

**KARADENİZ TEKNİK ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**

**BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**

**BİLGİSAYARLI DUDAK OKUMA**

**YÜKSEK LİSANS TEZİ**

**Zafer YAVUZ**

**Haziran 2007**

**TRABZON**

**KARADENİZ TEKNİK ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**

**BİLGİSAYARLI DUDAK OKUMA**

**Bilgisayar Müh. Zafer YAVUZ**

**Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsünce  
“Bilgisayar Yüksek Mühendisi”  
Unvanı Verilmesi İçin Kabul Edilen Tezdir.**

**Tezin Enstitüye Verildiği Tarih : 07.06.2007**

**Tezin Savunma Tarihi : 29.06.2007**

**Tez Danışmanı : Prof. Dr. Vasif V. NABIYEV**

**Jüri Üyesi : Prof. Dr. Rifat YAZICI**

**Jüri Üyesi : Yrd. Doç. Dr. Ayten ATASOY**

**Enstitü Müdürü : Prof. Dr. Emin Zeki BAŞKENT**

**Trabzon 2007**

## ÖNSÖZ

İnsan bilgisayar etkileşiminde bilgisayara insana özgü davranışlar kazandırılması son zamanlarda önemli çalışma alanlarından biridir. Bilgisayarlı dudak okuma fikri de bu amaç doğrultusunda ortaya çıkmıştır. Böylesine bir çalışma konusunu bana öneren ve beni yönlendiren, hiçbir zaman desteğini esirgemeyen, Yapay Zeka ve Algoritmalar gibi bilgisayar mühendisliği eğitiminde önemli bir yere sahip kitapların yazarı Sayın Danışman Hocam Prof. Dr. Vasif V. NABİYEV'e sonsuz teşekkür ve şükranlarımı sunarım.

Yüksek lisans çalışmalarım sırasında fikirlerine ve bilgilerine başvurduğum, başta Sayın Hocam ve Bilgisayar Mühendisliği Bölüm Başkanı Prof. Dr. Rıfat YAZICI olmak üzere diğer KTÜ Bilgisayar Mühendisliği Bölümü öğretim üyelerine teşekkürlerimi sunarım.

Çalışmalarım sırasında zorlu geçen süreçlerde yardımlarını esirgemedikleri ve gösterdikleri anlayıştan dolayı Öğretim Görevlisi Ömer ÇAKIR, Araştırma Görevlisi Eyüp GEDİKLİ, mezun öğrencilerimizden Esra YILMAZ ve diğer lisans ve yüksek lisans öğrencilerine teşekkür ederim.

Her şeyden önce benim bugünlere gelmemi sağlayan, her türlü fedakârlığı gösteren ve her zaman yanımda olan aileme saygı ve sevgilerimi sunarım.

Tez yazım aşamasında benden hiçbir zaman desteğini esirgemeyen ve her zaman yanımda olan sevgili eşim Zeynep YAVUZ'a gösterdiği sabır ve manevi desteğinden dolayı teşekkür etmeyi bir borç bilirim.

Bu çalışmanın hazırlanması esnasında bana desteği olan adını sayamadığım nice insana ve Karadeniz Teknik Üniversitesi'ne teşekkür ederim.

Zafer YAVUZ  
Trabzon, 2007

## İÇİNDEKİLER

	<u>Sayfa No</u>
ÖNSÖZ.....	II
İÇİNDEKİLER.....	III
ÖZET .....	VII
SUMMARY.....	VIII
ŞEKİLLER DİZİNİ.....	IX
TABLolar DİZİNİ.....	XII
1. GENEL BİLGİLER.....	1
1.1. Giriş.....	1
1.1.1. Dudak Okuma Gereksinimleri.....	3
1.1.2. Otomatik Dudak Okuma Sistemleri.....	4
1.1.3. Sistemin Çalışma Şeması.....	5
1.2. İnsan Dudağının Yapısı.....	7
1.3. Renk Uzayları.....	8
1.3.1. RGB Renk Uzayı.....	8
1.3.2. CIE Lab Renk Uzayı.....	10
1.3.3. HSI ya da HSV Renk Uzayları.....	11
1.3.4. CMY ve CMYK.....	11
1.3.5. YCbCr Renk Uzayı.....	12
1.3.6. YIQ ya da YUV Renk Uzayları.....	12
1.3.7. TSL Renk Uzayı.....	13
1.4. Görüntü İyileştirme Teknikleri.....	14
1.4.1. Ortalama Alma (Mean Filter).....	14
1.4.2. Bir Görüntü Üzerinde Maske Uygulama.....	15
1.4.3. Histogramın Çıkarılması.....	17
1.4.4. Gri Seviye Resimde Histogram.....	19
1.4.5. Renkli Resimde Histogram.....	19
1.4.6. Histogram Yumuşatma (Histogram Smoothing).....	20
1.4.7. Orta Değer Süzgeci (Median Filtresi).....	22
1.4.8. Normalleştirme.....	23

1.4.9.	Histogram Eşitleme .....	23
1.5.	Yatay ve Düşey İz düşüm .....	26
1.5.1.	Yatay İz düşüm .....	26
1.5.2.	İkili Resimde Yatay İz Düşüm .....	27
1.5.3.	Gri Seviye Resimde Yatay İz düşüm .....	27
1.5.4.	Renkli Resimde Yatay İz Düşüm .....	28
1.5.5.	Düşey İz Düşüm .....	29
1.6.	Morfolojik İşlemler .....	30
1.6.1.	Yayma (Dilation) .....	30
1.6.2.	Aşındırma (Erosion) .....	31
1.6.3.	Açma (Opening) İşlemi .....	32
1.6.4.	Kapama (Closing) .....	33
1.7.	Görüntü Eşikleme (Image Thresholding) .....	34
1.7.1.	Giriş .....	34
1.7.2.	Global Eşikleme .....	34
1.7.3.	Değişken (Dinamik) Eşikleme .....	35
1.8.	Bazı İstatistiksel Temeller .....	36
1.8.1.	Standart Sapma .....	37
1.8.2.	Varyans .....	38
1.8.3.	Kovaryans .....	38
1.8.4.	Kovaryans Matrisi .....	41
1.8.5.	Matris Cebiri .....	41
1.8.6.	Öz Vektörler .....	41
1.8.7.	Öz Değerler .....	42
1.8.8.	Öz Değer ve Öz Vektörlerin Hesaplanması .....	43
1.9.	Temel Bileşen Analizi (Principle Component Analysis, PCA) .....	44
1.9.1.	Analiz Edilecek Veri Kümesini Elde Etme .....	44
1.9.2.	Ortalamaların Elde Edilmesi .....	45
1.9.3.	Kovaryans Matris Hesabı .....	45
1.9.4.	Kovaryans Matrisinden Öz Değer ve Öz Vektörleri Elde Etme .....	46
1.9.5.	Öz Vektörleri Seçme ve Transformasyon Matrisini Oluşturma .....	46
1.9.6.	Yeni Veri Kümesinin Türetilmesi .....	47

2.	YAPILAN ÇALIŞMALAR, BULGULAR VE İRDELEME .....	49
2.1.	Giriş .....	49
2.2.	Resimde Yüzün Belirlenmesi .....	49
2.3.	Ten Rengi Bulma Algoritmaları .....	50
2.3.1.	Sınırlamalar Yardımıyla Ten Renginin Bulunması .....	50
2.3.2.	YCbCr Renk Uzayı ile Ten Renginin Belirlenmesi .....	50
2.3.3.	Sabit Arka Planlı (siyah) Resimlerde Yüzün Belirlenmesi .....	53
2.4.	Dudak Bölgesinin Bulunması .....	56
2.4.1.	Gözün Bulunması .....	56
2.4.2.	Gözün Bulunması .....	58
2.5.	Dudak Tespiti .....	60
2.5.1.	Dinamik Eşiklemeye Dayalı Dudak Tespiti .....	60
2.5.2.	Renkler Arasındaki Bağıntılardan Yararlanarak Dudak Tespiti.....	65
2.5.3.	HSI Renk Uzayı Kullanarak Dudak Tespiti .....	67
2.5.4.	Renk Bileşenlerinin Güçlendirilerek Dudağın Tespit Edilmesi .....	68
2.6.	Özellik Çıkarma.....	69
2.6.1.	Alt ve Üst Dudak Koordinatlarının Bulunması .....	69
2.6.2.	Sağ ve Sol Dudak Koordinatlarının Bulunması .....	70
2.6.3.	Ağız Açıklığının Bulunması.....	70
2.6.4.	Alt ve Üst Dişlerin Görünüp görünmediğinin İncelenmesi.....	71
2.6.5.	Dudağın Alanının Bulunması .....	72
2.6.6.	Dudağın Eninin ve Boyunun Bulunması .....	72
2.6.7.	Dudağın Dış Sınırlarının Bulunması .....	72
2.6.8.	Oluşturulan Özellik Matrisi.....	73
2.6.9.	Özellik Matrisindeki Sütunların Anlamları .....	74
2.7.	Dudak Bölgesi İzleme .....	75
2.7.1.	Giriş .....	75
2.7.2.	İzlenecek Dudak Bölgesinin Temsil Edilmesi.....	76
2.7.3.	Dudak Bölgesi Takibinin Gerçeklenmesi.....	76
2.8.	Temel Bileşen Analizi (PCA) Algoritması.....	78
2.9.	PCA'nın Bilgisayarlı Dudak Okumaya Uygulanışı .....	80
2.9.1.	Eğitim Aşaması .....	80

2.9.2. Dudak Okuma Aşaması .....	83
3. SONUÇLAR VE ÖNERİLER .....	85
3.1. Yüz Tespiti ile İlgili Sonuçlar .....	85
3.1.1. Sabit Arka Plana Göre Yüz Tespiti .....	85
3.1.2. YCbCr Renk Uzayı ile Ten Renginin Belirlenmesi .....	87
3.2. Dudak Bölgesi Bulma.....	88
3.3. Dudak Tespiti Sonuçları .....	90
3.4. Dudak Okuma Sonuçları .....	93
3.5. Dudak Okumayı Etkilen Faktörler ve Öneriler .....	97
4. KAYNAKLAR.....	99
ÖZGEÇMİŞ.....	101

## ÖZET

İnsan bilgisayar etkileşiminde konuşma tanıma sistemleri son yıllarda önemli bir çalışma konusu haline gelmiştir. Konuşan kişinin ses bilgisinin alınarak bu bilgilerin bilgisayarda işlenmesi karmaşık hesapsal süreçleri beraberinde getirmektedir. Bu karmaşıklık sesin tanınmasında önemli bir problem oluşturmaktadır. Bugüne kadar yapılan araştırmalara göre konuşma ile ilgili bilgiler, sadece ses işaretinde değil aynı zamanda görsel işaretlerde de bulunmaktadır. Görsel işaretler olarak beden dili, işaret dili, mimik ve jestler ile dudak hareketleri kabul edilebilir. Çalışmada bu görsel işaretlerden olan dudak hareketleri üzerinde durulmuştur ve bilgisayarlı dudak okuma sistemi geliştirilmiştir.

Bilgisayarlı dudak okuma, bir görüntüden hiçbir ses bilgisine ihtiyaç duymadan kişinin ne söylediğinin anlaşılması işlemidir. Tezde bilgisayarlı dudak okuma için gerekli çalışma şeması verilmiş ve sistemi oluşturan alt modüller ayrı ayrı gösterilmiştir.

Bilgisayarlı dudak okuma için öncelikle yüzün tespit edilmesi işlemi gerçekleştirilmiştir. Tespit edilen yüz üzerinde dudak okuma gerçekleştirebilmek için önce dudak bölgesi ve ardından dudağın ikili görüntüsü elde edilmiştir. Bu aşamada dudak bölgesi bulma ve dudak tespiti için yeni yöntemler önerilmiştir. Dudağın ikili görüntüsü elde edildikten sonra bilgisayarlı dudak okumada kullanılacak özellikler çıkarılmıştır. Türkçede sesli harflerden (a, e, ı, i, o, ö, u, ü) oluşan hecelerin söylendiği 56 farklı görüntü alınmıştır. Bu görüntüler kullanılarak PCA yöntemine dayalı bir tanıma sistemi gerçekleştirilmiştir. 56 görüntüden 24 tanesi eğitim için 32 tanesi ise test için kullanılmış ve sistemde %25 oranında başarımlar sağlanmıştır. Ayrıca test aşamasında yakın benzerlik gösteren sesliler (ı-i, o-ö, u-ü) düşünüldüğünde bu başarımlar %50 civarında olmaktadır.

**Anahtar Kelimeler:** Dudak Okuma, İşaret Dili, Dudak Tespiti, Temel Bileşen Analizi (PCA).



## SUMMARY

### **Automatic Lipreading**

Speech recognition systems have recently been one of the major study areas in human-computer interaction. Audio processing in computers consists of very complex operations. This complexity results in significant problems. Recent studies show that not only audio but also visual signs include information related with speech. Body language, sign language, gestures and lip movements may be accepted as visual signs. In this study, lip movements are examined, and an automatic lipreading system is implemented.

Automatic lipreading is a process of understanding what a person says from a scene without a need of audio information. In this thesis required flowchart for automatic lipreading is given and the modules of the system are shown separately.

First, face detection operation was done for automatic lipreading. After that in order to be able to perform lipreading, lip area was found and binary lip image was obtained. In this phase, new methods were suggested to find lip area and detect lip. Next, features for automatic lipreading were extracted. 56 different videos including Turkish vowels (a, e, ı, i, o, ö, u, ü) were recorded. A recognition system based on PCA method was realized by using these videos. 24 of these videos were used for training and 32 of them were used for testing and 25% of success was achieved. Also, in the test phases the success is increased to 50% if misrecognized similar vowels (o-ö, ı-i, u-ü) are not taken in to account.

**Keywords:** Lipreading, Sign Language, Lip Detection, Principle Component Analysis (PCA).

## ŞEKİLLER DİZİNİ

	<u>Sayfa No</u>
Şekil 1.1. Çeşitli dudak türleri .....	1
Şekil 1.2. Ses ve İşaret dilinin arasındaki ilişki .....	3
Şekil 1.3. Sistemin çalışma yapısı .....	6
Şekil 1.4. Çeşitli dudak türleri .....	8
Şekil 1.5. RGB renk küpü.....	9
Şekil 1.6. Orijinal resim ve kırmızı yeşil ve mavi bileşenleri .....	10
Şekil 1.7. 3x3 lük ortalama alma maskesi .....	15
Şekil 1.8. Orijinal resim ve resme ortalama alma işleminin uygulanması .....	15
Şekil 1.9. 5x5 lik ortalama filtresi .....	16
Şekil 1.10. Örnek histogram grafiği .....	18
Şekil 1.11. Gri seviye resimde histogram.....	19
Şekil 1.12. R, G ve B bileşenlerine ait histogram grafikleri.....	20
Şekil 1.13. Histogram yumuşatma işlemi .....	21
Şekil 1.14. Histogram yumuşatma işleminin sonucu işleminin sonucu .....	22
Şekil 1.15. Histogram eşitleme işlemini sonucu.....	24
Şekil 1.16. Histogram eşitleme işleminin görüntü üzerindeki sonucu .....	25
Şekil 1.17. Farklı resimler için histogram eşitleme grafikleri .....	25
Şekil 1.18. İkili resimde yatay iz düşüm.....	27
Şekil 1.19. Gri seviye resimde yatay iz düşüm.....	28
Şekil 1.20. Renkli resimde R bileşenine ait yatay iz düşüm grafiği .....	29
Şekil 1.21. İkili, gri seviye ve renkli resimlerde düşey iz düşümler.....	30
Şekil 1.22. Yayma işlemi.....	31
Şekil 1.23. Aşındırma işlemi .....	32
Şekil 1.24. Açma işlemi.....	33
Şekil 1.25. Kapama işlemi .....	33
Şekil 1.26. Global eşikleme .....	35
Şekil 1.27. Otsu algoritması sonucu elde edilen ikili resim .....	36

Şekil 2.1	Ten renginin tespit edilmesi .....	51
Şekil 2.2.	YCbCr alt uzayına geçiş ve Cb'Cr' uzayına dönüşüm(c ve d). .....	53
Şekil 2.3.	İkili resimde bölütleme işlemi .....	54
Şekil 2.4.	Sabit arka plana göre yüzün bulunması.....	55
Şekil 2.5.	Çeşitli resimlerden tespit edilen yüzler .....	55
Şekil 2.6.	Yüz görüntüsü ve yüz görüntüsünün düşey dağılım grafiği.....	57
Şekil 2.7.	Dağılım düzgünleştirme sonucu .....	58
Şekil 2.8.	Göz konumunun tespit edilmesi .....	59
Şekil 2.9.	Dudak bölgesinin bulunması için kullanılan oranlar.....	59
Şekil 2.10.	Ortalama alma işleminin uygulaması .....	61
Şekil 2.11.	Histogram dağılım grafiği .....	62
Şekil 2.12.	Elde edilen eşik değerine göre remin ikiliye çevrilmesi.....	62
Şekil 2.13.	Simetrik katlama işlemi .....	63
Şekil 2.14.	Dudak eni ve boyu.....	63
Şekil 2.15.	Dudağın elde edilmesi .....	64
Şekil 2.16.	Dudağın dış sınırlarının bulunması .....	64
Şekil 2.17.	Dış sınırları temsil edecek noktalar .....	65
Şekil 2.18.	Elde edilen noktaların ağırlık merkezine olan uzaklıklarının grafiği ....	65
Şekil 2.19.	Dudak resmi ve K faktörüne göre dudak şeklinin elde edilmesi.....	66
Şekil 2.20.	Farklı c değerlerine göre elde edilen dudak yapıları .....	66
Şekil 2.21.	Dudağa ait piksellerin belirlenmesi için kullanılan filtre .....	67
Şekil 2.22.	Alt ve üst dudak kalınlıkları .....	71
Şekil 2.23.	Elde edilen ikili dudak görüntüsü.....	71
Şekil 2.24.	Dudak üzerinde elde edilen 14 nokta .....	73
Şekil 2.25.	Sistemin örnek ekran görüntüsü .....	74
Şekil 2.26.	Kullanılan algoritmanın blok şeması.....	78
Şekil 2.27.	PCA yönteminin çalışma şeması .....	79
Şekil 3.1.	Dudak bölgesi bulmak için kullanılan örnek yüzler .....	90
Şekil 3.2.	Örnek yüzlerden elde edilen dudak bölgeleri.....	90
Şekil 3.3.	Dudak tespiti sonuçları .....	91
Şekil 3.4.	Farklı katsayıların uygulanması ve elde edilen dudak şekilleri.....	92
Şekil 3.5.	Örnek dudak tespiti sonuçları.....	93

## 1. GENEL BİLGİLER

### 1.1. Giriş

Bir bireyin ölçülebilen fizyolojik ya da davranışsal özelliklerine biyometrik (biometric) özellikler denilmektedir. En çok kullanılan biyometrik özelliklere örnek olarak parmak izi, retina, iris, el ve yüz yapısı, ses, yürüyüş, konuşma şekli, DNA yapısı gibi insana ait olan birçok özellik verilebilir[1]. Biyometrik sistemlerin kullanılmasındaki amaç bu yapıların kişiden kişiye değişen, bununla beraber transfer edilemeyen, kaybedilemeyen ya da unutulamayan, güvenilir bir yapıda olmasıdır. Kişiden kişiye değişen bir başka biyometrik özellik ise kişinin dudak hareketleridir. Şekil 1.1’de biyometrik özelliklerden olan çeşitli dudak türleri gösterilmiştir[2].



