlaklık değerleri grafik olarak verilmiştir her satırdaki tepe noktaları tespit edilerek tepe noktalarının indis değerleri “pp.append(indis)” komutu ile bir “pp” dizisine eklenmektedir. Sonra bu dizinin ortalaması “np.mean(pp)” komutu ile alınarak yeni tepe noktasının indisi hesaplanmaktadır. Hesaplanan yeni indis referans alınarak tepe noktaları arasındaki farka göre sinyal ötelenmekte ve Şekil 5.22 (b)'deki sinyal ve (d)'deki görüntü elde edilmektedir. Şekil 5.23'de bütün bantları düzeltilmiş jel görüntüsü görülmektedir. Şekil 5.24'te ise orijinal görüntü ile düzeltilmiş görüntünün yakından karşılaştırılmış hali gösterilmiştir.



Şekil 5.23. Smile etkisi düzeltilmiş görüntü.



Şekil 5.24. a) Smile etkisi düzeltilmemiş görüntü, b) Smile etkisi düzeltilmiş görüntü.

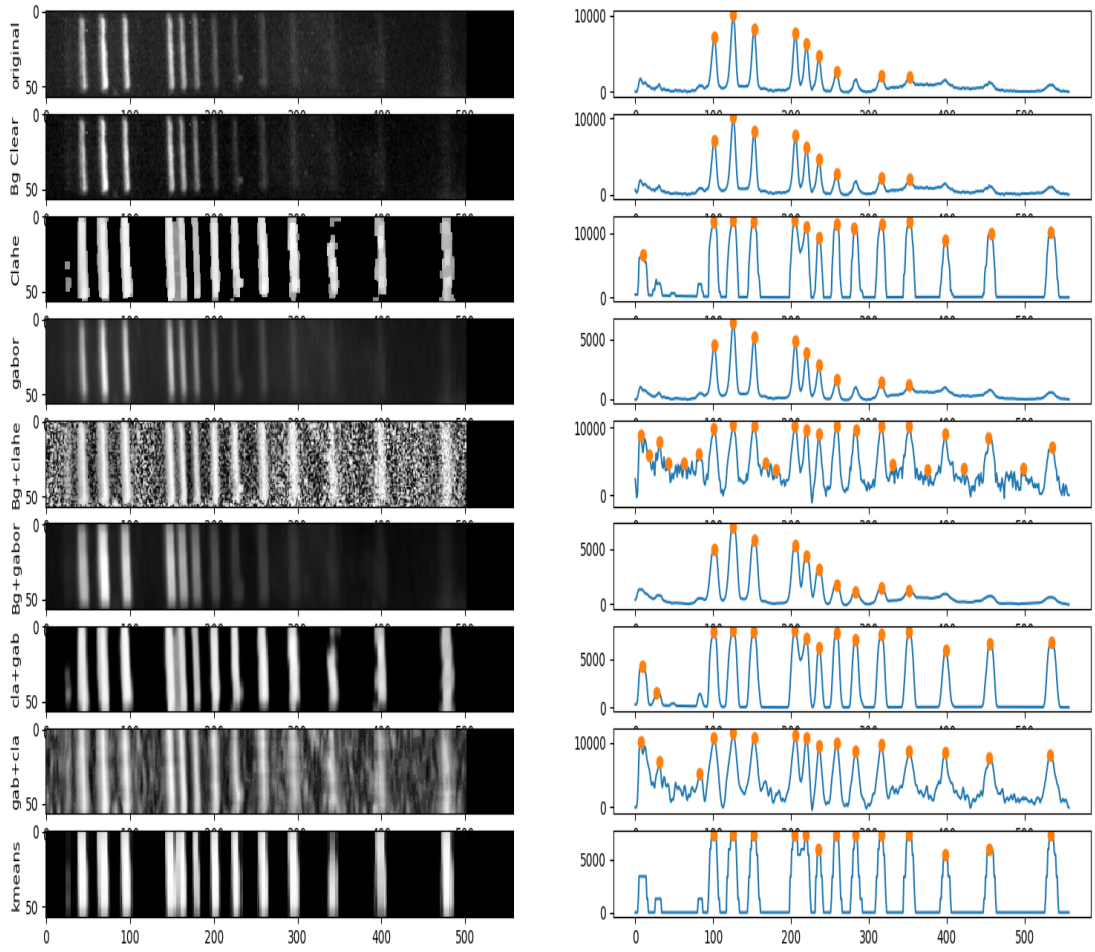
Genellikle fazla gerilimden kaynaklanan bu geometrik bozuklukları gidererek mini jel elektroforez sisteminde yürütülen örnekler daha fazla voltaj vererek yürütülme sürelerini kısaltmak mümkün olabilmektedir.

### 5.3. FİLTRELENMİŞ GÖRÜNTÜLERDE BANTLARIN BULUNMASI

Filtrelenmiş görüntülere teker teker bant bulma algoritmaları uygulayarak en iyi filtreleme yöntemini tespit etmek gerekmektedir. Bu filtre yöntemlerini karşılaştırarak hangi filtreleme yöntemlerinde daha çok bantın bulunduğu tespit edilecektir.

