

**DUYGU ANALİZİ VE YÜZ İFADESİ TANIMA İÇİN BİR DAĞITIK  
SİSTEMDE PARALEL PROGRAMLAMA KULLANILARAK  
YÜZ GÖRÜNTÜLERİNİN SINIFLANDIRILMASI**

**YÜKSEK LİSANS TEZİ**

**TİMUÇİN KORKMAZ**

**MERSİN ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**BİLGİSAYAR MÜHENDİSLİĞİ  
ANABİLİM DALI**

**MERSİN  
Temmuz - 2020**

**DUYGU ANALİZİ VE YÜZ İFADESİ TANIMA İÇİN BİR DAĞITIK  
SİSTEMDE PARALEL PROGRAMLAMA KULLANILARAK  
YÜZ GÖRÜNTÜLERİNİN SINIFLANDIRILMASI**

**YÜKSEK LİSANS TEZİ**

**TİMUÇİN KORKMAZ**

**MERSİN ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**BİLGİSAYAR MÜHENDİSLİĞİ  
ANABİLİM DALI**

**Danışman  
Prof. Dr. Hamza EROL**

**MERSİN  
Temmuz - 2020**

## ETİK BEYAN

Mersin Üniversitesi Lisansüstü Eğitim-Öğretim Yönetmeliğinde belirtilen kurallara uygun olarak hazırladığım bu tez çalışmada,

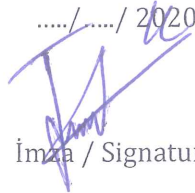
- Tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- Görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlâk kurallarına uygun olarak sunduğumu,
- Başkalarının eserlerinden yararlanması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- Atıfta bulunduğum eserlerin tümünü kaynak olarak kullandığımı,
- Kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- Bu tezin herhangi bir bölümünü Mersin Üniversitesi veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı,
- Tezin tüm telif haklarını Mersin Üniversitesi'ne devrettiğimi

beyan ederim.

## ETHICAL DECLARATION

This thesis is prepared in accordance with the rules specified in Mersin University Graduate Education Regulation and I declare to comply with the following conditions:

- I have obtained all the information and the documents of the thesis in accordance with the academic rules.
- I presented all the visual, auditory and written informations and results in accordance with scientific ethics.
- I refer in accordance with the norms of scientific works about the case of exploitation of others' works.
- I used all of the referred works as the references.
- I did not do any tampering in the used data.
- I did not present any part of this thesis as an another thesis at Mersin University or another university.
- I transfer all copyrights of this thesis to the Mersin University.

...../...../ 2020  


İmza / Signature

Timuçin KORKMAZ

Öğrenci Adı ve Soyadı / Student Name and Surname

## ÖZET

### **Duygu Analizi ve Yüz İfadesi Tanıma İçin Bir Dağıtık Sistemde Paralel Programlama Kullanılarak Yüz Görüntülerinin Sınıflandırılması**

Görüntü dosyalarından oluşan veritabanındaki büyük veriyi sınıflandırmak için yüksek performans bilgisayar sistemi gerekmektedir. Bu tez çalışmasında sabit (hard) diskte veritabanında kayıtlı yüz görüntü dosyaları yazılım hesaplama (soft computing) yöntemleriyle bir dağıtık sistem üzerinde paralel programlama kullanılarak işlenmiş ve sınıflandırılmıştır. Sınıflandırma işleminde duygu analizi yöntemleri ve yüz ifadesi tanıma teknikleri kullanılmıştır. Tasarlanan dağıtık bilgisayar sistemi ile yüksek performans hesaplama sistemi mimarisi ve ortamı oluşturulmuştur. Paralel programlama tekniği ile birim zaman içerisinde birden fazla işlem gerçekleştirilmiştir. Kullanılan dağıtık bilgisayar sistemi ile farklı bilgisayarlarda hesaplamalar yapılmıştır. Böylece duygu analizi ve yüz ifadesi tanıma için bir dağıtık sistemde paralel programlama kullanılarak yüz görüntülerinin sınıflandırılması gerçekleştirilmiştir. Tez çalışmasında farklı duygular içeren 58 görüntü yapılan testlerde yüksek başarı oranları elde edilmiştir. Dağıtık sistem üzerinde görüntüler paralel programlama yardımıyla ana bilgisayar ve diğer bilgisayar üzerinde paylaştırılarak zaman ve iş üzerinde kazanç sağlanmıştır. Yapılan çalışma sonucunda duyguların sınıflandırılmasında %91 doğruluk başarı oranı ve %95,30 F-ölçüsü başarı oranı sağlanmış, ayrıca görüntüler 0,673 saniyede işlenmiştir. Paralel programlama seri programlamaya göre %40 daha hızlı işlem yapmıştır.

**Anahtar Kelimeler:** Büyük veri, Dağıtık sistem, Duygu analizi, Görüntü sınıflandırma, Paralel programlama, Yüz ifadesi tanıma

**Danışman:** Prof.Dr. Hamza EROL, Mersin Üniversitesi, Bilgisayar Mühendisliği Anabilim Dalı, Mersin.

## ABSTRACT

### **Classification of Face Images Using Parallel Programming on a Distributed System For Emotion Analysis and Facial Expression Recognition**

A high performance computer system is required to classify large data in a database of image files. In this thesis, the face image files stored in the database on hard disk were processed and classified using parallel programming on a distributed system using soft computing methods. Emotion analysis methods and facial expression recognition techniques were used in the classification process. High performance computing system architecture and environment has been created with the distributed computer system designed. Multiple operations were performed in unit time with parallel programming technique. With the distributed computer system used, calculations were made on different computers. Thus, the classification of facial images was carried out using parallel programming in a distributed system for emotion analysis and facial expression recognition. High success rates were achieved in 58 images with different emotions in the thesis study. On the distributed system, the images are shared on the main computer and other computer with the help of parallel programming, allowing time and work to be saved. As a result of the study, 91% accuracy success rate and 95.30% F-measure success rate were achieved in the classification of emotions, and the images were processed in 0.673 seconds. Parallel programming is 40% faster than serial programming.

**Key Words:** Big data, Distributed System, Emotion analysis, Image classification, Parallel Programming, Face expression recognition

**Advisor:** Prof.Dr. Hamza EROL, Department of Computer Engineer, University of Mersin, Mersin.

## TEŞEKKÜR

Bu Yüksek Lisans Tez çalışması Mersin Üniversitesi Lisansüstü Eğitim-Öğretim Yönetmeliğinde belirtilen kurallara uygun olarak hazırlanmıştır. Tezde kullanılan bütün bilgi ve belgeler akademik kurallar çerçevesinde elde edilmiştir. Yüz ifadelerindeki değişikliklerin ya da değişimlerin analizinden yüz duygu sınıflandırma araştırılmıştır. Veritabanındaki yüz görüntü dosyalarının bir dağıtık sistem üzerinde paralel programlama kullanılarak sınıflandırması gerçekleştirilmiştir. Bu tez çalışmasında veritabanındaki görüntü dosyalarının sınıflandırılması için bir Yüksek Performans Hesaplama Sistemi oluşturulmuştur. Bu araştırmaların yapılması sırasında destek olan başta ailem olmak üzere, Yüksek Lisans Tez danışmanım Prof.Dr. Hamza EROL'a teşekkürlerimi sunarım.



## İÇİNDEKİLER

	Sayfa
İÇ KAPAK	ii
ONAY	iii
ETİK BEYAN	v
ÖZET	vi
ABSTRACT	vii
TEŞEKKÜR	viii
İÇİNDEKİLER	ix
TABLolar DİZİNİ	xi
ŞEKİLLER DİZİNİ	xii
KISALTMALAR ve SİMGELER	xiv
<b>1. GİRİŞ</b>	<b>1</b>
1.1. Yazılım Hesaplama	1
1.2. Paralel Programlama	3
1.3. Büyük Veri	4
1.3.1. Büyük Veri Analitiği	6
1.4. Dağıtık Bilgisayar Sistemi	7
1.4.1. Dağıtık Bilgisayar Sisteminin Yapısı ve Gereksinimleri	8
1.5. Duygu Analizi	9
1.5.1. Duygu Analizinde Yapılan İncelemeler	10
1.5.2. Duygu Analizinin Kullanıldığı Alanlar	10
<b>2. DUYGU ANALİZİ VE YÜZ İFADESİ TANIMA KULLANILARAK GÖRÜNTÜ SINIFLANDIRMA ALANINDA YAPILAN ÇALIŞMALAR</b>	<b>12</b>
2.1. Yüz Görüntülerinin İşlenmesi ve Sınıflandırılması	12
2.1.1. Gabor Filtresi	12
2.1.2. Yüz Mimik Kodlama Sistemi	14
2.1.3. Yüzlerdeki Durumu Sınıflandırma Tekniği	16
2.1.3.1. Bütünsel Analiz	16
2.1.3.2. Özellik Ölçme	17
2.1.3.3. Optik Akış	18
2.2. Yüz Görüntülerinde Yüz Dinamiklerinin Gösterimi ve Analizi	18
2.2.1. 3D Model Tabanlı İzleme	19
2.2.1.1. Deforme Edilebilir Yüz Modeli	19
2.2.1.2. Yüz Görüntülerinde Hareketli T Sıralaması	20
2.2.1.3. Yüz Görüntülerinde Tek Gözlü T Sıralaması	20
2.2.2. Yüz Gösteriminde İkili Taban Metodu	21
2.2.2.1. Yüz Gösteriminde Yerel İkili Kalıplar	21
2.2.2.2. Lineer Programlama Tekniği ile Yüz Gösterimi	23
2.3. Optimize Edilmiş Özellik Seçimi Kullanılarak Elde Edilen Yüz Görüntü Dizileri	23
2.3.1. Yüz Görüntüsü Veritabanı	24
2.3.2. Yüz Görüntüsü Ön İşleme	24
2.3.3. Yüz İfadesi Tanıma	26
2.3.3.1. Yüz İfadesi Tanıma Bulanık Yaklaşım	26
2.3.3.2. Bulanık Yaklaşım Kullanan Yüz Tanıma Sistemi	26
2.3.3.3. Yüz Özelliklerinin Belirlenmesi	26
2.3.3.4. Sınıflandırma	28
2.4. Yüz Algılama ve Konumlandırma Kenar Çizgi Belirleme Algoritmaları	28
2.4.1. Yüz Tespiti ve Lokasyonu	29
2.4.2. Yüz Görüntülerinde Ağız ve Göz Pozisyonları	29
2.4.3. Yüz Tanıma Sisteminde Kenar Algılama	30
2.4.3.1. Canny Kenar Algılama	30
2.4.3.1.1. EM Algoritması	31

	<b>Sayfa</b>
2.4.3.2. Laplace Kenar Algılama	33
2.4.3.3. Sobel Kenar Algılama	34
2.4.3.4. Robert Kenar Algılama	34
<b>3. MATERYAL ve YÖNTEM</b>	<b>36</b>
3.1. Dağıtık Bilgisayar Mimarisinde C# İle Paralel Programlama	36
3.2. Dağıtık Bilgisayar Mimarisinde İş Parçacıkları Oluşturulması	37
3.3. Dağıtık Bilgisayar Mimarisinde Çok İş Parçacıklı (Multi-Threaded) Programlama	39
3.3.1. Yüz Algılama	41
3.3.2. Veri Çıkarma	41
3.3.3. Yüz Görüntülerini İşleme	41
3.3.4. Yüz İfadelerinin Sınıflandırılması	41
3.4. MATLAB (Matrix Laboratory) ile Yüz Görüntüsü İşlenmesi	42
3.5. Dağıtık Bilgisayar Sisteminde MATLAB ve Paralel C# Programlama Kullanılarak Yüz Görüntülerinin Sınıflandırılması	44
3.5.1. Dağıtık Bilgisayar Sisteminde MATLAB ile Yüz Görüntülerinin İşlenmesi	44
3.5.2. Entity Framework ile Paralel C# Programlama Kullanılarak Yüz Görüntülerinin Sınıflandırılması	47
<b>4. BULGULAR ve TARTIŞMA</b>	<b>62</b>
4.1. Ruhsal Etkenlerin Duygu Analizi Üzerine Etkisi	62
4.2. Çevresel Etkenlerin Duygu Analizi Üzerine Etkisi	63
4.3. Duygu Analizinde Görüntü İşleme	64
4.4. Yüz İfadelerinin Sınıflandırılması	66
4.5. Yüz İfadelerinin Sınıflandırılmasında Doğruluk	70
4.6. Kullanılan Görüntünün Özellikleri	71
4.7. Seri ve Paralel Programlama	72
<b>5. SONUÇLAR ve ÖNERİLER</b>	<b>77</b>
KAYNAKLAR	78
ÖZGEÇMİŞ	81

## **TABLÖLAR DİZİNİ**

Tablo 2.1. Kenar Algılama Yöntemlerine Göre Elde Edilen Sonuçlar	34
Tablo 3.1. MATLAB Formül ve Kısaltmaları	42
Tablo 3.2. Uygulama Duygu Sınıflandırma Doğruluk Oranları Tablosu	61
Tablo 4.1. Confusion Matris	73



## ŞEKİLLER DİZİNİ

Şekil 1.1. Örnek Görüntü Akış Diyagramı	2
Şekil 1.2. Yüksek Seviye Dillerin Makine Diline Çevrilmesi	3
Şekil 1.3. Örnek Paralel Programlama Yöntem Gösterimi	4
Şekil 1.4. Büyük Veri İncelemede Kontrol Edilen Özellikler	5
Şekil 1.5. Dağıtık Bilgisayar Sistemi	7
Şekil 1.6. Örnek Dağıtık Sistem Şeması	9
Şekil 1.7. Duygu Analizlerinde Elde Edilen Duygu Halleri	11
Şekil 2.1. Gabor Filtre Bankası Sistemi	14
Şekil 2.2. FACS Kişideki Değişimlerin Analizi	15
Şekil 2.3. Kişilerden Alınan Farklı Yüz İfadelerindeki Örnekler	16
Şekil 2.4. İleri Beslemeli Ağın Mimarisi	17
Şekil 2.5. Kişinin Yüzündeki Kırışıklıklar ve Düzleştirilmiş Piksel Yoğunluğu	17
Şekil 2.6. Duygu Durumuna Göre Yüzdeki Değişimler	19
Şekil 2.7. Kişinin Yüzü ve Modelleme	19
Şekil 2.8. Kişinin 5 Farklı Resmi ve Vektör Çizimleri	20
Şekil 2.9. Yerel İkili Örüntü Operatörü	21
Şekil 2.10. Orijinal Resimler ve Ön İşlemede Özellik Çıkarma İşlemi ile Elde Edilen Görüntü Parçaları	22
Şekil 2.11. Görüntü Histogramı	22
Şekil 2.12. Kullanılan Görüntüler ve Elde Edilen Değerler	23
Şekil 2.13. CohnKanade Veri Tabanında Yer Alan Örnek Görüntü Dizisi Seti	24
Şekil 2.14. Görüntü Ön İşleme Yöntemi İşlem Akışı	25
Şekil 2.15. Görüntünün Yüz Tanımadaki Kenar Çıktısı	26
Şekil 2.16. Yüz Görüntüsünde Gözlerin Belirlenmesi ve Kırılması İşleminin Çıktısı	27
Şekil 2.17. Yüz Görüntüsünde Ağız Açıklık Belirlenmesi	27
Şekil 2.18. Kenar Algılama Algoritmaları Uygulanmış Görüntüler	30
Şekil 2.19. Piksel 8 Komşu Değerin Diyagramı	31
Şekil 2.20. EM Algoritması İşlem Akış Diyagramı	33
Şekil 3.1. Senkron ve Asenkron Programlamanın Gösterimi	36
Şekil 3.2. C# Paralel Programlama Örneği	38
Şekil 3.3. Sıralı İşlem ile Paralel İşlem Karşılaştırılması	39
Şekil 3.4. Tek İş Parçacığı (Thread) ve Çoklu İş Parçacığı (Multi Thread)	40
Şekil 3.5. MATLAB Elde Edilen Değerlerin Bir Bölümü	43
Şekil 3.6. Resmin Bir Bölümünün RGB Değer Gösterimi	44
Şekil 3.7. MATLAB Uygulamasına Gönderilen Görüntü	45
Şekil 3.8. Görüntünün RGB Çıktısı	46
Şekil 3.9. Entity Framework İş Akışı	47
Şekil 3.10. Entity Framework Veri Tabanı Bağlantısı	48
Şekil 3.11. Görüntülerin İşleme Alındığı Kod Satırları	49
Şekil 3.12. Paralel Programlama ile İşleme Alınan Görüntüler	50
Şekil 3.13. Görüntünün Özellikleri	50
Şekil 3.14. Sobel Filtreleme İşlemi	51
Şekil 3.15. Kullanılan Test Fotoğrafı	52
Şekil 3.16. Fotoğraftaki Sol Göz	52
Şekil 3.17. Fotoğraftaki Sağ Göz	52
Şekil 3.18. Göz Belirleme İşlemi	53
Şekil 3.19. Fotoğraftaki Ağız Şekli	53
Şekil 3.20. Ağızın Şeklini Bulmayı Sağlayan Kod Kısmı	54
Şekil 3.21. Fotoğraftaki Duygunun Sonucu	55
Şekil 3.22. Normal Yüz İfadesi	55
Şekil 3.23. Üzgün Yüz İfadesi	56
Şekil 3.24. Mutlu Yüz İfadesi	56

Şekil 3.25. Şaşırılmış Yüz İfadesi	57
Şekil 3.26. Kızgın Yüz İfadesi	57
Şekil 3.27. Dağıtık Sistem Mimarisinde Bağlantı Kontrolü	59
Şekil 3.28. Uygulama Bağlantı Gösterimi	59
Şekil 3.29. Görüntü ve Duygu İşleme	60
Şekil 3.30. Uygulama İşlem Bilgileri	60
Şekil 4.1. Farklı Seslerin Yüz İfadesinde Meydana Getirdiği Değişiklikler	63
Şekil 4.2. Görüntü İşleme Akış Şeması	64
Şekil 4.3. Farklı Duygularla Yüzdeki Değişimler	65
Şekil 4.4. Mutluluk Duygu Halinde Yüzdeki Değişimler	67
Şekil 4.5. Üzgün Duygu Halinde Yüzdeki Değişimler	67
Şekil 4.6. Korku Duygu Halinde Yüzdeki Değişimler	68
Şekil 4.7. Kızgın Duygu Halinde Yüzdeki Değişimler	68
Şekil 4.8. Şaşırma Duygu Halinde Yüzdeki Değişimler	69
Şekil 4.9. İğrenme Duygu Halinde Yüzdeki Değişimler	69
Şekil 4.10. Yüz Sınıflandırma Tekniklerinin Doğruluk Başarı Oranları	74
Şekil 4.11. Yüz Sınıflandırma Tekniklerinin F-Ölçüsü Başarı Oranları	74
Şekil 4.12. Sobel Filtre İçeren Sistemlerin Doğruluk Başarı Oranları	75
Şekil 4.13. Sobel Filtre İçeren Sistemlerin F-Ölçüsü Başarı Oranları	75
Şekil 4.14. Paralel Programlama Kullanılan Farklı Teknikler ve Süreleri	76

## **KISALTMALAR ve SİMGELER**

---

<b>Kısaltma/Simge</b>	<b>Tanım</b>
CPU	Merkezi İşlem Birimi
FACS	Yüz Eylem Kodlama Sistemi
CFA	Yüz Eylemini Sınıflandırma
UI	Kullanıcı Arabirimi
RGB	Renk Modeli(kırmızı-yeşil-mavi)
SN	Saniye



## 1. GİRİŞ

Bu yüksek lisans tezinde büyük veri analitiği çerçevesinde duygu analizi ve yüz ifadesi tanıma için sabit disk üzerindeki veritabanında kayıtlı görüntü dosyalarının bir dağıtık bilgisayar sistemindeki işlemcilerde paralel programlama kullanılarak oluşturulan yüksek performans hesaplama sisteminde yazılım hesaplama (soft computing) ile sınıflandırılması araştırılmıştır.

### 1.1. Yazılım Hesaplama

Gelişen teknoloji nedeniyle günümüzde görüntü kaydetme ve bu kaydedilen görüntülerin işlenebilmesiyle farklı türde bilgi elde etme ihtiyacı günden güne artmaktadır. Bu bilgilerin elde edilebilmesi için gereken yazılımların oluşturulması günden güne zorlaşmaktadır. Yenilenen teknolojiye de ayak uydurması gereken bu yazılımlar oluşturulurken birçok farklı teknik kullanılmaktadır. Kullanılan bu teknikler sistemlerin daha farklı alanlarda daha kapsamlı bilgi elde edilmesine olanak sağlamaktadır.

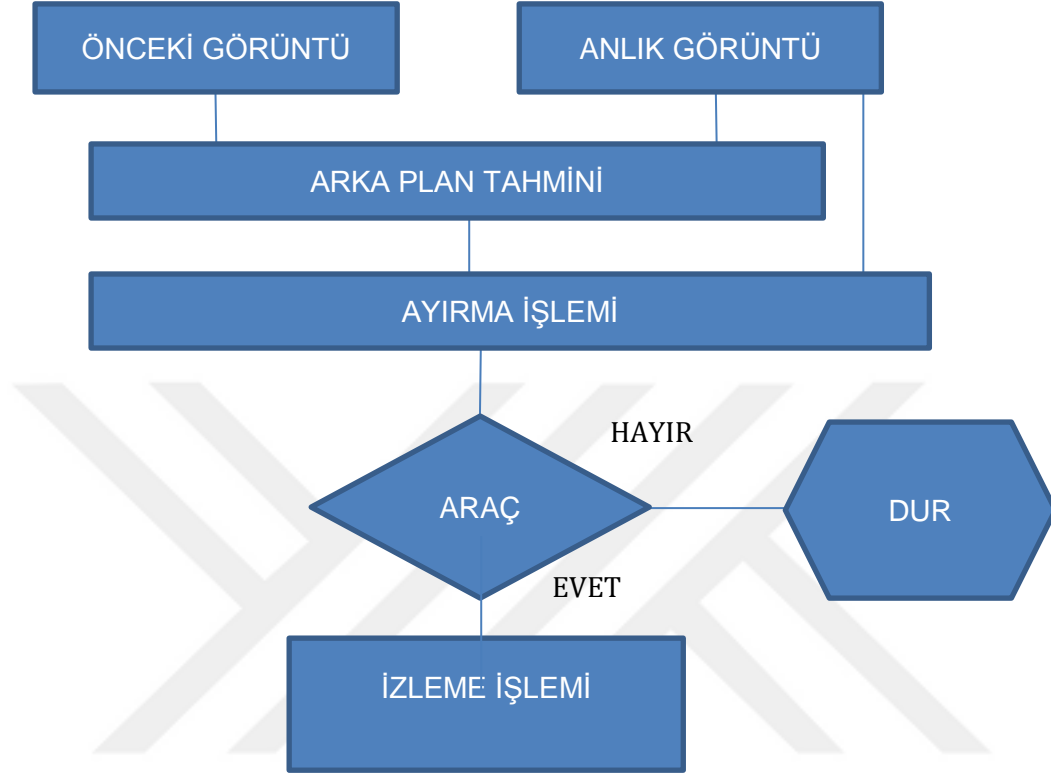
Yazılımlar oluşturulurken programlamadan yararlanılmaktadır. Programlama, günlük hayattaki karşılan problemlerin çözülmesi için kullanılacak makinelerin anlayacağı biçimde anlatma, problemin çözüm yollarının makinelere öğretilmesi, anlatılması için kullanılan, belirli yazım kuralları olan teknik terimler bütününe verilen genel isimdir.

Günlük hayatta karşılan problemlerin makineler bilgisayarlar yardımıyla çözülebilmesi için öncelikle makineye anlatmak gerekmektedir. Programlama konuyu makinanın anlayabileceği şekilde anlatma biçimidir. Programlama problemin tespiti ile başlayıp ürün ya da isteğin yerine getirilmesine kadar ki aşamaların bütününe verilen genel isimdir.

Bilgisayar tabanlı programlama yapabilmek için farklı teknolojiler takip edilebilir. Bu teknolojilerin sağladığı çözüm yollarıyla istenilen sonuçlara en kolay yoldan ulaşmak sağlanabilir. Programlamanın temelini algoritma oluşturmaktadır. Algoritma bir problemin çözümünde ilerlenecek yol haritasının akış şema simgeleri ile anlatılmasına denir. Problemin çözümünün temeli doğru algoritmanın kurulmasıdır [1].

Örnek görüntü verisinin işlenmesi ve görüntüden bilgi elde edilebilmesi için oluşturulacak algoritmanın akış diyagramı Şekil 1.1'de gösterilmiştir.

Örnek görüntü verisinin işlenmesi ve görüntüden bilgi elde edilebilmesi için programlama yapılırken teknolojilere göre farklı şekilde ifade biçimleri oluşturulmuştur. Bu ifade biçimleri programlama dillerini oluşmasını sağlamıştır.



**Şekil 1.1.** Örnek Görüntü Akış Diyagramı

Programcı, programlama problemlerinin çözümünü oluşturmak amacıyla algoritma hazırlar. Programlama dili programcının algoritmayı ifade etmek için kullandığı teknolojiye yönelik yazdığı komut satırlarıdır. Programcı farklı programlama dilleri kullanabilir. Farklı teknolojiler farklı dillerin oluşmasını ve gelişmesine olanak tanımıştır. Programlama dilleri verinin programcının isteği doğrultusunda işlenip programcının kullanabileceği şekilde dönüştürür, farklı koşullarda hangi işlemlerin yapılacağını anlatmasını sağlar.

Programlama dilleri insanların anlayabilme kolaylığına göre 3 gruba ayrılmıştır [2]. Birinci grup alt seviye programlama dillerinden oluşur. Alt seviye programlama dilleri makine koduna yakın bir dil kullanırken makine üzerinde hakimiyeti yüksek bir dildir. Bu dil genellikle mikro işlemcilerin programlamasında kullanılmaktadır. İkinci grup orta seviye programlama dillerinden oluşur. Orta seviye programlama dilleri anlaşılabilirlik açısından alt seviye dillere göre biraz daha gelişmiş dillerdir. Üçüncü grup üst seviye programlama dillerinden oluşur. Üst seviye programlama

dilleri nesne tabanlı diller olarak adlandırılmaktadır. Hızlı ve etkili diller olmasının yanında öğrenilmesi diğer seviye dillere göre daha kolaydır. Fakat bu diller makinenin anlayabileceği dillerden değildir. Derleyici ve yorumlayıcı yapıları kullanırlar.

Üst seviye programlama dillerinin makineler tarafından algılanıp istenen komutu yerine getirebilmesi için makine diline çevrilmesi gerekmektedir. Bu çevirme işlemi yazılan programlama dilinin derleyici ve yorumlayıcısı yerine getirmektedir. Derleyici ve yorumlayıcılar üst seviye dili makinenin anlayabileceği şekle dönüştürerek makinenin yapacağı işi anlamasını sağlamaktadır [2]. Yüksek seviyeli dillerin makine dillerine çevrilmesi Şekil 1.2'de gösterilmiştir.



Şekil 1.2. Yüksek Seviye Dillerin Makine Diline Çevrilmesi [2]

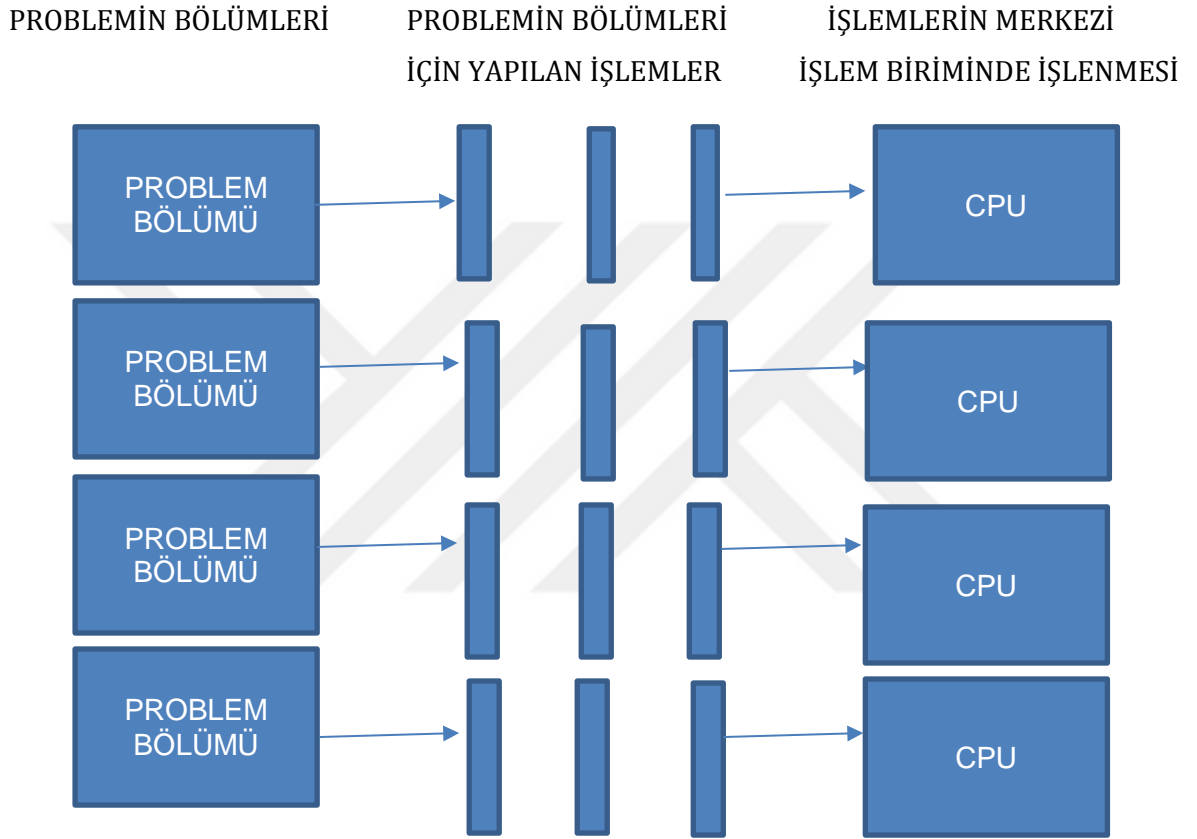
## 1.2. Paralel Programlama

Görüntülerden yeni bilgiler elde edilebilmesi için verilerin hızlı bir şekilde işlenebilmesi ve işlem sonuçlarının hızlı bir şekilde yorumlanması gerekmektedir. Bu işlemleri yaparken sisteminde işlemlerin gerçekleşmesinde süreklilik sağlaması gerekmektedir. Bir programın seri hesaplamalar şeklinde, bir işlemin ardından diğerinin başlaması olarak devam eden ve işlemlerin bir işlemci üzerinde birim zamanda bir komut işlenmesiyle oluşan programa paralel programlama denir. Paralel programlama bir problem çözümünde birçok hesaplamanın yapılarak sonucun elde edildiği programlama türüdür [3]. Paralel programlamanın sistematığı Şekil 1.3 'te gösterilmiştir.