Şekil 1.1. Çeşitli dudak türleri

Biyometrik özelliklerden olan dudak hareketinin izlenmesi ve yorumlanması çeşitli problemlerin çözümünde yardımcı olabilir. Bu problemlere örnek olarak insan-bilgisayar etkileşimini verebiliriz. İnsan bilgisayar etkileşiminde konuşma tanıma sistemlerinde görsel bilgilerin ses bilgisine tamamlayıcı bilgi olarak eklenmesi konuşma tanıma performansını arttırmaktadır. Ayrıca duyma engelli insanlar için dudak hareketlerinin yorumlanması, konuşulunun anlaşılması açısından oldukça önemlidir. Çalışmada, bilgisayarla dudak hareketlerinin izlenmesi ve yorumlanması işlemi ele alınmaktadır. İnsan bilgisayar etkileşiminde bu işlem otomatik dudak okuma olarak isimlendirilecektir.

Bilgisayarlı dudak okuma problemleri iki bağlamda ele alınmaktadır. Bunlar genellikle sesin tanınması ve işaret dilinin çözümlenmesi şeklindedir. İnsan iletişiminin önemli unsurlarından olan sesin tanınması insan-bilgisayar etkileşiminin temel öğelerinden birini oluşturmaktadır. Sadece ses dalgalarına dayalı sistemler, işaretin çeşitli gürültülerden güçlü şekilde etkilenmesi nedeniyle çok da güvenli değildir. Öte yandan birçok konuşmacı olduğu durumlarda o anki konuşanın tanınması ayrıca problem oluşturmaktadır. Bu

nedenle ses sinyalinin ve video bilgisinin(dudak hareketlerinin) dikkate alınması sesin tanınmasını daha da güçlendirilebilir. Bugüne kadar yapılan çalışmalarda görsel bilgilerin ses bilgisine eklenerek yapılan iletişimin sadece ses bilgisi kullanarak yapılan iletişimden daha güçlü olduğu ispatlanmıştır[3].

Dudak okuma ile ilgili bir diğer uygulama alanı işaret dilinin çözümlenmesidir. Bir çok çalışmada genellikle sağırlara yönelik işaret dilinin bilgisayarlı çözümlenmesinde ve dudak okuma üzerinde durulmaktadır. İşitme engelliler iletişimde genellikle üçlü bilginin etkileşiminden faydalanmaktadır. Bunlar;

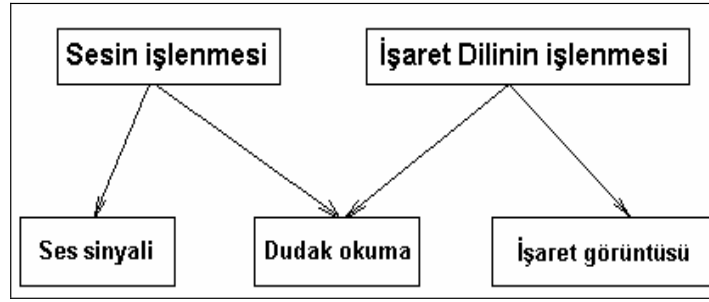
- Daktilo veya parmak alfabesi
- Dudak okuma
- İşaret dili

şeklinde ele alınmaktadır. Daktilo alfabesi, 16. yy'da İspanyol filozoflar Carden, Pons, Bonet tarafından önerilmiştir. Daktilo alfabesinde her bir harf tek elle ya da iki elle kodlanarak kelimeler oluşturulur. Bu kelimelerden ise cümleler elde edilerek iletişim gerçekleştirilir. Düşünülenin aksine dünyada 40'tan fazla çeşitli parmak alfabesinin olduğu bilinmektedir [20,21]. Bunlara örnek olarak Amerikan İşaret dili olan Ameslan (ASL- Ameslan- **American Sign Language**) ve Yetişkinler İçin Türkçe İşaret Dili Kılavuzu gösterilebilir.

Sağırların iletişim yöntemlerinden ikincisi dudak okuma yöntemidir. Dudak okuma ile ilgili ilk defa I.K.Amman tarafından çeşitli yöntemler geliştirilmiştir[20]. Daha sonra yapılan çalışmalarda dudak okurken sesli harflerin daha kolay bir şekilde anlaşıldığı ortaya çıkarılmıştır. Bunun nedeni ise dildeki fonemlerin farklılık göstermesidir. Konuşulanları anlamaları için sağırların dilin gramer yapısına hakim ve gerekli kelime haznesine sahip olmaları gerekmektedir. Dudak okumanın sağlıklı bir şekilde yapılmasında konuşmanın, yani dudak hareketlerinin abartısız, net ve kesin bir biçimde ifadelendirilmesi gerekir. Ayrıca sağırların iletişimde sinonim kelimeler de bulunmaktadır.

Sağırların iletişimde kullanılan üçüncü yöntem, sesli iletişimde de kullanılan mimikler ve jestlerdir[20,22]. Mimikler ve jestler aynı zamanda cisimlerin sıfatları da olmaktadır. Çoğu mimikler ve jestler tek kelimeye tekabül eder. Genellikle 150-200 sayıda jest iletişim için yeterlidir. Yüksek öğretim görmüş sağırlarda kelime haznesi arttığı için binlerce jest kullanılabilir. Jestlerle ilgili en önemli kaynak Jestuno adında 4 ciltlik bir sözlüktür[20,23].

İletişim sırasında genellikle işaret dili dudak okuma ile birlikte kullanılmakta, yalnız bilinmeyen kelimelerin ifadelendirilmesi gibi bazı özel durumlarda parmak alfabesinden faydalanılmaktadır. Bu açıdan işaret dili sesli veri girişi ile bir benzerlik göstermektedir. Aralarındaki fark sadece ses sinyalinin işaret görüntüsü ile birlikte değişmesidir[2]. Şekil 1.2’de sesin işlenmesi ve işaret dilinin işlenmesinin birbirleri ile olan ilişkileri görülmektedir.



Şekil 1.2. Ses ve İşaret dilinin arasındaki ilişki.

Şekil 1.2’de de görüldüğü gibi dudak okuma işlemi, sesin işlenmesi ve işaret dilinin işlenmesi işlemlerinin her ikisinde de ortaktır. Bu bakımdan otomatik dudak okumada hem ses sinyali güçlendirilmekte hem de işaret dili tanınması gerçekleştirilmektedir.

### 1.1.1. Dudak Okuma Gereksinimleri

Dudak okumaya gerek duyulan alanlardan bir tanesi uzaktan sesli ve görsel iletişimdir. Anlatılmak istenen ifadenin ses bilgisi içerisinde aranması bir takım sorunlara yol açmaktadır. Bu sorunlara örnek olarak ortamdaki gürültü, ses dalgaları arasındaki girişimler, mikrofondaki bozukluklar verilebilir. Sesin tanınmasında ortamdaki gürültü her kayıt esnasında farklı olması ve bu gürültünün önceden tahmin edilemeyen yapıda olması sesin tanınmasını zorlaştırmaktadır. Bu durumda ses bilgisini güçlendirecek diğer bilgilere başvurmak yararlı olacaktır.

İnsan konuşmasının algılanmasında dudak okuma gibi “görsel konuşma bilgileri” nin oldukça yararlı olduğu iyi bilinen bir gerçektir[1]. Duyma engelli insanlar, dudak okumayı iletişimde birincil kaynak olarak kullanmaktadırlar. Ayrıca uzman dudak okuyucular akıcı bir konuşmayı sadece dudak hareketlerinden anlayabilme yeteneğine sahiptirler. Normal duyan insanlar için bile, konuşan kişinin dudak hareketlerini görmek, konuşmanın

anlaşılabilmesi açısından oldukça önemlidir. Bugüne kadar yapılan araştırmalarda görsel bilginin iletilen mesajın 3'te birini içerdiği vurgulanmaktadır[6,7,8].

Birçok otomatik konuşma tanıma sistemi (ASR), akustik ses sinyali üzerine yoğunlaşmakta ve bunun sonucu olarak ortamdaki gürültüden kolayca etkilenmektedir. Ayrıca bazı sesler ses uzayında birbirine karıştırılabilirken görsel uzayda bu sesler birbirinden kalaylıkla ayırt edilebilmektedir. Örneğin “b” ve “v” ile “m” ve “n” sesleri bu şekildedir[4]. Bu sebeplerden dolayı görsel bilgilerin otomatik konuşma tanıma (ASR) sistemlerinde tamamlayıcı bilgi olarak kabul edilmesi dikkatleri “otomatik dudak okuma” sistemlerinin üzerine çekmektedir.

### **1.1.2. Otomatik Dudak Okuma Sistemleri**

Dudak okuma alanında ilk yapılan çalışma Petajan adlı doktora öğrencisi tarafından 1984 yılında gerçekleştirilmiştir. Bu çalışmada Petajan, sadece siyah beyaz renklerden oluşan bir görüntüden dudak biçimini çıkarmayı başarmış, sadece görsel verileri işleyen ve “dinamik zaman esnetmesine” (dynamic time warping) dayanan bir tanıma sistemi geliştirmiştir[5].

Devam eden yıllarda dudak okuma için dudağın dış sınırlarının önemli bir parametre olduğu anlaşılmış ve dış sınırların elde edilmesi ile ilgili çeşitli yaklaşımlar geliştirilmiştir. Örneğin Sum ve arkadaşları tarafından 2001 yılında yapılan çalışmalarda Aktif Şekil Modeli (Active Shape Model, ASM) kullanarak dudağın dış sınırları bulunmuş ve bu dış sınırları temsil eden 14 Nokta Modeli oluşturmuştur. Daha sonra takip eden görüntüler için de 14 nokta çıkarılarak konuşma esnasında dudağın dış sınırları izlenmiştir[6].

2002 yılında Matthews ve arkadaşları [4] çalışmalarında dudak okuma için ciddi bir çalışma içine girmiş ve dudak okuma için gerekli özelliklerin çıkarılması üzerinde durmuştur. Çalışmasında konuşma için gerekli dudak hareketleri, vücut hareketleri, kafa hareketleri gibi parametreleri birleştirmiştir ve bu parametrelerin konuşma tanımadaki nasıl tamamlayıcı bilgi olarak kullanılacağını göstermiştir. Dudak görüntü dizilerini parametrik ifade etmek için 3 yöntem kullanılmış ve bu yöntemleri Saklı Markov Modelle karşılaştırmıştır. Matthews, çalışmasında tümevarım ve tümdengelim yaklaşımları da incelemiş ve bu yaklaşımlarla elde ettiği sonuçları test etmiştir[4].

Devam eden yıllarda Rogozan ve arkadaşları, dudak okuma için Hibrit Saklı Markov Modellerine ve Yapay Sinir Ağlarına dayalı bir öğrenme önermişlerdir. Çalışmasında

dudak şeklinin geometrik özelliklerini çıkarmış ve geliştirdiği öğrenme sistemini eğiterek dudak okuma sistemini gerçeklemeye çalışmıştır[6].

2004 yılında Lei Xie ve arkadaşları RoHiLTA adını verdikleri ve dudak okuma için kullanılabilir bir dudak sınırlarını izleme sistemi geliştirmişlerdir. Bu sistemde yüzün çeşitli özelliklerinden yararlanarak ağız bölgesi bulunmuş ve bu bölgede dudak sınırlarının elde edilmesi için ASM kullanılmıştır. Dudak dış sınırları bulunduktan sonra dudak izleme işlemini gerçekleştirmişlerdir[7].

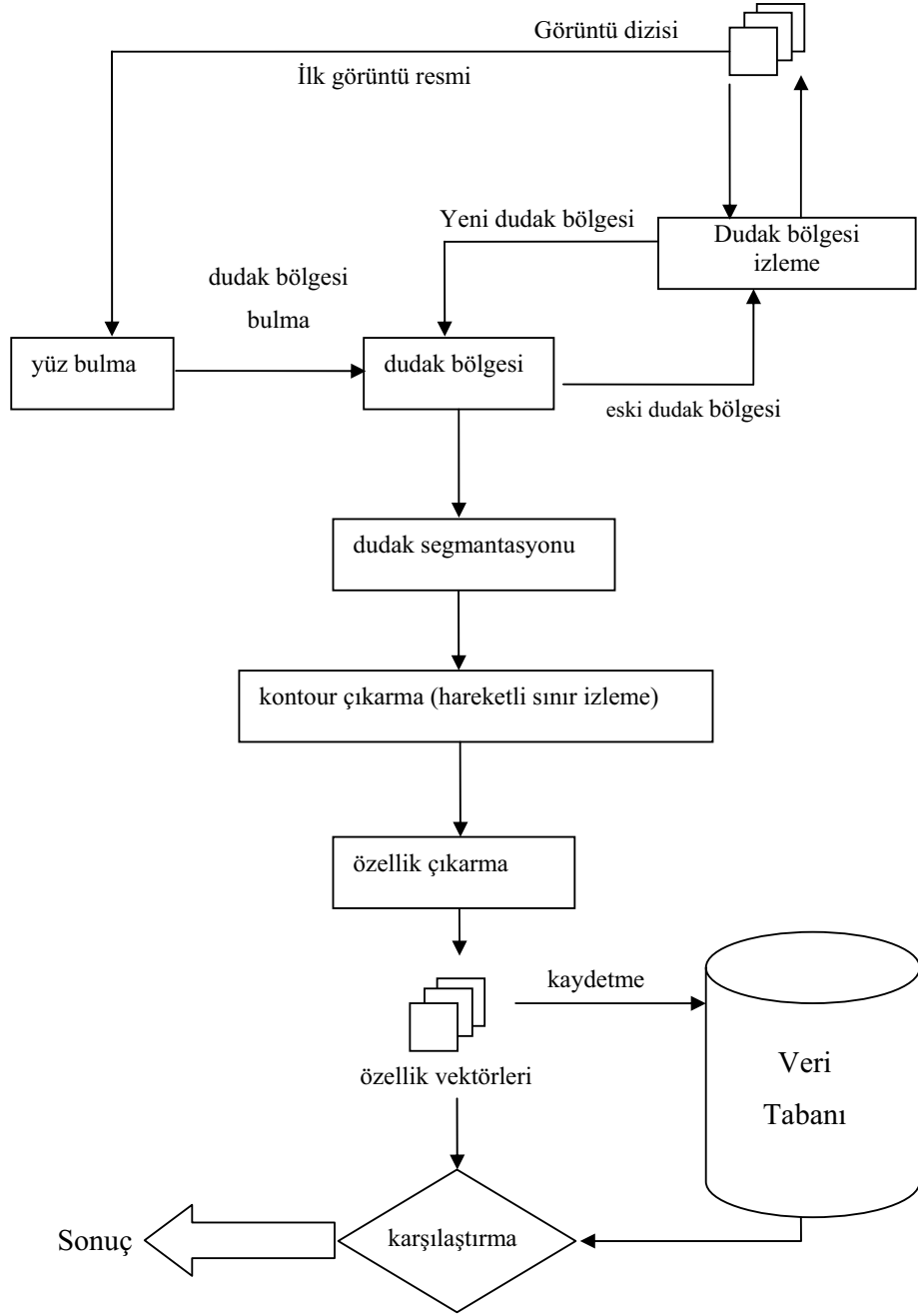
Yine 2004 yılında Wang ve arkadaşları çalışmalarında birbirinden ayrı ayrı alınan 0'dan 9'a kadar olan İngilizce rakamların görüntülerini alarak görsel bilgilerden dudak okuma sistemi gerçekleştirmişlerdir. Dudağın dış sınırlarını temsil etmek için 14 nokta modeli kullanılmış, ayrıca ağız içi parametreleri de belli ölçüde dikkate alınmıştır. Ağız içi ve ağız dışından elde edilen parametrelerden belli bir özellik vektörü oluşturulmuş ve bu özellik vektörünün eğitilmesi için Saklı Markov Modeli kullanılmıştır[8].

Literatürde birçok araştırmacının dudak okuma üzerinde çeşitli çalışmalar gerçekleştirmesine rağmen Türkçede bu konuda doyurucu bir çalışma yapılmamıştır.

### **1.1.3. Sistemin Çalışma Şeması**

Bilgisayarlı dudak okuma işlemi birçok karmaşık hesapsal süreçleri içermektedir. Bu yüzden sistem alt modüllere parçalanarak her bir alt modül kendi içinde ayrı ayrı incelenmiş ve gerçekleştirilmiştir. Şekil 1.3'te, bilgisayarlı dudak okuma sistemi için önerilen ve gerçekleştirilen çalışma şeması verilmektedir.





Şekil 1.3. Sistemin çalışma yapısı

Çalışmada öncelikle işlenecek olan görüntü dizisi kaynak olarak alınır. Bu görüntü dizisi her biri bir çerçeveden oluşan görüntülerdir. Bu görüntüler RGB renk uzayındadır. Bu çerçeve görüntü dizisi üzerinde dudak okuma gerçekleştirmek için her bir dizi elemanı

üzerinde sırasıyla şu işlemler gerçekleştirilmesi gerekir:

1. Yüz bulma
2. Dudak bölgesi bulma
3. Dudak bölgesinde dudak tespiti
4. Özellik çıkarma
5. Veri tabanı kullanarak karşılaştırma yapma ve dudak okumayı gerçekleştirme

Şekil 1.3'te de görüldüğü gibi görüntü dizisinin ilk çerçeve resmi alınarak bu resim üzerinde önce yüz bulma ve ardından yüz üzerinde dudak bölgesi bulma işlemi gerçekleştirilir. Dudak bölgesi, dudağı içeren bölge olarak düşünüldüğünde bu bölgenin bulunması dudak üzerinde işlem yapmaya başlamadan önce önemli bir aşamadır. Ayrıca bu bölge üzerinde ilerleyen aşamalarda sadece dudak değil aynı zamanda mimik ve jestler de incelenebilecektir.

Dudak bölgesi bulunduktan sonra bu bölgenin bundan sonraki resimler için bulunmasına gerek kalmayacaktır. Bunun yerine ilk çerçeve resimde bulunan dudak bölgesi, nesne takibi algoritmaları ile takip edilerek işlem yükünün azaltılması amaçlanmıştır. Dudak bölgesi içerisinde dudak okuma işleminde en fazla etkiye sahip dış sınırların bulunması işlemi gerçekleştirilir. Dudak sınırlarının bulunması işlemi dudak bölütlenmesi olarak adlandırılmaktadır. Bu işlem de gerçekleştirdikten sonra dudağın dış sınırlarını temsil eden noktalar bulunur ve bu noktalara göre özellik vektörü çıkarılır. Çıkarılan bu özellik vektörü sadece o anki çerçeve içindir. Dudak tespiti ve özellik vektörü çıkarma işlemi takip eden çerçeveler için de çıkarılarak 2 boyutlu bir özellik matrisi oluşturulur. Bu matris PCA yöntemi ile eğitilerek veritabanına kaydedilir. Daha sonra tanıma için gelen diğer dudak görüntüleri de aynı işlemlerden geçerek veritabanına kaydedilen özellik matrisi ile uygun şekilde karşılaştırılarak karar verilir.