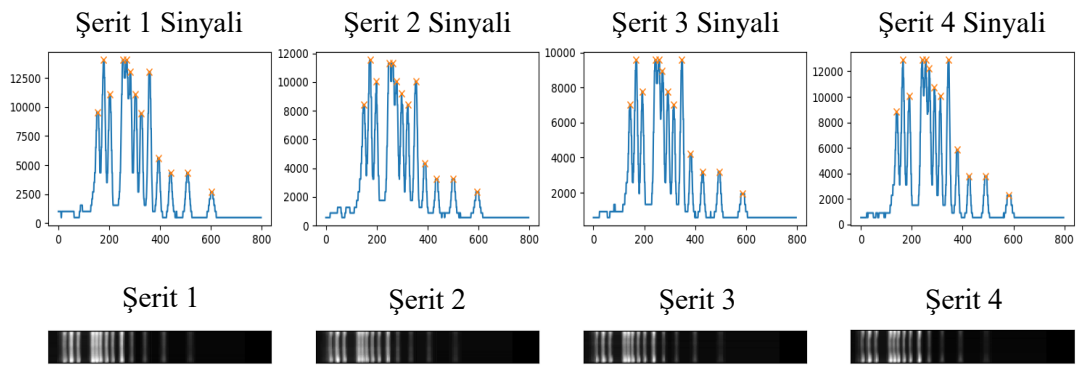
En verimli olan filtreleme yöntemiyle bulunan bantların konumları tespit edilecektir. Bant bulma algoritması Şekil 5.25’de farklı türlerde filtreler uygulanmış şerit görüntüsü gösterilmektedir. Algoritma ilk önce orijinal görüntüde ve görüntüden arka alan gürültüsü (BG Clear) kaldırılmış görüntüye uygulanmış iki görüntüde de sondaki parlak olmayan bantları bulamamıştır. CLAHE, BG Clear+CLAHE, CLAHE+Gabor filtreleri uygulanan görüntülerde tüm bantları ve bazı istenmeyen lekeleri de bulmuştur.

Yine Şekil 5.25’de görüldüğü üzere uygulanan filtreler neticesinde sinyaldeki tepe noktaları belirginleşmiş bazı görüntüler mevcut olsa da görüntünün kendisine baktığımızda bantlarda bozunmalar olduğu gözlemlenmiştir. Clahe uygulanıp bantlar belirginleştirilmiş, ardından eşikleme uygulanıp (Threshold Value = 120) zemin siyah olarak ayarlanmış sonrasında morfolojik açma uygulanıp parazitli istenmeyen veriden kurtulmak mümkün olmuştur. Sonrasında gabor filtre uygulayarak desene uygun olmayan parazitlerden görüntü arındırılmıştır. Son olarak da K-means kümeleme yöntemiyle dağınık olan bölgeler birleştirilip bantlar gerçek boyutlarına en yakın ebatlara getirilmiştir. Bantların belirginleştirilmesinin ardından bantların konumlarının tamamı tespit edilmiştir.



Şekil 5.25. Farklı filtrelerle elde edilen görüntüleri bant bulma algoritmasının uygulanması.

Şekil 5.26'da ise K-means kümelemesi ve algoritmayı kullanarak bütün Şeritlerdeki bantların tamamı tespit edilmiş, şeritler ve şeritlerin sinyalleri verilmiştir.



Şekil 5.26. K-Means Kümeleme sonrası algoritma ile bantların bulunması.

### 5.3.1. Mesafeye Göre BP Hesaplayan Fonksiyon Oluşturma

Bantların konumları piksel olarak tespit edildikten sonra ladder DNA'nın yanında verilen veya üreticisinin sitesinden ulaşılabilen bilgi sayfalarından Lader DNA baz çifti sayıları bantların yürüme sırasına göre polinomal regresyon yapılarak belirlenmektedir.

Çizelge 5.5. Algoritma ile bulunan bantlar ile baz çifti sayılarını gösteren çizelge.

Band No	Şerit 1	Şerit 2	Şerit 3	Şerit 4	BP Sayıları
1	212	207	202	200	3000
2	236	231	226	223	2000
3	261	256	251	248	1500
4	312	308	302	298	1000
5	326	321	315	312	900
6	341	336	330	327	800
7	362	357	351	348	700
8	384	380	374	370	600
9	417	412	406	402	500
10	452	447	441	437	400
11	499	494	487	483	300
12	566	559	553	549	200
13	661	653	645	641	100

Çizelge 5.5 incelendiğinde BP sayısı azaldıkça gidilen mesafe arttığı görülmektedir. Baz çifti sayısı fazla olan bantların aralarındaki mesafe kısa iken baz çifti sayısı azaldıkça iki bant arasındaki mesafenin de daha fazla açıldığı görülmektedir. Çizelgeden, piksel değerleri ile baz çifti arasında arada oluşturulacak fonksiyonun lineer olamayacağını çıkartılabilmektedir. Piksel değerleri ile Baz çifti sayıları arasında polinomal regresyon yaparak, yürünen piksel mesafesine göre baz çifti sayılarının tahminini yapan bir fonksiyon oluşturulabilmektedir.

Programla tespit edilen bantların piksel bazında konumları ile polinomal regresyon (Eşitlik 5.3), üs regresyon (Eşitlik 5.4) ve üstel regresyon (Eşitlik 5.5) olmak üzere 3 farklı eğri uydurulmuştur. Polinom ve üs regresyon piksel değerleri ile oluşturulurken, üstel regresyon için en küçük baz çiftine sahip bandın gittiği konum referans tutulur ve diğerlerinin ona oranla ne kadar yol kat ettiği yüzde olarak hesaplanmaktadır. Bu