Bilgisayar sistemleri tasarlanırken paralel programlama kullanılmasının birçok tercih nedeni bulunmaktadır. Bunlardan başlıcaları;

- i. Bilgisayarlarda bir işe daha fazla kaynak ayrıldığında işlem daha kısa süreceğinden zamandan tasarruf sağlar.
- ii. Süper bilgisayarlara göre paralel bilgisayarlar maliyet açısından daha düşük maliyetli olduğundan tercih edilirler.

- iii. Büyük problemlerin çözümünde çok sayıda bilgisayar olması çözümü olanaklı kılmaktadır.
- iv. Çok sayıda bilgisayar olması birim zamanda farklı bilgisayarlarda farklı işlemler yapılabilir.
- v. Bilgisayarlar network ağıyla farklı lokasyonlardaki bilgisayarların kaynaklarını kullanabilmektedir [3].



Şekil 1.3. Örnek Paralel Programlama Yöntem Gösterimi

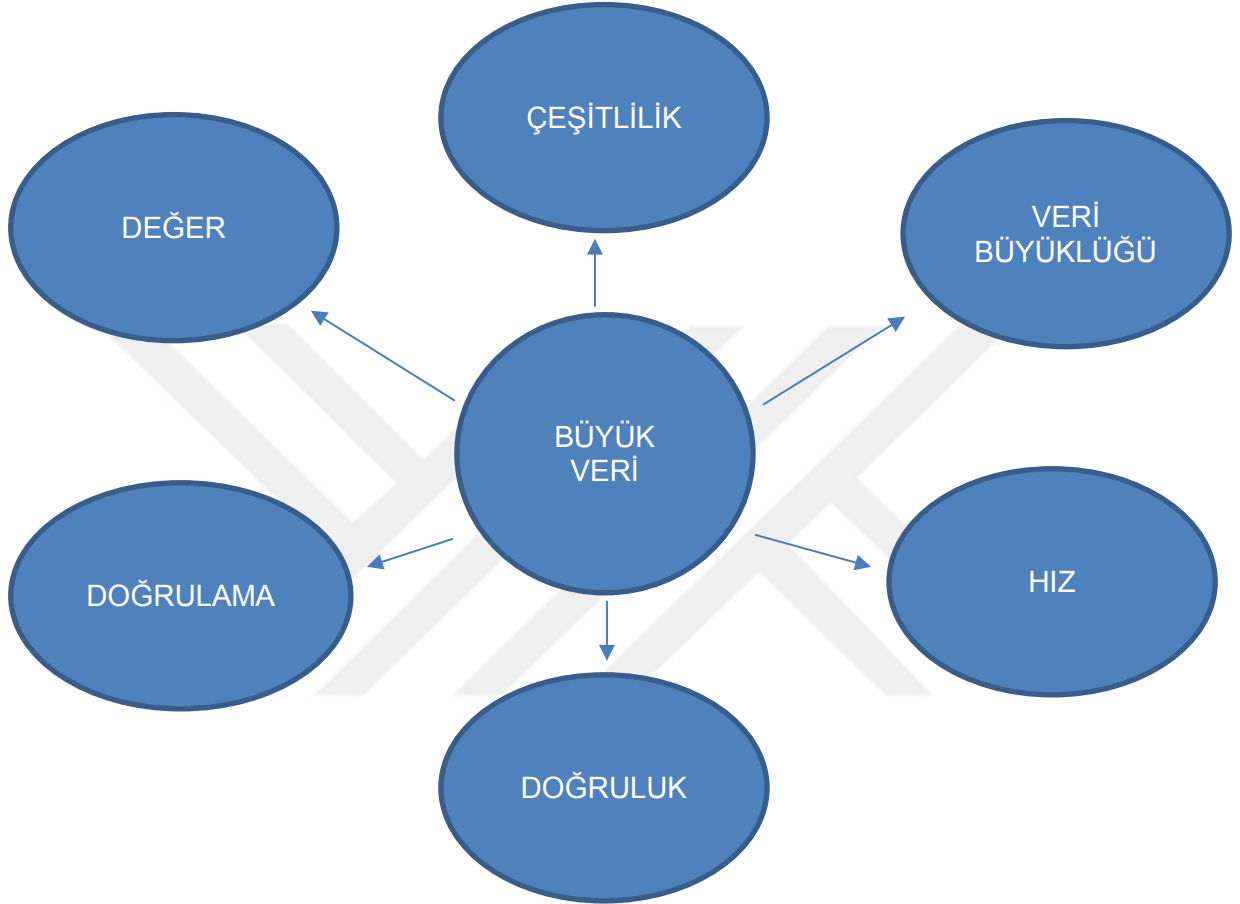
### 1.3. Büyük Veri

Görüntülerden anlamlı bilgi çıkarmak için geliştirilen yazılımlarda çeşitli algoritmalar kullanılmaktadır. Bu algoritmalar içinde AR-GE çalışmaları yapılmaktadır. Bu yapılan çalışmalarda anlamlı bilgi elde edilmesi için yapılan işlemler bütününe Büyük Veri denilmektedir. Büyük veri katmanlı veridir. Büyük veride yapılan işlemlerde kontrol edilen özellikler;

- i. Veri büyüklüğü
- ii. Veriyi işleme hızı
- iii. Verinin doğruluğu

- iv. Verinin doğrulanması
- v. Verideki gözlem değerleri
- vi. Verideki çeşitlilik

şeklinde. Büyük veride yapılan işlemlerde kontrol edilen özellikler Şekil 1.4'te gösterilmiştir.



**Şekil 1.4.** Büyük Veri İncelemede Kontrol Edilen Özellikler [4]

Büyük Veri diskte çok yer kaplayan işlenmemiş veri yığınlarıdır [5]. Büyük veri, yeni ve popüler bir kavram olmasına rağmen, ulusal literatürde kuramsal nitelikte çok fazla araştırma bulunmamaktadır. Büyük verinin kuramsal yönünü işleyen az sayıda yayın olmakla birlikte, teknoloji ile ilgili pek çok kurum ve kuruluş bu konuya son derece önem vermektedir. Büyük verinin analitiği için çok büyük yatırımlar yapmaktadırlar. Bu tez çalışmada büyük veri kavramsal olarak ele alınmış, pek çok kavramla olan ilişkisi, büyük veri teknolojileri ve büyük veri işlenirken kullanılan yöntemler incelenmiştir. Büyük veri ile ilgili farkındalık yaratan kuruluşlarla ve dünyada büyük verinin kullanım alanları ile ilgili farklı örnekler ve uygulamalar verilmiştir.[6]

### 1.3.1. Büyük Veri Analitiği

Büyük veri analitiği veri analizinden ve veri modellemeden oluşmaktadır.

**Veri analizi:** Veri analizi, veriden ihtiyaç duyulan yararlı bilgileri keşfetmek, sonuç çıkarmak ve karar vermeyi desteklemek amacıyla uygulanır. Verileri incelemek, temizlemek, dönüştürmek ve modellemek veri analizinde kullanılan bir süreçlerdir. Veri analizi, farklı iş, bilim ve sosyal bilim alanlarında çeşitli isimler altında çeşitli teknikleri kapsayan çok yönlü ve farklı yaklaşımlara sahiptir [30].

**Veri İşleme:** Başlangıçta elde edilen veriler analiz için işlenmeli ve düzenlenmelidir. Alınan veriler, bir elektronik tablolama veya istatistiksel yazılım gibi daha ileri analizler için bir tablo formatında satırlara ve sütunlara yerleştirilir.

**Kestirimsel modelleme:** Öğrenme analitiğinin bir amacı da kestirimsel modellemedir. Kestirimsel modellemede matematiksel bir model geliştirilir. Bu model çerçevesinde muhtemel sonuçlar tahmin edilir. Bu model aynı zamanda sonuçları geliştirmek için yapılacak uygulamalarda kullanılır [31].

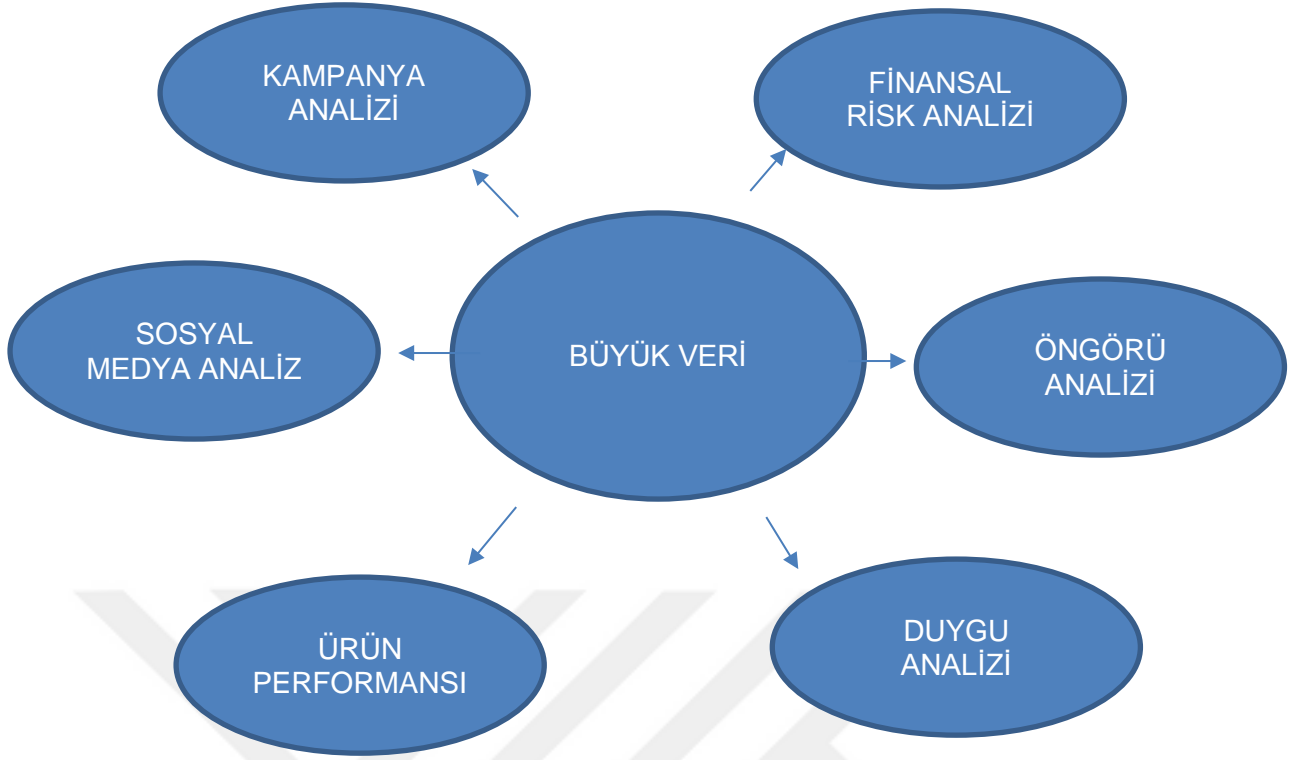
**Sosyal ağ analizi:** Sosyal ağ analizi birçok disiplinden ve teknikten faydalanan bir yöntemler bütünüdür. Buna göre sosyal ağ analizinde düğümler ve bu düğümler arasında gerçekleşen ilişkiler incelenir [7].

**Kullanım/kullanıcı takibi:** Öğrenme yönetim sistemleri ve sanal öğrenme ortamlarında gerçekleşen öğrenen etkinliklerinden veya bilgisayar üzerinde gerçekleşen her türlü uygulamadan toplanan veri öğrenme analitiği için kullanılır [32].

**İçerik analizi ve semantik analiz:** Öğrenenler tarafından üretilen nitel veri toplanır. Doğal dil işleme (natural language processing) süreci ve örtük semantik analiz (latent semantic analysis) teknikleri kullanılarak öğrenenler tarafından üretilen nitel veri zengin ve anlamlı bir biçimde analiz edilir [33].

**Öneri motoru:** Öneri motorları bireylere birçok kullanıcının benzer davranışlarını inceleyerek bireylere ilgilendikleri konulara dair önerilerde bulunan bilişim tabanlı araçlardır [7].

Büyük veri analitiğinin bazı uygulama alanları Şekil 1.5'te gösterilmiştir.



Şekil 1.5. Büyük Veri Analitiğinin Uygulama Alanları [8].

#### 1.4. Dağıtık Bilgisayar Sistemi

Bu tez çalışmasında sistem tasarımı yapılırken oluşturulacak yapıda dağıtık sistem kullanılması planlanmaktadır. Dağıtık sistem, birden fazla bilgisayar ya da sunucunun aralarında bağlantı kurarak iletişim sağlamasıyla bir ağ bütünü şeklinde çalışması işlemine verilen isimlendirmedir [9].

Dağıtık sistem bir ağa kurulmaktadır. Yazılımla ağdaki cihazlar arasında uyumluluk sağlanır. Kullanıcı açısından kullanım kolaylığı oluşturulur. Dağıtık sistemin diğer ağ yapılarından farkı donanım değil yazılımsal olarak yapılan değişikliklerdir. Dağıtık sistem için oluşturulan bu ağ yapısı kullanılan yazılım vasıtasıyla tek bir bilgisayar gibi davranır. Ayrıca birden çok cihaz olduğundan performans açısından çok güçlüdür [9].

Dağıtık sistemlerin en önemli amacı kaynakları paylaşmaya duyulan ihtiyaçtır. Bu kaynaklar donanımsal olabileceği gibi yazılım, belge, veri tabanları, dosyalar, dosyalama sistemleri olabilmektedir. Dağıtık sistemlerin bazı özellikleri aşağıda belirtilmiştir.

1. Dağıtık sistem bütünleşik bir sistemdir. İzole değildir.
2. Bulut yapısının temelini oluşturmaktadır.
3. Senkron çalışmaktadır. İş bölümü vardır.
4. Yedekli olarak çalışmaktadır.

Dağıtık sistemler birden fazla cihazdan oluştuğu için cihaz sayısı kadar Merkezi Yönetim Birimi (CPU)'dan oluşur. Kullanıcı bunu tek CPU gibi görmektedir. Dağıtık sistemin kullanıcıları çalıştırdığı programların ve dosyaların nerede tutulduğu hakkında bilgisi yoktur. Bu tür işlemler işletim sisteminin kontrolündedir [9].

#### **1.4.1. Dağıtık Bilgisayar Sisteminin Yapısı ve Gereksinimleri**

Dağıtık sistemlerdeki cihazların birbirleriyle senkron ve uyumlu bir şekilde çalışabilmesi için bazı gereksinimler bulunmaktadır [9]. Bu gereksinimler aşağıda belirtilmiştir.

1. İşletim Sistemi
2. Donanım mimarileri
3. İletişim mimarileri
4. Programlama dilleri
5. Yazılım ara yüzleri
6. Güvenlik ölçüleri
7. Bilgi gösterimleri

Bu özellikler dağıtık sistemlerdeki cihazların senkron bir şekilde çalışması için başlıca gereksinimlerdir. Dağıtık sistemlerin avantaj ve dezavantajları bulunmaktadır. Bunlar aşağıda maddeler halinde belirtilmiştir.

#### **Avantajları:**

- i. Maliyetin düşük olması
- ii. Erişim kolaylığı - Hız performansı
- iii. Hesaplama ve depolama alanı
- iv. Ölçeklenebilir olması
- v. Güvenilir olması
- vi. Dağıtım kolaylığı
- vii. Güvenlikli olması

#### **Dezavantajları:**

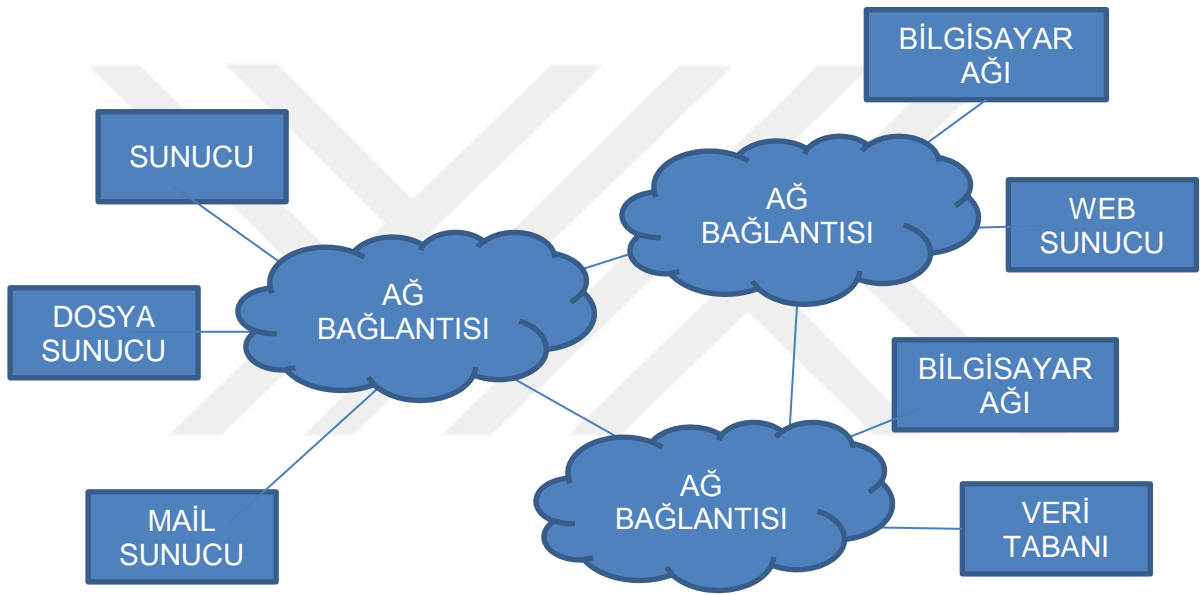
- i. Merkezi yapı zorunluluğu

- ii. Her dilde yazım desteğinin olmaması
- iii. İletişimdeki zorluklar

Bir dağıtık sistem için aşağıdaki kurallar geçerlidir.

1. Kaynaklar ve sistemler ağ üzerinde dağıtık olduğu için yeterince şeffaf değildir.
2. Yazılımların kendine has kuralları vardır. Bu yazılımların kendi sentaks yapıları olmasından dolayı sistemler yeterince açık değildir.

Bir dağıtık sistem mimarisi ve bileşenleri Şekil 1.6'da gösterilmiştir.



Şekil 1.6. Örnek Dağıtık Sistem Şeması [9]

### 1.5. Duygu Analizi

Bu tezde dağıtık sistemler kullanılarak görüntülerden duygu analizi yapılacaktır. Duygu analizi bir konu hakkında toplumda yapılan araştırmaların sonucunun konu hakkında toplumdan edinilen bilginin olumlu ya da olumsuz olup olmadığı konusunda bilgi veren analizlerdir. Duygu analizleri günümüzde çok kullanılmaktadır. Örneğin bir marka hakkında konuşan kitlenin marka hakkındaki düşüncesinin olumlu mu olumsuz mu olduğu konusunda yapılan analizdir [10].

### 1.5.1. Duygu Analizinde Yapılan İncelemeler

Duygu analizinde yapılan işlemler aşağıda belirtilmiştir [10].

- i. **Öznellik Sınıflandırması:** Metinde yapılan incelemelerle metnin herhangi bir fikir içerip içermediğine bakılır.
- ii. **Duygusal Sınıflandırma:** Seçilen bir cümlenin olumlu, olumsuz veya nötr olması durumları değerlendirilir.
- iii. **Fikrin yardımcı olma ihtimali:** Fikir içeren metnin incelemesi yapılarak kişiye fikir oranında ne kadar yardımcı olacağı belirlenir.
- iv. **İstenmeyen fikir taraması:** Bir fikrin kötü amaçla yazılması durumunu incelemektedir. Bu durum yönlendirme amacı olarak kullanılabilir. Örneğin; bir markanın ürün reklamında marka kendi ürününü öne çıkarmaktadır.
- v. **Fikir Özetleme:** Fikir içeren metnin fikir veya fikirlerini kısa bir şekilde ifade edilebilmesi için yapılır. Örneğin; makalelerde bulunan anahtar kelimelerin çıkarılması bu yöntemle yapılmaktadır.
- vi. **Karşılaştırmalı fikirlerin çıkarılması:** İki veya daha fazla fikir içeren karşılaştırılmalı fikirlerde, fikirlerin hangi yönlerden karşılaştırıldığı ve hangi fikrin diğerine oranla durumunun belirtilmesidir.

Duygu analizinin kullanıldığı bir alan fikir madenciliğidir [10]. Fikir Madenciliğinde;

- i. Bakış tabanlı fikir madenciliği teknikleri kullanılmaktadır.
- ii. Frekans veya ilişki tabanlı fikir madenciliği teknikleri incelenmektedir.
- iii. Model tabanlı fikir madenciliği teknikleri uygulanmaktadır.

### 1.5.2. Duygu Analizinin Kullanıldığı Alanlar

Duygu Analizi günümüzde birçok alanda kullanılmaya başlanmış bir analiz çeşididir. Duygu Analizleri günümüzde daha çok sosyal medya verilerinden yararlanılarak yapılmaktadır. Duygu analizinde analiz edilmek istenen veri büyük veriler arasından ayıklanarak ortaya çıkarılmaktadır. Duygu Analizi;

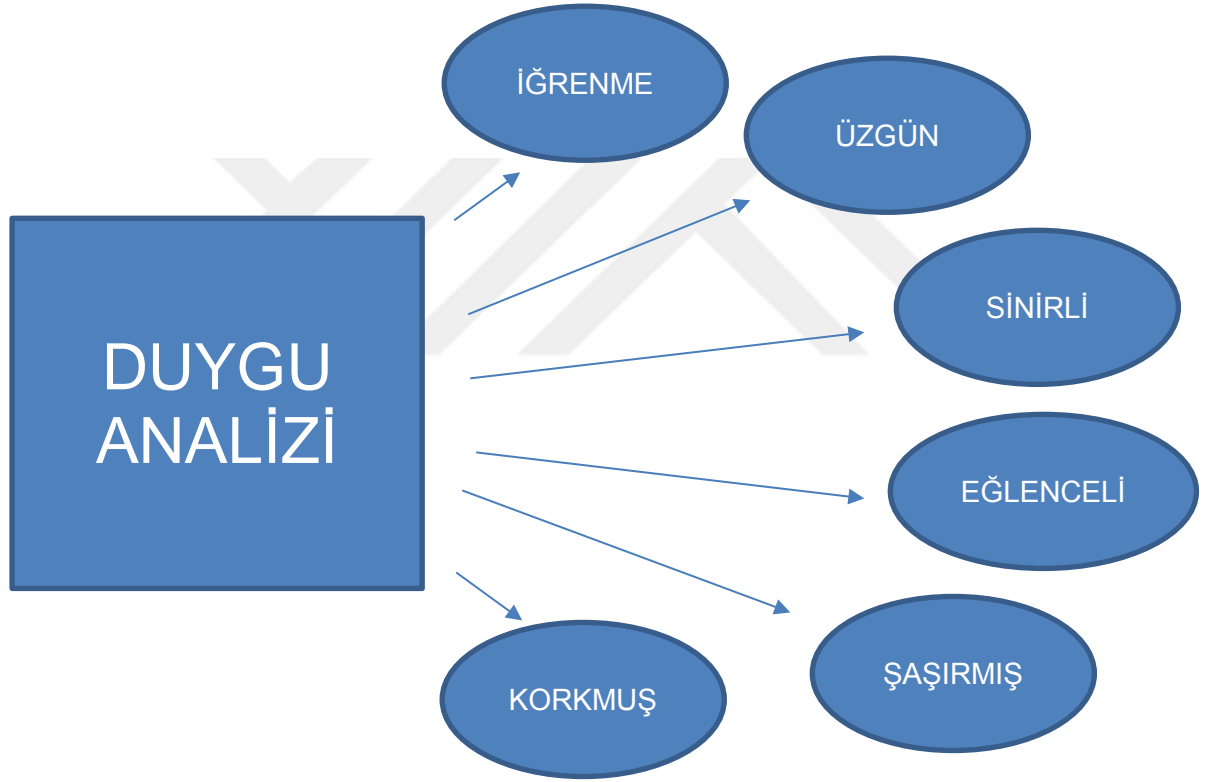
- i. Analiz yapılan konunun sosyal medya üzerinde ne kadar konuşulduğu hakkında bilgi edinilmesini sağlar.
- ii. Elde edilen veriler kriz anlarında kullanılarak krizin gittiği noktanın tahmin edilebilmesini sağlar.
- iii. Yapılan bir değişikliğin toplumda yaptığı etkiye bakılarak olumlu/olumsuz sonuçlarının çıkarılmasını kolaylaştırır.

Duygu analizinde ana duygu halleri 6 tanedir. Bunlar;

1. İğrenme
2. Üzgün
3. Sınırlı
4. Eğlenceli
5. Şaşırmış
6. Korkmuş

şeklindedir.

Duygu analizinde tanımlanan ana duygular Şekil 1.7’de gösterilmiştir.



Şekil 1.7. Duygu Analizlerinde Elde Edilen Duygu Halleri [11]

## 2. DUYGU ANALİZİ VE YÜZ İFADESİ TANIMA KULLANILARAK GÖRÜNTÜ SINIFLANDIRMA ALANINDA YAPILAN ÇALIŞMALAR

Duygu analizi konusu üzerine birçok çalışma yapılmıştır. Bu çalışmaların her birinde farklı teknikler, algoritmalar ve yöntemler kullanılmıştır. Kullanılan tekniğe göre farklı sonuçlar ve bilgiler elde edilmiştir.

### 2.1. Yüz Görüntülerinin İşlenmesi ve Sınıflandırılması

Bu bölümde yüz görüntüsü işlemede kullanılan terimler ve kavramlar açıklanacaktır.

#### 2.1.1. Gabor Filtresi

Gabor filtresi görüntü işlemede kullanılan yöntemlerden biridir. Gabor filtresi sayesinde görüntüde yanlara doğru uzayan nesnelere fark edilebilmektedir. Görüntüde bulunan ayrıntılar tespit edilmektedir. Bu özelliği sayesinde gabor filtresi günümüzde yüz tanıma, plaka tanıma, iris tanıma gibi birçok alanda kullanılmaktadır. Gabor filtresi 2.1 ve 2.2 denklemleri ile

$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \cos\left(2\pi \frac{x'}{\lambda} + \psi\right) \quad (2.1)$$

$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \sin\left(2\pi \frac{x'}{\lambda} + \psi\right) \quad (2.2)$$

ifade edilmektedir [12]. Burada x ve y fonksiyonlardaki değişkenleri;  $\lambda, \theta, \Psi, \sigma, \gamma$  fonksiyonlardaki parametreleri göstermektedir.

**X ve Y değerleri:** Görüntü için çekirdek matrisin hesaplaması sırasında kullanılmaktadır. Çekirdek matris genellikle n\*n boyutunda kare matris olmaktadır. Kare matris  $[-(n-1)/2, (n-1)/2]$  kapalı aralığında x ve y değerleri verilerek elde edilmektedir.

**X' ve Y' değerleri:** Bu değerler yukarıda bulunan matrisin dönüşüm matrisi yardımıyla transpozunun alınmasıyla elde edilmektedir.

**Yönelim Açısı (Teta):** Bu açı tespit edilmek istenen nesnenin yönünü belirlemek amacıyla kullanılmaktadır. Açı değeri değiştirildikçe farklı yön ve doğrultulardaki nesnelere tespit edilebilmesi sağlanmaktadır [12].

**Standart Sapma (Sigma):** Fonksiyonda yer alan 2. çarpan gaussian fonksiyonu olduğundan bu değer fonksiyonun standart sapma değeridir. Bu değer band genişliği ile dalga boyu parametreleri kullanılarak hesaplanan bir değerdir.

**Dalga Boyu(Lambda):** Kosinüs çarpanının dalga boyunu belirleyen değerdir. Birimi pikseldir. Bu parametreye en az 2 değeri verilmelidir.

**Standart Sapma ve Band genişliği ile dal boyu ilişki:** Band genişliği sıfırdan büyük bir sayıdır. Düşük band genişliği, büyük standart sapmaya sebep olmaktadır. Büyük standart sapma da Filtreli görüntüde paralel şeritlerin sayısının artmasına sebep olmaktadır [12].