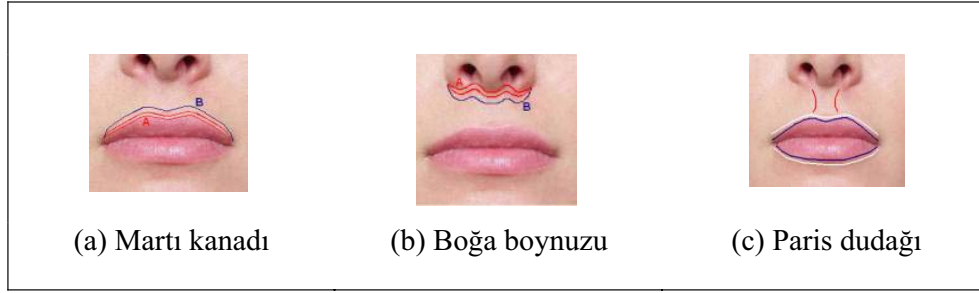
Bilgisayarlı dudak okuma sistemi, basit ve ileri seviye görüntü işleme algoritmaları kullanmaktadır. Bu algoritmalar çeşitli renk uzaylarında uygulanmakta ve verimli sonuçlar elde edilmektedir. İlerleyen bölümlerde sistemde kullanılan renk uzayları ve görüntü işleme teknikleri hakkında bilgiler verilmiştir.

## 1.2. İnsan Dudağının Yapısı

Dudak okuma konuşma dilinin görsel ipuçları olduğundan dudağın dıştan görünen yapısı bilgisayarlı dudak okuma için oldukça önemlidir. İnsan dudağı yapısal olarak

incelendiğinde çeşitli dudak yapıları göze çarpmaktadır. Bunlar yaygın olarak bilinen martı kanadı ve boğa boynuzu dudak türleridir. Bu dudak türleri genellikle estetik cerrahide kullanılmaktadırlar.

Martı kanadı dudak türünde dudak geniştir ve dudağın üst kısmı martı kanadına benzemektedir. Ayrıca dudağın pembe kısmı yukarı doğru kalkık durmaktadır. Paris dudak türünde ise dudak kenarları ve filtrum denilen üst dudakta burun altında kalan çukurluğun kenarları belirgin ve dolgundur. Bu dudak türlerinden başka boğa boynuzu gibi başka dudak türleri de bulunmaktadır. Burun altında kalan kıvrımların boğa boynuzuna benzemesi sonucu bu ismi almıştır. Ayrıca normal duran dudakların farklı duruşlarında da insan dudağı farklı yapıda olabilmektedir. Şekil 1.4'te çeşitli dudak yapıları verilmektedir.



Şekil 1.4. Çeşitli dudak türleri

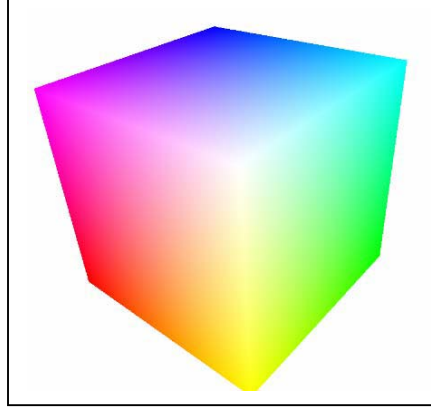
### 1.3. Renk Uzayları

Renk uzayları renk bilgisinin nasıl temsil edileceğini belirleyen uzaylardır. Renkli görüntüyü ifade etmek için, saklama ortamları ve uygulamaların gereksinimlerine göre uygun olacak şekilde çeşitli renk uzayları kullanılmaktadır. Hangi renk uzayının kullanılacağı yapılacak uygulamaya bağlıdır. Bir rengi herhangi bir uzayda temsil etmek için genellikle temel renk bileşenleri, parlaklık, renk tonu doygunluk gibi kavramlar kullanılır.

#### 1.3.1. RGB Renk Uzayı

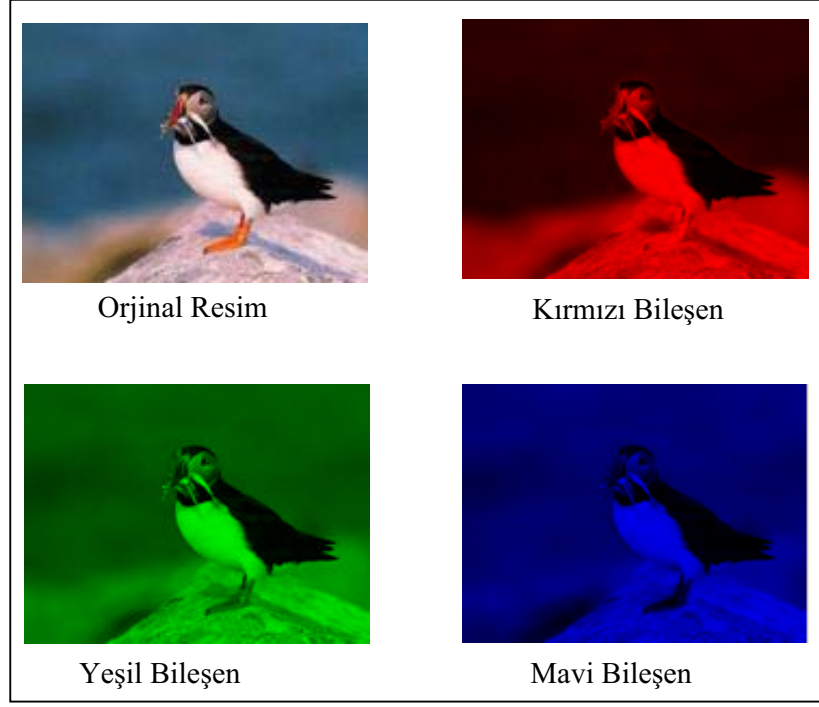
RGB renk uzayı renklerin ekranda gösterilmesi için kullanılan basit ve sağlam bir renk uzayıdır. RGB rengi temsil etmek için üç sayısal bileşen kullanır. Bu renk uzayı her bir bileşenin bir eksene karşılık geldiği 3 boyutlu uzaya benzer. R kırmızı, G yeşil ve B

mavi rengi temsil eder. Bu renk uzayı insan gözü için en elverişli uzaydır. RGB toplamalı bir renk uzayı olduğundan istenen bir rengi elde etmek için R, G ve B değerlerinin belli oranda değerlerinin toplamı bulunur. Her bir renk bileşeni bilgisayarda genellikle 8 bitle temsil edilir. Her renk 0 ile 255 arasında bir parlaklık değeri alır. R, G ve B bileşenlerinin her bir değeri 0 olduğunda siyah, her bir renk bileşeni 255 olduğunda ise beyaz rengini elde edilir. Örneğin bu üç rengin aynı parlaklık seviyelerinin toplanmasıyla gri tonlar elde edilirken üç rengin saf hallerinin toplanması ile beyaz renk elde edilmiş olur. Aşağıdaki şekilde bu üç rengin bileşiminden oluşan renk küpü verilmiştir. Şekil 1.5'te bu 3 rengin bileşiminden oluşan renk küpü verilmektedir.



Şekil 1.5.RGB renk küpü

Ayrıca RGB renk uzayında R, G ve B bileşenlerinin her biri uygulama alanına göre farklı bilgiler içermektedir. Şekil 1.6'te orijinal görüntünün kırmızı, mavi ve yeşil renk bileşenleri verilmiştir.



Şekil 1.6. Orjinal resim ve kırmızı yeşil ve mavi bileşenleri

RGB renk uzayı özellikle bilgisayar grafiklerinde ve görüntü işleme tekniklerinin uygulanmasında kullanılmaktadır.

### 1.3.2. CIE Lab Renk Uzayı

CIE Lab renk uzayı 1976 yılında görsel medya için tasarlanıp oluşturulmuştur. CIE Lab renk uzayı bileşenlerinden L değeri parlaklığı a değeri R/G değerini ve b değeri ise Y/B değerini göstermektedir.

$$L=0.299R+0.587G+0.114B \quad (1)$$

$$a=R/G \quad (2)$$

$$b=Y/B \quad (3)$$

Günümüzde CIE Lab renk uzayı çeşitli alanlar için standart renk uzayı olarak seçilmiştir ve bugün pek çok uygulamada kullanılmaktadır.

### 1.3.3. HSI ya da HSV Renk Uzayları

HSI renk uzayı ilk olarak renk özelliklerinin sayısal olarak ifade edilmesine ihtiyaç duyulduğunda ortaya atılmıştır. Kırmızı, mavi ve yeşil renkleri doğrusal olmayan bir şekilde yeniden organize edilmiştir. Bu renk uzayı görüntü işleme açısından oldukça kullanışlıdır. Çünkü HSI renk uzayında renklerin renk bileşenlerinin ayrılış biçimi insanın görme sisteminin algılama şekli ile örtüşmektedir. Özellikle deri rengi ile ilgili çalışmalarda oldukça başarılı sonuçlar vermektedir. H (hue ) renk tonunu yani baskın olan rengi, S(saturation) renk doyumunu yani renk miktarını ve I(intensity ya da value) ise rengin parlaklığını temsil etmektedir. RGB renk uzayından HSI renk uzayına geçişteki bağıntılar aşağıda verilmiştir[15].

$$\left. \begin{aligned} H &= \cos^{-1} \left\{ \frac{\frac{1}{2}[(R-G) + (R-B)]}{\sqrt{[(R-G)^2 + (R-B)(G-B)^2]}} \right\} \\ S &= 1 - \frac{3}{(R+G+B)} [\min(R, G, B)] \\ I &= \frac{1}{3}(R+G+B) \end{aligned} \right\} R, G, B \in [0,1] \quad (4)$$

### 1.3.4. CMY ve CMYK

Bu renk uzayı daha çok kağıt üzerinde renklerin daha canlı görünmesi için kullanılmaktadır. CMYK(Çiyan, Magenta, Sarı, Siyah) renkler kağıt üzerinde daha iyi bir şekilde yansıdığı için bu renk uzayı özellikle yazıcılarda kullanılmaktadır. Ayrıca sayısal sanat yaparken bu renk uzayı renklerin daha canlı gözükmesini sağlayacaktır.

Bu renk uzayı daha çok baskı sistemleri için kullanılmaktadır. Cihaz-bağımlı bir renk uzayıdır. CMY ailesine dahil iki renk uzayı bulunmaktadır.

CMY : çok yaygın olarak kullanılmamaktadır. Sadece düşük kapasiteli renkli yazıcılarda kullanılmaktadır.

CMYK : baskı sistemlerinde oldukça yaygın olarak kullanılmaktadır.

CMYK renk uzayında Çiyan, Magenta ve Sarı üç ana renktir. Kırmızı (R), Yeşil (G) ve Mavi (B) bileşenleri ise bu renk uzayında ikincil renkler olarak adlandırılmaktadır.

Teorik olarak siyah renge ihtiyaç yoktur. Ancak Çiyan, Magenta ve Sarı renkleri kağıt üzerinde tam olarak birleştiğinde siyah elde edilmesi gerekirken koyu kahverengi bir renk elde ediliyor. Böylece daha iyi bir görüntü elde etmek için siyah renk(yazıcılarda mürekkep olarak düşünülmemektedir.) kullanılmaktadır. Aşağıda RGB renk uzayından CMYK renk uzayına dönüşüm yaparken kullanılan bağıntılar verilmektedir.

$$C = 1 - R \quad (5)$$

$$M = 1 - G \quad (6)$$

$$Y = 1 - B \quad (7)$$

Burada kullanılan R, G, B değerleri [0-1] arasına iz düşürülmüş kırmızı, yeşil ve mavi bileşenlerdir.

### 1.3.5. YCbCr Renk Uzayı

YCbCr renk uzayı RGB bileşenlerinden elde edilen ve Avrupa televizyon stüdyolarında görüntü sıkıştırma işlemleri için yaygın olarak kullanılan bir renk uzayıdır. Y değeri RGB değerlerinin ağırlıklı toplamlarından ve Cb ve Cr değerleri ise renk farklarından oluşmaktadır. (8)'de RGB renk uzayından YCbCr renk uzayına geçiş için kullanılan bağıntılar verilmiştir[13].

$$Y = 0.299R + 0.587G + 0.114B$$

$$Cr = R - Y \quad (8)$$

$$Cb = B - Y$$

Bu renk uzayı görüntü işlemede daha çok deri rengi modellemelerinde kullanılmaktadır.

### 1.3.6. YIQ ya da YUV Renk Uzayları

RGB uzayında saklanan görüntünün çok yer kaplaması nedeniyle, televizyon yayıncılığında kullanılan bant genişliğini en aza indirmek için donanıma bağlı olarak YIQ ya da YUV uzayları kullanılır. Y, rengin parlaklık bilgisini taşır ve renkli görüntünün gri seviyeli halini ifade eder. IQ ya da UV ise resmin renk bilgisini taşımaktadır. YIQ ya da YUV değerleri ile RGB arasındaki ilişki (9) ve (10)'da görülmektedir.

$$\begin{aligned}
Y &= 0.299 R + 0.587 G + 0.114 B \\
I &= 0.596 R - 0.274 G - 0.322 B \\
Q &= 0.211 R - 0.523 G + 0.312 B
\end{aligned} \tag{9}$$

$$\begin{aligned}
R &= 1.000 Y + 0.956 I + 0.621 Q \\
G &= 1.000 Y - 0.272 I - 0.647 Q \\
B &= 1.000 Y - 1.106 I + 1.703 Q
\end{aligned} \tag{10}$$

Avrupa'daki televizyon yayıncılığında yaygın olarak kullanılan PAL (Phase Alternating Line) sistemlerde YUV uzayı kullanılırken Amerika'da kullanılan NTSC (National Television Dtandaarts Committee) sistemlerde YIQ tercih edilmektedir.

YUV uzayının yeni teknolojilerde kullanılmasının nedenlerinden bir tanesi insanın görme duyusunun, renkliliğe parlaklıktan daha az duyarlı olmasıdır. Bu nedenle YUV uzayında UV renk bileşenleri parlaklık bileşenine göre daha az örneklenir. Y bileşeni ise görüntünün siyah beyaz televizyonlardaki bileşenini taşımaktadır.

### 1.3.7. TSL Renk Uzayı

TSL renk uzayı normalize RGB değerleinden elde edilmekte ve anlamsal olarak HSI renk uzayına benzemektedir. Aşağıda RGB uzayından TSL uzayına geçiş için gerekli bağıntılar verilmiştir[13].

$$S = \sqrt{\frac{9}{5}(r'^2 + g'^2)} \tag{11}$$

$$T = \begin{cases} \frac{\tan^{-1}\left(\frac{r'}{g'}\right)}{2\pi} + \frac{1}{4} & , g' > 0 \\ \frac{\tan^{-1}\left(\frac{r'}{g'}\right)}{2\pi} + \frac{3}{4} & , g' < 0 \\ 0 & , g' = 0 \end{cases} \tag{12}$$

$$L = 0.299R + 0.587G + 0.114B \tag{13}$$

Burada,



$$r' = r - \frac{1}{3}, \quad g' = g - \frac{1}{3} \quad (14)$$

$$r = \frac{R}{R+G+B}, \text{ normalize R}, \quad g = \frac{G}{R+G+B}, \text{ normalize G} \quad (15)$$

Deri rengi ile ilgili yapılan çalışmalarda TSL renk uzayının modelleme için en iyi uzay olduğunu görülmüştür. Ancak RGB uzayından TSL uzayına çift taraflı dönüşüm bağıntıları ağır matematiksel ifadeler içerdiğinden, gerçek zamanlı uygulamalar için pek uygun değildir. Bu nedenle TSL renk uzayına anlamsal açıdan en yakın HSI renk uzayını kullanmak daha uygun olmaktadır.

#### 1.4. Görüntü İyileştirme Teknikleri

Birçok görüntü işleme uygulamalarında görüntüler üzerinde tanıma, eğitim, özellik çıkarma gibi işlemler uygulanmaktadır. Bu işlemleri gerçekleştirebilmek için görüntüler üzerinde bir takım işlemler yapılmaktadır. Bu işlemler genellikle ön işlem olarak adlandırılır. Ön işlemlere örnek olarak görüntü iyileştirme teknikleri verilebilir. Görüntü iyileştirme teknikleri, görüntü üzerinde işlemlere başlamadan önce yapılması gereken iyileştirme işlemleridir. İlerleyen bölümlerde bilgisayarlı dudak okumada kullanılan görüntü iyileştirme teknikleri ayrıntılı bir şekilde incelenmektedir.

##### 1.4.1. Ortalama Alma (Mean Filter)

Ortalama alma tekniği, resimleri yumuşatmak için basit, sezgisel ve kolay gerçekleştirilen bir yöntemdir. Bu yöntemle komşu iki piksel arasındaki parlaklık değişimlerinin değerleri azaltılarak gürültü azaltma işlemi gerçekleştirilir.

Ortalama alma yönteminin temel mantığı, herhangi bir pikselin değerinin kendisi dahil komşu piksel değerlerinin ortalaması şeklinde ifade edilmesine dayanır. Ortalama alma yönteminin uygulanması ise resim üzerinde bir maskenin gezdirilmesine dayanır. Maske uygulama işlemi bir sonraki bölümde ayrıntılı olarak açıklanmıştır. Uygulanacak maske Şekil 1.7'deki gibidir[9]. Maske uygulanması sonucu elde edilen sonuç Şekil 1.8'de verilmiştir.

$$\begin{pmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{pmatrix}$$

Şekil 1.7. 3x3 lük ortalama alma maskesi



Şekil 1.8. Orijinal resim ve resme ortalama alma işleminin uygulanması

#### 1.4.2. Bir Görüntü Üzerinde Maske Uygulama

Bir görüntü üzerinde maske uygulanması herhangi bir 3x3 lük operatör matrisin bu resim üzerinde gezdirilmesine karşılık gelmektedir. Maske görüntünün sol üst köşesinden başlayarak tüm görüntüye uygulanır. Maskenin resim üzerinde iz düştüğü pikseller ile maskenin pikselleri çarpılarak toplanır ve son durumda ortalaması alınır. Elde edilen yeni değer maskenin çekirdeğine karşılık gelen pikselin yeni değeri olur. Bu işi gerçekleştiren C++ kodu aşağıda verilmiştir.

```
// görüntü üzerinde maske gezdirme
for( i=1;i<n-1;i++)
  for( j=1;j<m-1;j++)
  {
    int toplam=0;
    for(int k=0;k<3;k++)
      for(int l=0;l<3;l++)
```

```

        toplam+=resim[i-1+k][j-1+l]*mask[k][l];
        tmp[i][j]=toplam;
    }
// geçici diziyi orijinal görüntü dizisine kopyalama
for(i=0;i<n;i++)
    for(j=0;j<m;j++)
        resim[i][j]=tmp[i][j];

```

Maske gezdirmeye işlemi uygulanırken orijinal görüntünün sınır değerleri üzerinde işlem yapılmaz. Çünkü maskenin orta elemanı tam sınırdaki diğer elemanları görüntü dışına taşımaktadır. Bu soruna sınır değerleri üzerinde maske uygulamak yerine bu değerleri aynen almak şeklinde bir çözüm getirilebilir.