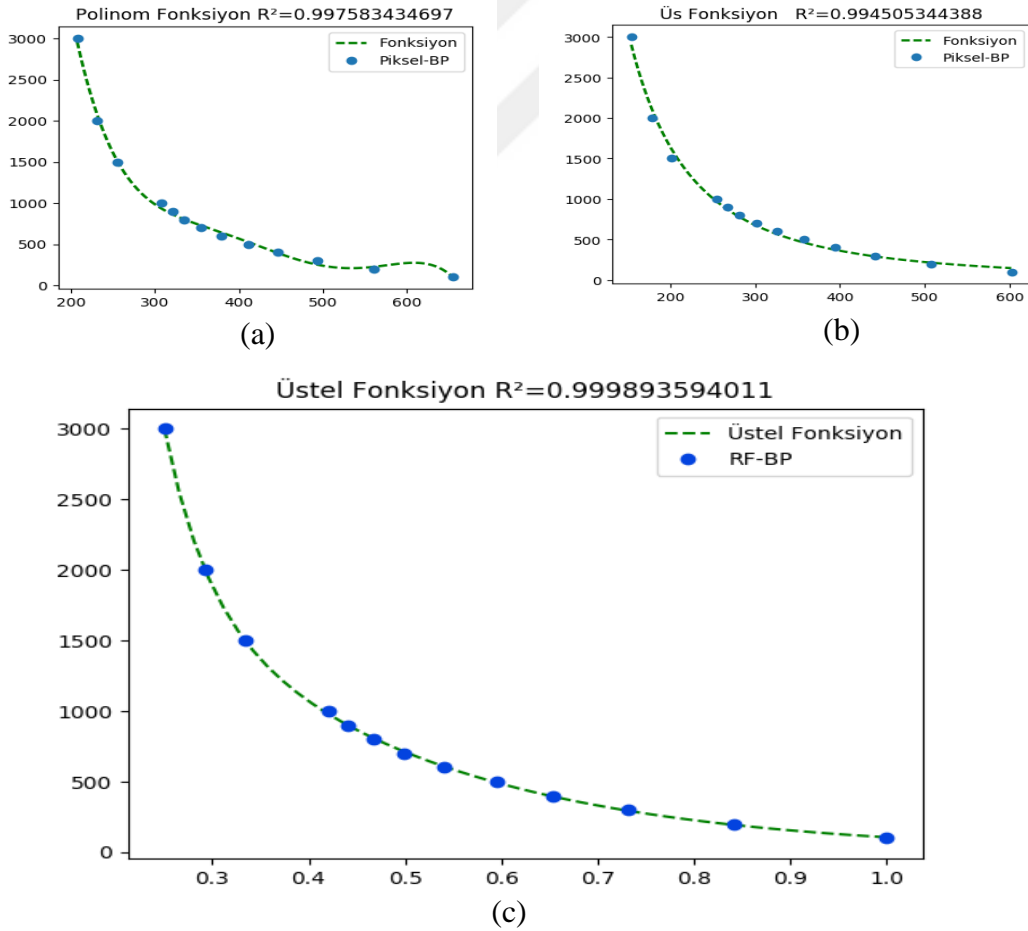
işleme ön bağıl (relative front:RF) denilmektedir. Her bant için bir RF değeri hesaplanmaktadır. Bulunan bu RF değerlerini ile üstel fonksiyon oluşturulmaktadır.

$$f(x) = -2,81 \cdot 10^{-9} \cdot x^5 + 5,44 \cdot 10^{-6} \cdot x^4 - 4,12 \cdot 10^{-3} \cdot x^3 - 2,84 \cdot 10^2 \cdot x + 2,2 \cdot 10^4 \quad (5.3)$$

$$f(x) = 1,69339694 \cdot 10^8 \cdot x^{-2,1787} \quad (5.4)$$

$$f(x) = 4692,579 \cdot e^{-3,7841x} + 35690,25 \cdot e^{-22,8097x} \quad (5.5)$$

Bu fonksiyonların grafikleri ve Piksel-BP grafikleri Şekil 5.27’de verilmiştir. Grafiklerde bakıldığında  $R^2$  değerleri sıralaması olarak üs<polinom<üstel olduğu görülmektedir. Buradan üstel fonksiyonun  $R^2$  değerinin 1’e en yakın olan fonksiyon olduğu çıkarılmaktadır.



Şekil 5.27. Piksel-Baz Çifti grafiği, uydurulan eğri fonksiyonlarının karşılaştırılması a) Polinom fonksiyon, b) Üs fonksiyon, c) Üstel fonksiyon

Fonksiyonlar arasındaki farkları daha iyi görebilmek için piksel konumları bulunan bantların, değerlerini fonksiyonlara girip tahmin edilen BP sayıları ile gerçek BP sayıları arasındaki hata oranları Çizelge 5.6’da gösterilmektedir. Çizelgede görüldüğü üzere  $R^2$  değerleri büyük olsa bile polinom ve üs fonksiyonlarının tahmin ettiği baz sayılarında dalgalanmalar olduğu anlaşılmaktadır. Üstel fonksiyonla oluşturulan eğri de ise gerçek değerlerine çok yakın değerleri tahmin ettiği görülmektedir.

Çizelge 5.6. Oluşturulan fonksiyonlarla hesaplanan BP değerleri ile gerçek BP değerleri arasındaki hataları gösteren çizelge.