Bant genişliği;

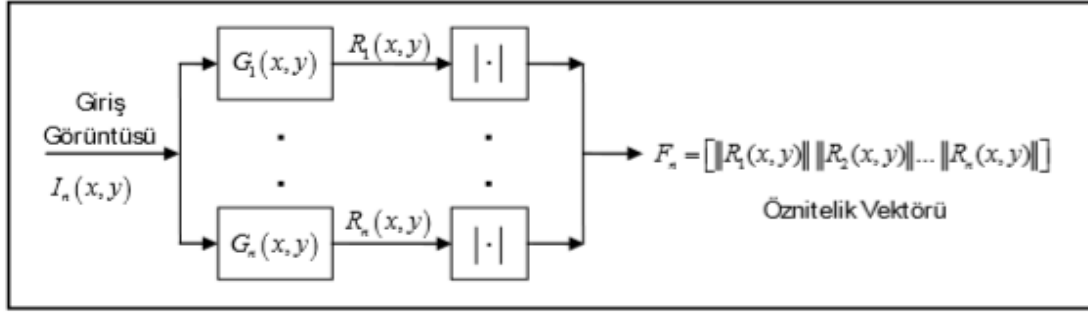
$$Band\ Genişliği = \log_2 \frac{\frac{\theta}{\lambda} \pi + \sqrt{\frac{\ln 2}{z}}}{\frac{\theta}{\lambda} \pi - \sqrt{\frac{\ln 2}{z}}} \quad (2.3)$$

2.3'teki formül ile tanımlanır. Bu üç değer arasında formüldeki gibi bir bağlantı bulunmaktadır. Genelme yapılacak olursa bant genişliği 1 kabul edilmektedir. Böylece standart sapma, dalga boyu ve band genişliği değerleri kullanılarak sigma değeri  $0,56 \cdot \lambda$  olarak elde edilmiş olur.

**En/Boy Oranı(Gama):** Gaussian fonksiyonunun çıktısı çan eğrisi biçiminde olmaktadır. En/boy oranı eğrinin enin boyuna oranlanması ile elde edilmektedir. Değerin değişmesi elde edilen nesnelerin boyutlarını değiştirmektedir.

**Faz Açısı(Fi):** Bu değer kosinüs çarpanının faz açısı olarak hesaplanır. Bu değer kosinüs açısının aldığı değerlere göre değişkenlik göstermektedir. Kosinüs değeri 0 ve 180 olduğu durumlarda gaussian eğrisi merkezde yer almaktadır. Faz açıları grafiğin x eksenine üzerinde hareket etmesini sağlamaktadır. Ayrıca değer eğrinin y eksenine asimetric olmasını da sağlamaktadır.

Gabor filtresinde değişik açı ve frekans değerleri kullanılarak değişik özellikler içeren filtre bankası oluşturmak mümkündür. Bu durum Şekil 2.1'de gösterilmiştir.



Şekil 2.1. Gabor Filtre Bankası Sistemi [12]

### 2.1.2. Yüz Mimik Kodlama Sistemi

Yüz tanıma sistemlerinde kullanılan sistemlerden biri de Mimik Kodlama Sistemi (Facial Actions Coding Systems) sistem kısa adıyla FACS'tır. FACS, Carl-Herman Hjortsjö tarafından geliştirilmiş bir kodlama sistemidir. FACS sistemi insan yüz hareketlerini yüzündeki görünümüyle kıyaslaması üzerine yapılmıştır [13].

FACS sistemi kullanılarak insan yüzleri manuel olarak kodlanabilmektedir. FACS sistemi kodlamayı işlem birimlerine aktarabilmektedir. Sistem gelen veriyi işleyerek önceden tanımlanmış olan karar verme yapılarına göre işlemektedir. FACS sistemi örneğin kaslarda meydana gelen kasılma ve gevşemeleri kontrol ederek yapılan hareketin türünü ayırt etmekte kullanılabilir. FACS sistemi kas hareketlerinde değişiminin kullanımına örnek vermek istersek iki gülümsemeyi karşılaştırabiliriz. Buna göre:

- i. Samimiyetsiz ve zorlanarak yapılan bir kişinin gülümsemesi sırasında zigomatik marjör kası kasılır.
- ii. Samimi ve farkında olmadan yapılan bir gülümsemede ise göz kapaklarına etki eden kasın yanı sıra zigomatik marjör kası kasılır.

Buradan yapılan gülümsemenin durumu, yüz kaslarında kasılmalar incelenerek durumun incelenmesine örnek oluşturmaktadır [13].

FACS sistemi kodlamasında işlemler belirli bir numaraya dayandırılarak ve buna karşılık gelen kas grubu yazılarak tanımlanmıştır. Burada tanımlanan her bir numara bir hareketi anlatmakta ve bu hareketin yapılması sırasında kullanılan kas veya kas grupları belirtilmektedir. Verilen örnek samimi bir gülümsemedeki değişimlerle, samimiyetsiz bir gülümsemedeki değişimlerin aynı olmadığı anlatılmıştır.

FACS sistemi ayrıca değişimlerdeki farklarda sıralama yapmıştır. Buna göre değişimlerdeki fark sıralamasını Küçük bir iz, hafif iz, belirgin değişiklik vb. gibi yoğunluk farkları şeklinde yapmıştır. Bu sıralamaları A ile E arasında harflerle belirterek yoğunlukları belirtmiştir. 1A zayıf bir iz iken, 1E bir kişideki en belirgin değişiklik olarak belirlenmiştir. Diğer harflerdeki değişimlere göre bu seviyedeki değişikliğin yoğunluğu büyüktür. Bu durum Şekil 2.2’de gösterilmiştir.



**Şekil 2.2.** FACS Kişideki Değişimlerin Analizi [14]

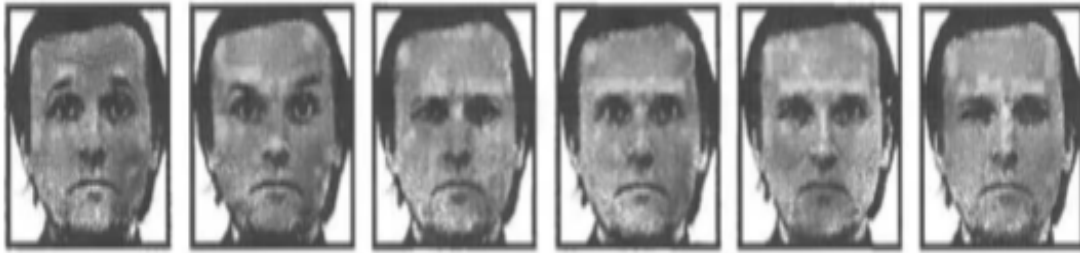
Şekil 2.2’de görüldüğü gibi kişinin yüzünde belirli değişimler gerçekleşmiş, meydana gelen değişiklikler belirginlik seviyelerine göre sıralanmıştır. Mimik kodlama sisteminde kullanılan özellikler:

- i. İç kaş kaldırma
- ii. Dış kaş kaldırma
- iii. Alın düşürme
- iv. Göz kapağı yükseltme
- v. Göz kapağı sıkıştırma
- vi. Dudak gerilmesi
- vii. Çene inmesi

şeklinde tanımlanır.

### 2.1.3. Yüzlerdeki Durumu Sınıflandırma Tekniği

Ekman ve Friesen tarafından geliştirilmiş olan FACS sistemi yüzdeki kas kasılmalarından yola çıkarak kişinin yüz ifadesini anlamaya yönelik yapılmıştır. Bu bölümde tanımlanan yüzlerdeki durumu sınıflandırma (Classifying Facial Action - CFA) sistemde ise yüz eylemlerinden yola çıkılarak yapılan eylem tespit edilir ve bu eylemler sınıflandırılarak yüz ifadesini bulmaya yönelik bir tekniktir [15]. CFA sisteminin hazırlanması sırasında 24 kişinin oluşturduğu bir denek grubu oluşturulmuştur. Bu denek grubundan 150'den fazla yüz eylemi alınmış, 1100'den fazla görüntü dizisiyle bir veri tabanı oluşturulmuştur. Kişilerden alınan farklı yüz ifadelerindeki örnekler Şekil 2.3'te gösterilmiştir.



Şekil 2.3. Kişilerden Alınan Farklı Yüz İfadelerindeki Örnekler [15]

Kişilerin yüz örnekleri alınırken en az 6 farklı yüz görüntüsü alınmıştır. Şekil 2.3'te görüleceği gibi kişinin bakışlarında değişiklikler oluşmuş bu da kaşlarında, yüz şekillerinde değişikliklere neden olmuştur.

#### 2.1.3.1. Bütünsel Analiz

Bütünsel analiz (Holistic Spatial Analysis - HSA) sistemi, geri yayılım tarafından eğitilmiş ileri beslemeli ağlarla elde edilen temel bileşenlere dayanan bütünsel gösterim şeklidir. Yapılan çalışmalarda sistem cinsiyetleri yüz görüntüleriyle başarılı bir şekilde sınıflandırabileceğini ortaya koymuştur.

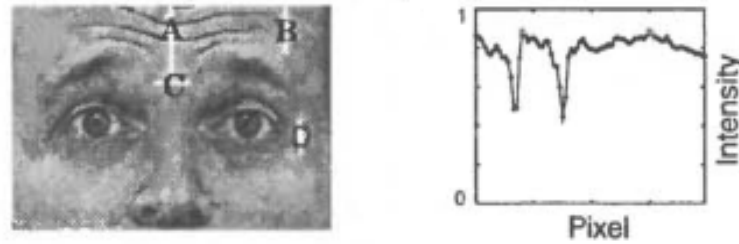
HSA sistemi 10 gizli birim içeren 3 katmalı bir sinir ağı ve 6 birim işlem içeren işlemlerden oluşmaktadır. Ağ ileri beslemeli olarak beslenmekte ve bir hiperbolik tanjant fonksiyonu ile bağlantılı bir sistemle eğitilmiştir. HSA sistemi 50 bileşen içeren bir veri setiyle test edilmiş ve %88,6 doğruluk vermiştir [15]. İleri beslemeli ağın mimarisi Şekil 2.4'de gösterilmiştir.



**Şekil 2.4.** İleri Beslemeli Ağın Mimarisi [15]

### 2.1.3.2. Özellik Ölçme

Veri setiyle yapılan deneylerde kişinin farklı durumlarda yüzündeki değişiklikleri, kasların kasılmasıyla oluşan kırışıklık yüz tanınmasında önemli bir özelliktir. Yapılan işlemlerde aynı kişinin farklı yüz pozisyonlarında kırışıklıkların daha belirginleşmesi, yüz hatlarının ortaya çıkması gibi sonuçlar elde edilmiştir. Başlangıçta belirlenen ana görüntüyle en fazla değişikliğin olduğu görüntü arasındaki değer farkları ölçümlerle belirlenmiştir [15]. Bu belirlenirken düzeltilmiş piksel yoğunluğu grafiği de kullanılır. Kişinin yüzündeki kırışıklıklar ve düzeltilmiş piksel yoğunluğu grafiği Şekil 2.5'te gösterilmiştir.



**Şekil 2.5.** Kişinin Yüzündeki Kırışıklıklar ve Düzleştirilmiş Piksel Yoğunluğu [15]

Kişinin yüzündeki kırışık oranı 2.4' teki

$$P = \sum(I_i - I_{i-1})^2 \quad (2.4)$$

formülü ile hesaplanmıştır. Bu eşitlikte P değeri pikseli, I değeri derinliği ifade etmektedir.

### 2.1.3.3. Optik Akış

Bu işlem yalnızca yüz hareketinin düzenini temel alarak sınıflandırmaya çalışır. Hareket, ana görüntü ile karşılaştırılacak görüntü çiftlerinden çıkarılır. Görüntülerde parlaklık ölçümleri için parlaklık kısıtlama denklemine göre görüntü noktasındaki hız hesaplanır. Hızlar;

$$\vartheta_x \frac{\partial I(x,y,t)}{\partial x} + \vartheta_y \frac{\partial I(x,y,t)}{\partial y} + \frac{\partial I(x,y,t)}{\partial t} = 0 \quad (2.5)$$

2.5 denklemine göre hesaplanır.

Yukarıda 2.1.3 bölümünde ele alınan özellik birleştirilerek oluşan sistemde çalışmalar yapılmıştır. Yapılan çalışmada kullanılan veri seti düşük ve orta dereceli kas faaliyeti olan bir veri setidir. Yapılan çalışmada özellikler ayrı ayrı kullanılarak elde edilen doğruluklar sırasıyla %89, %57 ve %85 olarak hesaplanmıştır. Fakat 3 sistemin ortak kullanılmasıyla %92'lik bir doğruluk oranı hesaplanmıştır [15].

## 2.2. Yüz Görüntülerinde Yüz Dinamiklerinin Gösterimi ve Analizi

Duygu analizi sisteminin temelini oluşturan düşünce kafa takibi ile başlamaktadır. Kişinin kafa takibi ile yüzünün özellik takibi yapılabilmektedir. Kafa takibinde iki önemli özellik bulunmaktadır. Bunlar;

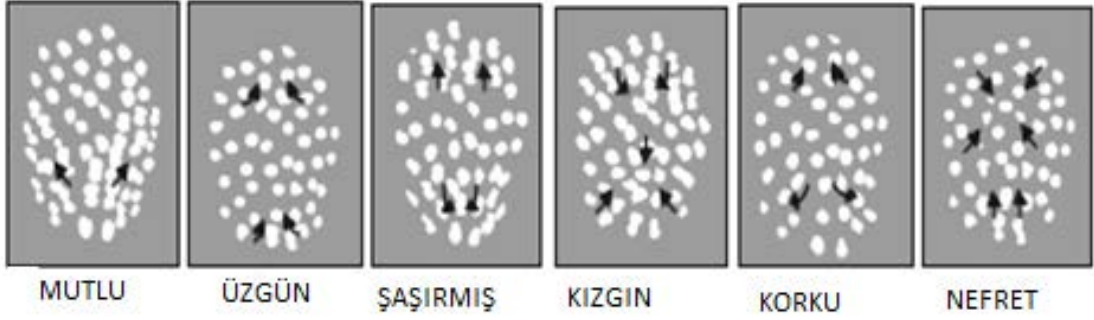
- i. Özellik sınırları belirleme
- ii. Özellik bölge takibi

şeklinde.

**Özellik Sınırları Belirleme:** Yüz özelliklerinden şekillerin izlenerek doğru bir şekilde ayırma işlemini yapmaya çalışır.

**Özellik Bölge Takibi:** Yüzü çevreleyen sınırları baz alarak yüzdeki durumların incelenmesine denilmektedir.

Yüz özellikleri konuşma sırasında alt çenenin hareket etmesinden dolayı birkaç tane yüz ifadesi oluşturmaktadır. Buradaki duygu analizi sisteminin ana fikri kişinin yüzünde oluşan ifade değişikliklerine göre ifadenin yüzde gerçekleştirdiği yönüne uygun olarak optik akış hesaplaması kullanılmıştır. Şekil 2.6'da bazı duygu durumlarında yüzde gerçekleşen değişimler oklar ile gösterilmiştir [16].



Şekil 2.6. Duygu Durumuna Göre Yüzdeki Değişimler (Basili) [16]

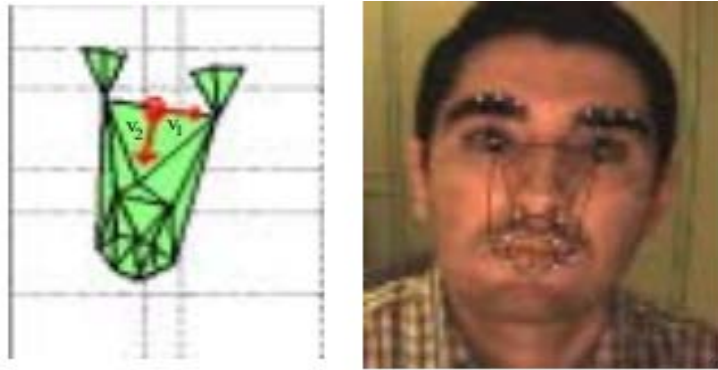
Psikolojide yapılan araştırmalara göre en az 6 ana duygunun yüzde değişim yaptığı kanaatine varılmıştır. Bu altı ana duygu; mutluluk, üzüntü, sürpriz, korku, öfke ve iğrenme olarak kabul edilmiştir. Şekil 2.6'da da bu duygular gösterilmiştir. Basili [16] tarafından yapılan işlemler temel yüz hareketlerini belirlemekte önemli bir rol oynamıştır. Basilinın yaptığı bu çalışmada yüzdedeki değişimlerin yönleri gösterilerek duygu tanımlama işlemi gerçekleştirilmiştir.

### 2.2.1. 3D Model Tabanlı Yüz İzleme

Bu bölümde 3D Model Tabanlı Yüz İzleme Yöntemi incelenecektir.

#### 2.2.1.1. Deforme Edilebilir Yüz Modeli

N toplam çerçeve değerini göstermek üzere Şekil 2.7'deki yüz görüntüsü üzerinde N=19 olarak belirlenmiştir. Buradaki yüz referans çevresini toplam yüz çevresi olarak kabul edilmektedir [17]. Kişinin yüz modeli Şekil 2.7'de gösterilmiştir.



B: Kişinin Yüz Modeli

A: Kişinin Yüzü

Şekil 2.7. Kişinin Yüzü ve Modelleme [17]

Duygu analizi sisteminde hesaplamalar yapılırken yüzler arasındaki geometrik şekil farkı 0 iken, çizilen vektörlerin değerleri yaklaşık olarak 1'dir. Sistemin değerleri hesaplanırken

$$\vec{X} = \vec{X}_0^u + \sum_{k=1}^p a_k \vec{X}_k \quad (2.6)$$

2.6 formülü kullanılmaktadır. Burada p nokta sayısı, X değerleri ise koordinat vektörleridir.

### 2.2.1.2. Yüz Görüntülerinde Hareketli T Sıralaması

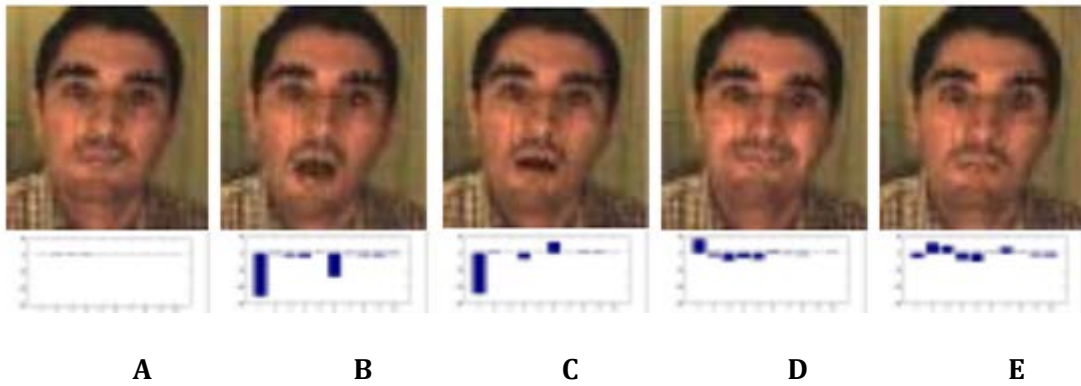
Hareketli T sıralaması (Stereo T Ranking) sisteminde verilen stereodaki temel hareket vektörleri tahmin edilebilmektedir. Kullanıcının yüz resminden Şekil 2.7 de görüldüğü gibi 19 noktadan oluşan stereo izlemeden akış içeren bir dizi veri elde edilmiştir. Stereo izlemeden sonra x vektörlerinin değerleri 3D şekil yörünge matrisinin Tekil Değer Ayrışımı (Singular Value Decomposition - SVD) kullanılarak hesaplanabilir. Bu şekilde farklı kişilerin yüzlerinin görüntüleri alınarak kullanılmak üzere benzer değerler hesaplanır. Hesaplanan değerler nötr değerinden çıkarılarak  $X \sim u$  verisi elde edilir. Bu veriler 2.7'deki

$$\vec{M} = [\widetilde{X}^1(1) \quad \widetilde{X}^1(2) \quad \dots \quad \widetilde{X}^1(N_1) \quad \widetilde{X}^2(1) \quad \dots \quad \widetilde{X}^K(N_K)] \quad (2.7)$$

matrisini elde etmek için kullanılır.

### 2.2.1.3. Yüz Görüntülerinde Tek Gözlü T Sıralaması

Tek Gözlü T Sıralaması (Monocular T Ranking) sistemi optik akış takibi art arda gelen iki görüntüdeki bir noktanın yer değiştirmesi işlemidir. Bu işlem 3B model aracılığıyla tüm noktalar parametrikleştirilerek birleştirilir. Model yapılandırılmasıyla görüntü tahmini gerçekleştirilir [17]. Kişinin 5 farklı resmi ve vektör çizimleri Şekil 2.8'de gösterilmiştir.



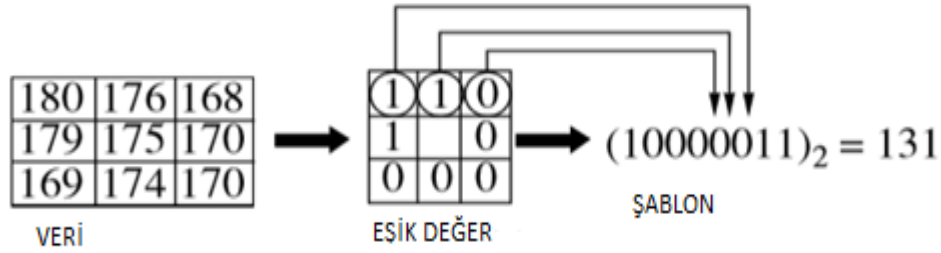
Şekil 2.8. Kişinin 5 Farklı Resmi ve Vektör Çizimleri [17]

Tek gözlü T sıralaması sisteminin en büyük katkısı başın hareke etmesine rağmen 3D modellemenin yapılabilmesidir. 3D izleme sayesinde yüz ifade tanıma sistemi gelişmiştir. Sistemin yapımında kullanılan destek vektör makinelerinin de güçlü sınıflandırma yapabildiği görülmüştür. Şekil 2.8'de yapılan (normal görüntü, ağız açma/kapama, kaş yükseltme ve gülümseme) hareketlerinden %91 başarı sağlanmıştır [17].

## 2.2.2. Yüz Gösteriminde İkili Taban Metodu

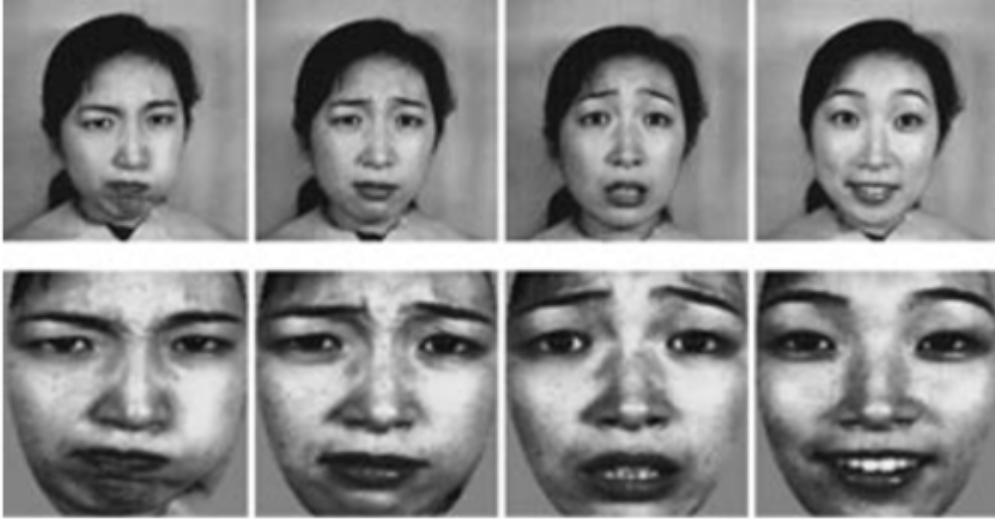
### 2.2.2.1. Yüz Gösteriminde Yerel İkili Kalıplar

Bu kısımda yerel ikili kalıplarla yüz gösterimi ele alınacaktır. Yerel ikili örüntünün şematik hali Şekil 2.9'da gösterilmektedir. Üçlü komşu değer merkez piksel değeri ile eşiklenerek bir ikili model kodu üretilmesi sağlanır [18]. Şekil 2.9'da veri eşik değeri, şablon, üçlü komşu değeri ve üretilen ikili model kodu yer almaktadır.



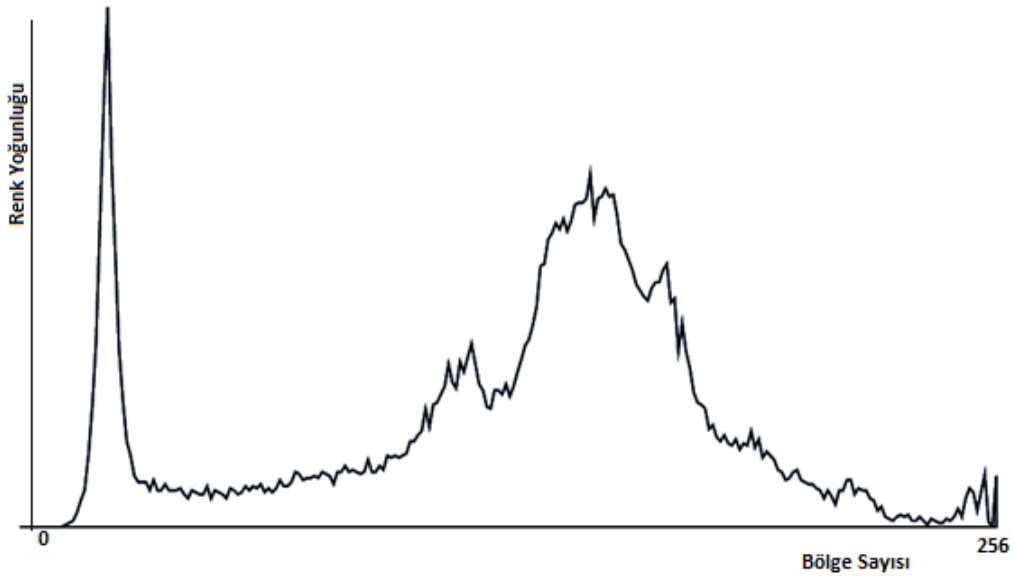
Şekil 2.9. Yerel İkili Örüntü Operatörü

Veri bölümünde yer alan komşu merkez pikselin kodu ikili sisteme verilerek ikili sistem kodu elde edilir. İşlenen operatörün değerine bağlı olarak her piksel bu kodla etiketlenir. Etiketlerin 256 bölmeli histogramı elde edilerek belirlenen bölgenin görüntüsü elde edilmiş olur. Özellik çıkarma işlemi Şekil 2.10'da gösterildiği gibi uygulanmaktadır. Yüz görüntüsü birbirleriyle örtüşmeyen parçalara ayrılma işlemi yapılır.



**Şekil 2.10.** Orijinal Resimler ve Ön İşlemede Özellik Çıkarma İşlemi ile Elde Edilen Görüntü Parçaları [18]

Görüntüler elde edilirken tüm histogramların elde edilmesinin ardından hepsi birleştirilerek bir bütün haline getirilir. Daha iyi görüntü ortaya çıkarmak için bir eşik ortalama değeri bulunur. Bu değer tüm histogramlardan edilen değerlerin ortalamasıdır. Bu değer altında kalan desenler atılarak daha iyi bir görüntü elde edilir. Görüntülerin histogramları Şekil 2.11'de gösterilmiştir.



**Şekil 2.11.** Görüntülerin Histogramı

### 2.2.2.2. Linear Programlama Tekniği İle Yüz Gösterimi

Linear programlama ile yüz gösterimi oluşturulurken iki noktaya ait ortalama değerlerin toplamında oluşan hataları en aza indirmek için bir doğrusal programlama formülü kullanılmaktadır. Bunun için A ve B olmak üzere iki matris tanımlanmıştır. A matrisinin boyutları  $m * n$ , B matrisinin boyutları  $k * n$ 'dir [18]. A ve B matrislerinden hataları azaltmak için 2.8'deki

$$\begin{cases} -A\omega + e\gamma + e \leq y \\ B\omega - e\gamma + e \leq z \\ y \geq 0, z \geq 0. \end{cases} \quad (2.8)$$

denklemler ile elde edilir. Elde edilen denklemdeki değerler oluşturacak görüntüdeki hata payını en aza indirmek için hesaplanmış eşitliklerdir. Bu eşitlikler kullanılarak farklı sınıflandırma grupları elde edilir. Elde edilen sınıflandırma yöntemlerinde duygular ayırt edilmeye çalışılmıştır. İşlemler yapılırken 7 farklı duygu durumu (öfkeli, korkmuş, şaşkın, iğrenmiş, mutlu, üzüntülü ve normal) kullanılmıştır. Şekil 2.12'de örnek görüntü ve duygular gösterilmiştir.