Şekil 1.8’de 3x3 lük bir operatör matrisi gösterilmiştir. Farklı uygulamalarda farklı boyutlardaki operatör matrisleri kullanılabilir. Örneğin 5x5 lik bir operatör matrisi Şekil 1.9’da verilmiştir.

$$\begin{vmatrix} 1 & 1 & 1 & 1 & 1 \\ 25 & 25 & 25 & 25 & 25 \\ 1 & 1 & 1 & 1 & 1 \\ 25 & 25 & 25 & 25 & 25 \\ 1 & 1 & 1 & 1 & 1 \\ 25 & 25 & 25 & 25 & 25 \\ 1 & 1 & 1 & 1 & 1 \\ 25 & 25 & 25 & 25 & 25 \end{vmatrix}$$

Şekil 1.9. 5x5 lik ortalama filtresi

Operatör matris boyutu arttıkça görüntüdeki yumuşama daha da artmaktadır. Ancak bu ilkemin sisteme getirdiği yük daha fazladır. Görüntü işleme uygulamalarında genellikle 3x3 lük boyut kullanılmaktadır. Ancak farklı tür uygulamalarda 5x5 ya da 7x7 şeklindeki boyutlarda kullanılabilir.

### 1.4.3. Histogramın Çıkarılması

Bir resmi oluşturan piksellerin o resim içinde hangi sıklıkla geçtiğini veren istatistiksel bilgiye histogram denir. Renkli resimlerde ve gri seviye resimlerde olmak üzere iki türlü histogram çeşidi vardır. Çalışmada daha çok gri seviye histogram kullanılmıştır. Basit olarak histogram şu şekilde çıkarılır.

Elimizde farklı değerlerden oluşan bir vektör olsun. Bu değerler aşağıdaki değerlerden oluşsun:

$$v_i = \{1,2,2,2,3,3,5,5,5,5,9,9,9,10,10,12,12,13,14\}$$

Bu değerlerin sıklık(frekans) larını çıkaracak olursak

$$(v_i, f_i) = \{1,1\}, \{2,3\}, \{3,2\}, \{5,4\}, \{9,3\}, \{10,2\}, \{12,2\}, \{13,1\}, \{14,1\}$$

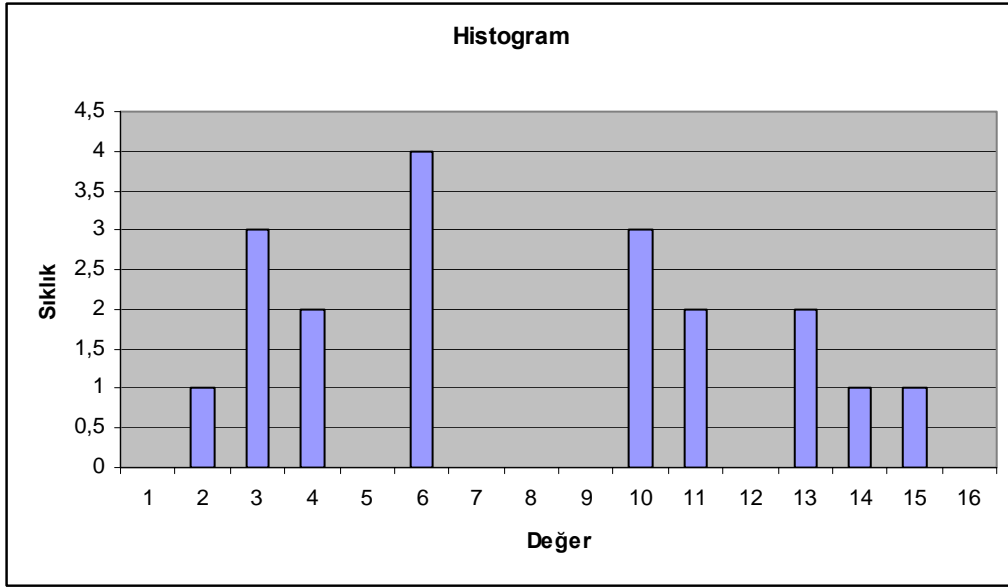
Şeklinde bir vektör ikilisi elde ederiz. Örneğin  $\{5,4\}$  şeklindeki vektör ikilisi 5 değerinin dizide 3 kez rastlandığını gösterir. Yukarıda minimum değer 1 ve maksimum değer ise 14 olduğu görülür ve aradaki bazı değerlerin de frekanslarının sıfır olduğu görülür. Bu sıfır frekans değerli elemanlar da araya yerleştirilir ve aşağıdaki şekil elde edilir.

$$(v_i, f_i) = \{1,1\}, \{2,3\}, \{3,2\}, \{4,0\}, \{5,4\}, \{6,0\}, \{7,0\}, \\ \{8,0\}, \{9,3\}, \{10,2\}, \{11,0\}, \{12,2\}, \{13,1\}, \{14,1\}$$

Yukarıda görüldüğü gibi 14 değer vardır. Bu 14 değer 2'nin üstel katlarına tamamlanması gerekir. Bu yüzden diğer değerler de eklenerek 16 elemanlı  $\{v_i, f_i\}$  elde edilir.

$$(v_i, f_i) = \{0,0\}, \{1,1\}, \{2,3\}, \{3,2\}, \{4,0\}, \{5,4\}, \{6,0\}, \{7,0\}, \\ \{8,0\}, \{9,3\}, \{10,2\}, \{11,0\}, \{12,2\}, \{13,1\}, \{14,1\}, \{15,0\}$$

Bu vektör iki boyutlu x-y düzleminde çizilebilir.



Şekil 1.10. Örnek histogram grafiği

Şekil 1.10'da da görüldüğü gibi histogram dizisi (frekans dizisi)

$f_i = \{0, 1, 3, 2, 4, 0, 0, 0, 3, 2, 0, 2, 1, 1, 0\}$

şeklinde olmaktadır. Bu dizideki değerler her bir değer için veri dizisinde kaç kez rastlandığını gösterir. Aynı şekilde görüntü dizileri içinde bu şekilde histogram dizisi çıkarılabilir.

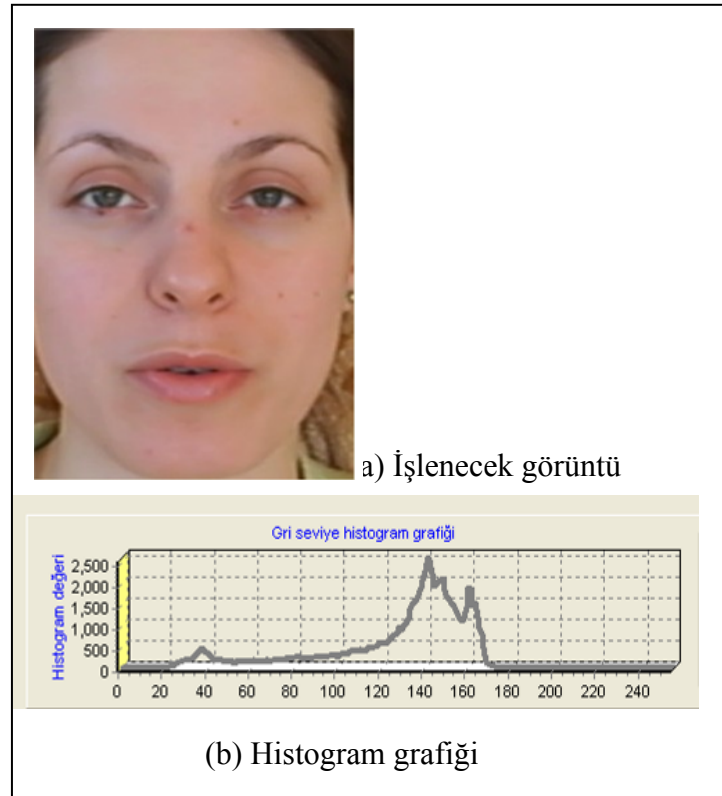
Gri seviye için histogram çıkaran C++ kodu aşağıdaki şekilde gibidir.

```
for(int i=0;i<n;i++)
  for(int j=0;j<m;j++)
  {
    gray=resim[i][j];
    histogram[gray]++;
  }
```

Burada resim dizisi 256 farklı seviyeden oluşan ve boyutları (n,m) olan gri seviye resimdir. Histogram dizisi ise 256 elemanlı tek boyutlu bir dizidir.

#### 1.4.4. Gri Seviye Resimde Histogram

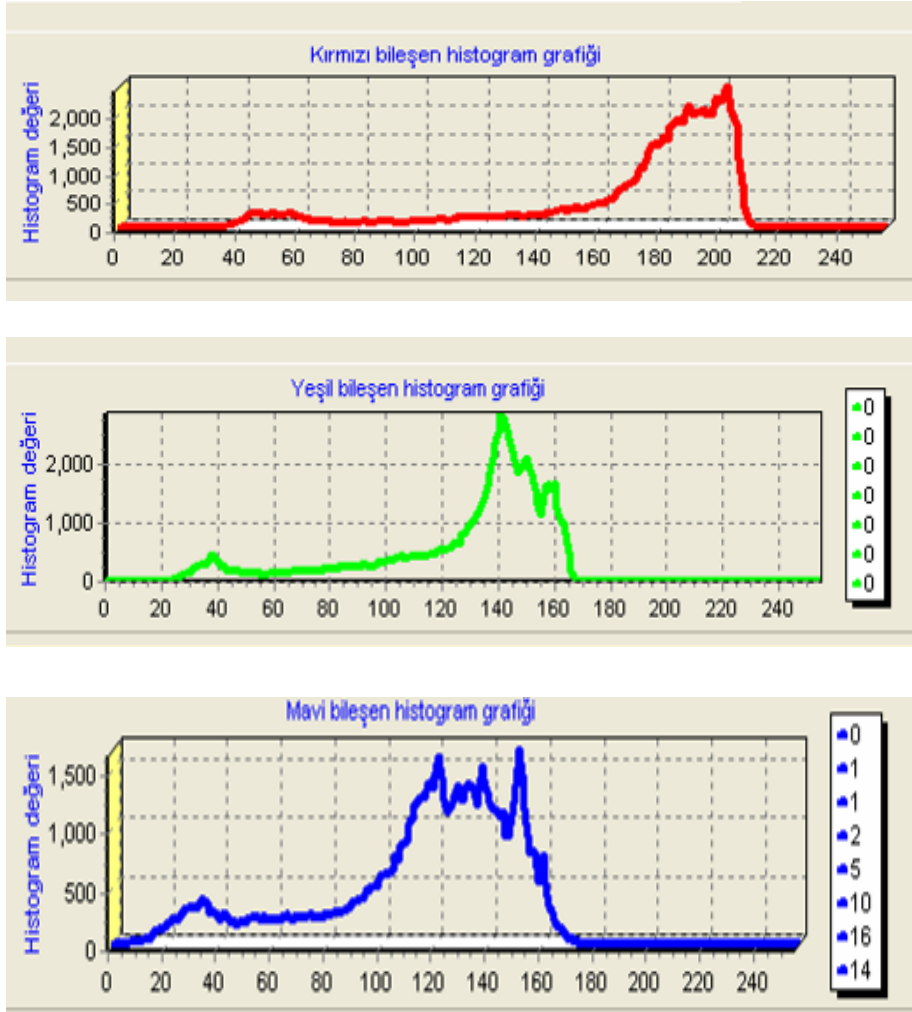
Resim gri seviyeye çevrildikten sonra elde edilir. Tüm resim taranarak 0 değerine sahip piksellerin sayısı tespit edilir ve 0 değerli pikselin frekansı (sıklığı) olarak belirlenir. 1 değerine sahip pikseller sayılır ve 1 değerli pikselin sıklığı olarak belirlenir. Bu şekilde 0-255 arası tüm piksellerin frekansları bulunarak histogram dizisi elde edilir. Şekil 1.11’de orijinal görüntü ve bu görüntünün histogramı verilmektedir.



Şekil 1.11. Gri seviye resimde histogram

#### 1.4.5. Renkli Resimde Histogram

Renkli resimde histogram elde etmek için her bir renk kanalı ayrı ayrı değerlendirilir. RGB değerlerinin her biri için gri seviye resimde olduğu gibi ayrı bir histogram dizisi elde edilir. Yani R, G ve B değerlerinin ayrı ayrı frekanslarının bulunması gerekmektedir. Şekil 1.12’de Şekil 1.11a’daki orijinal görüntünün renkli histogram grafikleri verilmiştir.

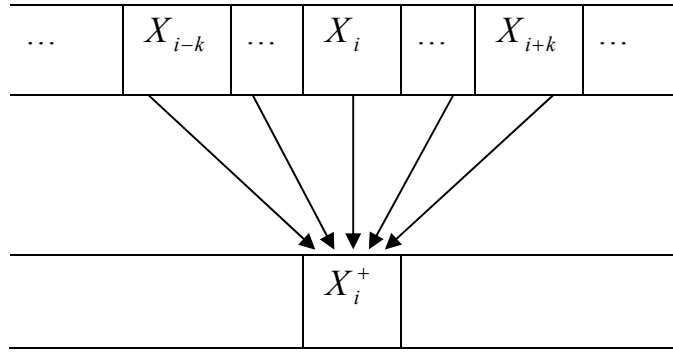


Şekil 1.12. R, G ve B bileşenlerine ait histogram grafikleri

#### 1.4.6. Histogram Yumuşatma (Histogram Smoothing)

Histogram yumuşatma işlemi aslında ortalama alma işlemi ile benzer özellikler taşır. Yumuşatma işlemini uygulamadan önce histogramın çıkarılması gerekir.

Histogram dizisinin çıkarılmasından sonra bu dizinin yumuşatılması işlemine geçilir. Histogram yumuşatma işlemi ilke olarak ortalama alma işlemi uygulamaya benzemektedir. Histogram dizisinde yeni oluşacak her bir değer kendisi ve komşu değerlerin ortalamasından oluşur. Aşağıdaki şekilde bu algoritma açıklanmaktadır.



Şekil 1.13.Histogram yumuşatma işlemi

Kullanılan bağıntı ise şu şekildedir.

$$X_i^+ = \frac{X_{i-k} + \dots + X_{i-1} + X_i + X_{i+1} + \dots + X_{i+k}}{2 \cdot k + 1} = \frac{\sum_{t=i-k}^{i+k} X_t}{2 \cdot k + 1} \quad (16)$$

Bu işlem için gereken C++ kodu ise aşağıdaki şekildedir.

```
for(int i=k+1;i<n-k;i++)
{
    float toplam=0,ort=0;
    int j;
    for(j=i-2;j<=i+k;j++)
        toplam+=histogram[j];
    ort=toplam/(2*k+1);
    tmp[i]=(int)ort;
}
```

Burada  $n = 256$  değerini,  $k$  ise histogram yumuşatma işleminin kalitesini göstermektedir. Histogram yumuşatılmış dizi **tmp** dizisidir. Bu dizi daha sonra orijinal histogram dizisine kopyalanarak işlemlere devam edilir. Şekil 1.13'teki histogram yumuşatma işleminde kullanılan kalite katsayısı  $k=2$  olarak verilmiştir.

Histogram yumuşatma sonucu Şekil 1.14'te elde edilmiştir.





Şekil 1.14.Histogram yumuşatma işleminin sonucu işleminin sonucu

#### 1.4.7. Orta Değer Süzgeci (Median Filtresi)

Orta değer alma işlemi, ortalama alma işlemine benzer şekilde resmi daha yumuşak hale getirmek için kullanılır. Bu işlemin ortalama alma işleminden farkı sadece bir pikselin değerinin komşu piksellerin ortalamasından oluşmasından ziyade orta değer alınmasına dayanır[9]. Bunun için önce görüntü üzerinde gezdirilen 3x3 lük operatör matrisine karşılık gelen alandaki piksel değerleri sıralanır ve ortadaki değer alınır. Alınan bu değer o pikselin yeni değeri olur. Gerekli C++ kodu aşağıdaki gibidir.

```
int t[9],ti=0,temp;
for(int i=1;i<n-1;i++)
{
    for(int j=1;j<m-1;j++)
    {
        for(int k=j-1;k<=j+1;k++) // 3 x 3 luk matris tek boyutlu diziye al
            for(int l=i-1;l<i+1;l++)
                t[ti++]=resim[k][l];
        for(int k=0;k<9;k++) // 9 elemanlı diziyi sırala
            for(int l=k+1;l<9;l++)
                if(t[k]>t[l])
                {
                    temp=t[k];
                    t[k]=t[l];
                    t[l]=temp;
                }
        tmp[i][j]=t[4]; // orta değer alınıyor
    }
}
```

#### 1.4.8. Normalleştirme

Normalleştirme işlemi RGB renk biçimindeki piksellerin R, G, ve B değerleri arasındaki farkı normale çekme işlemi olarak ele alınır. RGB renklerinin parlaklıklarındaki gürültülerden kaynaklanan değişimleri süzmek için kullanılır. R, G ve B değerleri aşağıdaki bağıntılara göre yeniden düzenlenir[10].

$$r = \frac{R}{R+G+B}, g = \frac{G}{R+G+B}, b = \frac{B}{R+G+B} \quad (17)$$

#### 1.4.9. Histogram Eşitleme

Histogram Eşleme, görüntüdeki piksellerin dağılımlarını daha düzenli hale getirmek için kullanılan bir tekniktir. Koyu renkli görüntüleri daha açık hale getirmek, açık renkli görüntüleri koyulaştırmak ya da normal resimdeki kontrastı artırmak için kullanılır. Bu teknik bir görüntünün tümüne ya da belli bir kısmına uygulanabilir.

Histogram Eşleme, parlaklık dağılımlarını yeniden düzenler. Görüntü dağılımında önceden kaç tane tepe noktası ya da vadi bulunursa histogram eşlemeden sonra da o kadar tepe ya da vadi oluşur. Ancak bu dağılımların aralarındaki açıklık değişir. Histogram eşleme sonucu histogram belli bir aralıktan tüm aralığa yayılır ve böylece dağılım daha düzenli hale gelir. Şekil 1.16'da histogram dağılımı sonucu resmin ne kadar değiştiği görülmektedir.

Histogram Eşleme için kullanılan yöntem aşağıdaki gibidir. Histogram Eşleme yapacağımız değer aralığı 0-7 olsun. Minimum değer 0 ve maksimum değer ise 7 dir. Bu durumda görüntü 3 bitle temsil edilmektedir. Kullanılan histogram dizisi (18)'deki gibidir[9].

$$h = \{10,8,9,2,14,1,5,2\} \quad (18)$$

*Adım 1:* Histogram toplamları hesaplanır. Örneğin ilk değer 10, ikinci değer  $10+8=18$ , üçüncü değer  $10+8+9=27$  ve bu şekilde tüm seдеğerle hasaplanır.