Bant No	Baz Çifti Sayısı	Polinom			Üs			Üstel		
		Tahmin	Hata	Hata%	Tahmin	Hata	Hata%	Tahmin	Hata	Hata%
1	3000	2960	40	1	2960	40	1	2999	1	0
2	2000	2071	71	4	2071	71	4	2004	4	0
3	1500	1509	9	1	1509	9	1	1497	3	0
4	1000	932	68	7	932	68	7	982	18	2
5	900	862	38	4	862	38	4	902	2	0
6	800	802	2	0	802	2	0	809	9	1
7	700	728	28	4	728	28	4	714	14	2
8	600	644	44	7	644	44	7	607	7	1
9	500	527	27	5	527	27	5	494	6	1
10	400	392	8	2	392	8	2	395	5	1
11	300	251	49	16	251	49	16	295	5	2
12	200	225	25	13	225	25	13	194	6	3
13	100	97	3	3	97	3	3	106	6	6
		Ortalama Hata=		5	Ortalama Hata=		5	Ortalama Hata=		2

#### 5.4. SONUÇLAR VE TARTIŞMA

Şeritlere uygulanan filtreler ve yöntemlerin sonuçları ve kombinasyonlarının sonuçlarında elde edilen görüntülerdeki bantların bulunmasını sağlayan algoritmaların çalıştırılması ile tespit edilen bant konumları karşılaştırılmıştır. Bu karşılaştırma neticesinde en etkili yöntemin K-means kümeleme yöntemi ile elde edildiği görülmüştür. Yazılım arayüzüne jel görüntüsüne göre alternatif olarak CLAHE+Threshold, BG+CLAHE, CLAHE+Gabor yöntemleri de eklenebilir. Böylelikle farklı jel görüntülerinde farklı filtrelemeler seçilerek en verimli yöntemi kullanıcının oluşturmasına imkân tanımak mümkün olabilmektedir.

Eđri uydurmada kullanılan fonksiyonlar karşılaştırıldığında polinom ve üs fonksiyonlarının  $R^2$  değerleri büyük olmasına rağmen üstel fonksiyonun  $R^2$  değeri 1'e diğerlerinden daha yakın olduğundan dolayı gerçeđe daha yakın tahminlerde bulunmaktadır. Bu sebepten dolayı programda uygulanması gereken eđri uydurma fonksiyonu olarak üstel fonksiyon kullanılmaktadır.