**Şekil 2.12.** Kullanılan Görüntüler ve Elde Edilen Değerler [18]

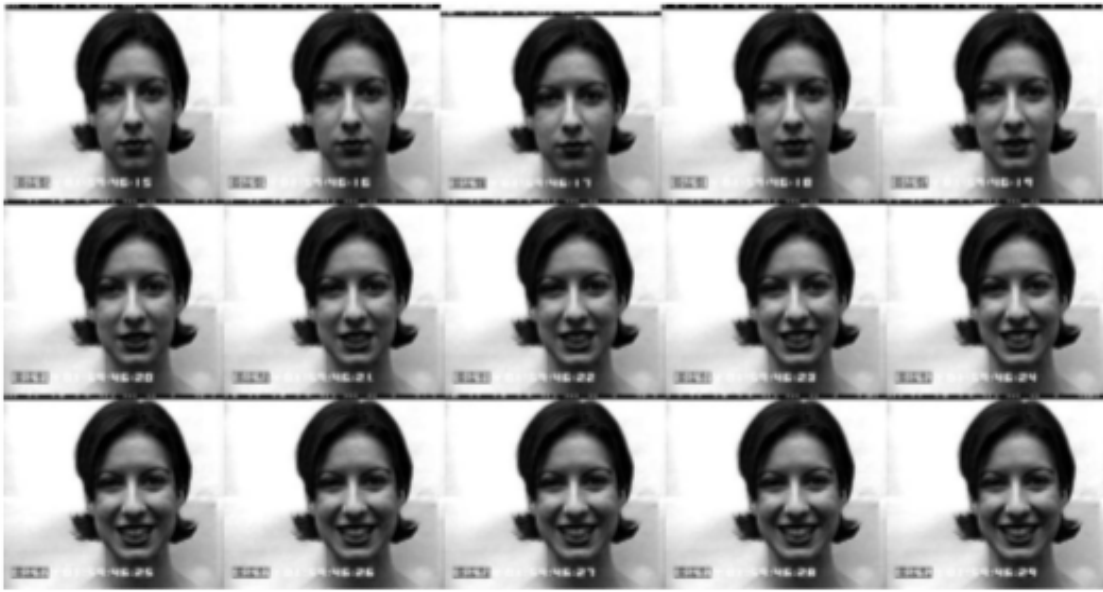
İşlemler sonucunda birbirinden farklı 21 sınıflandırma metodu elde edilmiş. Elde edilen metotlarla yukarıda belirtilen duygu durumlarına yönelik işlemler gerçekleştirilmiştir. Yapılan çalışmalar sonucunda ortalama olarak %93,8'lik bir başarı elde edilmiştir [18].

### 2.3. Optimize Edilmiş Özellik Seçimi Kullanılarak Elde Edilen Yüz Görüntü Dizileri

Bu kısımda optimize edilmiş özellik seçimi kullanılarak görüntü dizileri konusu incelenecektir.

### 2.3.1. Yüz Görüntüsü Veritabanı

Optimize edilmiş özellik seçimi kullanılarak görüntü dizeleri sistemde yüz verilerini eğitmek için CohnKanade veri tabanında [19] bulunan yüz ile ilgili görüntü dizeleri kullanılmıştır. Görüntü verisi 100 farklı kişinin yüz görüntüsünden oluşmaktadır. Görüntü dizeleriyle birlikte 388 görüntü içermektedir. Görüntü dizeleri 12-16 arası kare içermektedir. Görüntü veri setindeki kişilerin yaşları 18 ile 30 arasındadır. Görüntü veri tabanında bulunan görüntülerde öfke, iğrenme, korku, mutluluk, üzüntü ve şaşırma olmak üzere altı temel duygu yer almaktadır [19]. Şekil 2.13'te örnek görüntüler gösterilmiştir.



Şekil 2.13. CohnKanade Veri Tabanında Yer Alan Örnek Görüntü Dizisi Seti [19]

### 2.3.2. Yüz Görüntüsü Ön İşleme

Görüntü ön işleme, sistemde bulunan görüntüler üzerinde normelleştirme yapmaktan oluşur. Görüntülerin normelleştirilmesiyle sadece normalize yoğunluğa sahip yüz görüntülerini içeren, sade görüntü dizeleri elde edilir. Görüntüler histogram eşitlemesi yapılarak normelleştirilmektedir. Veri tabanındaki görüntülerde yüz özellikleri seçilirken aşağıdaki kurallar uygulanır.

- i. Haar-like benzer özelliklerine bakılır [19],
- ii. Adobost öğrenme algoritmasına dayalı Viola-Jones Yöntemi kullanılır [19].

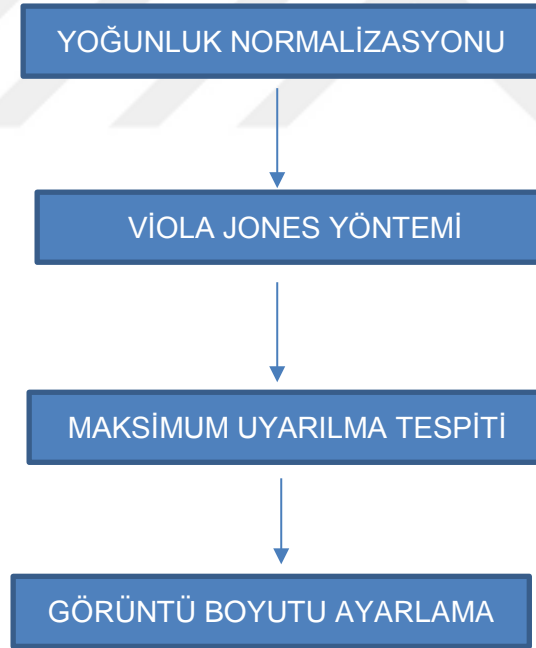
Burada;

**Viola Jones Yöntemi** gerçek zamanlı olarak rekabetçi nesne algılama oranları sağlayan bir algoritma sistemidir. Yüz tanımlama sistemleri için tasarlanmıştır.

**Adabost Algoritması** yüksek performans sağlayabilmesi için diğer birçok algoritma ile çalışabilecek öğrenme algoritmasıdır. Sistem, yinelemeli kullanımda bir ağırlık dağılımını güncelleyerek yanlış sınıflandırdığı örneklerin ağırlıklarını artırır. Bu ağırlık artırma işlemi, verilere sistemin daha dikkatli sınıflandırma yapmasını sağlamaktadır.

Ön işleme işlemi gerçekleştirilirken yapılan bir diğer aşama duygunun tespit edilebilmesidir. Bunu içeren görüntülere minimum karşılıklı bilgi (MI) kriteri kullanılmıştır [19].

İşlem gerçekleştirilirken görüntü dizisinin her karesinde ayrı ayrı işlem yapılır. Mevcut çerçeve ile ilk çerçeve verileri karşılaştırılır. Elde edilen veriler doğrultusunda minimum karşılıklı bilgi içeren çerçeve maksimum uyarılma ile ilgili duyguyu içeren çerçeve olarak kabul edilir. Görüntü ön işleme yönteminin işlem akışı Şekil 2.14'te gösterilmiştir.



**Şekil 2.14.** Görüntü Ön İşleme Yöntemi İşlem Akışı [19]

Bu kısımda incelenen sistem, bir sınıflandırma sistemi olarak uygulanmış ve test edilmiştir. Kullanılan yöntem, işlemleri otomatik olarak gerçekleştirmektedir. Sistem aynı zamanda otomatik olarak duygu seçimi için en doğru kareyi bulmaktadır. Optimum özellik seçimi, uygulanan

sınıflandırmaların doğru sınıflandırma yüzdesini %79,5'a yükseltmiştir. Aynı zamanda CPU'da gerçekleştirilen işlem süresini 1.71 saniyeye kadar indirmeyi başarmıştır [19].

### 2.3.3. Yüz İfadesi Tanıma

#### 2.3.3.1. Yüz İfadesi Tanımadaki Bulanık Yaklaşım

Bu kısımda bulanık mantık yaklaşımı incelenecektir. Bu yaklaşımda sistem, göz ve ağız durumları için bulanık yaklaşım kullanılarak duygu tanıma yapmak amacıyla tasarlanmıştır. Bu sistemde duygu tanıma için sistemin yüz tanıma yapması, yüzün özelliklerini belirlemesi ve sınıflandırma yapması gerekmektedir.

#### 2.3.3.2. Bulanık Yaklaşım Kullanan Yüz Tanıma Sistemi

Bulanık yaklaşım kullanan sistemde yüz tanıma işlemi Matlabın görüntü işleme fonksiyonları kullanılarak gerçekleştirilir. Bu fonksiyonlardan YCbCr renk uzayındaki kırmızı chroma bileşenleri ile görüntüdeki yüz kenarları belirlenmektedir. Ayrıca performansı arttırmak için 1-D medyan filtresi, sobel kenarı tespiti, lineer yapı elemanlı morfolojik işlemler de kullanılır. Şekil 2.15 görüntünün yüz tanımadaki kenar çıktısıdır [20].

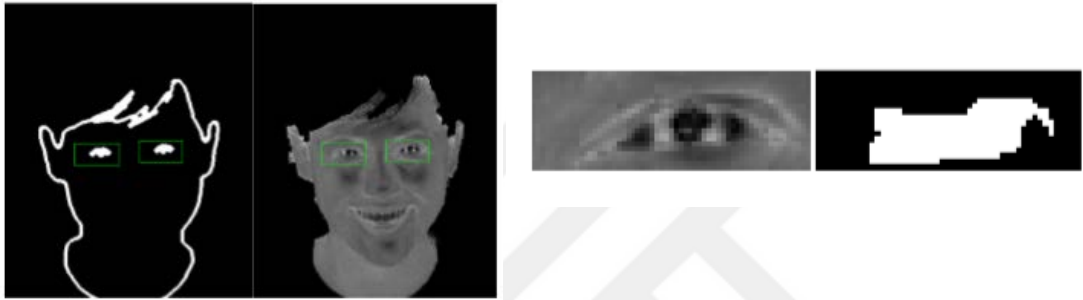


Şekil 2.15. Görüntünün Yüz Tanımadaki Kenar Çıktısı

#### 2.3.3.3. Yüz Özelliklerinin Belirlenmesi

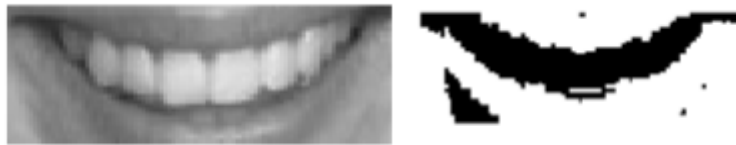
Yüz özelliklerinin belirlenmesinde göz açma boyutu, ağız açma boyutu ve göz açıklığı özellikleri belirlenecektir.

1. **Göz Açma Boyutu:** Matlabtaki YCrCb renk uzayı kullanılırken arařtırmacılar genellikle Cb Chroma bileřenlerine dikkat ederek arařtırmalarına bařlamaktadır. Kenar tespiti, morfolojik iřlemler ve ikili baęlı bileřen fonksiyonları ięeren Matlab yazılım fonksiyonlarının kullanılması, incelenen görüntünün göz bölgelerinin belirlenmesini saęlamakta ve bu bölgeleri alanlara ayırmaktadır. Daha sonra bu bölgelerde HSV renk uzayı kullanılarak S bileřeni ięeren bölgeler belirlenerek bu bölgeler kırılır. Gözün aęıklık belirleme iřlemi gözün kenarından merkez çizgisindeki piksellerin sayılmasıyla belirlenir. Yüz görüntüsünde gözlerin belirlenmesi ve kırılması iřlemi Őekil 2.16'da gösterilmiřtir.



**Őekil 2.16.** Yüz Görüntüsünde Gözlerin Belirlenmesi ve Kırılması İřleminin Çıktısı

2. **Aęız Aęıklığı Boyutu:** Aęız aęıklığı tespiti  $a \times V$  parametresi ile yapılmak istenmiřtir. Aęız aęıklığı boyutu  $a \times V$  ile hesaplanır.  $L^*a^*b^*$ 'de  $a$  ve HSV renk alanlarında  $V$  aęız rengi olarak tanımlanmıřtır. Aęız bölgeleri morfolojik olarak aęık ikili görüntü kullanılarak tespit edilir. Aęız aęıklığı  $Cr$  boyutu parametresinden elde edilmiř görüntünün kırılmasıyla hesaplanır [20]. Yüz görüntüsünde aęız aęıklığı boyutunun belirlenmesi Őekil 2.17'de gösterilmiřtir.



**Őekil 2.17.** Yüz Görüntüsünde Aęız Aęıklık Belirlenmesi

3. **Göz Aęıklığı:** Göz bölgesi sınırı ięinde göz aęıklığı göz geniřlięine bölünerek elde edilir.
4. **Aęız Geniřlięi:** Aęız bölgesini sınırlayan alandan belirlenir.

#### 2.3.3.4. Sınıflandırma

Bulanık yaklaşım kullanılarak yüzden duygu tanımı ve sınıflandırma için yüz nitelikleri ve duygu uzayının bulunması gerekir. Bu işlemde trapezoidal fonksiyonda 94 bulanık kuralı ve Mamdani-Tip ilişkileri kullanılarak işlemler gerçekleştirilir [20]. Yüz Sınıflandırma işleminde aşağıda belirtilen kategoriler kullanılmıştır.

- i. Ağız açıklığı ölçümleri 3 kategoriye ayrılarak yapılmıştır. Bunlar; düşük, orta ve yüksek olarak belirlenmiştir.
- ii. Göz açma bulanık kümeleri çok düşük, düşük, orta, yüksek, çok yüksek ve ekstra yüksek olmak üzere 6 kategoriye ayrılarak sınıflandırılmıştır.
- iii. Göz açma genişlik oranı çok düşük, düşük, orta, yüksek ve çok yüksek olmak üzere 5 kategoriye ayrılarak sınıflandırma yapılmıştır.
- iv. Ağız genişliği düşük, orta, yüksek ve çok yüksek olmak üzere 4 kategoriye ayrılarak sınıflandırılmıştır.

Bulanık sistem yüzdeki duyguların sınıflandırılmasında kullanılmıştır. Duygunun mutlu ve kızgın olarak sınıflandırılması için kurallar aşağıda verilmiştir.

**Kural 1:** Ağız açıklığı yüksek, göz açıklığı yüksek, göz açıklık/genişlik oranı çok düşük ve ağız genişliği yüksek ise duygu mutlu olarak belirlenmiştir.

**Kural 2:** Ağız açıklığı düşük, göz açıklığı çok düşük, göz açıklık/genişlik oranı yüksek ve ağız genişliği düşük ise duygu kızgın olarak belirlenmiştir.

Bulanık mantık kullanılarak yapılan sistemde Ebner veri tabanı [20] ile karşılaştırılmıştır. Bulanık sistemle ortalama olarak %78,8 doğruluk elde edilmiştir [20]. En iyi uyum mutlu ve korku durumları için gözlemlenmiştir. En kötü duygu normal olarak belirlenmiştir. Cohn-Kande veri tabanı kullanılarak yapılan işlemlerde mutlu, korku ve üzgün duygular için elde edilir. Normal ve iğrenme konusunda sonuçlar iyi değildir. Önerilen yöntem mutlu ve korku duygularında iyi iken; iğrenme duygusunun tespitinde zayıf olduğu belirlenmiştir.

#### 2.4. Yüz Algılama ve Konumlandırma Kenar Çizgi Belirleme Algoritmaları

Görüntüde yüz algılama ve konumlandırma işleminde birçok etken mevcuttur. Görüntünün alındığı ortamın ışık düzeyi, görüntü alan cihazın görüntü kalitesi ve yüz pozunu bu etkenlerden bazılarıdır. Böyle etkenler mevcutken doğruluk saptanması olayı değişkenlik göstermektedir [21].

### 2.4.1. Yüz Tespiti ve Lokasyonu

Yüz tanıma sistemi oldukça karmaşık ve zor bir iştir. Yüz tanıma teknolojisi göz önüne alındığında yüz tanıma sisteminde iki yöntem kullanılmaktadır. Bunlar istatistiksel yöntemlere ve bilgi yöntemine dayalı tekniklerdir.

İstatistiksel yöntemeye dayanan yüz tanıma sisteminde sinir ağının kendi kendine öğrenme algoritması, alt uzay boyutsallığı azaltma yöntemi gibi yöntemler kullanılmaktadır. Bu yöntemler güvenilir olmasına rağmen hesaplama süresi uzunluğu, büyük miktarda veri işlendiği ve fazla hesaplamalar yapıldığı için gerçek zamanlı kullanıma uygun olmamaktadır [21].

Bilgi temelli yüz tanıma sistemi insan yüzünün birçok özelliği ele alınarak model oluşturmak için bir standart oluşturma işlemidir. Kullanılan yöntemlerden bazıları:

- i. Renk ve doku yöntemi
- ii. Hareket yöntemi
- iii. Organ dağıtım yöntemi
- iv. Simetrik yöntem.

olarak ifade edilir.

Yüz tanıma sistemi için elde edilen değerler bilgisayar ortamına aktarılarak etkin karakteristik parametreler elde edilir. Bilinen yüz özellikleri ile elde edilen parametre değerleri karşılaştırılır.

Yüz tanıma sisteminde standart yüz özellikleri olarak: iki göz; ağız ve gözler, kaşlar ve gözler veya göz ve diğer sabit yüz özellikleri arasındaki mesafeler kullanılmaktadır. Yüzün ana özellikleri olarak renk, ten rengi tespiti, HIS renk alanı, YCbCr renk alanı ve CIE gibi renk alanları kullanılır [21].

Yapılan çalışmalar ve araştırmalarda elde edilen sonuçlara göre YCbCr diğer birkaç renk uzayına göre karşılaştırıldığında yüz tanıma için daha uygun bir renk uzayı olduğu sonucu elde edilmiştir.

### 2.4.2. Yüz Görüntülerinde Ağız ve Göz Pozisyonları

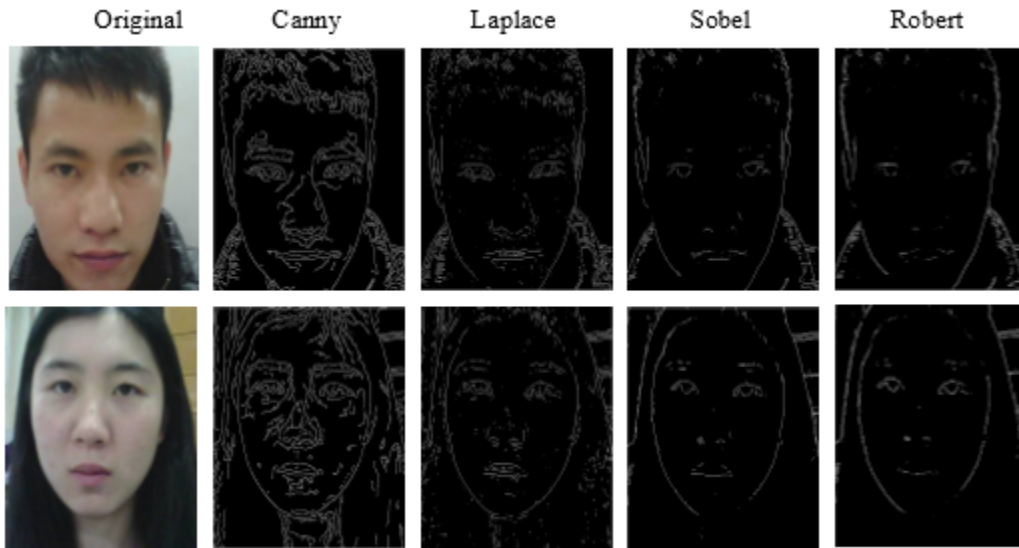
Yüz tanıma sisteminde yüzü tanımlamada, tanıma ve analiz etmede gözlerle ağız arasındaki mesafenin büyük önemi bulunmaktadır. Aradaki bu mesafenin değeri kişiden kişiye farklılık göstermekle birlikte, analizlerde büyük kolaylıklar sağlamaktadır.

### 2.4.3. Yüz Tanıma Sisteminde Kenar Algılama

Yüz tanıma sisteminde kenarlar görüntülerin hepsinde bulunan en temel özelliklerden biridir. Kenar, belirli yer ve pozisyonlardaki piksellerin dizilimlerinin toplamıdır. Kenar, görüntü algılama ve görüntü bölütleme tekniklerinin kullanıldığı özelliklerin başında gelmektedir. Darbe, şeklin kenar profilinin bir araya getirilmesi ile oluşan, değerlerinde yükselme ve düşme olan, gri değer gösteriminin ince bir şeridine karşılık gelen durumdur.

Belirli bir nesnenin sınırındaki piksellerindeki değişim o bölgenin değişim bölgesi olmasını sağlar. Bu değişimin iki önemli özelliği bulunmaktadır. Bunlar gri seviye değişimi ve yönün varyans oranıdır. Bu değerler gradyan vektörünün büyüklüğü ve yönü ile ifade edilmektedir [21].

Kenar çizgi algılama yöntemi, birçok pikseli ve komşusundaki pikselleri kontrol eder. Pikseller için yönlü türev maskesi konvolüsyonuna dayanan işlemler yapılması, piksellerin yönü dahil olmak üzere gri seviyesinin varyans oranını ölçmek içinde kullanılmaktadır. Sıklıkla kullanılan kenar algoritmaları: Canny, Laplace, Sobel ve Robert olarak bilinen algoritmalarlardır. En çok kullanılan kenar algılama yöntemleri Şekil 2.18'de gösterilmiştir. Yöntemlerden en kolay anlaşılmanı Robert iken en karmaşık olanı Canny'dir. Laplace ve Sobel yöntemleri anlaşılması zor olanlardır [21].



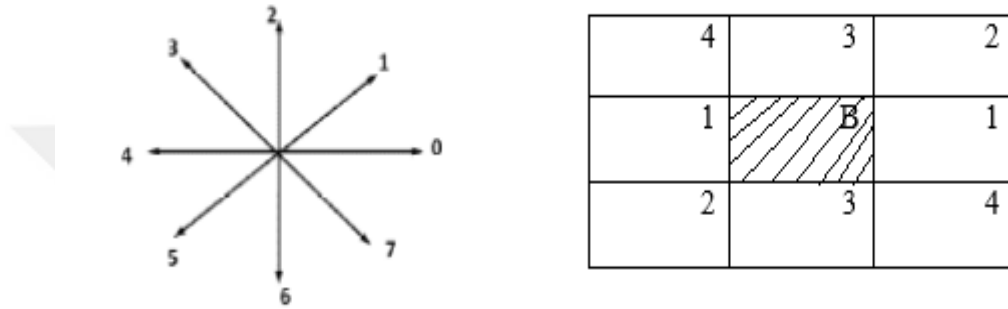
Şekil 2.18. Kenar Algılama Algoritmaları Uygulanmış Görüntüler [21]

#### 2.4.3.1. Canny Kenar Algılama

Canny kenar algılama yöntemi kullanılırken gözler ve dudaklar baz alınmaktadır. Gözler yüzün üst kısmında ve ortada yer almaktadır. Dudaklar da yüzün alt kısmının ortasında bulunduğu

için araştırma algoritması yüzün belirli bir bölümünde yapılır. Bu işlem hızını olumlu yönde etkiler [21]. Yüz tanıma sistemi öncelikle yüzdeki gözlerin ve dudakların yerlerini tespit eder. Yer tespiti yapıldıktan sonra bu yerde canny kenar algılama formülü kullanır. İlk olarak Gauss formülü kullanılarak dudaklar belirlenir. Gauss modeli ile birlikte EM algoritması kullanılır.

EM algoritmasında sonlu birinci dereceden kısmi türev aracılığı ile gradyan büyüklüğü ve yönü hesaplanması işlemi gerçekleştirilir. EM algoritmasının E adımında yön belirlenir, M adımında büyüklük hesaplanır. Gradyan büyüklüğünde maksimum olanın tarafına yönün kaymasının engellenmesi işlemi Şekil 2.19'da gösterilmiştir.



**Şekil 2.19.** Pikseldeki 8 komşu Değerin Diyagramı.

İşlemin gerçekleşmesi maksimum değer bulunmasının ardından olur. Maksimum değer sistemi kaydırmaması için maksimum gradyanın genliği ve yönüyle birleştirilir. Sonra kısmi türevlerin pikselleri karşılaştırılır. En uygun değer olarak maksimum değer olduğu nokta 0 noktası olarak ayarlanır [21].

#### 2.4.3.1.1. EM Algoritması

Literatürde beklenti maksimizasyonu algoritması olarak bilinmektedir. Bu algoritmanın amacı örnek küme üzerinde işlemler yaparak bazı sonuçlar elde etmek ve bu sonuçlardan bir kısmı eksik veya hatalı parametrelerle temsil edildiği durumlarda kullanılabilir. Bu algoritma istatistiksel yöntemler kullanılarak elde edilmiştir.

Algoritmanın sistemde uygulanabilmesi için ilk olarak hatalı parametrelerin tespit edilmesi gerekmektedir. Bu işlemden sonra parametrenin örnek uzayda kaç elemana ait olduğu tespit edilir. Tespit edilen bu sayıların kullanım amacı örnek kümeyle ait olasılık yoğunluk fonksiyonunun içinde kullanılacağıdır. Bu türdeki dağılımlar tekrar eden dağılımlardır [22].

$$f(x, p) = \frac{100!}{10!x65!x15!x10!} (0,2)^{x1+x3} (0,6-0,5p)^{x2} (0,25p)^{x4+x5} \quad (2.9)$$

örnek bir  $f(x,p)$  fonksiyonu 2.9'da verilmiştir.

EM algoritması açıklamak gerekirse şu şekilde ifade edilebilir. Tüm veriler yapılacak olan işlemlerde kullanılmak üzere tanımlanan bir değişkene eşitlenir. Hatalı verilerin sonucunun bulunması için hatalı değerler iki değişkene eşitlenerek işlemlere devam edilir.

Örnek küme uzayına alınan veriler incelenerek bu verilere ait olan olasılık dağılımları oluşturulur. Elde edilen olasılık dağılımlarında hatalı veya eksik değerlerin elde edilebilmesi için dağılımlarda bir harf değişkeni (örnek değişken  $p$ ) kullanılır. Bu değişken sayesinde beklenen değerlerin elde edilebilmesi sağlanmış olur [22].

Bu işlemler yapılarak olasılık dağılımları elde edilmiş olur. Olasılık dağılımları kullanılarak olasılık yoğunluk fonksiyonu oluşturulur. Oluşturulan olasılık fonksiyonu örnek olarak verilen  $f(x,p)$  fonksiyonudur. İşlem yapılırken faktöriyel olan kısmın hesaplanmasına gerek yoktur, çünkü türev alma işleminin ardından bu kısmın değeri 0 olarak elde edilecektir.

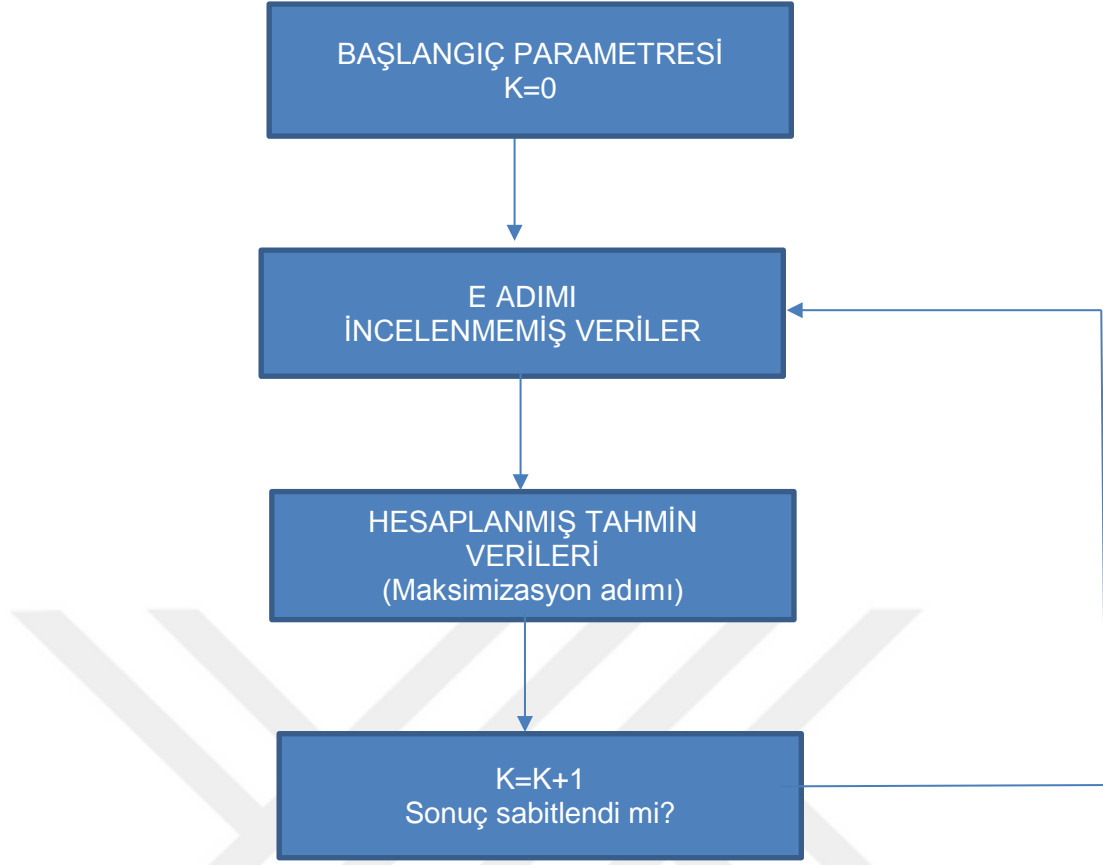
Olasılık yoğunluk fonksiyonunun işlemlerde kolay kullanmak için fonksiyonun logaritması alınarak fonksiyon daha sade bir hale gelmesi sağlanmış olur. Logaritma işleminin ardından fonksiyonun beklentisi alınır. Beklenti işleminin ardından  $f(x,p)$  fonksiyonu;

$$\ln(f(x,p)) = \ln A(x_1 + x_3) \ln(0,2) + x_2 \ln(0,6 - 0,5p) + (x_4 + x_5) \ln(0,25p) \quad (2.10)$$

2.10 fonksiyonu elde edilmiş olur. Logaritması alınan fonksiyonda  $x_3$  ve  $x_4$  değerleri dışındaki  $x$  değerleri problemin başında tanımlanmış değerlerdir. Sonucu bulabilmek için fonksiyonun türevi alınır.