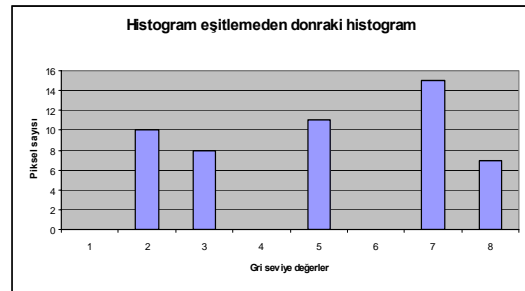
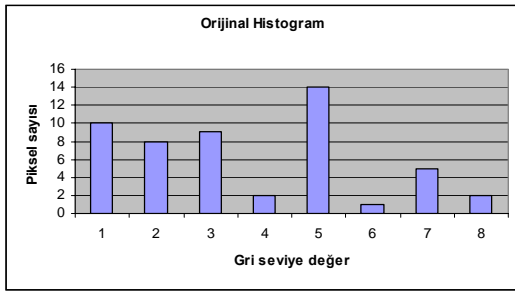
*Adım 2:* Adım 1 de elde edilen değerler toplam piksel sayısına bölünerek normalleştirilir. Toplam piksel sayısı  $10+8+9+2+14+1+5+2=51$  olarak hesaplanır. Normalleştirilen değerler sırasıyla  $10/51$ ,  $18/51$ ,  $27/51$ ,  $29/51$ ,  $43/51$ ,  $44/51$ ,  $49/51$ ,  $51/51$  şeklinde olur.

*Adım 3:* Adım 2 de elde edilen değerler, maksimum gri seviye değeri (bu örnekte 7 değeri) ile çarpılır ve şu değerler elde edilir. 1, 2, 4, 4, 6, 6, 7, 7.

*Adım 4:* Adım 3'te elde edilen değerler birebir eşleştirilir. Yani 0 değerli pikselin yerine 1 değeri, 1 değerli pikselin yerine 2 değeri, 2 değerli pikselin yerine 4 değeri konulur ve bu şekilde devam edilir. Tablo 1.1'de yapılan işlem adımları görülmektedir. Şekil 1.15'te ise eşitlenmiş histogram grafiği verilmektedir.

Tablo 1.1. Histogram eşleme işlemi

Gri seviye değeri	Histogram	Histogram Toplamı	Histogram eşitlenmiş değeri
0	10	10	$(10/51)*7=1$
1	8	18	$(18/51)*7=2$
2	9	27	$(27/51)*7=4$
3	2	29	$(29/51)*7=4$
4	14	43	$(43/51)*7=6$
5	1	44	$(44/51)*7=6$
6	5	49	$(49/51)*7=7$
7	2	51	$(51/51)*7=7$

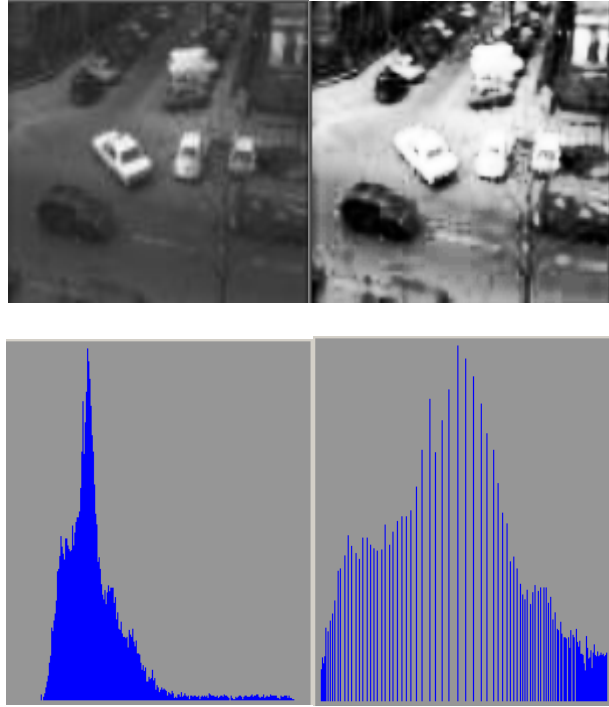


Şekil 1.15. Histogram eşitleme işlemi sonucu



Şekil 1.16. Histogram eşitleme işleminin görüntü üzerindeki sonucu

Aşağıda farklı resimlerin histogram eşitlenmiş halleri bulunmaktadır.



Şekil 1.17. Farklı resimler için histogram eşitleme grafikleri

Histogram eşitleme için kullanılan C++ kodu aşağıdaki gibidir.

```
int K[256],S[256];
K[0]=histogram[0]; // histogram değerleri önceden hesaplanmıştır.
for(int i=1;i<256;i++)
    K[i]=histogram[i] + K[i-1];
for(int i=0;i<256;i++)
    S[i]=(255 * K[i]) / K[255];
unsigned char tmp;
for(int i=0;i<n;i++) // nxm lik bir resim
    for(int j=0;j<m;j++)
    {
        tmp=resim[i][j];
        resim[i][j]=(unsigned char) S[tmp];
    }
```

### 1.5. Yatay ve Düşey İz düşüm

Görüntü işleme uygulamalarında yaygın olarak kullanılan bir başka yöntem yatay ya da düşey iz düşüm yöntemleridir. Yatay ve dikey iz düşüm yöntemleri herhangi bir görüntüde (imge) yoğunlaşmanın nerde olduğunu tespit etmede oldukça yararlı bilgiler vermektedir. Yatay ve dikey iz düşümler, ikili resimde, gri seviye resimde ya da renkli resimde ayrı ayrı bulunabilmektedir.

#### 1.5.1. Yatay İz düşüm

Yatay iz düşümde görüntüye soldan sağa doğru yatay olarak bakılır ve her bir satırın izdüşümü elde edilir. Her bir satırın iz düşümü bulunurken o satır boyunca tüm değerler toplanarak bir değer elde edilir. Bulunan bu değer o satırın yatay iz düşüm değeri olarak kabul edilir. Bazı durumlarda iz düşüm elde edilirken değerlerin toplanması oldukça büyük değerler üreteceğinden bu durumda değerlerin ortalamasının alınması yoluna da gidilebilir.

Yatay iz düşüm yöntemi farklı seviyedeki görüntülere farklı şekilde uygulanmaktadır.

### 1.5.2. İkili Resimde Yatay İz Düşüm

İkili resimde yatay iz düşüm alınırken o satır boyunca karşılaşılan siyah piksellerin sayısı o satırın iz düşüm değeri olarak alınır. Bu şekilde tüm satırların izdüşümleri alınarak yatay iz düşüm vektörü elde edilir. Şekil 1.18’de ikili resimde alınan yatay iz düşümün grafiği verilmiştir.



Şekil 1.18. İkili resimde yatay iz düşüm

### 1.5.3. Gri Seviye Resimde Yatay İz düşüm

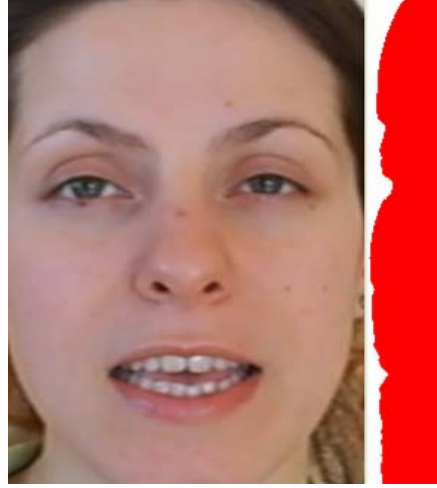
Gri seviye resimde yatay iz düşüm alınırken o satır boyunca olan tüm piksellerin parlaklık değerleri (gri seviye değerleri) toplanır ve o satırın yatay iz düşüm değeri olarak alınır. Bu şekilde tüm satırlar için aynı işlem tekrar edilerek yatay iz düşüm vektörü elde edilir. Şekil 1.19’da şekilde gri seviye resim için elde edilen yatay iz düşüm sonucu gösterilmiştir.



Şekil 1.19. Gri seviye resimde yatay iz düşüm

#### 1.5.4. Renkli Resimde Yatay İz Düşüm

Renkli resimde yatay iz düşüm alınması ikili resim ya da gri seviye resme göre biraz farklılık göstermektedir. Renkli resimde R, G ve B gibi farklı renk kanalları olduğu için her bir renk kanalı için farklı bir yatay iz düşüm grafiği elde edilmelidir. Örneğin R renk kanalı için yatay iz düşüm elde edilirken o satır boyunca olan her bir pikselin R bileşenleri toplanarak o satırın R yatay iz düşüm değeri elde edilir. Bu şekilde tüm satırlar için aynı işleme devam edilerek tüm satır boyunca yatay iz düşüm değerleri elde edilir ve kırmızı renk için yatay iz düşüm grafiği elde edilir. Aynı işlemler yeşil ve mavi bileşenler için de devam ettirilerek yatay iz düşüm vektörleri elde edilir. Şekil 1.20’de örnek olarak kırmızı bileşen için elde edilen yatay iz düşüm verilmektedir.

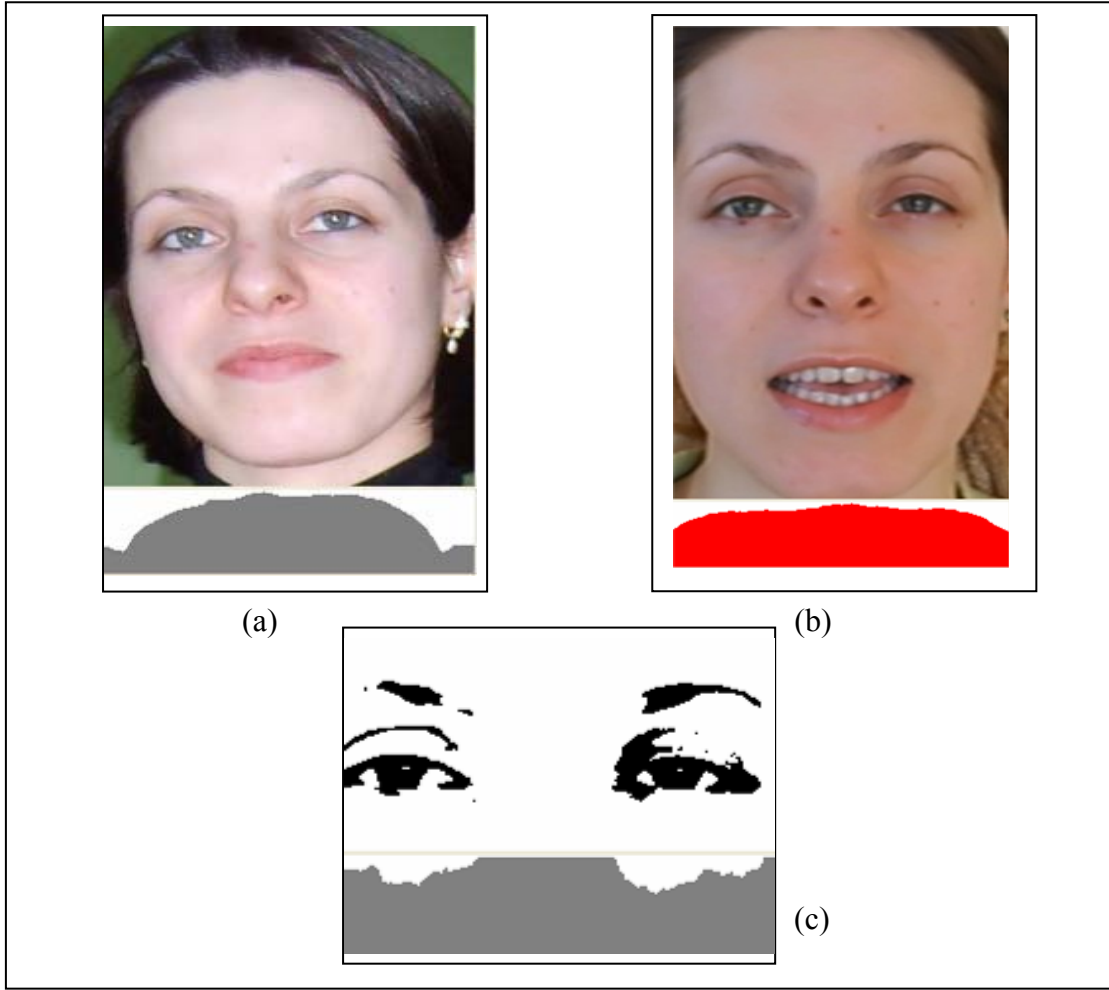


Şekil 1.20. Renkli resimde R bileşenine ait yatay iz düşüm grafiği

#### 1.5.5. Düşey İz Düşüm

Düşey iz düşüm de yatay iz düşüm gibi aynı işlemlerden oluşmaktadır. Arasındaki tek fark yatay iz düşüm de her bir satır için yatay iz düşüm değeri hesaplanırken düşey iz düşümde her bir sütun için bir iz düşüm değeri hesaplanır. Bunun dışındaki tüm işlemler yatay iz düşüm ile aynıdır. Düşey iz düşüm için de aynı şekilde ikili resim için, gri seviye resim için ve renkli resim için hesaplanan düşey iz düşüm (projeksiyon) grafikleri Şekil 1.21'de verilmiştir.





Şekil 1.21. İkili, gri seviye ve renkli resimlerde düşey iz düşümler

## 1.6. Morfolojik İşlemler

### 1.6.1. Yayma (Dilation)

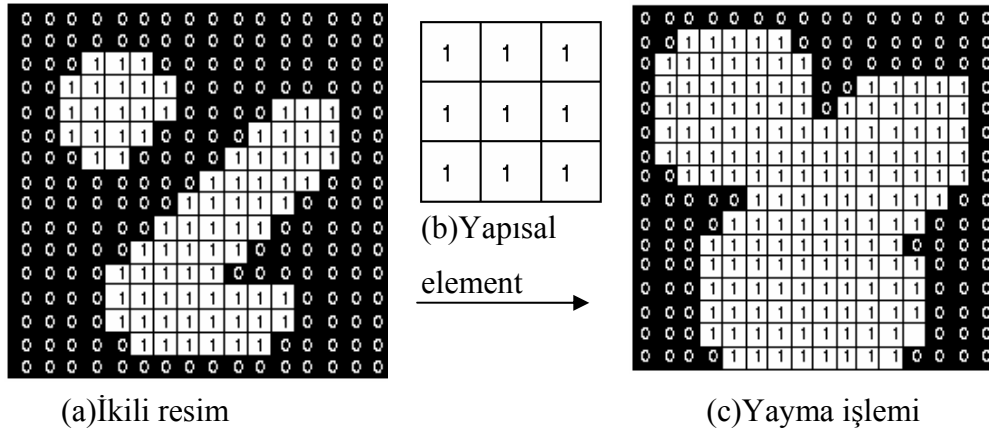
Görüntü işleme uygulamalarında kullanılan 2 temel morfolojik işlem vardır. Bunlardan biri aşındırma (erosion) diğeri ise yayma (dilation) işlemleridir. Genellikle bu işlemler ikili resimlere uygulanmaktadır. Yayma işleminin temel etkisi ikili bir resimdeki şekilleri genişletmektir. İkili resimde şekilleri oluşturan piksellerin beyaz ve arka planı oluşturan piksellerin siyah olduğu düşünülürse yayma işlemi sonucu beyaz piksellerin sayısı artarken siyah piksellerin sayısı azalmaktadır[25].

Yayma işlemi temelde iki giriş bilgisi alır. Birinci giriş, yayma işleminin uygulanacağı ikili resim ve ikinci giriş ise yayma işlemi için kullanılacak yapısal

element(operatör matris)'tir.

Basit olarak yayma işlemi şu şekilde uygulanır.

- Yapısal element ikili resim üzerinde gezdirilir.
- Yapısal element ikili resim üzerine iz düşürüldüğünde ikili resim üzerinde oluşan 3x3 lük bölgedeki piksellerin herhangi biri beyaz ise (şekle ait ise) o zaman merkezdeki eleman da beyaz olarak ayarlanır.
- İkili resim üzerinde oluşan 3x3 lük bölgedeki piksellerin tamamı siyah ise (arka plana ait ise) o zaman merkeze karşılık gelen piksel de siyah olur. Şekil 1.22'de yayma işleminin sonucu görülmektedir.



Şekil 1.22. Yayma işlemi

Yayma işleminde yapısal element olarak farklı türler kullanılabilir ve değişik sonuçlar elde edilebilir. Ancak en yaygın olarak kullanılan yapısal element Şekil 1.22.b'deki gibidir.

### 1.6.2. Aşındırma (Erosion)

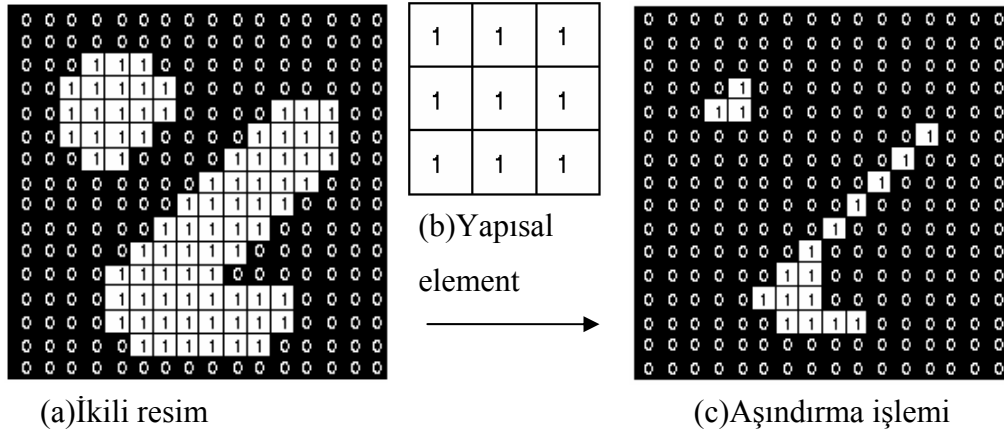
Görüntü işlemede kullanılan ikinci temel morfolojik işlem aşındırma işlemidir. Aşındırma işleminin temel etkisi şekli oluşturan piksellerin aşındırılmasıdır. Bu işlem sonucu ikili resimde beyaz piksellerin (şekli oluşturan piksellerin) sayısı azalırken, arka plandaki piksellerin (şekli oluşturan piksellerin) sayısı artmaktadır. Aşındırma işlemi de yayma işlemine benzer şekilde iki giriş bilgisi alır. Bunlardan birincisi ikili resim ve diğer giriş ise yapısal elementtir. Yapısal element, yayma işleminde kullanılan yapısal elementle

aynıdır.

Aşındırma işlemi aşağıdaki şekilde uygulanır:

- Yapısal element ikili resim üzerinde gezdirilir.
- Yapısal element ikili resim üzerine iz düşürüldüğünde ikili resim üzerinde oluşan 3x3 lük bölgedeki piksellerin herhangi biri siyah ise (arka plana ait ise) merkeze karşılık gelen piksel de siyah olarak ayarlanır.
- Eğer yapısal element ikili resim üzerine iz düşürüldüğünde ikili resim üzerinde oluşan 3x3 lük bölgedeki piksellerin tamamı beyaz ise (şekle ait ise) merkeze karşılık gelen eleman beyaz olarak kalır.