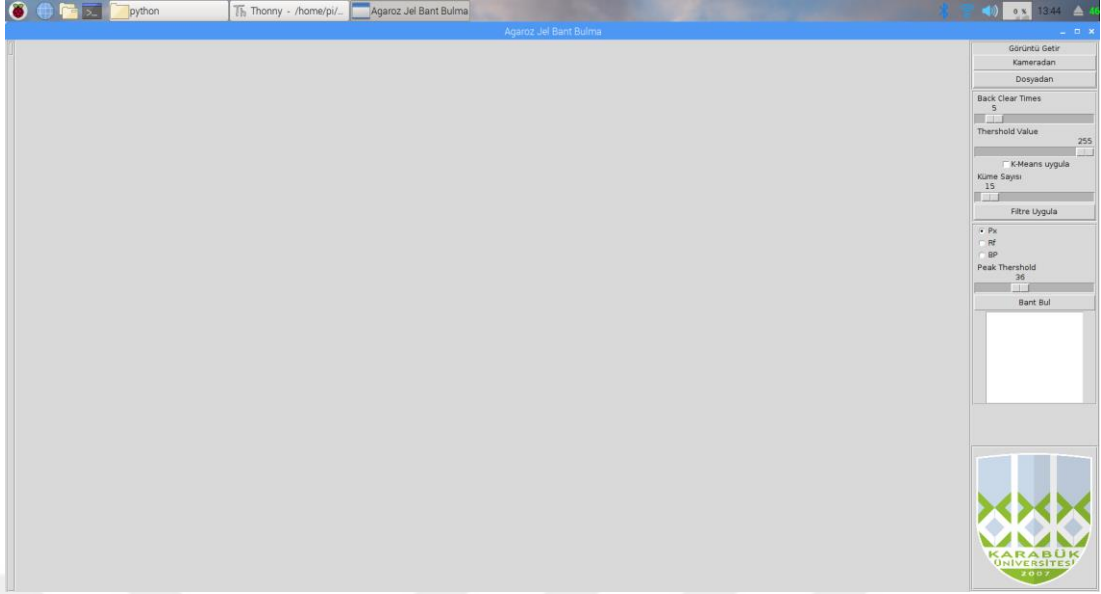
## BÖLÜM 6

### ARAYÜZ

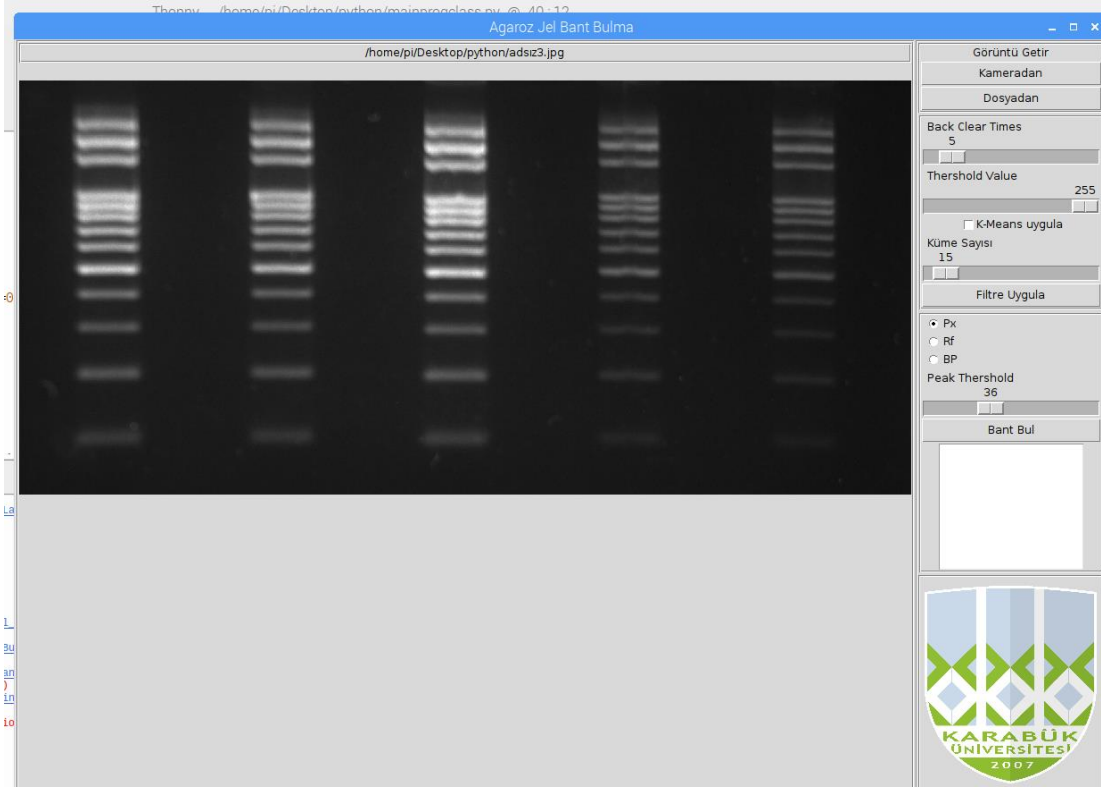
Bu bölüme kadar filtrelemeler ve bant konumlarının tespiti hakkında bilgiler verilmiştir. Filtreleme işlemleri ve bant konumlarının bulunması ile ilgili işlemler her görüntü için farklı parametreler içerdiğinden dolayı bu algoritma ve filtreleri uygulamak için bir arayüze ihtiyaç duyulmaktadır. Her kullanıcı kodlardaki bu parametreleri yüzlerce satır arasından bulup eliyle değiştiremeyeceği için ve kodlama bilgisine sahip olması gerekmediğinden dolayı bir arayüz tasarlanmıştır. Bu arayüz sayesinde programa kameradan ve dosyadan görüntü girişi sağlanabilmektedir. Yüklenen görüntüler üzerinde filtrelerin ve algoritmaların parametrelerini her kullanıcının rahatlıkla değiştirebileceği şekilde uyarlanmıştır.

#### 6.1. ARAYÜZ KULLANIMI

Arayüz kullanıcıyı komutların karmaşasından kurtarmak için basite indirgenmiş buton ve kaydırma çubuğu nesnelere ile programın kontrolünü kolaylaştıran bir yapı halinde tasarlanmıştır. Şekil 6.1’de programın ana ekran görüntüsü verilmiştir buradan yeni görüntü almak için kamera uygulamasına geçilebilmektedir. Ayrıca önceden kaydedilmiş bir görüntüyü de dosyadan aktarma imkânı bulunmaktadır. Dosyadan görüntü aktarmak için “Görüntü Getir” çerçevesinde bulunan “Dosyadan” butonuna basılarak açılan göz at penceresinden istenilen dosya seçilir. Görüntü ekrana Şekil 6.2’deki gibi gelmektedir.



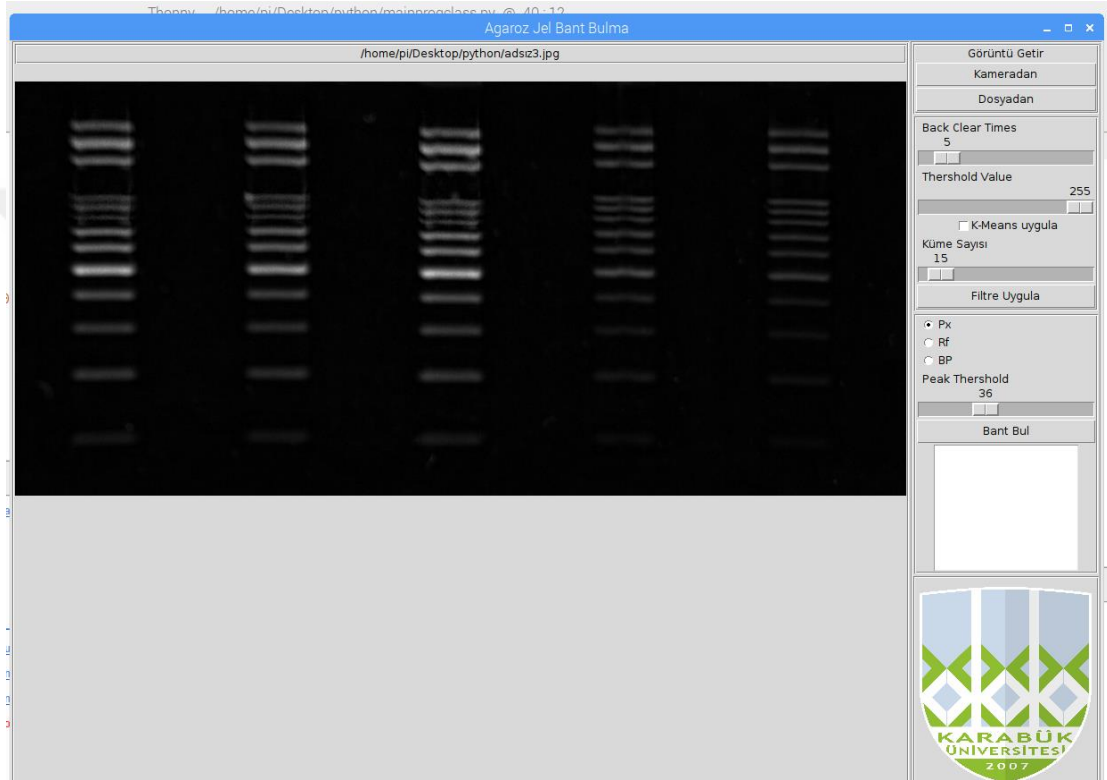
Şekil 6.1. Programın ana ekranı.