Türevi alınan fonksiyon 0'a eşitlenerek fonksiyonda bulunan "p" değerleri bir tarafta toplanır. Bu işlemin yapılmasındaki amaç yakınsayan  $p$  değerleri için parametre sonuçlarının elde edilmesidir. Bu adıma maksimizasyon adımı denilmektedir.

Elde edilen fonksiyonda  $x_4$  değeri problemde bulunan olasılık dağılımları kullanılarak  $x_4$  değeri bulunur.  $x_4$  değerini bulunması amacıyla  $p$  değerlerine 1'den küçük değerler verilerek işlem sonuçları elde edilir [22]. Bu şekilde işlemler yapılmaya devam edilir. İşlemden bir yerden sonra sabit değerler elde edilir. İstenilen değer elde edilmiş olan sabit değerdir. Algoritmanın genel şekli Şekil 2.20'de gösterilmiştir.



Şekil 2.20. EM Algoritması İşlem Akış Diyagramı [22]

#### 2.4.3.2. Laplace Kenar Algılama

Laplace filtresinde ilk olarak görüntü biraz bulanıklaştırılır. Bulanıklaştırma işlemi görüntüde gauss kıvrılması oluşturur. Gauss kullanılmasının amacı; gauss düz yüzeyleri daha belirgin hale getirir. Filtreleme işlemi yapıldığında dalgalanmalar oluşur. Bu dalgalanmalar kenar olarak algılanır. Görüntüdeki her dalgalanmanın kenar olarak algılanması nedeniyle yanlış kenar tespitleri yapılır [34].

Yanlış kenar tespitlerinin azaltılması için Laplace'ın sıfır değerlerinin elde edilmesi gerekir. Görüntünün yerel varyans hesaplaması yapılır. Laplace'ın sıfır geçişleri ile yerel varyans değerleri arasında bir karşılaştırma yapılır. Belirlenen eşik değeri geçen sonuçlar kenar olarak algılanır [34]. Yapılan işlemler sonucunda temiz bir görüntü elde edilir.

### 2.4.3.3. Sobel Kenar Algılama

Sobel filtresi bilimsel olarak basit kenar algılama (edge detection) filtresi olarak adlandırılır. Sobel filtresi kenarları hem dikey hem de yatay olarak ortaya çıkarır. Sobel kenar algoritmasında gri renk oranına göre işlem yapılmaktadır. Sobel filtresi uygulanacak görüntü ilk olarak siyah- beyaz formata çevrilir.

Sobel filtre işleminde yatay ve dikey kenarlar konvolüsyon matrisleri ile elde edilir. Matrisler görüntüye dikey ve yatay olarak uygulanır. Sobel operatörü ile görüntüdeki yüksek frekans içeren bölgeler tespit edilir. Bu bölgeler kenarları temsil eder [35].

### 2.4.3.4. Robert Kenar Algılama

Robert kenar algılama algoritması resim üzerinde bulunan 2 boyutlu geçişleri ölçer. Siyah-beyaz görüntü kullanılır. 2x2'lik matrisler kullanılarak işlemler yapılır. Görüntünün çok bulanık olması durumunda kenar algılama yapılamaz. Kullanım olarak sobel kenar algılama algoritmasına benzerlik göstermektedir. Robert kenar algılama algoritması keskin köşelerin bulunması sağlamaktadır [35]. Çift eşik algoritmasının kullanımı sırasında kenar çizgisi net bir şekilde çekilir. Kenarın arkasındaki yüksek değerleri kenar işleminden kaldırılır. En son elde edilen değer kenar eşik bağlantısından elde edilen değerle karşılaştırılır ve değer doğruluğu kontrol edilir.

Yüz tanıma sistemi için 4 farklı kenar algılama kullanılarak Xiaoming Chen ve Wushan Cheng tarafından duygu analizi yapılmıştır. Oluşturdukları sistemi eğiterek her bir kenar algoritmasında görüntülerle sınıflandırma yapılmıştır. Xiaoming Chen ve Wushan Cheng tarafından elde edilen duygu sınıflandırma sonuçları ve başarı yüzdeleri Tablo 2.1'de gösterilmiştir.

**Tablo 2.1.** Kenar Algılama Yöntemlerine Göre Elde Edilen Sonuçlar [21]

Yüz İfadesi Sınıflandırma Sonuçları (%)	Normal	Üzgün	Mutlu	Şaşırmış
Canny	100	91,3	99,4	95,7
Laplace	93,2	87,1	89,5	85,6
Sobel	63,4	49,7	66,8	64,9
Robert	49,8	30,1	42,5	48,2

Yüz Tanıma sisteminde en iyi sonuçlar Canny operatörü ile elde edilmiştir. Canny operatörü Şekil 2.18'de de görüldüğü gibi göz ve dudak şekilleri net bir biçimde tespit edilmiştir. Değerin bu kadar iyi elde edilmesinde anti- gürültü yeteneğine sahip kenar algoritmasının kullanılması da etkili olmuştur [21].



### 3. MATERYAL VE YÖNTEM

Bu bölümde dağıtık bilgisayar mimarisinde paralel programlama kullanılarak oluşturulan yüksek performans hesaplama sisteminde veritabanında kayıtlı yüz görüntü dosyalarının sınıflandırılması açıklanacaktır.

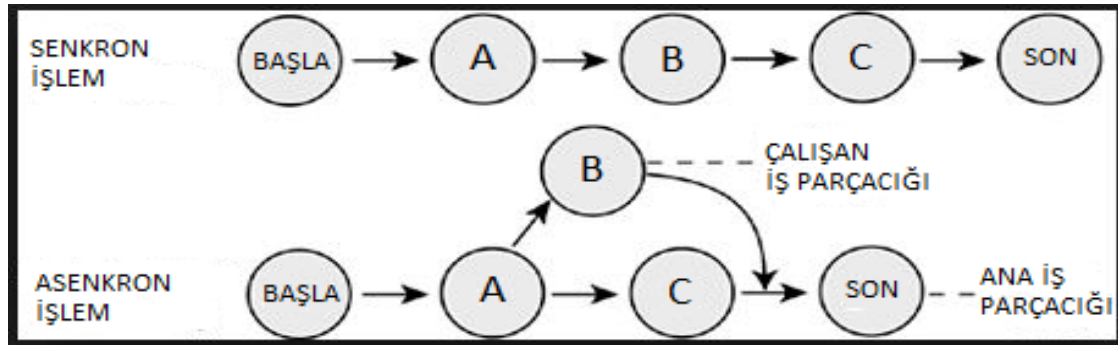
#### 3.1. Dağıtık Bilgisayar Mimarisinde C# ile Paralel Programlama

Teknolojik olarak bilgisayar işlemcisi dünyasındaki gelişmeler ve değişimler ile bilgisayarların gücü günden güne artmaya devam etmektedir. Bilgisayarın hızının artması programlarında hızının artmasını sağlamaktadır. Fakat programların hızlanması sadece bilgisayar özelliklerine bağlı değildir. Programların daha hızlı çalışması için yazılımların işlemciyi iyi kullanabilecekleri şekilde yapılması gerekmektedir. Bilgisayar sisteminde işlemci sayısının artması ve işlemlerin işlemciler arasında parçacıklara bölünerek yapılması süreci hızlandırmaktadır. İşlemlerin işlemciler arasında bölünmesi paralel programlama ile gerçekleştirilir. Paralel programlama yüksek performans hesaplamada kullanılan yazılım hesaplama tekniğidir.

Paralel programlama iki önemli kavram içermektedir [36].

1. Senkron (eşzamanlı) programlama
2. Asenkron (eşzamansız) programlama

Senkron ve asenkron programlama Şekil 3.1'de gösterilmiştir.



**Şekil 3.1.** Senkron ve Asenkron Programlamanın Gösterimi [36]

Senkron(eşzamanlı) yürütmeye, program tüm görevleri sırasıyla gerçekleştirir. Bu, her görev tetiklendiğinde bir görevin tamamlanmadan diğer göreve başlamayacağı anlamına gelir.

Asenkron (eşzamansız) yürütmeye program görevleri sırasıyla çalıştırmaz. Tüm görevler aynı anda çalıştırılır. İşlemlerin tamamlanması beklenir.

Senkron ve asenkron işlem yapmak farklı avantaj ve dezavantajlar içerir. Senkron işlem görevleri sırasıyla yürüterek işlem yapmaktadır. Senkron programlama işlemi asenkron programlama işlemine göre daha kolaydır. Senkron işlem yapmanın dezavantajları vardır [36]. Bunlar;

- i. İşlemlerin tamamlanması daha uzun sürer.
- ii. Kullanıcı arabirimi (UI) işlemin durdurulmasını sağlayabilir. İşlem kullanıcı arabirimi ile kesilirse programın yanıt vermemesi durumu oluşabilir.
- iii. Programlar işlemci sayısı bağımsız çalıştığından çoklu işlemcilerin yapısından faydalanamaz.

Asenkron programlama senkron programlamaya göre işlemleri daha kısa sürede tamamlar. İşlemci sayısına göre daha hızlı çalışabilir [36]. Fakat asenkron işlem yapmanın bazı zorlukları vardır:

- I. Asenkronda görevler senkronize edilmelidir. Başlayan tüm görevler bitmeden yeni bir işleme başlanmamalıdır.
- II. Aynı anda farklı görevlerde kullanılan kaynaklarının değişmemesi ve kaybolmaması gerekmektedir.
- III. Asenkron işlem başladıktan sonra ilk hangi görevin tamamlanacağı görevlerin işlem süresi ile ilgilidir.

### **3.2. Dağıtık Bilgisayar Mimarisinde İş Parçacıkları Oluşturulması**

İş parçacıkları (thread), işletim sisteminde görevleri yapan en küçük kod birimidir. Bir görev iş parçacıklarına bölünerek işlemcide işlem görür. Sistemde birden fazla işlemci varsa iş parçacıkları paralel olarak çalışabilir. İşlemlerin daha kısa sürede tamamlanmasını sağlar. C# ile paralel programlama işlemleri .Net Framework 4.0 ile TLP (Task Parallel Library) yapısı ile ortaya çıkmaya başlamıştır. Bu yapı sayesinde işlemler paralel bir şekilde yürütülebilir. Paralel programlama için TLP komutları System.Threading ve System.Threading.Task kütüphanelerinde bulunmaktadır [38].

TLP kütüphanesi her bir süreci görev olarak değerlendirmektedir. TLP Kütüphanesi görevleri mantıkla yönetmektedir. TLP Kütüphanesi ile;

- a) Çoklu görevler yapabilmektedir. Bu görevler üzerinde işlemlerin yapılması mümkündür.
- b) Çoklu görevler mantıklı biçimde organize edildiği için bir görev biterken yeni bir görev, o görevden sonra başlatılabilir.
- c) Çoklu görevlerin sonuçlarında değer döndürme işlemleri yapılabilir.
- d) Görev yapılırken kendi içinde alt görevler oluşturabilir. Bu görevlerin yapılması sağlanabilir.
- e) Çoklu görevler yapılırken işlemler iş parçacıkları (threadler) tarafından yerine getirilir.

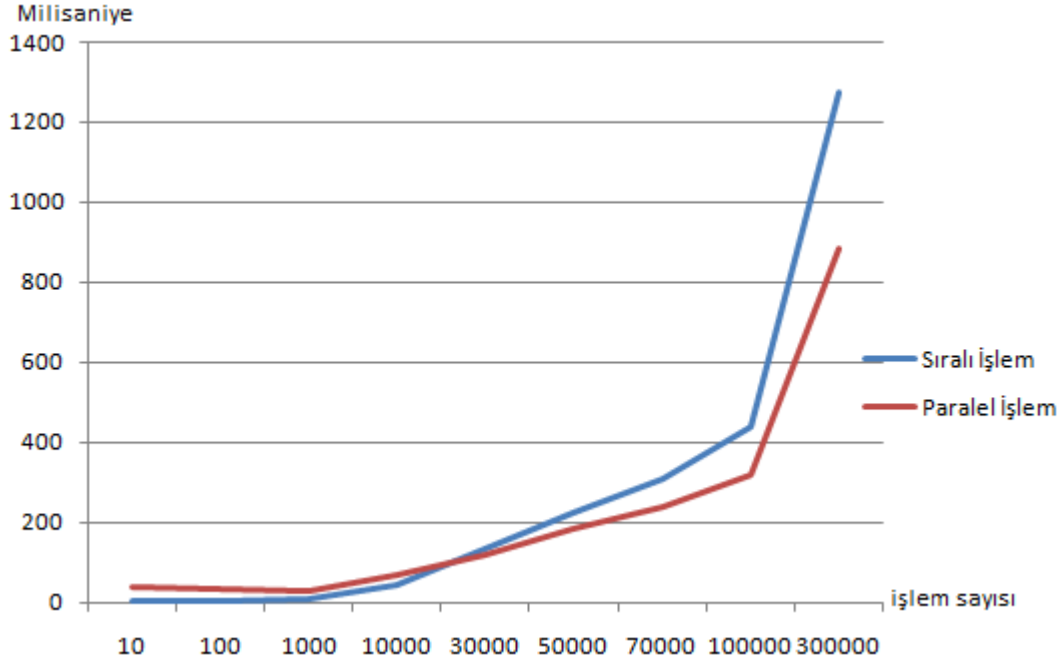
Paralel programlama yapılırken işlemlerin başarılı bir şekilde tamamlanması işlemi için Görev(task) beklenir. Görevlerin tamamlanma süresini işlemcinin özellikleri, sayısı ve kodlama etkilemektedir.

Paralel programlamada işlemler veya görevleri iş parçacıkları(threadler) yerine getirir. Threadler programda iş yapan parçacıklardır. İş parçacık sayısı değişkenlik gösterebilir. Bir veya daha fazla olabilirler. Bir iş parçacığı bir veya daha fazla görevi yapabilir ya da birden fazla iş parçacığı(thread) bir görevi yerine getirebilir. Paralel programlama, seri programlamaya göre zaman kazandıran çözümler sağlamaktadır. Seri programlama da bir işlem bitmeden diğeri başlamadığı için işlem süreleri paralel programlamaya göre daha uzun sürmektedir. Paralel programlamada aynı zamanda bir veya daha fazla iş yapılabilir [38]. Paralel programlama sayesinde işlemler daha kısa sürede tamamlanmış olur. C# ile ilgili paralel programlama örneği Şekil 3.2'de gösterilmiştir.

```
Parallel.For(2, 30, i =>{
    var result = sum(i);
    Console.WriteLine("root {0} : {1}, Thread:{2}",
        i,
        result, System.Threading.Thread.CurrentThread.ManagedThreadId);
});
```

**Şekil 3.2.** C# Paralel Programlama Örneği [38]

Örnek olarak paralel ve seri yöntemle yapılan işlem sayısı ve süre ile ilgili grafik aşağıdaki gibi oluşmuştur. İşlem sayısı her adımda biraz daha arttırılarak grafik elde edilmiştir. Seri ile paralel programlama arasındaki ilişki Şekil 3.3'te gösterilmiştir.



**Şekil 3.3.** Sıralı İşlem ile Paralel İşlem Karşılaştırılması [38]

Yapılan işlem sayısı ve harcanan süreç eğrisine göre 30000 işlem sıralı işlemle paralel işlemin kesişim noktasıdır. 30000 işleme kadar seri işlem paralel işleme göre daha kısa sürede biterken 30000 işlemden sonra paralel işlem süresi seri işlem süresinin yarısı kadar sürelerde tamamlanmaya başlanmıştır [38].

Yukarıdaki işlemde paralel programlama işlemi Parallel.For kullanılarak yapılmıştır. Seri işlem ise normal for kullanılarak yapılmıştır. Parallel.For ile seri for arasındaki en büyük fark seri for bir değeri alıp işlem adımına giderken paralel.for iş parçacıkları yardımıyla aynı anda birkaç değeri alarak işlem adımına gider. Böylece paralel for işlemi yapılırken birden fazla değer aynı anda işlenip sonuçlar diğer işlemin bitmesi beklenmeden elde edilmiş olur.

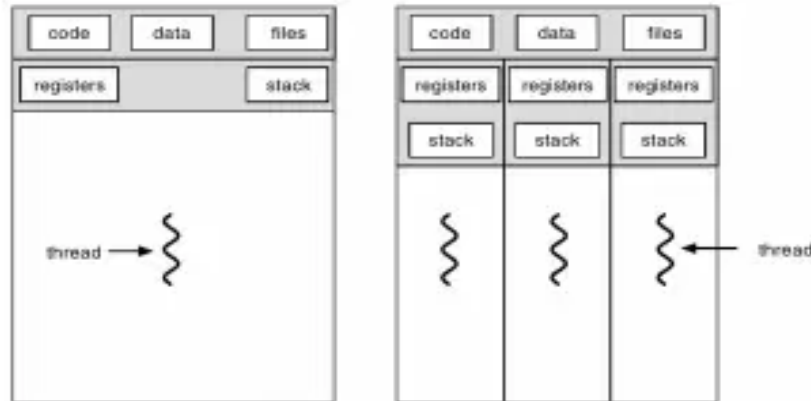
### 3.3. Dağıtık Bilgisayar Mimarisinde Çok İş Parçacıklı (Multi-Threaded) Programlama

Çok kanallı ya da iş parçacıklı programlama (Multi Threaded Programming) bir programın çalışırken aynı anda birden fazla işin yapılabilmesi demektir. Diğer bir ifadeyle bir kod parçası kendi işlemini gerçekleştirirken buna paralel başka bir kod parçasının da işlem yapabilmesidir. Birbirine paralel olarak çalışan her bir iş parçacığı bir çalışma kanalı oluşturur [39].

Uygulamaların ilk metodu olan main metot bir iş parçacığının(thread) işleme başlamasıyla çalışmaktadır. Fakat işlem sırasında tek bir iş parçacığı çalıştığı için yazılımcı yazdığı kodda bu iş parçacığının çalıştığını fark etmez. İş parçacıkları programlarda aynı zamanda birden fazla iş yaptığı için çoklu iş yapılmasını sağlamaktadırlar. İş parçacıkları(thread) ile yapılan programlarda;

- i. Kaynak paylaşımı yaparak birim zamandaki iş miktarını artırır.
- ii. Aynı zamanda daha çok iş yaparak işlemin tamamlanma süresini kısaltır.
- iii. İşlemciyi daha fazla kullanarak donanımın daha efektif kullanılmasını sağlar.

İş parçacıkları işlem için oluşturulduklarında start komutu gelmedikçe işlem yapmaya başlamazlar. İş parçacığının işleme başlayabilmesi için start komutunun çalıştırılması gerekmektedir. İş parçacıkları işlemler arasında beklemeyebilmesi için bekleme(sleep) komutu bulunmaktadır. Bekleme (sleep) komutunun içine girilen süre kadar program işlemi bekletir. Bekleme komutunda süre milisaniye cinsinden girilir. Bir iş parçacığını yok etmek için durdur(stop) komutu kullanılır. Durdur(stop) komutu iş parçacığını sonlandırır. İş parçacığı örnekleri Şekil 3.4'te gösterilmiştir.



**Şekil 3.4.** Tek İş Parçacığı (Thread) ve Çoklu İş Parçacığı (Multi Thread) [39]

İş parçacıkları aynı yapıyı kullanmalarına rağmen verileri farklı dosya konumlarında tutarlar. Bu şekilde aynı kodu kullanarak aynı zamanda farklı değerleri işleyerek sonuçları karışıklık olmadan sistemlerinde tutabilirler. Çoklu iş parçacıkları işlem sırasında sorunlar yaşayabilir. Yaşanabilecek sorunlar arasında senkronizasyon ve eş zamanlılık sorunu vardır. Bu sorunlar yaşanırsa kodlamada çeşitli metotlar kullanılarak çözülebilir.

Yapılan çalışma yüz ifadelerinin tanınması için yüz görüntülerinin sınıflandırılmasıdır. Yüz ifadelerinin tanınması için işlemler 4 ayrı başlık altında incelenecektir. Bu dört aşama aşağıda belirtilmiştir;

1. Yüz Algılama (Tanıma)
2. Veri Çıkarma
3. Yüz Görüntülerini İşleme
4. Yüz İfadesi Sınıflandırması

olarak başlıklara ayrılmıştır.

### **3.3.1. Yüz Algılama**

Yüz algılama işlemi için ilk adım önışlemdir. Yüz görüntüsü, sınıflandırmada tek tip boyuta ve şekle sahip olmalıdır.

Eğitimli sınıflandırmada kullanılan yüz görüntü veri tabanlarından kontrol ve test görüntüleri elde etmek için kırılmıştır. Test görüntüleri elde etmek için kırılma işleminde Matlab yazılımı kullanılmıştır.

### **3.3.2. Veri Çıkarma**

Yüz algılama işlemimde yüz ifadesinin tanınması için sınıflandırılmasındaki ikinci adım, veri çıkarma işlemidir. Kırılmış resimler Matlab programı yardımıyla piksellerine ayrılma işlemi yapılmaktadır. Her piksellik görüntüler RGB değerleri elde edilerek tüm resmin piksel olarak RGB değer listesi elde edilmiştir.

### **3.3.3. Yüz Görüntülerini İşleme**

Yüz ifadesinin tanınması için yüz algılama işleminde üçüncü adım görüntülerin düzenlenmesi işlemidir. Görüntü işleme yapılırken amaç görüntülerin daha az hata ile işlenmesini sağlamaktır. RGB değerlerine ayrılan görüntüdeki resmin kenarlarında bulunan hatalı değerler Matlab programı yardımıyla kesilir.

### **3.3.4. Yüz İfadesinin Sınıflandırılması**

Yüz ifadelerinin tanınması işlemindeki son adım yüz ifadesi sınıflandırmasıdır. Görüntülerin sınıflandırılması için test görüntü değerleri programa girilir. Görüntülerin doğru sınıflandırılabilmesi için her duyguyu içeren en az bir test değeri girilmelidir.

Program görüntülerin duygu seçimini yaparken veri tabanı kullanır. Seçilen görüntüdeki sağ ve sol gözdeki açıklık, ağızdaki genişlik yöntemlerle analiz edilerek programın veri tabanına kaydedilmektedir.

### 3.4. MATLAB (Matrix Laboratory) ile Yüz Görüntüsü İşlenmesi

Matlab temel olarak çok paradigmatlı sayısal hesaplama yazılımı ve kullanımı çok kolay, daha az kod yazarak işlem yapan, hazır şablonlar ve formüller içeren, bunun yanında grafiksel gösterim yapabilen teknik ve bilimsel hesaplamalar yapılabilmesi için yazılmış yüksek performansa sahip bir yazılımdır. Matlab programı genel olarak matematik ve hesaplama işlemleri, algoritma geliştirme, modelleme, simülasyon, veri analizi ve görsel efektlerle ilgili gösterimler, bilimsel ve mühendislik ile ilgili işlemler, uygulama geliştirme gibi birçok alanda kullanılmaktadır [23].

Matlab işlemleri gerçekleştirirken gönderilen tüm verileri bir dizi olarak işleme alır. Bu veriler dizi, skaler, vektör, metin, matris gibi bir karakter dizisi olabilmektedir.

Matlab işlemlerin kolay ve hızlı yapılabilmesi için birçok kısayol ve hesaplama metotları içermektedir. Bunlara birkaç örnek olarak Tablo 3.1'de gösterilmektedir.

**Tablo 3.1.** MATLAB Formül ve Kısaltmaları [23]

Değişken Adı	Açıklama
ans	Bir işlemin son cevabı ( <u>ans</u> wer) veya bir ifadenin değeri
pi	pi sabiti: $\pi=3.1415926535897\dots$
i veya j	imajiner birim, $\sqrt{-1}$
eps (epsilon yani e)	Kayan-nokta bağıl doğruluğu (relative precision), $2e^{-52}$
realmin	En küçük kayan-nokta (floating-point) sayısı, $2e^{-1022}$
realmax	En büyük kayan-nokta (floating-point) sayısı, $(2-e)^{1024}$ yani $2e^{1024}$ den çok az küçük
inf	Sonsuz ( <u>inf</u> inity) yani realmax dan daha büyük bir sayı
NaN	Sayı değil (Not-a-Number)

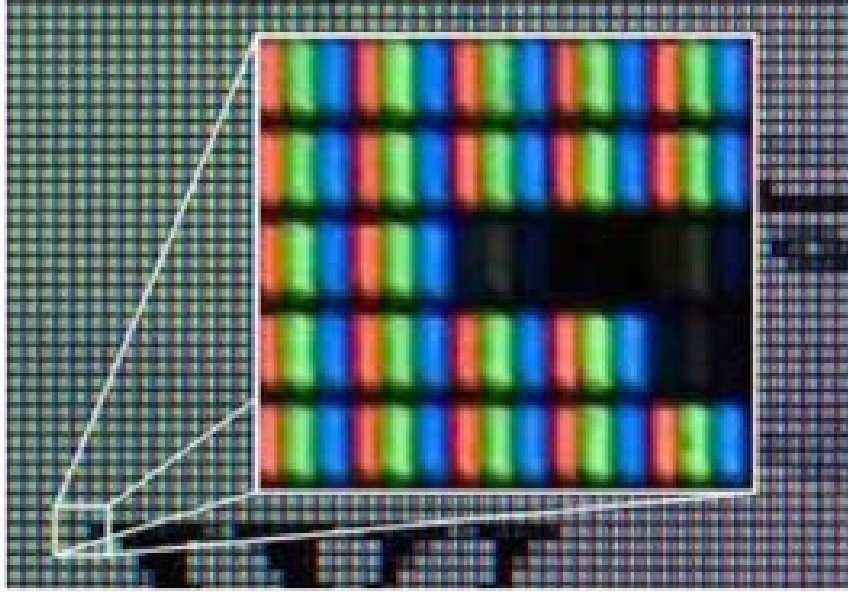
Bu tez çalışmamızda Matlab yazılımı kullanılmıştır. Matlab yüz ifadesini belirlemek için kullanacağımız resmin piksel ayrıştırmasında kullanılmıştır. Piksellerin satır ve sütun değerleri yazılan programla elde edilmiştir. Görüntülerin rgb değerleri elde edilmektedir. Şekil 3.5'te görüntüden elde edilen rgb değerleri gösterilmektedir.

Yazdığımız Matlab kodundan rgb değerlerini almayı sağlayan kod bölümü aşağıdaki gibidir;

```
A=imread('C:\Users\TK\Desktop\deneme.jpg');  
N=size(A,2);  
R = A(:,1);  
G = A(:,2);  
B = A(:,3);
```

```
264. satır,1020. sutun değeri R:26 G:11 B:18  
264. satır,1021. sutun değeri R:31 G:16 B:21  
264. satır,1022. sutun değeri R:36 G:20 B:21  
264. satır,1023. sutun değeri R:37 G:21 B:21  
264. satır,1024. sutun değeri R:37 G:21 B:21  
264. satır,1025. sutun değeri R:46 G:27 B:29  
264. satır,1026. sutun değeri R:53 G:34 B:36  
264. satır,1027. sutun değeri R:38 G:19 B:21  
264. satır,1028. sutun değeri R:48 G:29 B:31  
264. satır,1029. sutun değeri R:40 G:21 B:23  
264. satır,1030. sutun değeri R:43 G:24 B:26  
264. satır,1031. sutun değeri R:42 G:23 B:25  
264. satır,1032. sutun değeri R:45 G:27 B:27  
264. satır,1033. sutun değeri R:40 G:22 B:22  
264. satır,1034. sutun değeri R:46 G:28 B:28  
264. satır,1035. sutun değeri R:56 G:36 B:37  
264. satır,1036. sutun değeri R:59 G:39 B:40  
264. satır,1037. sutun değeri R:42 G:22 B:23  
264. satır,1038. sutun değeri R:38 G:18 B:19  
264. satır,1039. sutun değeri R:40 G:20 B:21  
264. satır,1040. sutun değeri R:46 G:25 B:24  
264. satır,1041. sutun değeri R:48 G:27 B:24  
264. satır,1042. sutun değeri R:60 G:36 B:32  
264. satır,1043. sutun değeri R:58 G:34 B:30  
264. satır,1044. sutun değeri R:70 G:46 B:42  
264. satır,1045. sutun değeri R:65 G:41 B:37
```

**Şekil 3.5.** MATLAB Elde Edilen Değerlerin Bir Bölümü



**Şekil 3.6.** Resmin Bir Bölümünün RGB Değer Gösterimi [24]

Şekil 3.5'te elde edilmiş olan veri seti Şekil 3.6'da şematik olarak verilen resimdeki bir bölümün satır ve sütun karşılığı olarak rgb değerlerini göstermektedir.