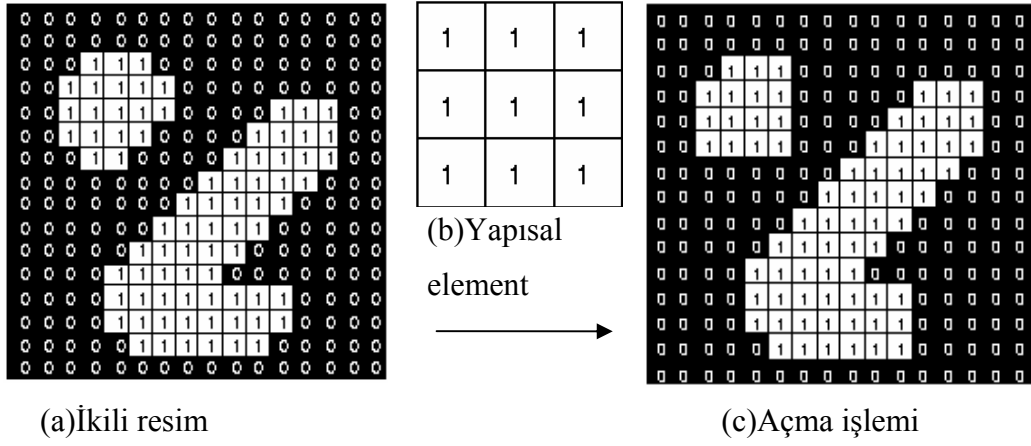
Şekil 1.23'te aşındırma işlemi sonucu elde edilen resim görülmektedir.



Şekil 1.23. Aşındırma işlemi

### 1.6.3. Açma (Opening) İşlemi

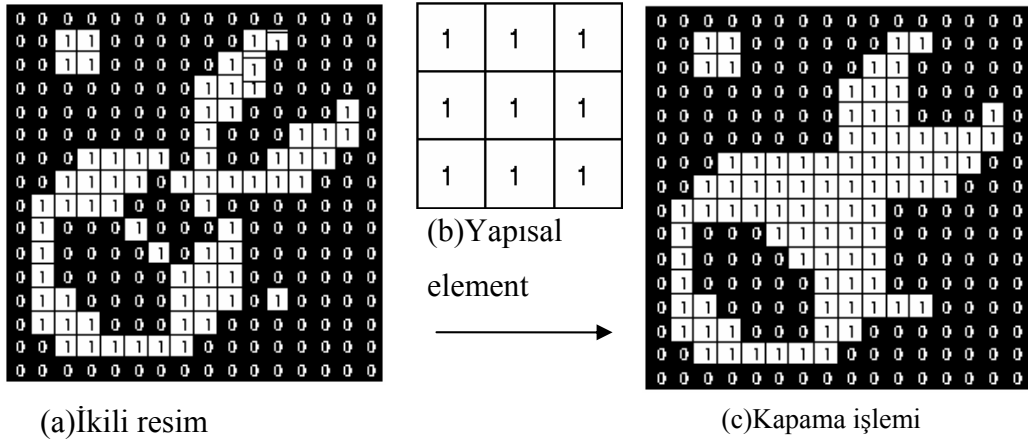
Görüntü işleme tekniklerinde yaygın olarak kullanılan bir diğer morfolojik teknik ise açma (opening) tekniğidir. Bu teknik temelde aşındırma ve yayma işlemlerine dayanır. Açma işlemi, yayma işleminin ardından aşındırma işleminin uygulamasına karşılık gelmektedir. Şekil 1.24'te açma işleminin sonuçlarını verilmiştir.



Şekil 1.24. Açma işlemi

#### 1.6.4. Kapama (Closing)

Kapama işleminin uygulanışı açma işleminin uygulanışının tam tersi olarak düşünülebilir. Yani önce aşındırma işlemi daha sonra da yayma işlemi uygulandığında bu işlem kapama işlemi olarak isimlendirilir. Şekil 1.25'te kapama işleminin sonucu verilmiştir.



Şekil 1.25. Kapama işlemi

## 1.7. Görüntü Eşikleme (Image Thresholding)

### 1.7.1. Giriş

Görüntü işlemede resimleri daha kolay işleyebilmek ve örüntü tanımada özellikler çıkarabilmek için resimler ikili hale dönüştürülmelidir. Resmin ikili hale dönüştürülmesi, belli bir eşik değerinin seçilerek bu değer altında kalan piksellerin siyah ve eşik değerinin üzerinde kalan piksellerin beyaz olarak seçilmesi ile gerçekleştirilir. Görüntü eşikleme işlemi ise bu eşik değerinin seçilmesine karşılık gelmektedir.

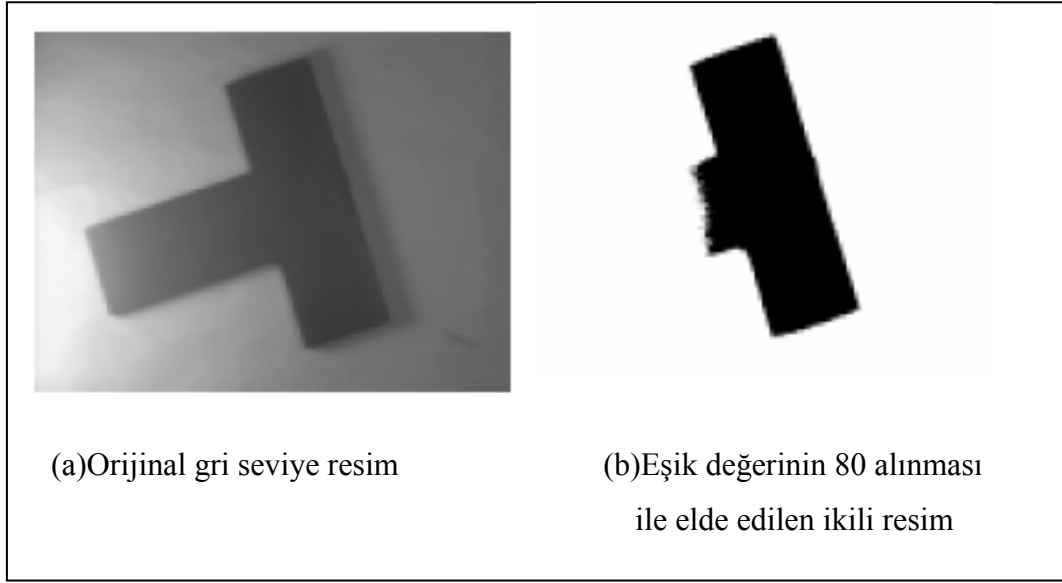
### 1.7.2. Global Eşikleme

Global eşikleme bir resmi gri seviye görüntüden ikili resme çevirmek için kullanılan bir yöntemdir. Bu yöntemin global eşikleme olmasının nedeni, uygulanacak eşik değerinin tüm resme uygulanmasıdır.

Eşikleme tek ya da çok seviyeli olabilir. Yukarıda da bahsedildiği gibi resmi ikili hale çevirmek için tek seviyeli eşikleme kullanılmalıdır. Tek bir eşik değeri kullanıldığında, resimdeki bir pikselin parlaklık değeri eşik değerinden düşükse o piksel değeri siyah ( ya da beyaz) olarak belirlenir ve benzer şekilde piksel değeri eşik değerinden düşükse o piksel değeri siyah (ya da beyaz) olarak belirlenir.

Bazı durumlarda eşikleme yapılırken belli bir bant aralığı seçilir. Piksel değeri bu bant aralığında ise o piksel değeri beyaz (ya da siyah) seçilirken piksel değeri aralığın dışında ise o piksel değeri siyah (ya da beyaz) olarak belirlenir.

Eşik değerinin belirlenmesi işlemi karmaşık bir işlemdir. Çoğu görüntü işleme uygulamalarında önceden belirlenen bir eşik değeri kullanılmaktadır. Şekil 1.26'da eşik değeri 80 olarak seçilmiş ve elde edilen sonuç verilmiştir.



Şekil 1.26. Global eşikleme

### 1.7.3. Değişken (Dinamik) Eşikleme

Global eşikleme, tüm resim üzerinde önceden belirlenen bir eşik değerinin uygulanmasına dayanır. Ancak çoğu durumda global eşikleme, farklı parlaklık ve durumlarda alınan resimler için uygun sonuçları vermez. Bu yüzden dinamik olarak eşik değerinin belirlenmesi gerekir. Eşik değerinin dinamik olarak belirlenmesi için Otsu algoritması kullanılmıştır[11]. Otsu algoritması, görüntüdeki piksel değerlerinin dağılımlarına göre bu piksellerin kümelendirilmesini sağlamaktadır. Görüntünün piksel değerleri  $L$  ( $[1,2...L]$ ) gri seviye ile temsil edilirse,  $i$ . seviyedeki piksel sayısı  $n_i$  olduğunda toplam piksel sayısı aşağıdaki şekildedir.

$$N = n_1 + n_2 + n_3 \dots + n_i + \dots + n_L \quad (19)$$

İki seviyeli eşik değeri seçiminde pikseller  $[1,2,...t]$  ve  $[t+1,...L]$  gri seviye değerlerini içeren  $C_1$ ,  $C_2$  gibi iki sınıfa ayrılmaktadır.

$$p_i = n_i / N \quad (20)$$

olarak gösterilirse  $C_i$  kümeleri aşağıdaki bağıntılarla verilebilir.

$$\begin{aligned} C_1 &: p_1 / \omega_1(t), p_2 / \omega_1(t) \dots p_t / \omega_1(t) \\ C_2 &: p_{t+1} / \omega_2(t), p_{t+2} / \omega_2(t) \dots p_L / \omega_2(t) \end{aligned} \quad (21)$$

Burada

$$\omega_1(t) = \sum_{i=1}^t p_i \quad \omega_2(t) = \sum_{i=t+1}^L p_i \quad (22)$$

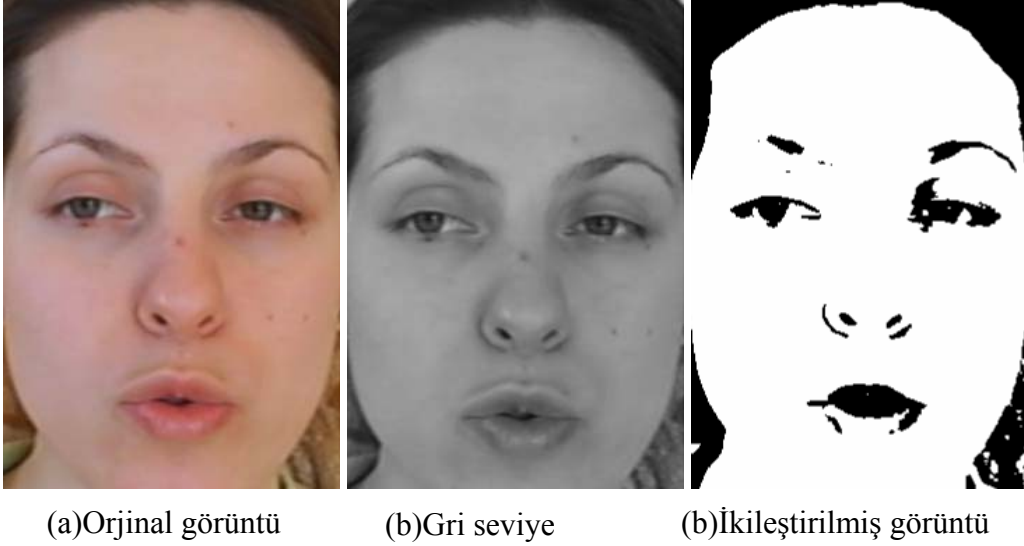
olarak alınır. Otsu algoritması, sınıflar arası dağılımı ( $\sigma_B^2$ ) maksimum yapacak optimum  $t^*$  eşik değerini belirlemektedir.

$$\sigma_B^2 = \omega_1 \omega_2 (\mu_2 - \mu_1)^2 \quad (23)$$

$$\mu_1 = \sum_{i=1}^t ip_i / \omega_1(t) \quad \mu_2 = \sum_{i=t+1}^L ip_i / \omega_2(t) \quad (24)$$

$$t^* = \underset{1 \leq t \leq L}{\text{ArgMax}} \{ \sigma_B^2(t) \} \quad (25)$$

Şekil 1.27’de dudak okuma için uygun giriş görüntüsünün akıllı eşiklenme aşamaları gösterilmiştir. Bulunan eşik değeri  $t^*=101$ ’dir.



Şekil 1.27. Otsu algoritması sonucu elde edilen ikili resim

### 1.8. Bazı İstatistiksel Temeller

Bu bölümde, otomatik dudak okuma için kullanılan Temel Bileşen Analizini daha iyi anlayabilmek için gerekli bazı istatistik temelleri verilecektir. Daha sonra PCA için kullanılacak belli başlı matris işlemlerine değinilecektir.

İstatistik, büyük veri kümeleri ile bu veri kümeleri arasındaki ilişkileri inceleyen bir bilim dalıdır. Bu bölümde bu ilişkilerin istatistik bilimi tarafından nasıl incelendiği konusuna değinilecektir.

### 1.8.1. Standart Sapma

Standart sapma, bir veri kümesindeki verilerin ne kadar dağınık olduğunu gösteren istatistiksel bir bilgidir. Örneğin aşağıdaki gibi bir veri kümemiz olsun.

$$X = [25 \ 36 \ 15 \ 87 \ 42 \ 78 \ 63 \ 54 \ 41 \ 35] \quad (26)$$

Veri kümemizdeki standart sapmayı hesaplamak için öncelikle bazı değerlerin hesaplanması gerekir. Örneğin veri kümesindeki değerlerin ortalaması, hesaplanması gereken değerlerden biridir. Ortalama basit bir şekilde şu formülle bulunur.

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{n} \quad (27)$$

Burada  $\bar{X}$  (X bar olarak adlandırılır) veri kümesinin ortalamasını gösterir.

Ortalama, verilerin orta değerinden başka herhangi bir bilgi içermemektedir. Örneğin birbirinden farklı veri kümelerinin ortalamaları aynı olmalarına rağmen veriler birbirinden tamamen bağımsızdır.(28)'de verilen veri kümeleri buna bir örnektir.

$$[6 \ 1 \ 10 \ 7] \text{ ve } [3 \ 5 \ 7 \ 9] \quad (28)$$

veri kümelerinin ortalamalarının ikisi de 6 olmasına karşın veri kümeleri birbirinden oldukça farklıdır.

Bu veri kümelerinin birbirinden farklı veriler olduklarını anlayabilmek için dağılıma bakmak gerekir. Bir veri kümesinin ne kadar yayılım gösterdiği standart sapma ile gösterilir. Standart sapma (29)'daki formülle hesaplanır.

$$s = \sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{(n-1)}} \quad (29)$$

Burada s, veri kümesinin standart sapmasını göstermektedir. Veri kümesindeki elaman sayısı da n olarak verilmektedir.



Tablo 1.2. Standart sapma hesabı

$X_i$	$(X - \bar{X})$	$(X - \bar{X})^2$	$X_i$	$(X - \bar{X})$	$(X - \bar{X})^2$
6	0	0	3	-3	9
1	-5	25	5	-1	1
10	4	16	7	1	1
7	1	1	9	3	9
Toplam		42	Toplam		20
(n-1) e böl.		14	(n-1) e böl.		6.6666
Standart s.		3.7416	Standart s.		2.5806

(28)'de verilen ve ortalaması aynı çıkan iki veri kümesinin standart sapması Tablo 1.2'de verilmiştir. Görüldüğü gibi bu iki veri kümesinin ortalamaları aynı olmasına rağmen standart sapmaları farklıdır. Bu durumda standart sapma bilgisi de veri kümeleri arasında herhangi bir ilişki vermemektedir.

### 1.8.2. Varyans

Varyans, bir veri kümesinin dağılımı ile ilgili bilgi veren bir diğer ölçümdür. Varyans, standart sapmanın karesi alınarak hesaplanır.

$$s^2 = \frac{\sum_{i=1}^n (X_i - \bar{X})^2}{(n-1)} \quad (30)$$

Standart sapma hesaplanırken karekök alma işlemi gerçekleştirilmezse bulunan değer varyans olur.

### 1.8.3. Kovaryans

Önceki bölümlerde verilen ölçümler bir boyutlu ölçümlerdi. Ancak bir çok veri kümesi birden fazla boyuta sahiptir ve istatistiksel olarak bu boyutlar arasındaki ilişkilerin incelenmesi gerekmektedir. Örneğin bir okulda öğrencilerin çalıştıkları saat miktarları ve aldıkları notlar olmak üzere iki veri kümemiz olsun. Kovaryans bu iki veri arasındaki ilişkinin ne olduğunu bize verir.

Standart sapma ve varyans ölçümleri sadece 1 boyut için hesaplanabildiğinden 2 boyutlu veriler arasındaki ilişkiyi hesaplayabilmek amacıyla her veri için gerekli varyans

birbirinden bağımsız olarak hesaplanması gerekmektedir.

Kovaryans ise daima 2 boyut arasındaki ilişkiyi ölçer. Eğer tek boyutlu bir veri incelenecekse varyans hesaplanır. Eğer  $(x, y, z)$  veri kümeleri arasındaki ilişki hesaplanacaksa bu durumda  $x$  ile  $y$  arasında,  $y$  ile  $z$  arasında ve  $x$  ile  $z$  arasında kovaryans hesabı yapmak gerekir.

Kovaryans hesabı, varyans hesabına benzemektedir. Bu benzerliği daha iyi anlayabilmek için varyans formülünü şu şekilde de yazabiliriz.

$$var(X) = \frac{\sum_{i=1}^n (X_i - \bar{X})(X_i - \bar{X})}{(n-1)} \quad (31)$$

Kovaryans ise varyansın 2 boyutlu hali olduğuna göre kovaryans formülü de şu şekildedir.

$$cov(X, Y) = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{(n-1)} \quad (32)$$

Görüldüğü gibi varyans formülündeki ikinci  $X$  'lerin yerini ikinci boyut olan  $Y$  'ler almıştır. Yukarıda bahsedilen örnekte öğrencilerin çalıştıkları saat miktarlarından oluşan veri kümesi  $S$  ve aldıkları notlardan oluşan veri kümesi  $N$  olarak gösterilirse bu durumda bu iki boyut arasındaki ilişkiyi bulabilmek için  $cov(S, N)$  hesaplanması gerekir.

Kovaryansın anlamı iki boyut arasındaki ilişkinin ne olduğunu vermektedir. Kovaryans eğer pozitif çıkarsa bunun anlamı iki veri kümesinden biri artarken diğeri de artmaktadır. Kovaryans eğer negatif çıkarsa bu iki veri kümesinden biri artarken diğeri azalmaktadır. Kovaryans eğer 0 çıkarsa bu iki veri kümesinin birbirinden bağımsız olarak değiştiğini göstermektedir. Aşağıda bir örnekle kovaryans matrisinin nasıl hesaplandığı incelenmektedir.

*Örnek:*

Öğrencinin çalışma saatleri ve aldıkları notlardan oluşan bir veri kümemiz olsun.