Şekil 6.2. Yüklennmiş jel görüntüsü.

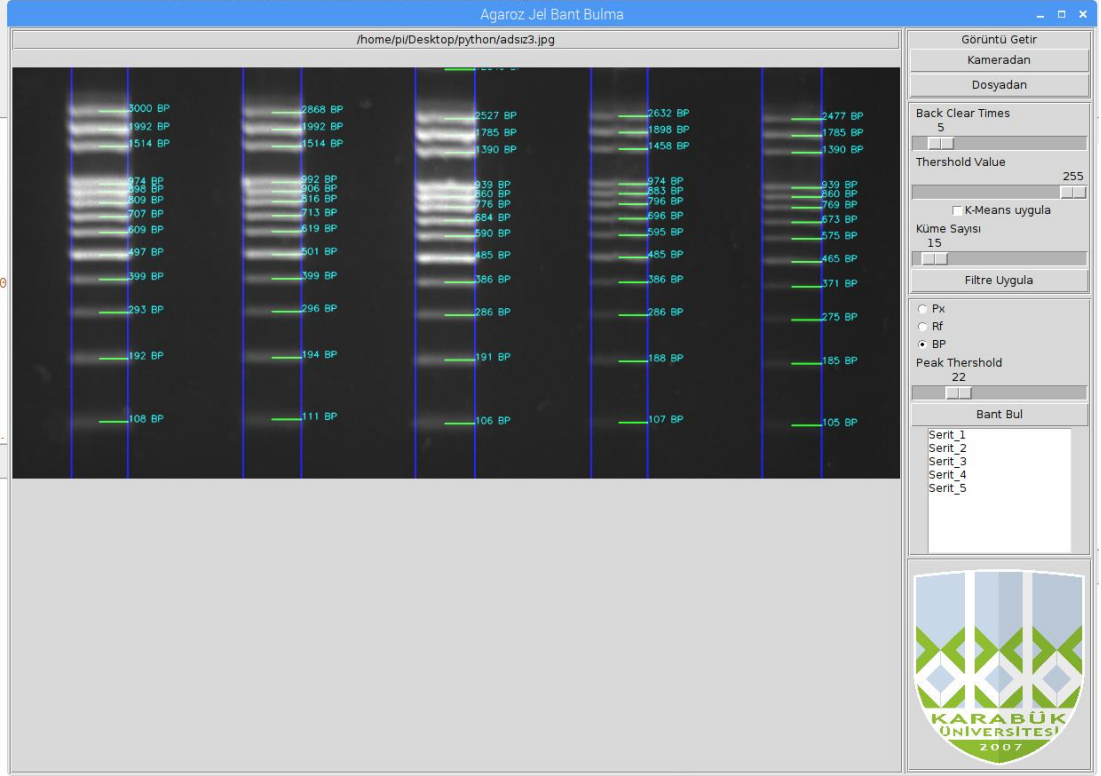
Görüntü yüklendikten sonra filtre çerçevesinde bulunan “Gürültü Temizleme Katsayısı” Arka plandaki gürültünün silinme işleminin kaç kere uygulanacağı ayarlanır. Bu değer arttıkça işlem tekrarı artacağından filtreleme işlemlerinde

gecikmeler oluşabilir. “Eşik Değeri” parametresi filtreleme esnasında oluşan görüntünün parlaklığını arttırıp azaltmada kullanılan ayar çubuğudur. “Küme Katsayısı” K-Means kümeleme işlemlerinde 0-255 arası olan değerler arasındaki histogramına kaç kümede toplanacağı parametresi ayarlanır. Bu ayar arttıkça filtreleme işlemi uzun sürmektedir. Şekil 6.3’da filtrelenmiş bir görüntüsü verilmektedir.



Şekil 6.3. Filtrelenmiş görüntü.

Bant bul komutu ise Şekil 6.4’deki gibi filtreleme işlemleri sonunda şeritlerin ve bantların konumlarını tespit edip hangi piksel üzerinde olduklarını görüntü üzerine yazdırma işlemi gerçekleştirilmektedir.



Şekil 6.4. Piksel değerlerinin jel görüntüsü üzerine yazılması.

## BÖLÜM 7

### SONUÇLAR VE ÖNERİLER

Elektroforez jel görüntülerinin gömülü sistemle analizinin gerçekleştirilmesi sonucunda, Eşikleme, BG Clear, CLAHE, HE, Gabor, K-means görüntü filtreleme yöntemleri uygulanmıştır. Bu yöntemler arasında bir eşik değerine göre filtreleme yaparak bantların tamamını arka alan gürültüsünden ayırmanın her görüntü için mümkün olmadığı gözlemlenmiştir.

Global histogram eşitleme yöntemi ise beyaz yoğunluklu noktaları aşırı parlatıp bantların sınırlarının birbirine geçmesine ve hatta yok olmasına neden olduğu gözlemlenmiştir. Global histogram eşitlemeye eşik değere göre eşikleme uygulandığında dahi bantlar arasındaki sınırların belirginleşmediği tespit edilmiştir.

CLAHE (kontrast sınırlamalı adaptif histogram eşitleme) yöntemi ile filtrelenmiş görüntü global histogram eşitleme yöntemine nispeten daha dengeli bir dağılım sağlamış ve ardından uygulanan eşik değerine göre eşikleme yöntemi uygulanan parazitlerde azalmalar gözlemlenmiştir.

Morfolojik geri çatma yönteminde ise görüntülerin arka alan gürültülerine göre tekrar miktarı değişen erozyon ve genişletme uygulanarak bantlar arka plandan ayrıştırılmıştır.