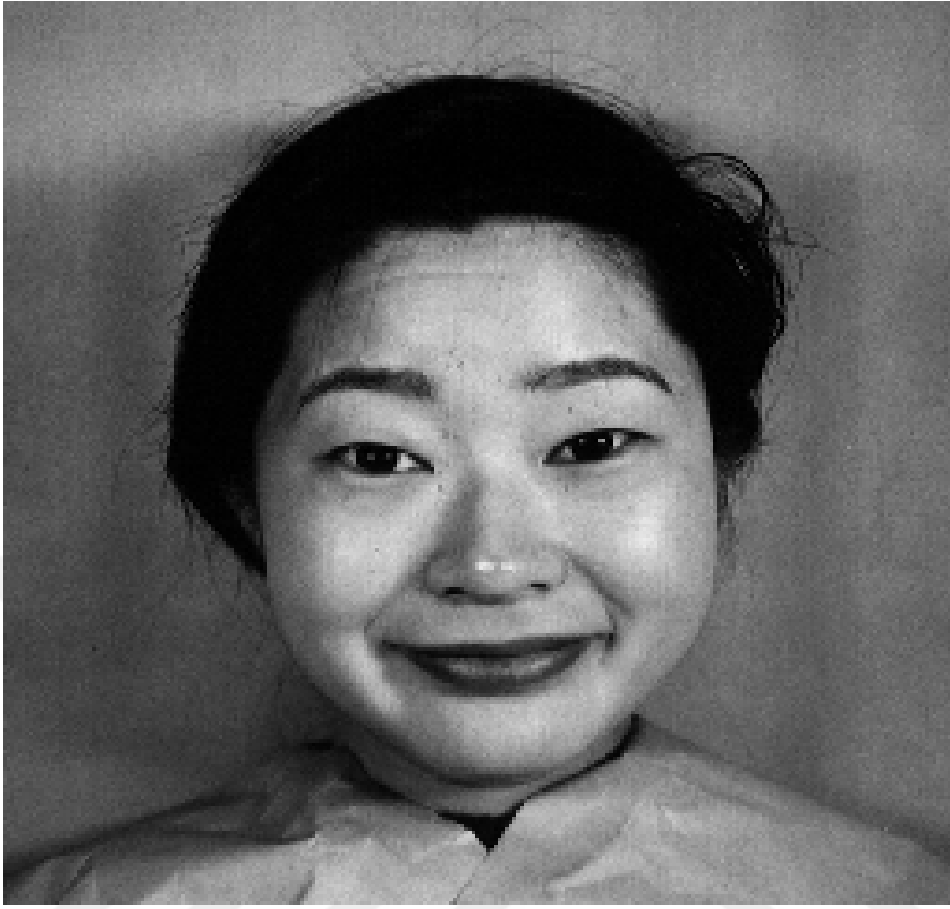
### **3.5. Dağıtık Bilgisayar Sisteminde MATLAB ve Paralel C# Programlama Kullanılarak Yüz Görüntülerinin Sınıflandırılması**

Çalışmamızı yaparken farklı kaynaklardan yararlandık. Görüntülerin daha uygun bir duruma getirilmesi için rgb olarak piksellere ayırdık. Bu görüntülerin kenarlarını uygulama ile keserek uygun görüntüyü elde ettik.

#### **3.5.1. Dağıtık Bilgisayar Sisteminde MATLAB İle Yüz Görüntülerinin İşlenmesi**

Uygulamamız MATLAB ile yazılmıştır. Uygulama sisteme gönderilen bir görüntüyü ilk olarak RGB olarak piksellerine ayırıyor. Her kenardan %3'lük bir kesim işlemi yaparak görüntüdeki yüzü öne çıkarıyor. Matlab programının imcrop fonksiyonuyla görüntünün kenarlarından kesim işlemi gerçekleştirilir.

Yapılan bu uygulama sayesinde işlem yapılacak görüntülerin kenarlarından kırpma işlemi yapılır. Kırpma yapılan görüntüdeki yüz öğeleri daha kolay belirlenir. Şekil 3.7'de Matlab uygulamasına gönderilen görüntü gösterilmiştir.



**Şekil 3.7.** MATLAB Uygulamasına Gönderilen Görüntü

Matlab uygulamasında sadeleştirilen görüntünün rgb değerlerinin bir kısmı Şekil 3.8'de gösterilmiştir.

77. satır,279. sutun değeri R:76 G:76 B:76  
77. satır,280. sutun değeri R:71 G:71 B:71  
77. satır,281. sutun değeri R:69 G:69 B:69  
77. satır,282. sutun değeri R:66 G:66 B:66  
77. satır,283. sutun değeri R:67 G:67 B:67  
77. satır,284. sutun değeri R:68 G:68 B:68  
77. satır,285. sutun değeri R:67 G:67 B:67  
77. satır,286. sutun değeri R:66 G:66 B:66  
77. satır,287. sutun değeri R:67 G:67 B:67  
77. satır,288. sutun değeri R:66 G:66 B:66  
77. satır,289. sutun değeri R:64 G:64 B:64  
77. satır,290. sutun değeri R:65 G:65 B:65  
77. satır,291. sutun değeri R:66 G:66 B:66  
77. satır,292. sutun değeri R:66 G:66 B:66  
77. satır,293. sutun değeri R:67 G:67 B:67  
77. satır,294. sutun değeri R:67 G:67 B:67  
77. satır,295. sutun değeri R:65 G:65 B:65  
77. satır,296. sutun değeri R:62 G:62 B:62  
77. satır,297. sutun değeri R:63 G:63 B:63  
77. satır,298. sutun değeri R:65 G:65 B:65  
77. satır,299. sutun değeri R:66 G:66 B:66  
77. satır,300. sutun değeri R:66 G:66 B:66  
77. satır,301. sutun değeri R:67 G:67 B:67  
77. satır,302. sutun değeri R:68 G:68 B:68  
77. satır,303. sutun değeri R:68 G:68 B:68  
77. satır,304. sutun değeri R:67 G:67 B:67  
77. satır,305. sutun değeri R:67 G:67 B:67  
77. satır,306. sutun değeri R:75 G:75 B:75  
77. satır,307. sutun değeri R:69 G:69 B:69  
77. satır,308. sutun değeri R:64 G:64 B:64  
77. satır,309. sutun değeri R:68 G:68 B:68  
77. satır,310. sutun değeri R:71 G:71 B:71

**Şekil 3.8.** Görüntünün RGB çıktısı

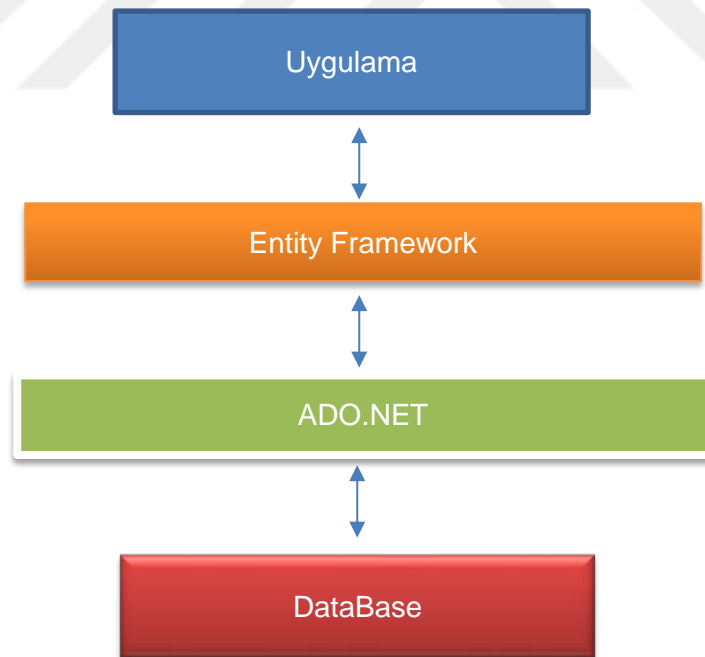
Duygu analiz uygulamamız C# tabanlı masaüstü uygulamasıdır. Uygulamada verilerin kontrol edilmesi ve sonuçların kaydedilmesi için veri tabanı kullanıldı. Veri tabanı bağlantısı için C#'ın bir frameworku olan Entity Framework kullanıldı. Entity Framework'un bir yöntemi olan Veri tabanı öncelikli model (database first) tekniği tercih edildi.

### 3.5.2. Entity Framework ile Paralel C# Programlama Kullanılarak Yüz Görüntülerinin Sınıflandırılması

Entity Framework ilişkisel veri tabanı ile nesneye yönelik programlamada verilerin veri tabanına gönderilmesini sağlayan frameworktur. Veriler Framework üzerinden veri tabanına iletilerek gerekli işlemlerin yapılması sağlanır. Entity framework ile ilişkisel veri tabanı işlemleri model yapısı kullanılarak yapılır [37].

Entity Framework Microsoft tabanlıdır. Veri bağlantıları yapılırken ADO.NET (ActiveX Data Objects.NET) alt yapısı kullanılır. ADO.NET Microsoft tabanlı bir veri erişim ve yönetim teknolojisidir.

Entity Framework ADO.NET teknolojisine göre daha gelişmiş bir yapıdır. ADO.NET'te tüm veri tabanı işlemleri için sorgu cümlecikleri kullanmak gerekir. Sorgu işlemi yapmak güncel teknolojilere göre daha çok işlem yapmayı gerektirir. Entity Framework'te tüm veri tabanı işlemleri; oluşturulan model üzerinden yapılır [37]. Entity Framework iş akışı Şekil 3.9'da gösterilmiştir.



Şekil 3.9. Entity Framework İş Akışı

Entity Framework 4 farklı yöntem kullanır [37]. Bunlar;

- i. **Önce Model (Model First):** Visual Studio ile yeni veri tabanı oluşturularak işlemler gerçekleşir. Veritabanı oluşturulması için model dosyası (.edmx) tanımlanır. Derleme aşamasında veritabanı oluşturulur.
- ii. **Önce Veritabanı (Database First):** Sistemde var olan veritabanı ile işlemler gerçekleşmektedir. Veritabanı sisteme entegre edilerek işlemler yapılır. Veritabanı tabloları entity framework sayesinde model dosyası olarak tanımlanır ve işlemler bu dosyalar üzerinden yapılır.
- iii. **Önce Kod (Code First):** Sisteme kodlama yapılarak oluşturulan veritabanıdır. Bu yöntemde veritabanının oluşması için sınıf (class) tanımlaması yapılarak sağlanır. Veritabanı için istenen tüm değişiklikler bu sınıf içinde tanımlanan metotlarla yapılır.
- iv. **Önce Kod (Code First):** Sistemde var olan veritabanı kullanılır. Veritabanında yapılması gereken değişiklikler sınıflardaki (class) metotlarla yapılır.

Entity Framework kullanmanın avantajları ve dezavantajları vardır [37]. Avantajları şu şekilde sıralanabilir;

- i. Nesne tabanlı proje geliştirmesini sağlar.
- ii. Veri tabanı bilgisi olmadan veri tabanında işlem yapılmasına olanak sağlar.
- iii. Tüm veri tabanı uygulamaları ile kullanılır.
- iv. Code first tekniği ile veri tabanı yedeklenmesine gerek kalmaz. İstenilen yerde veri tabanı oluşturularak işlem yapılabilir.
- v. Yazılım geliştirme süresine ve maliyetlerine olumlu yönde katkı sağlar.

Dezavantajları;

- i. ADO.NET'e göre işlem hızı daha yavaştır.
- ii. Tüm kontrol işlemleri Entity Framework tarafından yapılmaktadır. Buda bazen kolay bir kontrolde karmaşık sorgular oluşturur ve performansı olumsuz yönde etkiler.
- iii. Kullanım dili olarak Linq kullanılır.

Uygulamamızda Entity Framework veri tabanı bağlantısı Şekil 3.10'daki kod satırı ile sağlanmaktadır.

```
public GoruntuAktarEntities ent = new GoruntuAktarEntities();  
//entity framework bağlantı
```

**Şekil 3.10.** Entity Framework Veri Tabanı Bağlantısı

Uygulamada öğrenme tabanlı ve paralel programlama tekniği kullanılarak yapılmıştır. Uygulama yapılırken dağıtık hesaplama tekniği kullanıldı. İki bilgisayar birbirine ağ bağlantısı ile bağlanarak aynı veri setinin iki bilgisayarda da işlem görmesi sağlandı. İki bilgisayar arasında verilerin bir kısmı (1.pc ve 2.pc olarak adlandırıldı) 1.pcde, kalan veriler 2.pcde işlem görmüştür. Şekil 3.11'de görüntülerin işleme alındığı kod satırları gösterilmiştir.

```

if (openFileDialog1.ShowDialog() == DialogResult.OK)
//Görüntü seçme işleminin yapıldığı satır
{
    List<string> folders = new List<string>();
    foreach (var dosyaAdi in openFileDialog1.FileNames)
    {
        folders.Add(dosyaAdi);
    }

    Parallel.ForEach(folders, folder =>
    {
        var isim=dosyaAdi.Split('\\');
        ResimAd = isim[6];

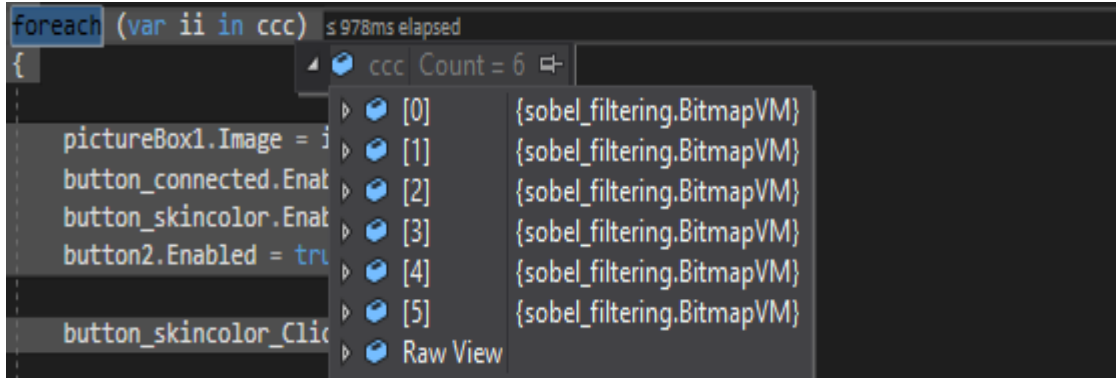
        var sonuc =
            ent.DataAktar.Where(u=>u.ResimAd==ResimAd).FirstOrDefault();
//veri tabanı kontrolü
if(sonuc==null){
        pictureBox1.Image = Image.FromFile(folder);
//işlem yapılan görüntünün gösterilmesi
        file_name = folder;
        DosyaVM dd = new DosyaVM { ThreadID =
            Thread.CurrentThread.ManagedThreadId, ThreadName = folder };
        tdosya.Add(dd);
        contrast_function();
        var c = bitmapdosya;
        var ccc = new List<BitmapVM>();

        foreach (var bmdosya in bitmapdosya)
//resimlerin liste olarak alınması
        {
            if (ccc.Where(u => u.BmID != bmdosya.BmID).Count() <= 0)
            {
                ccc.Add(bmdosya);
            }
        }
}
//işlem satırları

```

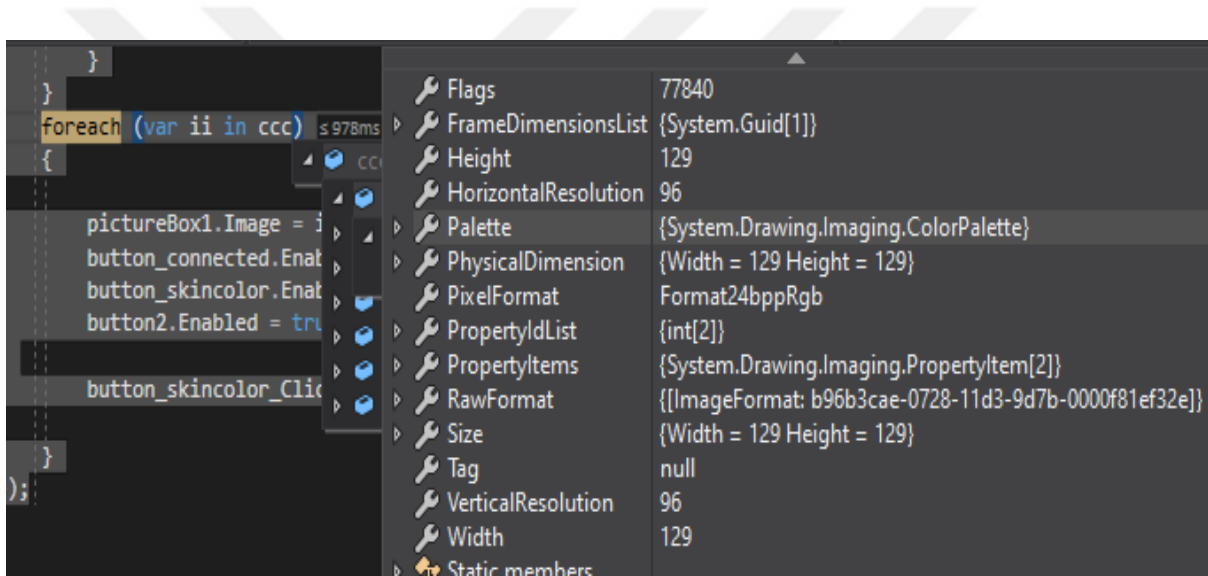
**Şekil 3.11.** Görüntülerin İşleme Alındığı Kod Satırları

Uygulamamız çalıştırıldığında seçilen görüntülerden bir kısmı paralel for ile alınarak işleme başlar. İşleme alınan görüntüler Şekil 3.12'de gösterilmiştir.



**Şekil 3.12.** Paralel Programlama ile İşleme Alınan Görüntüler

Alınan görüntülerden birinin özellikleri Şekil 3.13'te gösterilmiştir.



**Şekil 3.13.** Görüntünün Özellikleri

Uygulamadan alınan görüntülerle ilgili bazı bilgiler Şekil 3.13'te görülmektedir. Özelliklerden bazıları görüntünün bellekteki yeri, görüntünün ismi, genişliği, uzunluğu gibi bilgilerdir.

Uygulamamız fotoğraftaki kişinin sağ göz, sol göz ve ağız kısımlarını alarak bu bölgelerden duygu analizi yapmaktadır. Bu işlemler yapılırken bölgelerin belirlenmesinde sobel kenar bulma algoritması kullanılmıştır. Yatay ve dikey kenarlar matrislerle bulunmuştur.

Sobel filtresinin kullanıldığı kod satırları Şekil 3.14'te gösterilmiştir.

```

public Image sobel(Image im)
{
    int[,] gx = new int[,] { { -1, 0, 1 }, { -2, 0, 2 }, { -1, 0, 1 } };
    // Gx Matrisi
    int[,] gy = new int[,] { { 1, 2, 1 }, { 0, 0, 0 }, { -1, -2, -1 } };
    // Gy Matrisi
    b = (Bitmap)im;
    Bitmap b1 = new Bitmap(im);
    for (int i = 1; i < b.Height - 1; i++)
    // Görüntünün yükseklik piksellerinde dönme işlemi
    {
        for (int j = 1; j < b.Width - 1; j++)
        // Görüntünün genişlik piksellerinde dönme işlemi
        {
            float new_x = 0, new_y = 0;
            float c;
            for (int hw = -1; hw < 2; hw++) //Konvolisyon Matrisi
            {
                for (int wi = -1; wi < 2; wi++)
                {
                    c = (b.GetPixel(j + wi, i + hw).B +
                        b.GetPixel(j + wi, i + hw).R +
                        b.GetPixel(j + wi, i + hw).G) / 3;
                    new_x += gx[hw + 1, wi + 1] * c;
                    new_y += gy[hw + 1, wi + 1] * c;
                }
            }

            if (new_x * new_x + new_y * new_y > 128 * 128)
                b1.SetPixel(j, i, Color.Black);
            else
                b1.SetPixel(j, i, Color.White);
        }
    }
    return (Image)b1;
}

```

**Şekil 3.14.** Sobel Filtreleme İşlemi

Uygulama veri tabanındaki girilmiş değerlere göre sonucu bulan bir uygulamadır. Sisteme başta farklı duyguları içeren görüntüler ve bunların içerdiği duygular girilmektedir. Uygulama gelen görüntüde meydana gelmiş değişiklikleri veri tabanından eşleştirilerek en uygun duyguyu sonuç olarak döndürmektedir.

Uygulamanın duygu sonucunun tespit edilebilmesi için Şekil 3.15'teki test fotoğrafı kullanılmıştır.

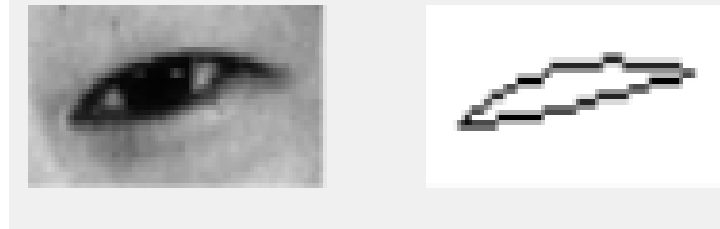


**Şekil 3.15.** Kullanılan Test Fotoğrafi

Kullanılan test görüntüsünden elde edilen sağ ve sol göz modellemesi Şekil 3.16 ve Şekil 3.17’de gösterilmiştir.



**Şekil 3.16.** Fotoğraftaki Sol Göz



**Şekil 3.17.** Fotoğraftaki Sağ Göz

Sağ ve sol gözün bulunmasını sağlayan kod bölümünün bir kısmı

```
private void eyelocal(Bitmap b)
{
    int w = b.Width;//gelen görüntünün genişliği
    int h = b.Height;//gelen görüntünün yüksekliği
    int ys1 = h, ye1 = h - 1, ys2 = h, ye2 = h - 1;
    int i, j, k;
    int mid = 0;//orta nokta
    int max = 0;

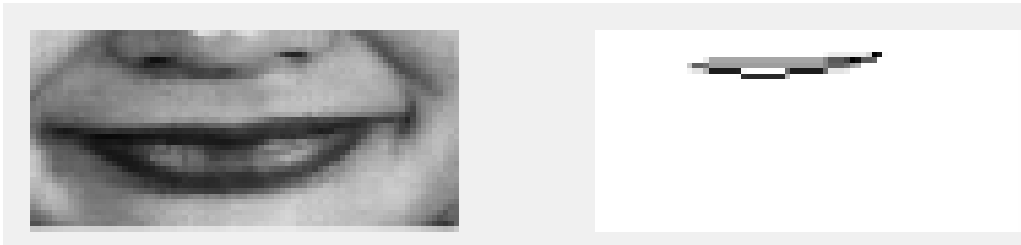
    for (i = 0; i < h; i++)
    {
        b.SetPixel(0, i, Color.White);
        b.SetPixel(1, i, Color.White);
        b.SetPixel(w - 1, i, Color.White);
        b.SetPixel(w - 2, i, Color.White);
    }

    for (i = w / 4; i < w - (w / 4); i++)
    // görüntünün 4 parçaya bölünerek sağ ve sol göz belirleme
    {
        for (j = spq; j < h; j++)
            if (b.GetPixel(i, j).R == 0 && b.GetPixel(i, j).B == 0 &&
                b.GetPixel(i, j).G == 0)
                break;

        if (max < (j - spq))
        {
            max = j - spq;
            mid = i;
        }
    }
    int tp1 = mid - 5, tp2 = mid + 5, mp1 = h - 1, mp2 = h - 1;
    ...
}
```

Şekil 3.18. Göz Belirleme İşlemi

Şekil 3.16 ve Şekil 3.17'deki göz şekilleri bulunurken işlemler Şekil 3.18'de gösterilen kod satırlarında yapılmıştır.



Şekil 3.19. Fotoğraftaki Ağız Şekli

Şekil 3.19'da fotoğraftaki kişinin ağız şekli çıkarılmıştır. Ağız belirleme işleminin yapılmasını sağlayan kod bölümünün bir kısmı Şekil 3.20'de gösterilmiştir.

```

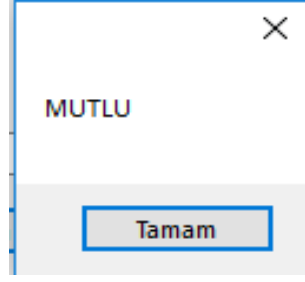
int xel, xe2;
xel = mid;
xe2 = mid;
df = 0;
for (i = mid - 1; i >= 0; i--)
//orta noktadan 1 piksel düşülerek sol ağız kenarı bulunuyor
{
    for (j = ys1; j < ye1; j++)
        if (b.GetPixel(i, j).R == 0 && b.GetPixel(i, j).B == 0 &&
            b.GetPixel(i, j).G == 0)
            //sol ağız kenarı RGB renk kontrolu
            {
                df = 1;
                break;
            }
    if (df == 1)
        // 3 deger 0 olması(siyah) durumunda döngüden çıkılıyor
        break;
    xel = i;
}

df = 0;
for (i = mid + 1; i < w; i++)
//orta noktadan 1 piksel artış yapılarak
// sağ ağız kenarı bulunuyor
{
    for (j = ys2; j < ye2; j++)
        if (b.GetPixel(i, j).R == 0 && b.GetPixel(i, j).B == 0 &&
            b.GetPixel(i, j).G == 0)
            //sağ ağız kenarı RGB Renk kontrolu
            {
                df = 1;
                break;
            }
    if (df == 1)
        // 3 deger 0 olması(siyah) durumunda döngüden çıkılıyor
        break;
    xe2 = i;
}
...

```

**Şekil 3.20.** Ağızın Şeklini Bulmayı Sağlayan Kod Kısmı

Bu ve buna benzer sonuçlar sisteme kaydedilerek sistemdeki fotoğrafları analiz edecek bir yapı oluşturulmaktadır.



**Şekil 3.21.** Fotoğraftaki Duygunun Sonucu

Test fotoğrafından Şekil 3.21'deki sonuç elde edilmiştir. Uygulamamızda yapılan işlemlerde aşağıdaki gibi sonuçlar elde edilmiştir.



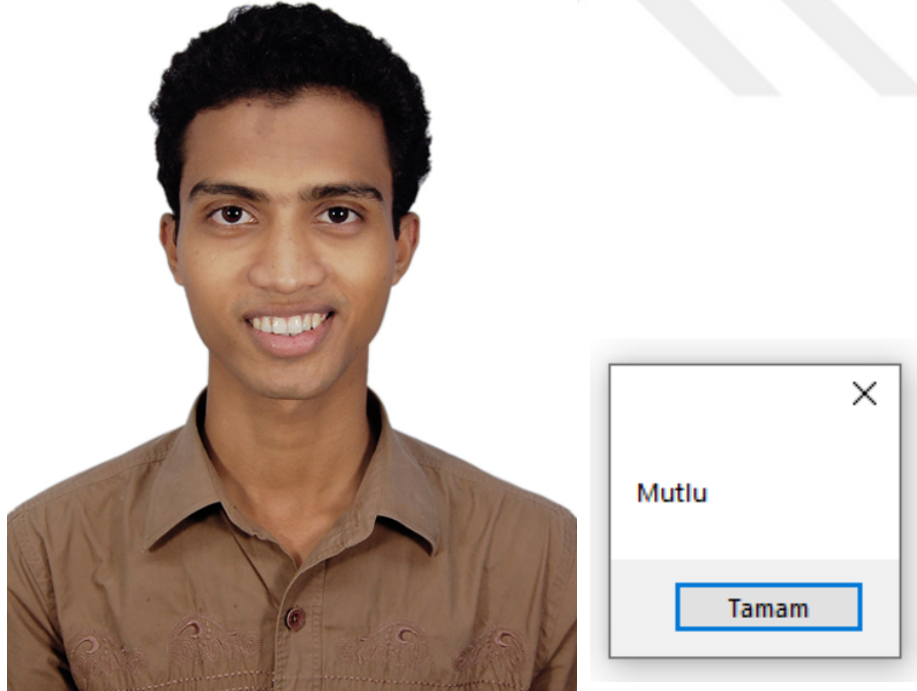
**Şekil 3.22.** Normal Yüz İfadesi

Şekil 3.22 'de uygulamamız normal duygu durumunu elde etmiştir.



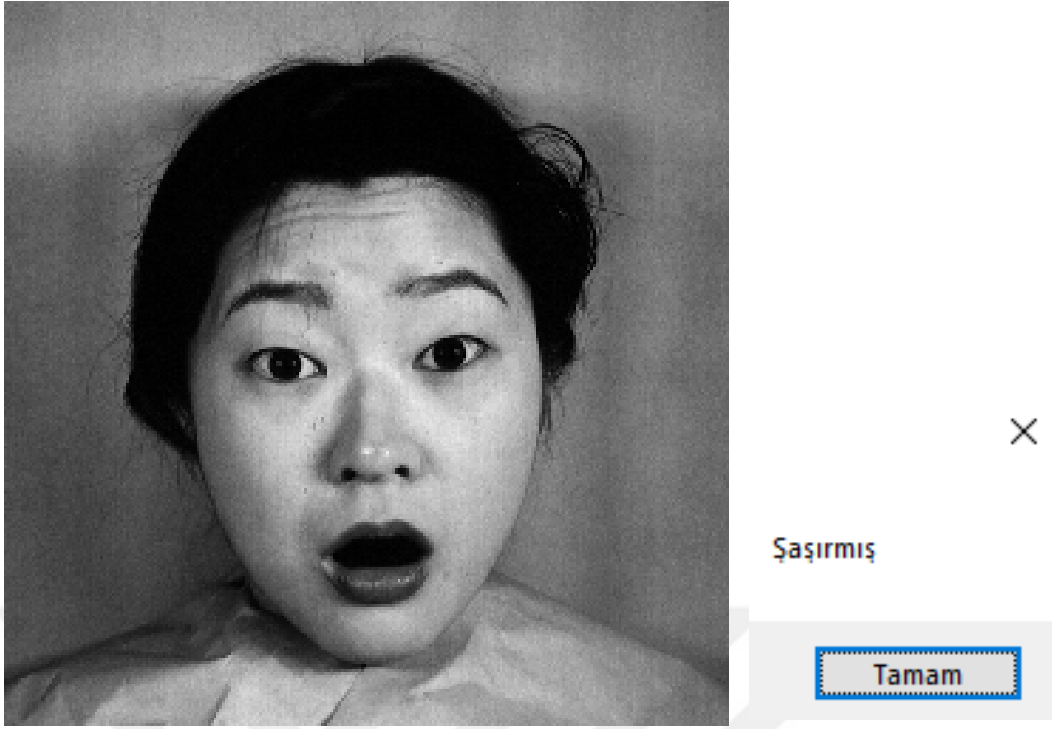
**Şekil 3.23.** Üzgün Yüz İfadesi

Şekil 3.23'te uygulamamız üzgün duygu durumunu elde etmiştir.