Tablo 1.3. Örnek Veri Kümesi

	Saat (S)	Not (N)
Veri	12	89
	10	45
	5	80
	2	25
	5	35
	0	14
	6	40
	14	52
	8	63
	7	30
Toplam	69	473
Ortalama	6.9	47.3

Verilen bu veri kümesinin kovaryans hesabı da aşağıdaki şekilde adım adım verilmektedir.

Tablo 1.4. Kovaryans hesabı

	Saat (S)	Not (N)	$(S_i - \bar{S})$	$(N_i - \bar{N})$	$(S_i - \bar{S})(N_i - \bar{N})$
Kovaryans	12	89	5.1	41.7	212.67
	10	45	3.1	-2.3	-7.13
	5	80	-1.9	32.7	-62.13
	2	25	-4.9	-22.3	109.27
	5	35	-1.9	-12.3	23.37
	0	14	-6.9	-33.3	229.77
	6	40	-0.9	-7.3	6.57
	14	52	7.1	4.7	33.37
				Toplam	545.76
				Kov.	77.96
	8	63	1.1	15.7	17.27
	7	30	0.1	-17.3	-1.73
Toplam				0	561.30
Ortalama (kovaryans)					56.13

Tablo 1.4'te elde edilen kovaryans değeri pozitif çıktığı için bu iki veri kümesi arasındaki ilişki beraber artan yönündedir.

#### 1.8.4. Kovaryans Matrisi

Kovaryansın 2 boyut arasındaki ilişkiyi ölçtüğü daha önce vurgulanmıştı. Eğer 2 den fazla veri kümesi varsa bu durumda her bir ikili arasında bir kovaryans değeri hesaplanır. Örneğin  $(x,y,z)$  gibi 3 veri kümesi olsun. Bu durumda boyut 3 olmaktadır. Aşağıda 3 boyut için hesaplanması gereken kovaryans matrisi görülmektedir.

$$C = \begin{pmatrix} cov(x,x) & cov(x,y) & cov(x,z) \\ cov(y,x) & cov(y,y) & cov(y,z) \\ cov(z,x) & cov(z,y) & cov(z,z) \end{pmatrix} \quad (33)$$

Burada C, elde edilen kovaryans matrisini göstermektedir. Dikkat edilmelidir ki  $cov(x,y)$  ile  $cov(y,x)$  değeri birbirine eşittir (Kovaryans işleminin değişme özelliği vardır). Bu durumda C matrisi simetrik bir matris olmaktadır.

Boyutun n olduğu düşünülürse hesaplanabilecek kovaryans değerleri sayısı (34) ile hesaplanır.

$$\frac{n!}{(n-2)!*2} = \frac{n*(n-1)}{2} \quad (34)$$

#### 1.8.5. Matris Cebiri

Bu bölümde Temel Bileşen Analizi için gerekli olabilecek matris işlemlerinde bahsedilmiştir. Temel matris işlemlerinin (toplama, çarpma, transpoz alma gibi) bilindiği varsayılarak özellikle öz değer ve öz vektör işlemleri üzerinde durulmuştur.

#### 1.8.6. Öz Vektörler

Birbirlerine boyut açısından iki matrisin çarpıldığını düşünelim. Aşağıdaki şekilde bunun bir örneği verilmiştir.

$$\begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix} \times \begin{pmatrix} 1 \\ 3 \end{pmatrix} = \begin{pmatrix} 11 \\ 5 \end{pmatrix} \quad (35)$$

$$\begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix} \times \begin{pmatrix} 3 \\ 2 \end{pmatrix} = \begin{pmatrix} 12 \\ 8 \end{pmatrix} = 4 \times \begin{pmatrix} 3 \\ 2 \end{pmatrix} \quad (36)$$

(35)'te normal bir matris çarpımı görülmektedir. (36)'da ise çarpımın sonucu orijinal vektörün 4 katı olacak şekilde çıkmıştır. (36)'daki  $[3 \ 2]^T$  vektörü 2 boyutlu düzlemde başlangıç  $(0,0)$  noktasından  $(3,2)$  noktasına olan bir vektörü temsil etmektedir. (36)'daki

diğer kare matris ise, transformasyon matrisi olarak düşünülebilir. Eğer soldaki bu matrisi başka bir vektörle çarparsak, aynı şekilde orijinal konumundan dönüştürülmüş başka bir vektör elde ederiz. (36)'da  $[3 \ 2]^T$  vektörü soldaki kare matrisin öz vektörü olarak bulunmuş olur.

Öz vektörler için aşağıdaki özellikler geçerlidir.

- Sadece kare matrisler için bulunabilir.
- Tüm kare matrislerin öz vektörleri olmayabilir. Yani her kare matrisin öz vektörü yoktur.
- $n \times n$  lik bir kare matrisin eğer öz vektörleri varsa bu öz vektörlerin sayısı  $n$  tanedir. Örneğin  $3 \times 3$  lük bir kare matrisin 3 adet öz vektörü bulunabilir.
- Öz vektörleri transformasyon matrisi ile çarpmadan önce belli bir sayı ile çarparsak yine aynı öz vektörü elde ederiz. Örneğin (37)'de  $[3 \ 2]^T$  öz vektörünü 2 ile çarptıktan sonra elde edilen yeni vektör  $[6 \ 4]^T$ , transformasyon matrisi ile çarpılması sonucu yine aynı  $[6 \ 4]^T$  vektörü elde edildi. Bunun nedeni, bir vektörün bir sayı ile çarpıldığında sadece uzunluğunun değişmesi ve yönünün değişmemesidir.

$$2X \begin{pmatrix} 3 \\ 2 \end{pmatrix} = \begin{pmatrix} 6 \\ 4 \end{pmatrix} \quad (37)$$

$$\begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix} X \begin{pmatrix} 6 \\ 4 \end{pmatrix} = \begin{pmatrix} 24 \\ 16 \end{pmatrix} = 4X \begin{pmatrix} 6 \\ 4 \end{pmatrix} \quad (38)$$

(37)'deki matrisin bir sabitle çarpılması sonucunda (38)'de bulunan öz değer değişmemektedir.

- Tüm bulunan öz vektörler ortogonal dir.
- Bulunan tüm öz vektörler birbirinden lineer bağımsızdır.

### 1.8.7. Öz Değerler

Öz değerler yukarıda anlatılan öz vektörlerle yakından ilişkilidir. Bir kare matrisin hesaplanan her bir öz vektörü için bir öz değeri vardır. Örneğin (36)'da hesaplanan öz vektör  $[3 \ 2]^T$  iken buna karşılık gelen öz değer ise 4 tür. Dolayısıyla öz değer ve öz vektörler bir (*öz değer, öz vektör*) çifti olarak anılırlar.

Örnek:

$$\begin{vmatrix} 3 & 0 & 1 \\ -4 & 1 & 2 \\ -6 & 0 & -2 \end{vmatrix} \quad (39)$$

matrisinin öz değer ve öz vektör çiftlerinden biri  $(1, [0 \ 1 \ 0]^T)$  dir.

$$\begin{vmatrix} 3 & 0 & 1 \\ -4 & 1 & 2 \\ -6 & 0 & -2 \end{vmatrix} \begin{vmatrix} X \\ 1 \\ 0 \end{vmatrix} = 1 \begin{vmatrix} X \\ 1 \\ 0 \end{vmatrix} \quad (40)$$

olarak sağlanması yapılabilir.

### 1.8.8. Öz Değer ve Öz Vektörlerin Hesaplanması

Önceki bölümlerde öz değer ve öz vektörlerin özelliklerinden kısaca bahsedilmişti. Bu bölümde bir kare matrisin öz değerlerinin nasıl bulunduğu ve bu öz değerlere karşılık gelen öz vektörlerin nasıl elde edildiğine değinilecektir.

A matrisi n boyutlu kare matris olsun.

$$AX = \lambda X \quad (41)$$

eşitliğini sağlayan  $\lambda$  değerine öz değer ve X vektörüne ise öz vektör denilmektedir. Bu değer ve vektörlerin bulunması (42) denklemini çözülerek hesaplanır.

$$|A - \lambda I| = 0 \quad (42)$$

Burada I, n boyutlu birim matris,  $\lambda$  ise bulunması gereken öz değerleri göstermektedir. A matrisi ise öz değerleri bulunacak kare matristir. (42)'deki denklem çözüldüğünde eşitlikten elde edilen denklem n. dereceden olacağından bulunan  $\lambda$  değerleri de n tane olacaktır.  $\lambda_i$  öz değerleri bulunduktan sonra bu değerler karşılık gelen  $X_i$  öz vektörleri de (43)'te verilen eşitlikle bulunur.

$$|A - \lambda_i I| X_i = 0, \quad i = 1, 2, \dots, n \quad (43)$$

Burada  $\lambda_i$  değeri i. öz değer ve  $X_i$  vektörü ise  $\lambda_i$  öz değerine karşılık gelen öz vektör olarak hesaplanır.

### 1.9. Temel Bileşen Analizi (Principle Component Analysis, PCA)

Temel bileşen analizi, yüz tanıma ve resim sıkıştırma gibi alanlarda kullanılan oldukça yararlı bir istatistiksel yöntemdir. PCA (Temel Bileşen Analizi), çok miktarda veriyi sınıflandırarak aralarındaki benzerlik ve farklılıkları bulduğundan otomatik dudak okuma için de kullanılabilir. PCA yöntemini daha iyi anlayabilmek için gerekli bazı istatistiksel yöntemler yukarıda verilmiştir.

PCA yöntemi çok miktarda veriler içinde örüntüleri, aralarındaki benzerlik ve farklılıkları çıkararak belirlemektedir. Aslında PCA, verileri analiz ederek bu verilerin temel bileşenlerini bulmaktadır. PCA yöntemi aşağıda verilmiştir[12].

- Analiz edilecek veri kümesini oluştur.
- Bu veri kümesinden ortalamaları bul.
- C kovaryans matrisini elde et.
- C kovaryans matrisinden öz değer ve öz vektörleri elde et.
- Kullanılacak öz vektörleri seç ve U transformasyon matrisini oluştur.
- Yeni veri kümesini türet.

Devam eden bölümlerde bu adımlar sırasıyla verilmektedir.

#### 1.9.1. Analiz Edilecek Veri Kümesini Elde Etme

Analiz edilecek veri kümesini elde etme işlemi PCA yönteminin birinci adımını oluşturmaktadır. Bu adımda analiz için kullanılacak veri kümesi oluşturulur. Bu adımda boyut miktarı ve kullanılacak örnek sayısı belirlenir. Örneğin elimizde 2 adet veri kümesi olsun. Bunlar,

$$X=[2.5 \ 0.5 \ 2.2 \ 1.9 \ 3.1 \ 2.3 \ 2 \ 1 \ 1.5 \ 1.1] \quad (44)$$

ve

$$Y=[2.4 \ 0.7 \ 2.9 \ 2.2 \ 3.0 \ 2.7 \ 1.6 \ 1.1 \ 1.6 \ 0.9] \quad (45)$$

olsun.

Bu durumda veri kümemiz 2 boyutlu (x ile y) ve örnek sayımız ise 10 olmaktadır.

### 1.9.2. Ortalamaların Elde Edilmesi

Ortalamaların elde edilmesi, PCA yönteminde ikinci adımı oluşturmaktadır. Bu adımda veri kümelerinin her bir boyutu için ortalamalar çıkarılır.  $x$  değerleri için  $\bar{x}$  ( $x$  değerlerinin ortalaması) ve  $y$  değerleri için  $\bar{y}$  ( $y$  değerlerinin ortalaması) elde edilir. Daha sonra her bir  $x$  değerinden  $\bar{x}$  değeri çıkarılır ve aynı şekilde her bir  $y$  değerinden  $\bar{y}$  değeri çıkarılır. Böylece yeni veri kümesi elde edilir. Bu aşamadaki işlemler Tablo 1.5'te verilmiştir.

Tablo 1.5. Ortalamaların elde edilmesi sonucu elde edilen veri kümesi

	x	y		$x - \bar{x}$	$y - \bar{y}$
Veri	2.5	2.4	Elde edilen	0.69	0.49
Kümesi	0.5	0.7	yeni veriler	-1.31	-1.21
	2.2	2.9		0.39	0.99
	1.9	2.2		0.09	0.29
	3.1	3.0		1.29	1.09
	2.3	2.7		0.49	0.79
	2	1.6		0.19	-0.31
	1	1.1		-0.81	-0.81
	1.5	1.6		-0.31	-0.31
	1.1	0.9		-0.71	-1.01

$\bar{x}=1.81$        $\bar{y}=1.91$

### 1.9.3. Kovaryans Matris Hesabı

Bu işlem daha önce bahsedilen kovaryans matris hesabındaki gibi yapılır. Son durumda 2x2 lik bir matris elde edilir. Yukarıdaki örneğe göre,

$$A = \begin{pmatrix} 0.69 & -1.31 & 0.39 & 0.09 & 1.29 & 0.49 & 0.19 & -0.81 & -0.31 & -0.71 \\ 0.49 & -1.21 & 0.99 & 0.29 & 1.09 & 0.79 & -0.31 & -0.81 & -0.31 & -1.01 \end{pmatrix}^T \quad (46)$$

olarak düşünülürse,

$$C = \frac{A^T \cdot A}{(n-1)} \quad (47)$$

çarpımından kovaryans matris elde edilir. Son durumda bulunan kovaryans matris (48)'deki gibidir.

$$C = \begin{pmatrix} 0.616555556 & 0.615444444 \\ 0.615444444 & 0.716555556 \end{pmatrix} \quad (48)$$



Görüldüğü gibi köşegen üzerinde olmayan elemanlar pozitif olduğundan  $x$  ve  $y$  değişkenlerinin birlikte arttığı söylenebilir.

#### 1.9.4. Kovaryans Matrisinden Öz Değer ve Öz Vektörleri Elde Etme

Kovaryans matris, kare matris olduğundan ve boyutu 2 olduğundan bu adımda 2 adet öz vektör ve bu öz vektörlere karşılık 2 adet de öz değer hesaplanır. Hesaplanan öz vektörler ve bu öz vektörlere karşılık gelen öz değerler (49) ve(50)'de verilmiştir.

$$\text{özdeğerler} = \begin{vmatrix} 0.0490833989 \\ 1.28402771 \end{vmatrix} \quad (49)$$

ve

$$\text{özvektörler} = \begin{vmatrix} -0.735178656 & -0.667873399 \\ 0.667873399 & -0.735178656 \end{vmatrix} \quad (50)$$

olarak elde edilir. Elde edilen öz değerlerin ilk elemanı öz vektörler matrisindeki ilk sütuna karşılık gelir[24].

#### 1.9.5. Öz Vektörleri Seçme ve Transformasyon Matrisini Oluşturma

Önceki bölümde elde edilen öz değerlerin birbirinden oldukça farklı olduğu görülmektedir. Bu değerlerin en büyüğü veri kümesinin temel bileşeni olarak kabul edilir. Bu en büyük değer veri kümesindeki farklı boyutlar arasındaki en önemli ilişkiyi belirtir.

Öz değerler bulunduktan sonra bu değerler büyükten küçüğe doğru sıralanır. Doğal olarak sıralama sırasında yer değiştiren öz değerlere karşılık gelen öz vektörler üzerinde de aynı şekilde yer değiştirme işlemi yapılır. Yani en büyük öz değer, öz değer vektörünün ilk elemanı olurken bu öz değere karşılık gelen öz vektör de öz vektörler matrisinin ilk sütununu oluşturur.

Öz değerler sıralandıktan sonra isteğe bağlı olarak küçük olan bazı değerler silinebilir. Ya da sıfıra yakın olan öz değerler elenebilir. Bu durumda bazı bilgi kayıpları olacaktır ancak küçük öz değerler elendiği için bu bilgi kayıpları anlamsız olacaktır ve fark edilmeyecektir. Resim sıkıştırmada bu bilgi kayıpları gözle görülür derecede fark edilmediği için bir sorun oluşturmamaktadır.

Öz değerlerin sayısı  $n$  iken bazı değerler elendikten sonra bu sayı  $p$  ye düşecektir. Elenen öz değerlere karşılık olan öz vektörler de eleneceğinden başlangıçta  $n$  olan boyut

sayısı şimdi  $p$  ye düşecektir. Elenen öz vektörlerden geriye kalan öz vektörler matrisin sütunlarını oluşturacak şekilde yeni matrise yerleştirilir. Bu matris PCA yöntemi için özellik vektörü olarak kabul edilir.

$$\text{ÖzellikVektörü} = \left| \begin{array}{cccc} \text{özvek}_1 & \text{özvek}_1 & \text{özvek}_3 & \dots & \text{özvek}_p \end{array} \right| \quad (51)$$

(51)'de öz değerler sıralandıktan sonra oluşan öz değer ve öz vektörler (52) ve (53)'te verilmiştir.

$$\text{özdegerler} = \left| \begin{array}{c} 1.28402771 \\ 0.0490833989 \end{array} \right| \quad (52)$$

ve

$$\text{özvektörler} = \left| \begin{array}{cc} -0.677873399 & -0.735178656 \\ -0.735178656 & 0.677873399 \end{array} \right| \quad (53)$$

Öz değerlerin küçük olanları elendikten sonra geriye özellik vektörü olarak sütunları öz vektörlerden oluşan bir matris kalmaktadır. Örneğimizde (54)'teki son durum elde edilir[24].

$$\text{özellik\_vektörü} = \left| \begin{array}{c} -0.677873399 \\ -0.735178656 \end{array} \right| \quad (54)$$

### 1.9.6. Yeni Veri Kümesinin Türetilmesi

Bu aşama PCA yönteminin son aşamasıdır. Bölüm 1.9.5'te özellik vektörü oluşturuldu. Oluşturulan bu özellik vektörü transformasyon matrisi olarak saklanır. Orijinal veri kümesindeki her bir örnekten oluşan matris ile bu transformasyon matrisi çarpılarak yeni veri kümesi elde edilir.

$$\text{yeni\_veri} = [\text{özellik\_vektörü}]^T \times [\text{orjinal\_veri\_satiri}] \quad (55)$$

Burada *özellik\_vektörü* olarak bölüm 1.9.5'te elde edilen matrisin transpozunu alınarak elde edilen matris kastedilmiştir. *Orijinal\_veri\_satiri* ise başlangıçtaki verilerden elde edilen satır vektörleridir. Yeni\_veri adında elde edilen yeni vektörler her bir örnek için hesaplanmaktadır. Bu durumda 10 adet örnek alınmışsa elde edilecek yeni\_veri vektörleri de 10 adet olacaktır. Yeni\_veri vektörleri artık veri tabanına kaydedilerek daha sonra karşılaştırmada kullanılmaktadır. Bu yeni\_veri vektörleri her bir orijinal örneğin imzası olmaktadır.