K-means yöntemiyle birbirine yakın piksel değerleri her görüntüye göre farklı küme sayısınca kümelenip görüntüde keskin farklar oluşturulmuştur. Bu sayede bantların konumları daha belirginleştirilmiştir.

Mini jel elektroforez cihazından [40] elde edilen görüntülere bu yöntem ve teknikler uygulanıp elde edilen görüntülerin üzerine algoritma uygulanıp bulunan bant konumları karşılaştırılmıştır. En verimli olacak yöntemler birleştirilerek görüntüdeki bantların yerlerinin tespit edilmesi sağlanmıştır. Uygulanan yöntemler neticesinde jel yapısından kaynaklı ışık patlamalarının her jel görüntüsünde aynı olmamasından dolayı oluşan arka planları kaldırmak için her görüntüye farklı parametre değerleri ile filtreleme uygulanması gerektiği belirlenmiştir. Bu gürültüleri kaldırmak için farklı filtreleme yöntemleri araştırılıp denenmiş, bazılarının kombinasyonları da uygulanarak en etkili olan yöntemin sırasıyla CLAHE, eşikleme, morfolojik açma, Gabor filtreleme uygulanmasının ardından K-Means yöntemi ve arka plan gürültüsünü kaldırmada kullanılan Morfolojik geri çatma yöntemlerinin sıralı olarak uygulanmasıyla etkili sonuçlar elde edilmiştir. Gürültülerin her görüntüye özgü olmasından ve her görüntüye standart bir filtre uygulanarak görüntüdeki bantların tespitinin mümkün olmamasından dolayı geliştirilen arayüze her kullanıcının parametreleri görüntüye özgü olarak ayarlayabilmesi için parametre değerlerini değiştirebilen kaydırma çubukları konulmuştur. Bu ayarlar sayesinde kullanıcı yazılım bilgisine ihtiyaç duymaksızın parametreleri istediği gibi değiştirip jel görüntülerinden en iyi görüntüyü elde edip bantların konumlarını piksel olarak veya BP sayısı olarak tespit ettirebilmektedir.

Gerekli filtrelemeler neticesinde bantların konumları tespit edilebilmektedir. İhtiyaçlar doğrultusunda program daha da geliştirilebilir. Analizlerin yapıldığı sistemin gömülü sistem olmasından dolayı elektroforez sisteminin kontrolü bu cihazla yapılabilir. Analiz programı ile kontrol programı birleştirilebilir. Smile etkisini düzelteren algoritma sayesinde gömülü sistem tabanlı mini jel elektroforez sistemine uygulanan gerilim artırılabilir. Gerilimin artması sayesinde ise DNA örnekleri daha kısa sürelerde ayrıştırılabilir. Gömülü sistemde internet bağlantısı ya da sunucu olarak kullanılabilme özellikleri de mevcut olduğundan dolayı bu elektroforez cihazı bir IOT (internet of things) cihazına dönüştürülebilir. Böylelikle sonuçlar istenilen sisteme, veri tabanına, internet sitesine veya mail adresine yönlendirilebilir. Sistemin hızı mevcut halinden çok daha hızlı bir hale getirilebilir ve daha kompakt bir sistem oluşturulabilir. Ayrıca sistem kendi analizini yapabilen bir elektroforez cihazı olarak ticari bir ürüne, çok düşük maliyetle dönüştürülebilir.

## KAYNAKLAR

1. Güneştutar Y. L. “Elektroforezde Yeni Yaklaşımlar”, *Journal of Engineering and Natural Sciences(Sigma)*, 151-160 (2009)
2. Intarapanich, A., “Automatic DNA Diagnosis for 1D Gel Electrophoresis Images using Bio-image Processing Technique. ”, *BMC Genomics* ,(2015)
3. Abadi, M. F., “Processing of DNA and Protein Electrophoresis Gels by Image processing”, *Fen Bilimleri Dergisi* , Cilt 36. ISSN: 1300-1949 , (2015 )
4. Mersinkaya, İ., “Gömülü Sistem Tabanlı Mini Jel Elektroforez Sistemi Tasarımı”, Yüksek Lisans Tezi, *Karabük Üniversitesi Fen Bilimleri Enstitüsü*, Karabük, (2019)
5. Smith A.D., “Oxford Dictionary of Biochemistry and Molecular Biology”, New York, (201,499),(2000)
6. Tiselius A., “A New Apparatus For Electrophoretic Analysis of Colloidal Mixtures”, *Transactions of the Faraday Society*, (33):524-531, (1937)
7. Ausubel, F.M. et al., “Current Protocols in Molecular Biology”, *Supplement* 23,USA(1989)
8. Bourzac, K. M. LaVine, L. J., “Analysis of DAPI and SYBR Green I as Alternatives to Ethidium Bromide for Nucleic Acid Staining in Agarose Gel Electrophoresis”, *Journal of Chemical Education*, (80):1292-1296(2003)
9. Green, M. R. And Sambrook J., “Agarose Gel Electrophoresis”, *Cold Spring Harbor Laboratory Press*,(2):87-93 (2019)
10. İnternet: GelAnalyzer. <http://www.gelanalyzer.com> (2019)
11. Heras J, Domínguez C, Mata E, Pascual V, Lozano C, Torres C, Zarazaga M. “GelJ-a tool for analyzing DNA fingerprint gel images”, *BMC Bioinformatics*, 16:270 (2015)
12. Pavel A, Vasile C. “PyElph-a software tool for gel images analysis and phylogenetics”. *BMC Bioinformatics*. 13:9 (2012)
13. Khakabimamaghani S, Najafi A, Ranjbar R, Raam M. “GelClust: a software tool for gel electrophoresis images analysis and dendrogram generation”, *Comput Methods Programs Biomed.* (111):512–518 (2013)