**Şekil 3.24.** Mutlu Yüz İfadesi

Şekil 3.24'te uygulamamız mutlu duygu durumunu elde etmiştir.



**Şekil 3.25.** Şaşırmiş Yüz İfadesi

Şekil 3.25'de uygulamamız şaşırmiş duygu durumunu elde etmiştir.



**Şekil 3.26.** Kızgın Yüz İfadesi

Şekil 3.26'da uygulamamız kızgın duygu durumunu elde etmiştir.

Uygulamamız veri tabanına girilen değerlere göre görüntüleri yorumlayarak sonuçları bulabilmektedir. Uygulamadaki doğruluk oranını arttırmak için;

- i. Farklı kişilerin görüntü sayısını arttırmak
- ii. Aynı duygu ifadesi içeren örnek sayısını arttırmak
- iii. Eklenen yüzdeki ifadelerin duygu halini tam yansıtması

gibi durumlar uygulamadaki başarı oranını doğrudan etkilemektedir. Bu işlemlerin yapılması uygulamadaki başarı oranını arttırmaktadır.

Duygu durumunun doğru bulunmasında kişinin yüzünde bu duyguyu içeren mimiklerin tam ve belirgin olması sonucu etkilemektedir. Kişinin yüzündeki mimiklerin tam oluşmaması ya da belirgin olmaması bulunan sonucun hatalı olabilmesine neden olmaktadır.

#### **Master Bilgisayar(pc1)**

**İşlemci:** Intel Core İ7-4720HQ CPU 2.60 GHz

**RAM:** 12 GB

**Sistem Türü:** 64 bit

#### **Slave Bilgisayar(pc2)**

**İşlemci:** Intel Core İ5-3210M CPU 2.50 GHz

**RAM:** 6 GB

**Sistem Türü:** 64 bit

Uygulamada dağıtık sistem kullanılmıştır. Dağıtık sistem işlemi 2 bilgisayar kullanılarak yapılmıştır. Dağıtık sistemde kullanılan bilgisayarların bilgileri yukarıdaki gibidir. Bilgisayarlardan biri master, diğeri slave olarak ayarlanmıştır. Uygulamanın veri taban bilgileri master bilgisayarda yer almaktadır. Yapılan işlemler ağ yolu ile master bilgisayardaki veri tabanına kaydedilmektedir.

Bilgisayarlar aynı ağ tabanında olduğu için birbirleri ile haberleşmekte, veri alış-verişi yapabilmektedir. Ana bilgisayar ip adresi:192.168.0.14 ve bağlanan bilgisayar ip adresi:192.168.0.12 'dir. Bağlantı kontrolü için cmd ekranından ping atılmıştır. Bağlantı sağlandığı Şekil 3.27'de görüntülenmektedir.

```
C:\Users\saris>ping 192.168.0.14

Pinging 192.168.0.14 with 32 bytes of data:
Reply from 192.168.0.14: bytes=32 time=3ms TTL=128
Reply from 192.168.0.14: bytes=32 time=12ms TTL=128
Reply from 192.168.0.14: bytes=32 time=6ms TTL=128
Reply from 192.168.0.14: bytes=32 time=4ms TTL=128

Ping statistics for 192.168.0.14:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 3ms, Maximum = 12ms, Average = 6ms
```

**Şekil 3.27.** Dağıtık Sistem Mimarisinde Bağlantı Kontrolü

Sistemde ağ erişiminde problem olmaması dağıtık sistemin düzgün çalışacağını gösterir.

**Şekil 3.28.** Uygulama Bağlantı Gösterimi

Bağlantı işlemi bilgisayarlar arasında Ethernet kablo yolu ile sağlanmıştır. Bilgisayarlar arasında bağlantı Şekil 3.28’de gösterilmiştir. Uygulama master (sol bilgisayar) ve slave (sağ bilgisayar) bilgisayarda birlikte başlatılmıştır. Birlikte işlem yaptıkları Şekil 3.29’da gösterilmiştir.

### Şekil 3.29. Görüntü ve Duygu İşleme

Uygulama çalıştırılarak veritabanına yapılan işlemlerle ilgili kayıtlar yapılmıştır. Yapılan kayıt bilgilerinde index olarak Id, işlenen görüntünün adı, duygu durumu, işlem yapılan bilgisayar ve işlem tarihi verileri kayıt edilmiştir. Elde edilen veriler Şekil 3.30'da gösterilmiştir.

	ID	ResimAd	DuyguDurum	AktaranPC	Tarih
11	11	22.JPG	Mutlu	pc1	2020-04-28 11:53:56.540
12	12	dg 21774 (8).JPG	Normal	pc2	2020-04-28 11:53:56.117
13	13	28.JPG	Mutlu	pc1	2020-04-28 11:53:56.963
14	14	301.JPG	Kızgın	pc2	2020-04-28 11:53:57.857
15	15	37.bmp	Mutlu	pc1	2020-04-28 11:53:58.113
16	16	dg 21774 (7).JPG	Normal	pc2	2020-04-28 11:53:57.600
17	17	312.JPG	Kızgın	pc1	2020-04-28 11:53:58.657
18	18	222.JPG	Mutlu	pc1	2020-04-28 11:53:59.070
19	19	2222.JPG	Şaşımış	pc1	2020-04-28 11:53:59.750

### Şekil 3.30. Uygulama İşlem Bilgileri

Veritabanı sonuçlarına göre işlemler bilgisayarlar arasında paylaşılmış, işlem gören resimle ilgili tekrar bir işlem yapılmamıştır. Görüntü analiz işlemleri bilgisayarda boş olan iş parçacıkları ile yapılmış, gönderilen tüm resimler analiz edilmiştir.

Duygu sınıflandırma işlemlerinde kullanılan görüntüler yüzün tam karşıdan alınmış halidir. Veri setinde bulunan görüntüler internet ortamından alınmıştır. Gönderilen veri setindeki görüntülerin duygu tipleri sınıflandırılmıştır. Uygulamadan alınan sonuçlara göre Tablo 3.2’de duygu sınıflandırma doğruluk oranları tablosu oluşturulmuştur.

**Tablo 3.2.** Uygulama Duygu Sınıflandırma Doğruluk Oranları Tablosu

<b>Duygular</b>	<b>Normal</b>	<b>Mutlu</b>	<b>Şaşırmış</b>	<b>Üzgün</b>	<b>Kızgın</b>
Duygu Sınıflandırma Doğruluk Oranları	%95	%92	%90	%92	%86

Paralel programlama tabanlı uygulamamızda yapılan testler sonucu yukarıdaki sonuçlar elde edilmiştir. Sonuçlar elde edilirken gönderilen veri setindeki duygu türlerindeki görüntü sayısı ile uygulamadan alınan duygu türlerinin sayısı karşılaştırılmıştır. Elde edilen sonuçların yüzdelik değerleri bulunarak tabloya eklenmiştir.

#### **4. BULGULAR VE TARTIŞMA**

Çalışmamızda yaptığımız duygu analizini işleminde duygunun elde edilmesi kısmında birkaç farklı yol kullanarak duygu verisini en doğru şekilde elde etmeye çalıştık. Duygu verisi elde edilirken birçok faktörün elde edilen veriyi etkilediğini görmüş olduk.

##### **4.1. Ruhsal Etkenlerin Duygu Analizi Üzerine Etkisi**

Günlük yaşantıda farklı duygu halleri içinde olunmaktadır. Birine kızıldığı zaman sinirlenmek, uzun zamandır görülmeyen bir akrabanın görüldüğü zaman sevinmek veya hiç beklemeyen bir zamanda beklenmeyen bir olayın ardından şaşırarak bu duygu hallerinden bazılarını örnek olarak verilebilir. Tabii ki içinde bulunulan ruh hali yüzdeki kaslar yardımıyla mimiklerden ya da yapılan hareketlerden anlaşılabilir.

Duygu analizi işleminde tezimizde daha çok kişinin yüz ve mimik hareketlerinden yararlanarak elde etme yöntemini kullanıldı. Bu yöntemde kişinin duygulara karşı yüzünde bazı kasların gerilip, bazı kasların ise gevşeyerek bu duyguya karşı tepki vermiş olduğunu görüldü. Örnek vermek gerekirse, kişi mutlu olduğu zaman yani gülme eyleminde göz kenarlarındaki kasların gerilerek gözleri kısması bunun yanında gözlerin yanlarında kırışıklar meydana gelmesi, ağız kenarındaki kasların gerilmesi, yanaklardaki kasların gerilerek yanakların belirginleşmesi gibi etkilerin meydana gelmiş olduğunu görüldü. Kişinin yüzünde olan bu değişiklikler kişinin duygu halinin belirlenmesinde kullanılmaktadır.

Kişinin ruhsal durumu duygunun belirlenmesinde önemli rol oynamaktadır. Kişinin ruh halinde olan değişiklikler hareketlerine, tavırlarına ve yüzüne yansımaktadır. Bu değişiklikler çoğu zaman gözle görülebilir derece olabilmektedir. Kişinin ruhsal halindeki değişim yüzündeki kaslarda değişim yapabileceğinden kişinin içinde bulunduğu ruh hali kişinin duygusu etkilemektedir. Ruhsal etkenler kişinin yüzünde hızlı değişimlere neden olabilmektedir. Örnek olarak kişi kızgınsa bu yüzünde kasların gerilmesine, alın kaslarının kasılmasına, dudak kaslarının büzülmesi gibi kişinin yüzünde değişiklikler oluşturmaktadır. Farklı örneklerde de farklı kaslarda olan değişiklikler yüze yansıtacağından kişinin içinde bulunduğu ruhsal durumdan duygusu elde edilebilecektir. Ruhsal durum yüzde değişikliklere neden olduğu için kişinin o an içinde bulunduğu duygu durumu tespit edilebilmektedir.

## 4.2. Çevresel Etkenlerin Duygu Analizi Üzerine Etkisi

Günlük yaşantı esnasında fark edilen ya da fark edilemeyen birçok çevresel etken hayatı yönlendirmektedir. Bu çevresel etkenlere birkaç örnek vermek gerekirse etraftaki sesler, ışıklar, farklı kokular gibi birçok örnek verebilir. Bu etkenler farkında olunmadan yaşantımızı birçok zaman kolaylaştırmak üzerine kullanılabilir. Bu etkenler aynı zamanda kişinin duygusal durumunu da etkilemektedir.

Çevresel etkenler kişide değişik duygu hallerinin oluşmasında olanak sağlamaktadır. Örnek olarak kulağa gelen güzel bir müzik mutlu edebilirken, bir korna sesi yada gürültü o anda sinirlendirebilmektedir.

Çevresel faktörler kişilerin yaşantısını etkilemektedir. Çevresel faktörlerin etkisi duygular üzerinde de hissedilmektedir. Bu faktörlerin etkileri kişinin yüzünde ve hareketlerinde değişikliklere neden olmaktadır. Kişinin yüzünde olan bu değişiklikler mimik hareketlerinden fark edilebilmektedir. Değişen mimikler kişinin o an içinde bulunduğu duygu hakkında bilgi vermektedir. Şekil 4.1’de kişilerin farklı duygular içinde olduğu gösterilmektedir.



**Şekil 4.1.** Farklı Seslerin Yüz İfadesinde Meydana Getirdiği Değişiklikler [25]

### 4.3. Duygu Analizinde Görüntü İşleme

Günümüzde görüntü işleme (image processing) birçok alanda kullanılmaya başlamıştır. Görüntü işleme görüntüyü dijital form haline getirmek, bazı işlemleri gerçekleştirmek, spesifik görüntü elde etmek veya bu görüntüden yararlanarak bilgi çıkarma yöntemine denir.

Bu yöntem için girilen veri bir fotoğrafa da görüntü olabilir. Fakat görüntünün çıktısı ise bilgi veya görüntünün dikkat edilmesi gereken kısmı olabilir [26]. Şekil 4.2’de görüntü işleme akış şeması gösterilmiştir.

Görüntü işleme işlemi 3 adımdan oluşmaktadır.

1. Görüntünün sisteme aktarılması
2. Görüntüye işlemler uygulayarak görüntünün analiz edilmesi
3. Görüntü çıktısının hazırlanarak kullanılabilir veri halinde alınabilmesi

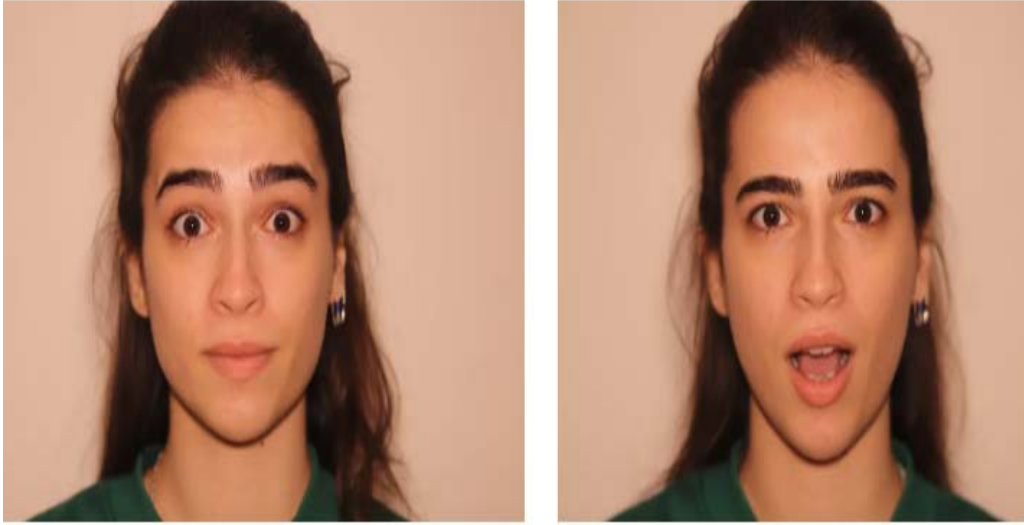


Şekil 4.2. Görüntü İşleme Akış Şeması [27]

Duygu analizinde ise görüntü işleme tekniğinin büyük önemi bulunmaktadır. Görüntü aktarılırken görüntünün iyileştirilmesi ve görüntüdeki eksik noktaların giderilerek aktarılması önemlidir. İyileştirilmiş olan görüntülerde detayların fark edilebilir olması önem arz etmektedir. Çünkü herhangi bir duygusal değişimde yüzdeki mimiklerde değişimler olacağı için bunların görüntü işlem çıktısında fark edilebilmesi gerekmektedir. Bu değişimlerin şekli ve yönü hangi duygu halinde bulunduğu konusunda bilgi vermektedir.

Değişimler için yapılan araştırmalar ve çalışmalar sonucunda bu hareketlerin yönleri, şekilleri ve hangi duyguyu içerdiği konusunda bir veri tabanı elde edilmiştir. Bu veri tabanından yararlanılarak duygunun türünün belirlenmesi sağlanmıştır.

Görüntülerde işlem yapılırken göz kenarlarındaki ve ağız çevresindeki değişimlerden yararlanılmaktadır. Gözlerde oluşan kırışıkların, ağız hareketlerinin görüntüde fark edilmesi daha kolay olmaktadır.



**Şekil 4.3.** Farklı Duygularla Yüzdeki Değişimler [28]

Şekil 4.3'te verilmiş olan örnekte farklı duygularda yüzde oluşan bazı değişimler görülmektedir.

#### 4.4. Yüz İfadelerinin Sınıflandırılması

Duygu analizi yapılırken yüzdeki hareketler o an hangi duygunun hissedildiği konusunda bilgi verebilmektedir. Genel olarak incelendiğinde kişilerin aynı duygulara verdiği tepkilerin yakın olduğu sonucu elde edilmiştir. Bu tepkiler belirlenerek sınıflandırılmış ve veri tabanına bu sınıflandırmaya uygun olarak eklenmiştir.

Sınıflandırma yapılırken kas hareketlerinden yararlanılmış ve duygu hallerine göre kaslarda meydana gelen değişiklikler sınıflandırılarak kaydedilmiştir.

Evrensel olarak duyguların yüzlere yansımaları çok yakın şekillerde meydana gelmektedir. Aşağıdaki örneklerde de görülebileceği gibi duygular yüzdeki belirli bölgelerdeki kas hareketleri ile meydana gelmektedir. Duygular ifade edilirken yüzdeki 3 bölgede belirgin değişiklikler meydana gelmektedir. Bu 3 bölge şu şekilde sıralanabilir;

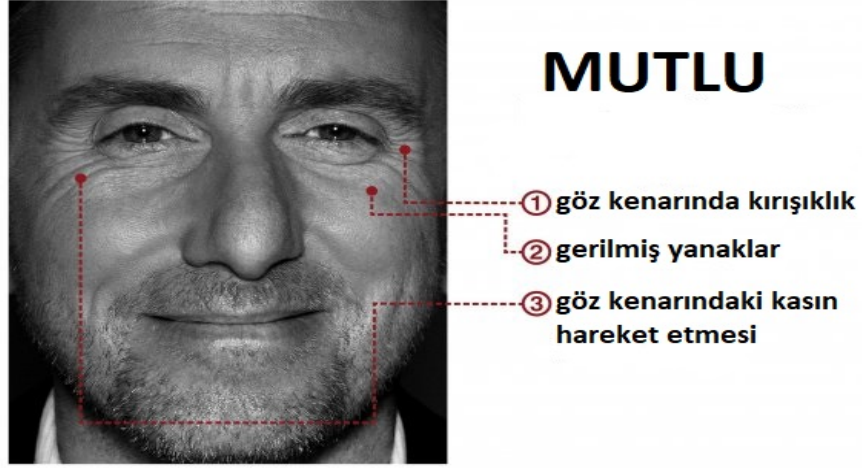
1. Kaş bölgesi
2. Göz ve göz kapaklarının bulunduğu bölge
3. Yüzün alt kısım (ağız, dudak) bölgesi

Bu bölgelerdeki kasların hareket etmesiyle yüzdeki duygu durumları oluşmaktadır.

Yüzlerde meydana gelen ifadelerin süreleri duygu durumunun algılanmasında belirgin bir rol oynamaktadır. Kısa sürede yüzden kaybolan ifadeler doğru algılanamayıp yanlış yorumlanabilir. Yüzlerde bazen çok küçük hareketler meydana gelebilir. Bu hareketler mikro ifade olarak isimlendirilmektedir [42]. Mikro ifadeler genellikle duygu kontrolü ya da duygunun hafif hissedilmesi durumunda yapılan küçük hareketlerdir. Örnek vermek gerekirse küçük bir espride sadece ağız kenarındaki küçük hareket mikro ifade olarak değerlendirilmektedir.

Evrensel olarak kabul edilen 6 duygu çeşidi bulunmaktadır. Bu duygu ifadelerinde yüzde bazı değişiklikler meydana gelmektedir. Bu değişiklikler hissedilen duyguya göre kişinin verdiği tepkilerdir. Bu tepkiler oluşurken yüzdeki kaslarda gerilmeler veya büzülme meydana gelmektedir. Duygu durumuna göre yüzdeki kas hareketleri şu şekilde gerçekleşmektedir.

**Mutluluk:** Gözler kenarında kaslar sayesinde kısılır, göz kenarlarında kırışıklıklar meydana gelir, yanaklar daha belirgin hale gelir [40]. Şekil 4.4'te mutluluk durumu gösterilmiştir.



Şekil 4.4. Mutluluk Duygu Halinde Yüzdeki Değişimler [40]

**Üzüntü:** Ağız kenarındaki kaslarda büzülme meydana gelir, dudak kenarları aşağı kayar, üst göz kapağı düşer, gözün odak noktasında kayma meydana gelir [40]. Şekil 4.5'te üzgün durum gösterilmiştir.



Şekil 4.5. Üzgün Duygu Halinde Yüzdeki Değişimler [40]

**Korku:** Dudaklarda gerilme meydana gelir, üst göz kapakları açılır, alt göz kapakları gerilir, kaşlar yukarı doğru kalkar [40]. Şekil 4.6'da korku durumu gösterilmiştir.



Şekil 4.6. Korku Duygu Halinde Yüzdeki Değişimler [40]

**Kızgın:** Dudaklar büzülerek ağız düz bir durum alır, Kaşlar aşağı iner çatılır ve gözler daha belirgin olarak ortaya çıkar [40]. Şekil 4.7'de kızgın durum gösterilmiştir.



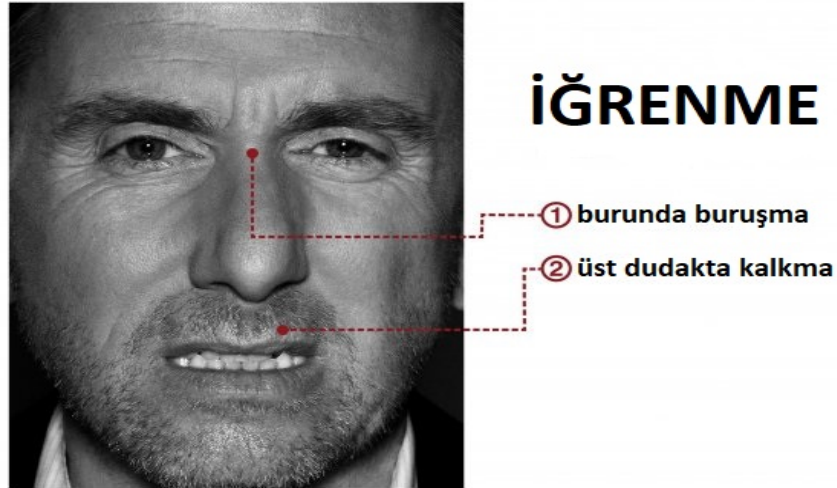
Şekil 4.7. Kızgın Duygu Halinde Yüzdeki Değişimler [40]

**Şaşırma:** Ağız açılarak duygu belli edilir. Kaşlar yukarı doğru kalkar ve göz kapakları iyice açılarak gözler belirgin hale gelir [40]. Şekil 4.8’de şaşırma durumu gösterilmiştir.



**Şekil 4.8.** Şaşırma Duygu Halinde Yüzdeki Değişimler [40]

**İğrenme:** Ağız yanlara doğru genişler, dudaklar kasılır üst dudak kalkar. Yüzdeki kaslar kasılır. Burunda buruma meydana gelir [40]. Şekil 4.9’da iğrenme durumu gösterilmiştir.



**Şekil 4.9.** İğrenme Duygu Halinde Yüzdeki Değişimler [40]

Yukarıda belirtilen özellikler başlıca duygularda meydana gelen yüzdeki kas hareketleridir.

#### 4.5. Yüz İfadelerinin Sınıflandırılmasında Doğruluk

Yüz ifadelerinin sınıflandırılmasında bize yardımcı olan programlar öğrenmeye dayalı olarak işlem yapabilmektedir. Buda programlarda belirli bir sayıda öğretici verilerin girilmesini zorunlu hale getirmektedir. Girilen öğretici veriler programlardaki veri tabanında tutulmaktadır. Gönderilen görüntüye göre işlem yapılmaktadır.

Yeni gelen veri programa öğretici nitelikte olarak girilmişse veri tabanına kayıt edilirken, analiz için girilen veri sonuçları veri tabanında aranarak girilen veriye en yakın sonuç bulunur ve o sonucun duygu durumu ekranda görüntülenir. Yapmış olduğumuz programın doğruluk oranını etkileyen iki ana unsur bulunmaktadır. Bunlar;

- i. Veri tabanında öğretici olarak kullanılan veri sayısı
- ii. Kaydedilen öğretici verilerdeki çeşitlilik

Programlardaki öğretici veri sayısı doğruluk yüzdesini etkilemektedir. Kullanılan öğretici veri sayısı arttıkça programın sonuçlarındaki doğruluk oranı da olumlu bir şekilde artış göstermektedir. Analiz işlemi sırasında gelen veriye göre yapılan sorgulamada bulunan en yakın sonuç en doğru olanı bulmada yardımcı olmaktadır. Veri sayısındaki çeşitlilik gelen veriye göre en yakın sonuçları bulma ihtimalini arttırdığı için en yakın duygunun tahmin edilerek görüntülenmesini sağlamaktadır.

Programdaki doğruluk oranını etkileyen diğer bir durum ise öğretici nitelikte kullanılan verilerdeki çeşitliliğidir. Kullanılan yüzlerin farklı olması programın veri tabanındaki çeşitliliği arttırdığı için program duygu sonuçlarını verirken daha doğru kararlar vermesini sağlamaktadır. Aynı duygular farklı yüzlerde yakın tepkiler vermektedir fakat verilen tepkinin boyutu yüzdeki kasların hareketine göre değişiklik göstermektedir. Veri tabanında aynı duygunun birçok yüzde oluşturduğu sonuçlardaki çeşitlilik analiz için gönderilmiş olan veriye en yakın sonucun bulunmasında kolaylık sağlar. Programdaki veri çeşitliliğinin artması programın doğru sonuçlar oluşturmasına etki etmektedir.

#### 4.6. Kullanılan Görüntünün Özellikleri

Uygulamanın daha doğru ve net sonuçlar verebilmesi için kullanılan görüntünün özellikleri önemlidir. Kullanılan görüntüdeki;

- i. Görüntünün kalitesi
- ii. Görüntü üzerindeki gürültüler
- iii. Zayıf aydınlatmalar

gibi özellikler kullanılan görüntünün kalitesini etkilemektedir. Kullanılan görüntüyü daha iyi hale getirmek için üzerinde iyileştirme çalışmaları yapılmaktadır.

Görüntü üzerinde iyileştirme yaparken görüntünün kontrast, parlaklık gibi değerlerinde düzenlemeler yapılmaktadır [41]. Görüntü iyileştirme işleminin ardından elde edilmiş olan görüntü orijinal görüntüden daha iyi ve anlaşılır durumdadır. Görüntü işlemede kullanılan yöntemler görüntüye öznedir. Her görüntüdeki bozulma aynı olmadığı için kullanılan yöntemlerde aynı değildir. Genel olarak görüntü iyileştirmede kullanılan bazı yöntemler [41];

- i. Histogram Eşleme
- ii. Histogram Denkleştirme
- iii. Yeğnlik Dönüşüm Fonksiyonları
- iv. Fourier Dönüşümleri

gibi teknikler kullanılmaktadır. Bu tekniklerin kullanılmasının ardından görüntüdeki değişimlerin kişilerin görsel algısıyla fark edilebilmektedir. Bu sonuç temel tekniklerle başarının sağlanabildiğini göstermektedir.

Görüntünün iyileştirilmesi işlemlerinden biride görüntünün onarım işlemidir. Görüntü onarım işlemi, bozuk veya bozulmuş görüntüyü ilk haline getirme işlemidir.

Görüntülerde bozulma işlemi görüntünün gönderim ve alım işlemi sırasında oluşmaktadır. Görüntülerdeki bozulmalar görüntüye gürültü eklenmesiyle ortaya çıkmaktadır. Görüntü onarım işleminde görüntüdeki gürültüyü tespit etmek ve bu gürültüyü azaltmak için çeşitli yöntemler bulunmaktadır. Görüntü onarımında kullanılan bazı yöntemler [41];

- i. Dalgacık tabanlı görüntü onarımı
- ii. Bulanık tabanlı görüntü onarımı
- iii. İç boyama kullanarak görüntü onarımı
- iv. Maske işleme veya Filtreleme
- v. Nokta işleme
- vi. Bazı temel yoğunluk dönüşümleri

- vii. Bazı temel gri-seviye dönüşümleri
- viii. Histogram

kullanılan başlıca onarım yöntemleridir.

Bu işlemler uygulanarak görüntüdeki bozulmalar düzeltilmektedir. Görüntülerdeki gürültüler azaltılarak daha iyi bir görüntü elde edilmektedir.

#### 4.7. Seri ve Paralel Programlama

Sistemde kullanılan paralel programlama yöntemi sayesinde aynı anda birden fazla görüntünün işlenmesi sağlanmıştır. Sistemde paralel programlama sayesinde aynı zaman dilimi içerisinde birden fazla görüntünün işlenmesi sağlanmıştır.

Sistemde seri programlama kullanılmış olsaydı işlemlerin bitiş süresinin daha uzun olduğu tespiti yapılmıştır. Seri programlama; aynı zaman diliminde bilgisayarın bir işlemcisinde tek bir işlem adımının işlem yapmasını sağlamaktadır. Seri programlamada bir işlem bitmeden başka bir işlem çalıştırılmaz.

Paralel programlama farklı bir işlemin başlaması için bir önceki işlemin bitmesi gerekmemektedir. İşlemler daha küçük parçalara bölünerek farklı işlemcilerde işlem görebilmektedir. Aynı anda farklı işlemciler üzerinde işlemler yapılarak birim zamanda daha fazla işlem yapılabilmektedir.

Görüntü işleme ve sınıflandırma işleminde aynı anda bir tek işlemin yapılması görüntü işlem süresini olumsuz etkileyecektir. Bundan dolayı sistemimizde paralel programlama yöntemi tercih edilmiştir.

Program C#'ta for ve parollefor komutları kullanılarak aradaki hız farkları ölçülmüştür. Ölçüm değerleri elde edilirken C# timer aracı kullanılmıştır. Ölçüm değerleri;

- i. Parallefor görüntü sınıflandırma işlem süresi 0,673 saniye değeri elde edilmiştir.
- ii. For işleminde görüntü sınıflandırma işlem süresi 0,952 saniye değeri elde edilmiştir.

Elde etmiş olduğumuz sonuçlara göre paralel görüntü sınıflandırma işlemi 1,4 kat daha hızlı işlem yaptığı tespit edilmiştir.

Literatürde duygu analizi ve sınıflandırma ile ilgili olarak farklı çalışmalar yapılmıştır. Yapılan çalışmalarda kas hareketlerinin yüzde meydana geldiği değişimler baz alınarak duygu analizi ve sınıflandırma işlemleri yapılmıştır.