14. Çabuk, U. C., “Gömülü Sistemler İçin Android Tabanlı Bir Mikroişlemci Programlama Yazılımı ve Arayüzü”, *Bilişim Teknolojileri Dergisi* ,(4):321-331 (2018)
15. İnternet: Raspberry Pi Foundation Uk Registered Charity, <https://www.raspberrypi.org/documentation/> (2019)
16. Kılıçdağı, A. “Raspberry Pi”, İstanbul, (2014)
17. İnternet: Raspberry Pi Foundation Uk Registered Charity, <https://www.raspberrypi.org/documentation/hardware/> (2019)
18. Gholamreza A., Ders Notları, “Image Processing”, *Universty of Tartu, Estonia*, (2019)
19. Buber, E., Şahingöz, O. K., ”Makine Öğrenmesi Sistemi ile Görüntü İşleme ve En Uygun Parametrelerin Ayarlanması”, *International Artificial Intelligence and Data Processing Symposium (IDAP)*, 2017
20. Özgül, F., “Her Yönüyle Python”, İstanbul, (2017)
21. İnternet: Python Software Foundation, <https://www.python.org/about/> (2019)
22. Kuyumcu, B., “OpenCV Görüntü İşleme ve Yapay Öğrenme”, İstanbul, 13 (2018)
23. İnternet: OpenCV Team, <https://opencv.org/about/> (2019)
24. Horgan, G.W. and Glasbey, C. A., “Uses of Digital Image Analysis in Electrophoresis”, *Electrophoresis*,(16):298-305 (1995)
25. Kusim, A.S., “An Image Processing Enhancement Approach to Extract Information in Gel Electrophoresis”, *International Conference on Technology, Informatics, Management, Engineering & Environment (TIME-E 2013)*, 9-13 (2013)
26. Meşe, A. K., “Tek Hücre Jel Elektroforezi için Görüntü Analizi”, *IEEE 2017 25th Signal Processing and Communications Applications Conference*, (2017)
27. Gonzalez, R.C. and Woods R. E., “Digital Image Processing”, New Jersey, 144-150, 161, 649-698, 760-763, (2008)
28. Chityala, R. And Pudipeddi, S., “Image Processing and Acquisition Using Python”, Minnesota,44, 99-103 (2014)
29. Goutsias, J. And Batman, S., “Morphological Methods for Biomedical Image Analysis”, Medical Imaging, Washington, 175-272 (2009)
30. Walker, J. M., “Chapter 20. Computer Analysis of 2D Electrophoresis Gels”, *New Protein Techniques*.
31. Bradski, G. and Kaehler, A. “Learning OpenCV”, Sebastopol, 115-124 (2008)

32. Fogel, I. and Sagi, D., “Gabor Filters as Texture Discriminator”, *Biological Cybernetics*, 61, 103–113 (1989)
33. Chen, C., “Multi-resolutional gabor filter in texture analysis”, *Pattern Recognition Letter 17*, 1069-1076 (1996)
34. İnternet: [https://medium.com/@anuj\\_shah/through-the-eyes-of-gabor-filter-17d1fdb3ac97](https://medium.com/@anuj_shah/through-the-eyes-of-gabor-filter-17d1fdb3ac97) (2019)
35. Minichino, J. And Howse, J., “Learning OpenCV 3 Computer Vision with Python Second Edition”, Birmingham, 152 (2015)
36. Wiley, J., “Clustering Using K-Means”, Python Machine Learning, First Edition, 221-242 (2019)
37. Akçıl M., Ders notları, “Veri Analizi”, *Başkent Üniversitesi*, (2019)
38. İnternet: <https://micam.soft112.com/download.html> (2019)
39. İnternet: [https://www.solisbiodyne.com/EN/products/dna\\_ladders/100\\_bp\\_dna\\_ladder\\_](https://www.solisbiodyne.com/EN/products/dna_ladders/100_bp_dna_ladder_) (2019)
40. Mersinkaya, İ., Kavsaoglu, A.R., Gömülü Sistem Tabanlı Mini Jel Elektroforez Sistemi, *IEEE 2018 26th. Signal Processing and Communications Applications Conference (SIU)*, İzmir, (2018)

## ÖZGEÇMİŞ

Kerim ÖZKARA 1983 yılında Aydın'da doğdu; ilk ve orta öğrenimini aynı şehirde tamamladı. Aydın Anadolu Ticaret Meslek Lisesi Bilgi-İşlem bölümünden 2001 yılında mezun oldu. 2003 yılında Pamukkale Üniversitesi Teknik Eğitim Fakültesi Bilgisayar Sistemleri Öğretmenliği Bölümü'nde öğrenime başlayıp 2007 yılında mezun oldu. 2010 yılında Gazi Üniversitesi Bilişim Enstitüsünde Bilişim Sistemleri ABD yüksek lisansa başlayıp 2013 yılında iyi dereceyle mezun oldu. 2013 yılında İstanbul Ataşehir Adıgüzel Meslek Yüksekokulunda Mekatronik Bölümünde öğretim görevlisi olarak göreve başladı. 2013 yılın Kasım ayında Sinop Üniversitesi Ayancık Meslek Yüksekokulu Elektronik ve Otomasyon bölümüne atanıp, halen aynı yerde görevine devam etmektedir.

### **ADRES BİLGİLERİ**

Adres : Sinop Üniversitesi  
Ayancık Meslek Yüksekokulu  
Yalı mah. Şehit Fatih ERER sk.  
Ayancık/SİNOP  
Telefon : (505) 640 5606  
E-posta : kerim.ozkara@sinop.edu.tr