Sınıflandırma başarı oranları hesaplamasında işlemler belirli kriterler doğrultusunda yapılmaktadır. Başarı işlemleri yapılırken doğruluk(accuracy) sonuçlarından yararlanır. Bazı doğruluk sonuçları için duyarlılık(recall) ve kesinlik(precision) özelliklerine bakmak gerekmektedir. Duyarlılık ve kesinlik işlemlerinde karmaşıklık matrisinden(confusion matrix) yararlanılmaktadır [43]. Karmaşıklık matrisi Tablo 4.1'de gösterilmiştir.

**Tablo 4.1.** Karmaşıklık Matrisi

	<b>Tahmin 0</b>	<b>Tahmin 1</b>
<b>Gerçek 0</b>	Doğru Negatif (TN)	Yanlış Pozitif (FP)
<b>Gerçek 1</b>	Yanlış Negatif (FN)	Doğru Pozitif (TP)

$$\text{Doğruluk(accuracy)} = \frac{TP+TN}{TP+TN+FN+FP} \quad (4.1)$$

Doğruluk formülü 4.1'de gösterilmiştir.

$$\text{Duyarlılık(recall)} = \frac{TP}{TP+FN} \quad (4.2)$$

$$\text{Kesinlik(precision)} = \frac{TP}{TP+FP} \quad (4.3)$$

Duyarlılık ve kesinlik formülleri 4.2 ve 4.3'te gösterilmiştir [43].

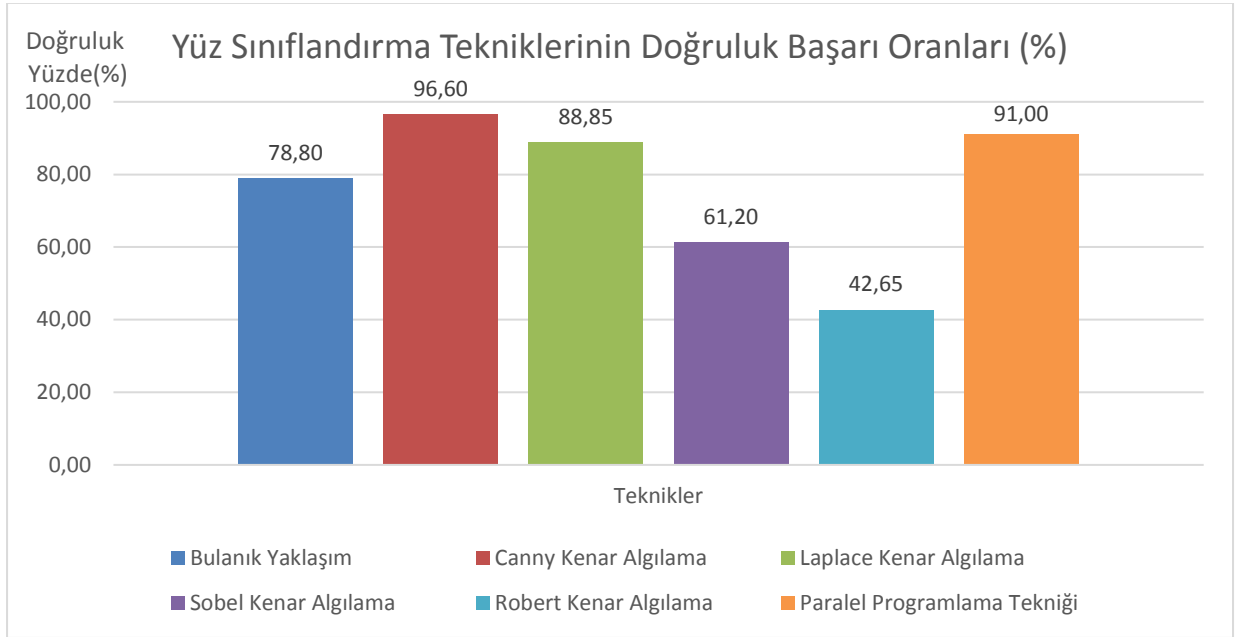
Duyarlılık ve kesinlik değerlerinin harmonik ortalamasının alınması işlemi F-Ölçüsü(F-Measure) formülü kullanılır. Harmonik ortalama sayesinde uç durumdaki değerlerde işleme alınır.

F-Ölçüsü Formülü;

$$F - \text{Ölçüsü}(F - \text{Measure}) = 2 * \frac{\text{Duyarlılık} * \text{Kesinlik}}{\text{Duyarlılık} + \text{Kesinlik}} \quad (4.4)$$

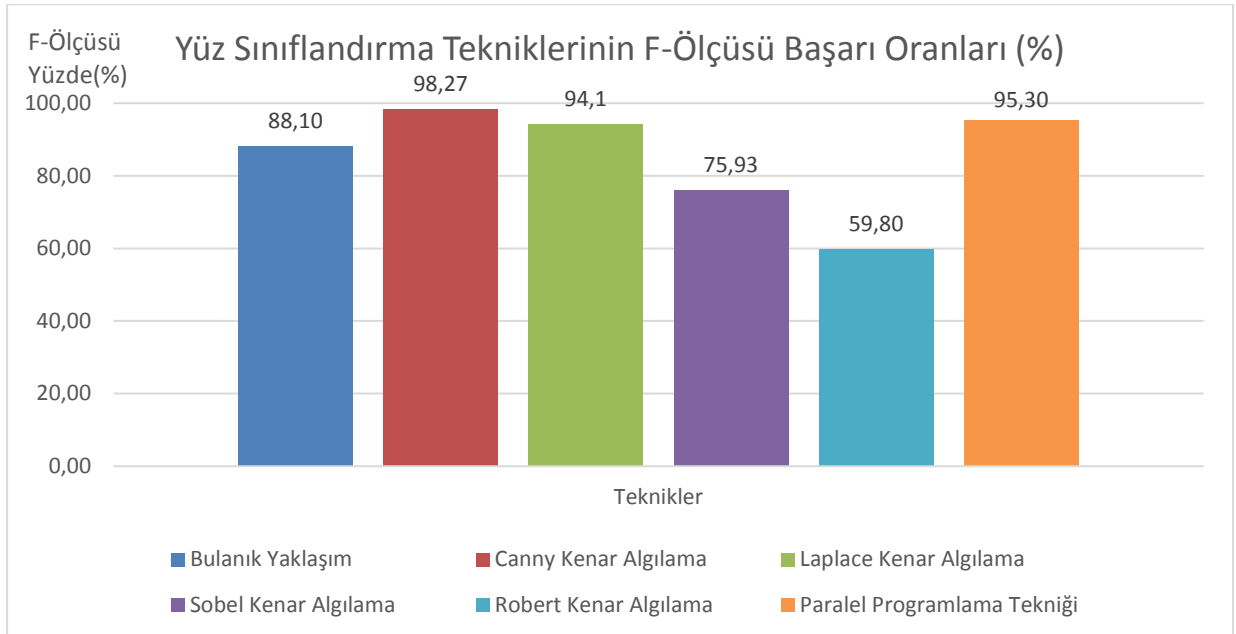
4.4'te gösterilmiştir [43].

Yüz sınıflandırma işlemlerinde kullanılan teknikler ve doğruluk başarı oranı tablosu Şekil 4.10'da gösterilmiştir.



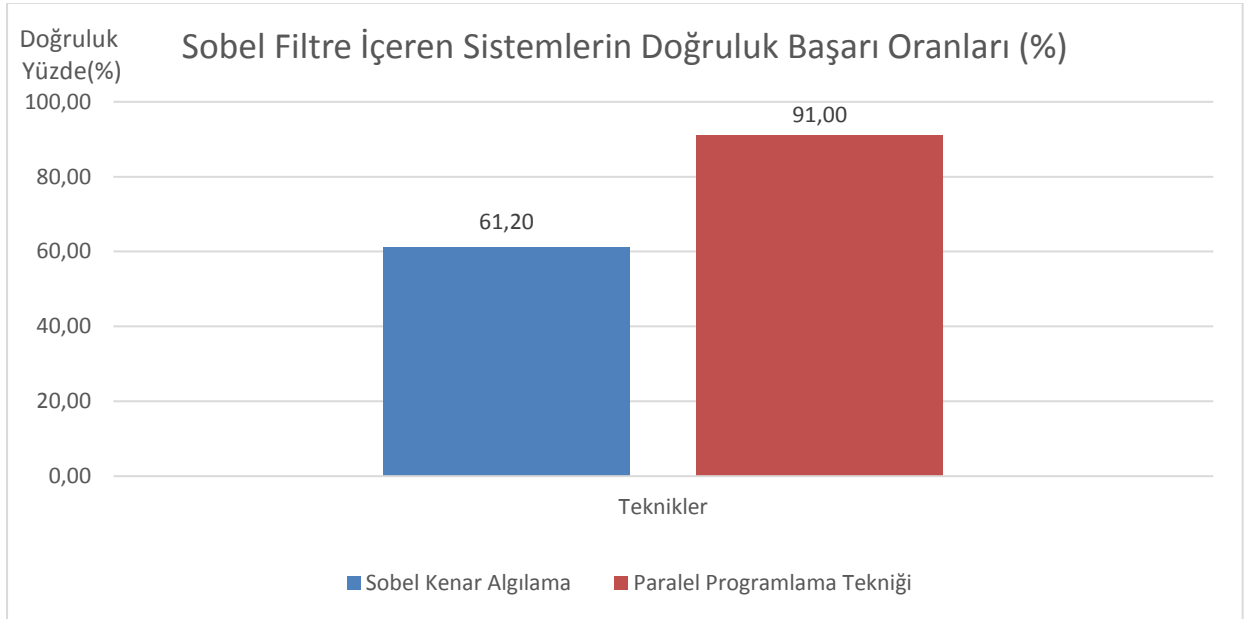
**Şekil 4.10.** Yüz Sınıflandırma Tekniklerinin Doğruluk Başarı Oranları

F-Ölçüsü ile hesaplanan yüz sınıflandırma oranları Şekil 4.11'de gösterilmiştir.



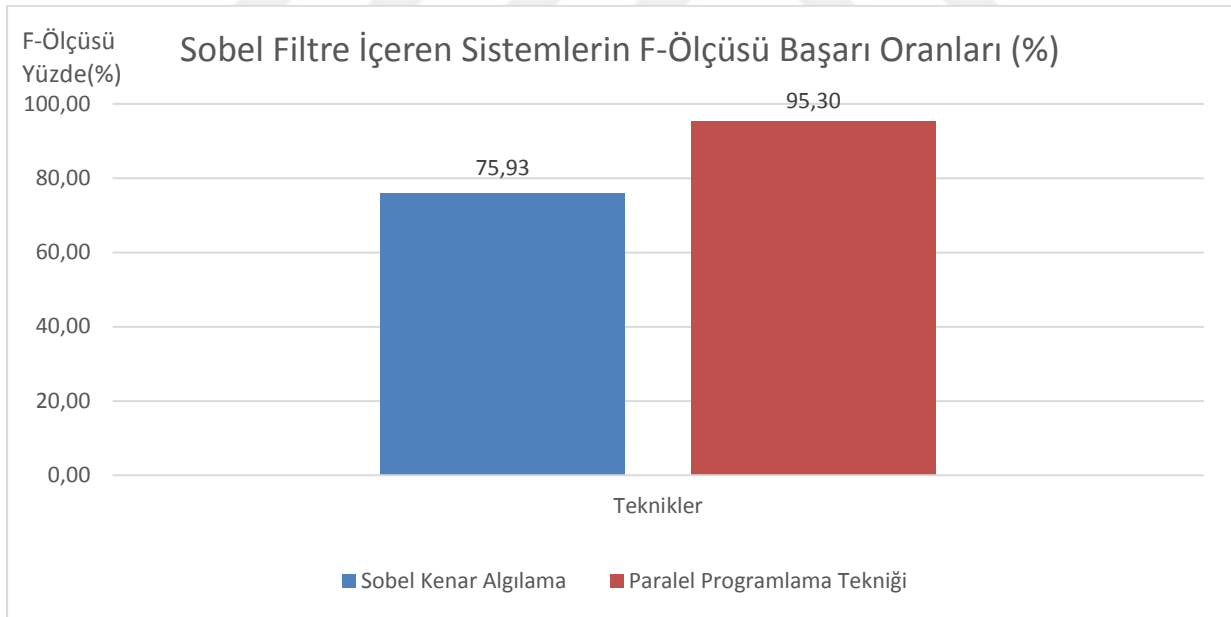
**Şekil 4.11.** Yüz Sınıflandırma Tekniklerinin F-Ölçüsü Başarı Oranları

Sistemde yüz tanıma işleminde sobel filtresi kullanılmıştır. Aynı filtre kullanan iki sistemin başarı oranları Şekil 4.12’de gösterilmiştir.



Şekil 4.12. Sobel Filtre İçeren Sistemlerin Doğruluk Başarı Oranları

F-Ölçüsü ile hesaplanan başarı oranları Şekil 4.13’te gösterilmiştir.



Şekil 4.13. Sobel Filtre İçeren Sistemlerin F-Ölçüsü Başarı Oranları

Paralel programlama sistemimizde işlemler bilgisayarın işlemcisi üzerinde yapılmıştır. İki görüntünün işlenmesi arasında geçen ortalama süre hesaplanmıştır. İki görüntünün işlenebilmesi için geçen ortalama süre 0,673 sn olarak hesaplanmıştır.

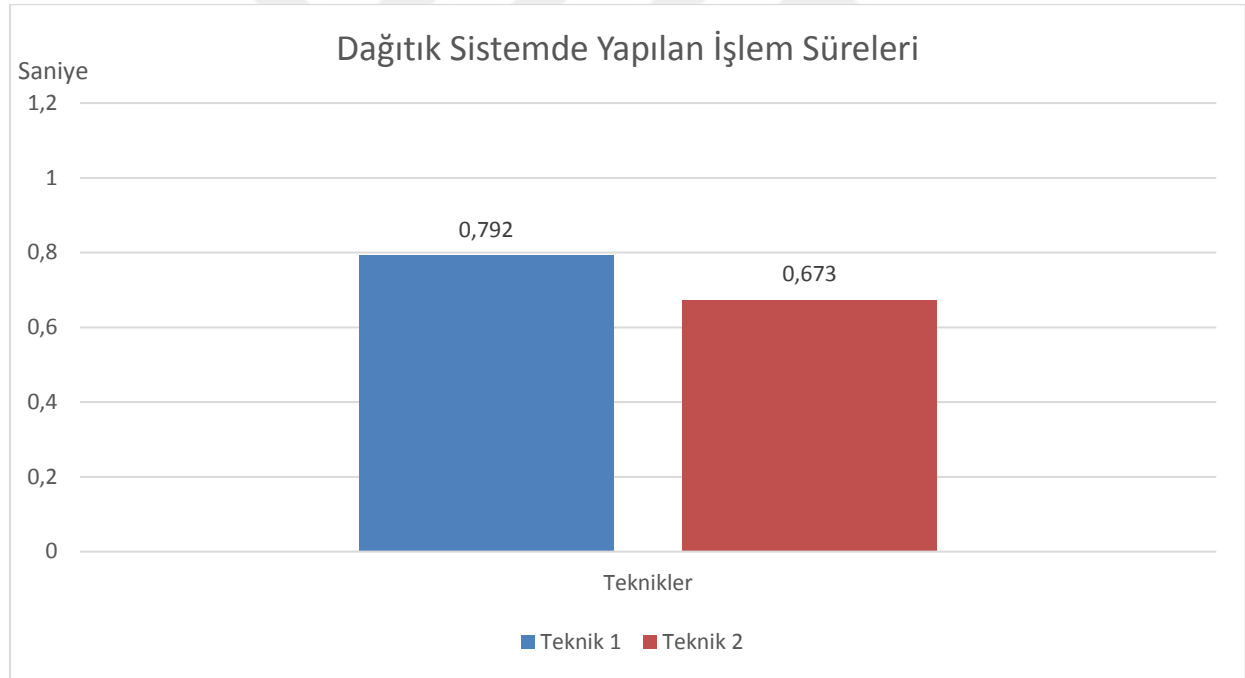
Sistemimizde dağıtık sistem kullanılmıştır. Dağıtık sistemdeki bilgisayarlar ana(master) bilgisayar ve diğer(slave) bilgisayar olarak ayarlanmıştır. Dağıtık sistem üzerinde iki farklı işlem tekniği uygulanmıştır.

Tekniklerin uygulama şekli ve sonuçları:

**Teknik 1:** Dağıtık sistemdeki ana bilgisayarda ve diğer bilgisayarda işlenecek olan görüntüler rasgele olarak işlem yapılmıştır.

**Teknik 2:** Dağıtık sistemdeki ana bilgisayar ve diğer bilgisayarda işlenecek olan görüntüler belirlenmiş ve ilgili bilgisayarda işlemler yapılmıştır.

Sistemler ayrı ayrı hazırlanarak çalıştırılmış ve ortalama süre hesaplaması yapılmıştır. Uygulanan teknikler arasında ortalama olarak 0,119 sn'lik bir zaman farkı elde edilmiştir. Elde edilen bu zaman farkının sebebi işlemler yapılmadan önce veri tabanından ilgili görüntünün sorgusunun yapılmasıdır. Şekil 4.14'te teknikler ve elde edilen süreler gösterilmiştir.



**Şekil 4.14.** Paralel Programlama Kullanılan Farklı Teknikler ve Süreleri

## 5. SONUÇLAR VE ÖNERİLER

Yapılan tez araştırmamızda duyguların oluşumunu birçok faktörün etkilediği sonuçlarını elde edildi. Bu faktörlerin çevresel ve psikolojik olarak ayrıldığı tespit edilmiştir. Kişinin içinde bulunduğu psikolojik halinin veya bulunduğu ortamın kişinin duygularına direk etki ettiği konusunda bilgiler elde edilmiştir. Kişi bu faktörlere karşı mimik ve hareketleriyle tepkiler oluşturduğu araştırmalar yardımıyla bulunmuştur.

Duyguların algılanmasında mimik hareketlerinin önemini anlaşılmış ve bu hareketlerden yola çıkarak kişinin o an hangi duygu içinde olduğu konusunda bilgi sahibi olunmuştur. Kişinin duygu durumuna göre verdiği tepkiler ile ilgili bilgiler toplamıştır. Bu toplanan bilgilerle veri setimiz oluşturulmuştur. Oluşturduğumuz veri setine farklı kişilerin farklı duygu durumlarındaki görüntüleri yer almaktadır. Kişilerin mimiklerindeki ifadeler alınarak görüntüdeki duygu durumu uygulamanın test değerlerine göre tahmin edilmiştir. Elde edilen sonuçlar uygulama başarı tablosunda gösterilmiştir.

Yapılan tez çalışmasında farklı programlama dilleri ve yöntemleri kullanılmıştır. Kullanılan programlama dilleri MATLAB ve C# 'tır. Bu alanlarla ilgili yaptığımız programlarda Matlab programı görüntüleri sadeleştirmiş, C# programı duygu durumlarının tahminlerini yapmıştır. C# uygulamasında paralel programlama tekniği kullanılmıştır. Dağıtık sistem kullanılarak işlemlerin bir kısmı ağa dahil olan diğer bilgisayarda yapılmıştır.

Sistemimizde kullandığımız filtrelerde sayesinde datalardaki bozulma ve gürültüleri en aza indirerek doğru sonuçları elde etme yolunda başarı sağlanmıştır. Bu tez çalışmasında sobel filtresi kullanılmıştır. Kullanılan sobel filtresi sayesinde gerçeğe yakın duygu değer tahminleri yapılmış, sonuçlarda yüksek doğruluk oranları elde edilmiştir.

## KAYNAKLAR

- [1] Cömert, Z. Programlama Nedir?(2015). 27 Kasım 2018 tarihinde <http://www.zafercomert.com/IcerikDetay.aspx?zcms=10> adresinden erişildi
- [2] Programlama Dilleri Özellikleri (2017). 23 Aralık 2018 tarihinde [http://www.robotiksistem.com/programlama\\_dilleri\\_ozellikleri.html](http://www.robotiksistem.com/programlama_dilleri_ozellikleri.html) adresinden erişildi.
- [3] Paralel Programlama Nedir? 25 Aralık 2018 tarihinde <https://www.hermesiletisim.net/dev/paralel-programlama-nedir-2#.XB6VulwzblU> erişildi.
- [4] Büyük Veri (2017). 29 Aralık 2018 tarihinde <http://docplayer.biz.tr/50652395-Yuksekogretimde-buyuk-veri-ve-ogrenme-analitikleri-guncel-teori-ve-uygulamalar.html> adresinden erişildi.
- [5] Big Data Nedir? (2018). 10 Ocak 2019 tarihinden <https://netvent.com/big-data-nedir/> adresinden erişildi.
- [6] Doğan K., Arslantekin S. (2016), Büyük Veri: Önemi, Yapısı ve Günümüzdeki Durum, *DTCF Dergisi* 56.
- [7] Sosyal Ağ Analizi nedir? 25 Ocak 2019 tarihinde <https://saaacikders.wordpress.com/> adresinden erişildi.
- [8] Big Data. 29 Ocak 2019 tarihinde <http://www.hightechnology.com.tr/sayfa.asp?id=604> adresinden erişildi.
- [9] Akça, Ş. Dağıtık Sistemler Programlama (2015). 05 Şubat 2019 tarihinde <https://www.slideshare.net/saho10/datk-sistemler-programlama> adresinden erişildi.
- [10] Seker, S.E. (2016). Duygu Analizi. 07 Şubat 2019 tarihinde <https://docplayer.biz.tr/34348590-Duygu-analizi-sentimental-analysis.html> adresinden erişildi.
- [11] Gündoğan, M. (2018). Duygu Analizi Nedir? 15 Şubat 2019 tarihinde <https://www.bilgidairesi.com/2018/06/19/sentimental-analiz-duygu-analizi-nedir/> erişildi.
- [12] Atasoy, H. (2011). Gabor Filtresi. 18 Şubat 2019 tarihinde <http://huseyinasoy.com/Gabor-Filtresi> adresinden erişildi.
- [13] Facial Action Coding System . 03 Mart 2019 tarihinde [https://en.wikipedia.org/wiki/Facial\\_Action\\_Coding\\_System](https://en.wikipedia.org/wiki/Facial_Action_Coding_System) adresinden erişildi.
- [14] Face Detection. 20 Mart 2019 tarihinde <http://mplab.ucsd.edu/grants/project1/research/face-detection.html> adresinden erişildi.
- [15] M. S. Bartlett, P. A. Viola, T. J. Sejnowski, B. A. Golomb, J. Larsen, J. C. Hager ve P. Ekman (1996), "Classifying facial action", *Advances in Neural Information Processing Systems* ,pp. 823-829
- [16] Y. Yacoob ve L. S. Davis (1996), "Recognizing human facial expressions from long image sequences using optical flow", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, cilt 18, no. 6, pp. 636-642.

- [17] S. B. Gokturk, J. Y. Bouguet, C. Tomasi ve B. Girod (2002), "Model-based face tracking for viewindependent facial expression recognition", Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition (FGR'02).
- [18] X. Feng, M. Pietikäinen ve A. Hadid (2005), "Facial expression recognition based on local binary patterns and linear programming", Pattern Recognition and Image Analysis, cilt 15, no. 2, p. 546-548.
- [19] S. M. Lajevardi ve M. Lech (2008), "Facial expression recognition from image sequences using optimized feature selection", 23rd International Conference Image and Vision Computing (IVCNZ), New Zealand.
- [20] F. S. Farahani, M. Sheikhan ve A. Farrokhi (2013), "A fuzzy approach for facial emotion recognition", 13th Iranian Conference on Fuzzy Systems (IFSC), Qazvin.
- [21] CHEN X., CHENG W. (2015), International Journal of Computer Science & Engineering Survey (IJCSSES) Vol.6, No.2, April 2015 DOI:10.5121/ijcses.2015.6201 FACIAL EXPRESSION RECOGNITION BASED ON EDGE DETECTION ,College of Mechanical Engineering, Shanghai University of Engineering Science, Shanghai 201620, China
- [22] EM Algoritması. 15 Eylül 2019 tarihinde <https://echo108.wordpress.com/tag/em-algoritmasi/> adresinden erişildi
- [23] MATLAB Nedir? 22 Eylül 2019 tarihinde [www.yildiz.edu.tr/~yates/Temel\\_Bilgiler.doc](http://www.yildiz.edu.tr/~yates/Temel_Bilgiler.doc) adresinden erişildi.
- [24]Karaman, A.C.(2018). Resim Düzenlemek. 10 Ekim 2019 tarihinde <https://slideplayer.biz.tr/slide/13348602/> adresinden erişildi.
- [25] Farklı Yüz İfadeleri. 17 Kasım 2019 tarihinde <https://www.genelsaglikk.com/> adresinden erişildi.
- [26] Görüntü İşleme (2017). 25 Kasım 2019 tarihinde <http://blog.udentify.co/04/2017/goruntu-isleme-nedir/> adresinden erişildi.
- [27] Mary, R. (2011). Intreoduction to Image Processing. 03 Aralık 2019 tarihinde [https://www.engineersgarage.com/article\\_page/introduction-to-image-processing/](https://www.engineersgarage.com/article_page/introduction-to-image-processing/) adresinden erişildi.
- [28] Emotions. 10 Aralık 2019 tarihinde <https://evrimagaci.org/duygular-ve-yuz-ifadeleri-yuzumuz-duygularimizi-nasil-ifade-eder-7555> adresinden erişildi.
- [30] Veri Analizi Nedir? 20 Aralık 2019 tarihinde <https://proente.com/veri-analizi-nedir/> adresinden erişildi.
- [31] Tahmini Analiz, 23 Aralık 2019 tarihinde <https://www.sahteyazilim.com/predictive-analytics-nedir-tahmini-analiz-yazilimlari/> adresinden erişildi.
- [32] A.BOZKURT (2016),"Öğrenme analitiği: e-öğrenme, büyük veri ve bireyselleştirilmiş öğrenme", Açıköğretim uygulamaları ve Araştırmaları Dergisi, Cilt 2 , Sayı 4 , sayfa 55-81.

- [33] İçerik Analizi. 28 Aralık 2019 tarihinde <https://prezi.com/3vkqpxf8pio/icerik-analizi/> adresinden erişildi.
- [34] Laplace Edge Detection. 10 Mart 2020 tarihinde <https://www.owlnet.rice.edu/~elec539/Projects97/morphjrks/laplacian.html> adresinden erişildi.
- [35] Kenar Bulma Algoritmaları. 16 Mart 2020 tarihinde [http://www.ibrahimcayiroglu.com/Dokumanlar/GoruntuIsleme/Goruntu\\_Isleme\\_Ders\\_Notlari-7.Hafta.pdf](http://www.ibrahimcayiroglu.com/Dokumanlar/GoruntuIsleme/Goruntu_Isleme_Ders_Notlari-7.Hafta.pdf) adresinden erişildi.
- [36] Paralel Programlama. 23 Mart 2020 tarihinde <https://www.csharpstar.com/parallel-programming-csharp/> adresinden erişildi.
- [37] Entity Framework. 07 Nisan 2020 tarihinde <https://medium.com/t%C3%BCrkiye/entity-framework-nedir-bff3943bec72> adresinden erişildi.
- [38] Paralel Programlama. 10 Nisan 2020 tarihinde <http://www.bayramucuncu.com/task-parallel-library-tlp-c/> adresinden erişildi.
- [39] Multithreading Programlama. 12 Nisan 2020 tarihinde <https://erdincuzun.com/ileri-python/multithreading-programlama/> adresinden erişildi.
- [40] Duygu Durum Görüntüleri. 23 Nisan 2020 tarihinde <http://www.fanpop.com/clubs/lie-to-me/images/3998280/title/emotions-photo> adresinden erişildi.
- [41] Görüntü Onarımı. 27 Nisan 2020 tarihinde [https://www.researchgate.net/publication/328192821\\_goruntu\\_iyilestirme\\_ve\\_goruntu\\_onarma\\_teknikleri\\_ile\\_yapilmis\\_uygulamalar](https://www.researchgate.net/publication/328192821_goruntu_iyilestirme_ve_goruntu_onarma_teknikleri_ile_yapilmis_uygulamalar) adresinden erişildi.
- [42] Mikro İfade. 03 Mayıs 2020 tarihinde <https://tr.wikipedia.org/wiki/Mikro-ifadeler> adresinden erişildi.
- [43] Duyarlılık ve Kesinlik. 06 Mayıs 2020 tarihinde [https://en.wikipedia.org/wiki/Precision\\_and\\_recall](https://en.wikipedia.org/wiki/Precision_and_recall) adresinden erişildi.

## ÖZGEÇMİŞ

**Adı ve Soyadı** : Timuçin KORKMAZ

**Doğum Tarihi** : 08.12.1991

**E-mail** : timucinkorkmaz33@gmail.com / t.korkmaz@kardelenyazilim.com

**Öğrenim Durumu** :

Derece	Bölüm/Program	Üniversite	Yıl
Lisans	Bilgisayar Mühendisliği	Karabük Üniversitesi	06-2015
Yüksek Lisans	Bilgisayar Mühendisliği	Mersin Üniversitesi	07-2020
Doktora			

## ESERLER (Makaleler ve Bildiriler)

1. JSON and XML Performance Comparisons in a Restful Web Service ALİ ŞENTÜRK, TİMUÇİN KORKMAZ (26.10.2016 - 28.10.2016 ) , Yayın Yeri:1st International Mediterranean Science and Engineering Congress , 2016

2. Android Adım Sayar Uygulaması, Timuçin Korkmaz, Yasin Ortakçı, Yayın Yeri: Akademik Bilişim 2015