Buraya kadar PCA yöntemi ile veri kümelerinin bir dönüşümü alınmış ve temel

bileşenler bulunmuştur. Başlangıçtaki verilerin ortalaması alınarak temel bileşenleri bulunmuş ve her bir verinin bir imzası oluşturulmuştur. Eğitim (PCA dönüşümü) aşamasında kullanılan örneklerin dışında bir veri geldiğinde bu veri bölüm 1.9.5'te elde edilen transformasyon matrisi ile çarpılarak aday imza elde edilir. Bu aday imza daha önce eğitim aşamasında oluşturulan ve veri tabanına kaydedilen orijinal imzalarla teker teker karşılaştırılarak her birine yakınlığı tespit edilir. Aday imza, orijinal imzalardan hangisine en yakın bulunursa o imzaya karar verilir. İkinci bölümde eğitim ve tanıma aşamaları daha ayrıntılı bir şekilde incelenmektedir.

## **2. YAPILAN ÇALIŞMALAR, BULGULAR VE İRDELEME**

### **2.1. Giriş**

Bu bölümde otomatik dudak okuma sistemi için gerekli olan işlemler üzerinde durulmaktadır. Otomatik dudak okuma sistemi için sırayla aşağıdaki işlemler gerçekleştirilmelidir.

- Renkli Resimde Yüzün tespiti
- Dudak bölgesi tespiti
- Dudak bölgesinde dudak tespiti
- Tespit edilen dudağın özellik vektörünün elde edilmesi
- Elde edilen özellik vektörüne göre otomatik dudak okuma

Devam eden bölümlerde bu işlemlerin otomatik dudak okuma için nasıl yapıldıkları incelenmektedir.

### **2.2. Resimde Yüzün Belirlenmesi**

Bu güne kadar insan yüzünün belirlenmesi ve izlenmesi için birçok çalışma yapılmıştır[13,14]. Bu çalışmalar günümüzde de devam etmektedir. Kesin ve tam bir sonuç üretmek için birçok sezgisel ve özellik tabanlı yaklaşım önerilmektedir. Özellik tabanlı bu çalışmalar arasından ten rengini tanıma aracı olarak kullanan yöntemler diğerlerine göre daha fazla önem kazanmaktadır. Ten renginin insanlar tarafından kolaylıkla ayırt edilebildiği düşünüldüğünde, bu özelliğin yüz tespiti için kullanılmasının daha doğru olduğu görülmektedir. Bu tür bir sistem tasarlarken aşağıdaki 3 sorun ile karşılaşılır.

- Hangi renk uzayı kullanılmalıdır?
- Tam olarak ten rengi olan pikseller diğerlerinden nasıl ayırt edilir?
- Renk ayırımından sonra bölütleşmiş resimden yüz nasıl bulunur?

Çalışmalarda yüz tespitinde bu soruların cevabı aranmış farklı renk uzaylarında piksel tabanlı çalışmalar incelenmiştir. Yüzün ayırt edilebilmesi için öncelikle ten renginin bulunması üzerinde durulmaktadır.

### 2.3. Ten Rengi Bulma Algoritmaları

#### 2.3.1. Sınırlamalar Yardımıyla Ten Renginin Bulunması

En basit olarak ten renginin diğer renklerden ayırt edilebilmesi için RGB renk uzayında bazı sınırlamalar kullanarak yüzün bulunması amaçlanmıştır. Örneğin RGB uzayında,

$$\begin{aligned} R > 95 \ \& \ G > 40 \ \& \ B > 20 \ \& \\ \max\{R,G,B\} - \min\{R,G,B\} > 15 \ \& \\ |R-G| > 15 \ \& \ R > G \ \& \ R > B \end{aligned} \quad (56)$$

şeklinde kısıtlamalar göz önüne alınarak ten rengi ayırt edilebilmektedir. [13]

Ten rengi olan pikseller bu sınırlamalar içine girebilir. Bu yöntemin gerçekleştirilmesinin çok kolay olduğu görülmekte ancak istenilen performansı gösterememektedir. Aynı sınırlamalar farklı uzaylarda da bulunabilir. Örneğin YIQ uzayı için

$$(60 < Y < 200) \ \vee \ (20 < I < 50) \quad (57)$$

şeklinde bir sınırlama ile ten renginin ayırt edilebilmesi amaçlanabilmektedir. Ancak bu tür sınırlamalar da tam olarak ten renginin ayırt edilebilmesini sağlamamaktadır.

#### 2.3.2. YCbCr Renk Uzayı ile Ten Renginin Belirlenmesi

İnsan yüzü ile ilgili işlemler yapan sistemlerde, en büyük sorun görüntüde yüzün bulunmasıdır. Genelde yüzün konumu konusunda bazı ön kabuller yapılarak arama yapılır, ancak bu ön kabuller çoğu zaman gerçek hayat uygulamaları için doğru sonuç vermemektedir. Hsu vd.[15] 2002’de geliştirdikleri yüz tanıma algoritması, renkli resimlerde yüzü bulurken ilk adım olarak görüntüde ten rengi olabilecek pikselleri başarıyla ayırt edebilmektedir. Renkli görüntüde ten rengi bulabilmek için yapılması gereken en önemli adım bahsedilen renk uzaylarından doğru olanı seçebilmektedir[15].

Ten rengi tanıma konusunda farklı renk uzaylarında birçok çalışma yapılmış ve en iyi sonuçların TSL uzayında elde edildiği belirlenmiştir. Ancak RGB uzayından TSL uzayına çift taraflı dönüşüm bağıntıları ağır matematiksel ifadeler içerdiğinden, gerçek zamanlı uygulamalar için pek uygun değildir. Bu nedenle Hsu vd.[15] rengin renklilik ve parlaklık bileşenlerini ayırma şekli TSL uzayına benzeyen YCbCr uzayını kullanarak ten rengi

tanıma uygulaması yapmışlardır. Yapılan çalışmaların çoğunda ten rengi olan pikseller için renklilik bileşenlerinin parlaklık bileşeninden bağımsız olduğu varsayımı yapılmıştır, ancak ten rengi tonları lineer olmayan bir şekilde parlaklığa bağlıdır. Hsu vd. [15] ise yüz veritabanından alınan ten rengi örneklerini YCbCr uzayında belirlenmiş ve bazı matematiksel fonksiyonlar yardımıyla dönüşümler yaparak bu bağımlılığın ortadan kaldırılması sağlanmıştır. Böylece YCbCr uzayında belirli sınırlar içinde kalan pikseller ten rengi olarak değerlendirilmiştir.

Yapılan ten rengi belirleme araştırmaları içinde en iyi sonucu bu yöntem verdiği için çalışmada bu yöntem kullanılmıştır. Şekil 2.1’de programın çıktısına ait örnek bir resim verilmiştir.



(a)Orijinal resim

(b)Ten rengi olan pikseller

Şekil 2.1. Ten renginin tespit edilmesi

YCbCr uzayında renklilik bileşenleri ( $C_b$  ve  $C_r$ ) parlaklık( $Y$ ) bileşenin fonksiyonu olarak  $C_b(Y)$  ve  $C_r(Y)$  şeklinde ifade edilirse,  $C_b'(Y)$  ve  $C_r'(Y)$  ise dönüştürülmüş renklilik bileşenleridir. Gerekli dönüşümler aşağıda verilmiştir.

$C_i$  ifadesi  $C_b$  ya da  $C_r$  ye karşılık gelir.

$$C_i'(Y) = \begin{cases} (C_i(Y) - \bar{C}_i(Y)) \cdot \frac{W_{C_i}}{W_{C_i}(Y)} + \bar{C}_i(K_h) & , Y < K \\ C_i(Y) & , Y \in [K_l, K_h] \end{cases} \quad (58)$$

$$W_{C_i}(Y) = \begin{cases} WL_{C_i} + \frac{(Y - Y_{\min}) \cdot (W_{C_i} - WL_{C_i})}{K_l - Y_{\min}} & , Y < K_l \\ WH_{C_i} + \frac{(Y_{\max} - Y) \cdot (W_{C_i} - WH_{C_i})}{Y_{\max} - K_l} & , K_h < Y \end{cases} \quad (59)$$

$$\bar{C}_b(Y) = \begin{cases} 108 + \frac{(K_l - Y) \cdot (118 - 108)}{K_l - Y_{\min}} & , Y < K_l \\ 108 + \frac{(Y - K_h) \cdot (118 - 108)}{Y_{\max} - K_h} & , K_h \leq Y \end{cases} \quad (60)$$

$$\bar{C}_r(Y) = \begin{cases} 154 - \frac{(K_l - Y) \cdot (154 - 144)}{K_l - Y_{\min}} & , Y < K_l \\ 154 - \frac{(Y - K_h) \cdot (154 - 132)}{Y_{\max} - K_h} & , K_h \leq Y \end{cases} \quad (61)$$

Sabit değerler:

$$W_{Cb}=46.97, WL_{Cb}=23, WH_{Cb}=14, W_{Cr}=38.76, WL_{Cr}=20, WH_{Cr}=10, K_l=125, K_h=188$$

olarak alınır.

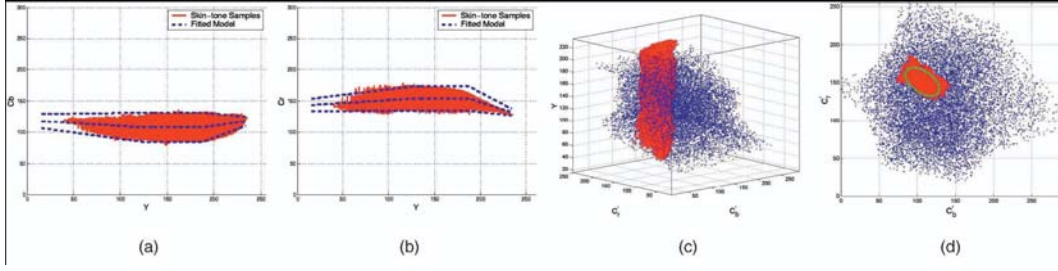
Bu sabitler veritabanındaki örnek ten rengi verilerinden elde edilen değerlerdir.  $Y_{\min}$  ve  $Y_{\max}$  sabitleri YCbCr uzayında sırasıyla 16 ve 236 değerlerine eşittir ve ten rengi olabilecek pikseller için eliptik model Cb'Cr' uzayına dönüştürülmüştür.

$$\frac{(x - ec_x)^2}{a^2} + \frac{(y - ec_y)^2}{b^2} = 1 \quad (62)$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} C'_b - c_x \\ C'_r - c_y \end{bmatrix} \quad (63)$$

Burada  $C_x = 109.8$ ,  $C_y = 152.2$ ,  $\theta = 2.53$ ,  $ec_x=1.60$ ,  $ec_y=2.41$ ,  $a=25.39$  ve  $b=14.03$  dir.

(62) ve (63) denklemini sağlayan x ve y değerleri ten rengi bölgelerinin Cb'Cr' uzayında oluşturduğu kümeye denk gelmektedir ve eliptik alan içinde kalan pikseller ten rengi olarak bulunmaktadır.



Şekil 2.2. YCbCr alt uzayına geçiş ve Cb'Cr' uzayına dönüşüm(c ve d).

Yapılan testlerde, görüntüdeki bazı piksellerin ten rengine çok benzemesi nedeniyle hatayla ten olarak belirlenmesi mümkün olmaktadır. Bulunan yüz pikselleri kahverenginin tonlarına yatkındır. Bu nedenle yanlışlıkla ten rengi olarak belirlenen alanlarında yüz bulunurken elenecektir.

### 2.3.3. Sabit Arka Planlı (siyah) Resimlerde Yüzün Belirlenmesi

Renkli resimlerde yüz bölgesinin belirlenmesinin bir diğer yolu ise sabit arka plana sahip görüntülerle çalışmaktır. Arka plan olarak yüzün rengine zıt bir renk kullanıldığında eşikleme yardımıyla cisim arka plandan kolaylıkla ayırt edilebilir.

Yüzün bulunması için yapılması gerekenler işlemler şu şekilde sıralanabilir:

- Gelen resim gri seviye resme dönüştürülür.
- Gürültülerin elenmesi için ortalama alma işlemi uygulanır.
- Gri seviyedeki resim Otsu algoritması kullanılarak ikili resme dönüştürülür.
- Bu aşamadan sonra yüzün sınırları belirlenmelidir. Resimde yüz, eller ve yanlış olarak bulunmuş sonuçlar olabilir. Bu alanlar ayırt edilmektedir

Önce bölütleme yapılarak gruplar etiketlenmelidir. Birbiri ile bağlantılı olan (komşu) 1 değerleri piksellere aynı etiket numarası verilir.



0	1	1	1	1	0	0	0	0
0	0	1	0	1	0	0	0	0
0	0	1	1	1	0	0	0	0
0	0	0	1	1	0	0	0	0
0	0	0	0	0	0	0	1	0
1	1	0	0	0	1	1	1	0
1	1	0	0	0	1	1	1	0
0	1	1	0	0	1	1	1	0
0	1	1	0	0	0	0	0	0

(a) İkili resim

	1	1	1	1				
		1		1				
		1	1	1				
			1	1				
							2	
3	3				2	2	2	
3	3				2	2	2	
	3	3			2	2	2	
	3	3						

(b) Bölütleme sonucu

Şekil 2.3. İkili resimde bölütleme işlemi

Bölütleme sonucunda kaç tane grup olduğu anlaşılmaktadır. Şekil 2.3'te verilen orijinal görüntünün bölütlenmesi sonucunda resimde 3 bölge olduğu tespit edilmiştir. Bölütlemeden sonra hangi grubun yüz olduğu belirlenmelidir. Bunun için her bir grupta aşağıdaki özellikler bulunmalıdır.

- Grubun alanı ( kaç tane 1 değerli piksel içerdiği )
- Alanın ağırlık merkezi
- Ortalama yükseklik ve genişlik hesaplanır.

Ortalama yükseklik hesaplanırken ağırlık merkezin üstünde kalan kısmın ağırlık merkezi  $C1$ ; ağırlık merkezinin altında kalan kısmın ağırlık merkezi  $C2$  bulunur.  $(C1+C2)*2$  değeri de ortalama yüksekliği vermiş olur.

Aynı işlem ortalama genişliğin bulunmasında da yapılır.

Hangi alanın yüz olduğu belirlenirken yüzün oranlarında faydalanılır. İnsan yüzünün boyunun enine oranı yaklaşık olarak altın orana eşittir.

*Ortalama\_yükseklik/ortalama\_genişlik* oranı bu değerler arasında çıkan grup yüz olarak belirlenir. Ancak hatalı olarak oluşmuş alanlarda bu değeri verebilir. Bu nedenle bu oranı sağlayan ve alanı en büyük olan bölüm yüzün bulunduğu bölüm olarak belirlenir ve çerçeve içine alınır. Burada çerçeve içerisinde yalnızca bir yüz olduğu ve dudak okuma için sadece tek bir yüz incelendiği varsayılmaktadır.

Şekil 2.4 ve Şekil 2.5'te yüz tespiti için elde edilen örnek çıktılar verilmektedir.



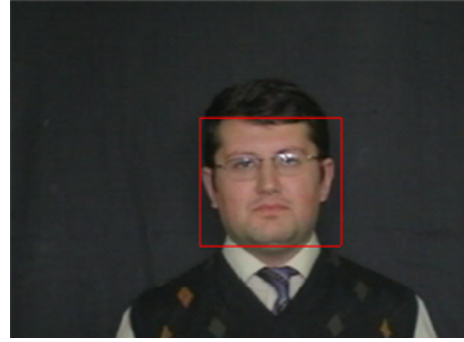
(a) Orijinal resim



(b) Bulunan yüz

Şekil 2.4. Sabit arka plana göre yüzün bulunması

Yüz belirleme programının çıktısından örnekler Şekil 2.5'te verilmiştir.



Şekil 2.5. Çeşitli resimlerden tespit edilen yüzler

## 2.4. Dudak Bölgesinin Bulunması

Bilgisayarlı dudak okuma uygulamalarında dudağın tespiti oldukça önemlidir. Dudağın tam olarak yüzün hangi bölgesinde olduğunun bulunması değişik yüz görüntüleri üzerinde farklılık göstermektedir. Ayrıca dudağın tespiti işlemi her defasında yüz üzerinde gerçekleştirilirse gereken işlemler oldukça uzun zaman alacaktır. Bu nedenlerle yüz üzerinde yaklaşık olarak dudak bölgesinin bulunması önem kazanmaktadır. Bir kere dudak bölgesi bulunduktan sonra artık dudağın tüm hareketleri bu bölgede sınırlı kalacak ve inceleme işlemi sadece bu bölge ile sınırlı kalacaktır.

Dudak bölgesi bulma işlemi basitlik açısından tam düz bakan yüzler için gerçekleştirilmiştir. Ancak hafif sağa ya da sola dönmüş yüzler için ise ayrıca gereken yaklaşımlar ilerleyen aşamalarda tartışılmıştır.

Dudak bölgesinin yereltilmesi işlemi de denilen dudak bölgesi bulma aslında dudağın içinde olduğu bölgenin bulunmasından ibarettir. Önerilen yaklaşım geometrik yaklaşımdır. Bu yaklaşıma göre dudak bölgesi yüz üzerinde belli sabit oranlar dikkate alındığında yatay olarak yüzün tam ortasında ve düşey olarak ise yüzün alt kısmındadır. Ancak bulunan yüz görüntülerinde alın açıklığı veya alnın üzerine saç ve benzeri gürültülerin gelmesi durumunda yüz üzerinde elde edilen geometrik oranlarda da sapmalar olacaktır. Bu yüzden dudak bölgesinin daha sağlıklı tespiti için önce yüz üzerinde gözlerin yerinin düşey olarak belirlenmesi daha uygun olacaktır. Dudak bölgesinin tam olarak tespiti için gereken yöntem aşağıda verilmiştir.

- Yüz üzerinde gözün konumunu bul
- Göz konumuna göre belli sabit oranlar alarak dudak bölgesini bul

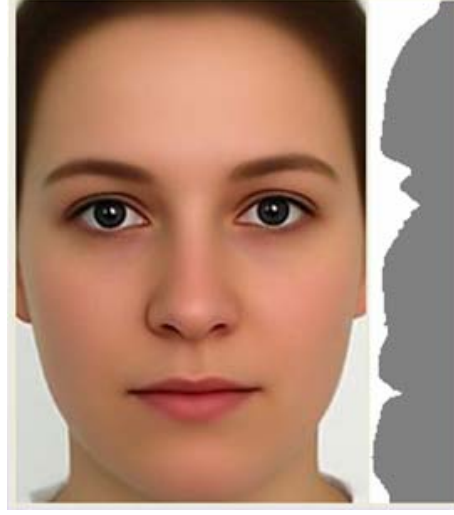
Önerilen yöntemeye göre öncelikle gözün düşey pozisyonunun belirlenmesi gerekmektedir.

### 2.4.1. Gözün Bulunması

Gözün bulunması işlemi gözün düşey olarak hangi konumda olduğunun tespit edilmesi işlemidir. Bunun için öncelikle yüzün düşey dağılımı incelenir. Düşey dağılımın bulunması şu şekilde gerçekleşmektedir. Yüzün her bir satırındaki görüntü elemanlarının (piksellerin) parlaklık değerleri toplanır.

- Her bir satır için hesaplanan bu parlaklık değerleri [0-1] arasına normalize edilir.

- Normalized values are processed on the face image for each row to obtain the graph in Figure 2.6.

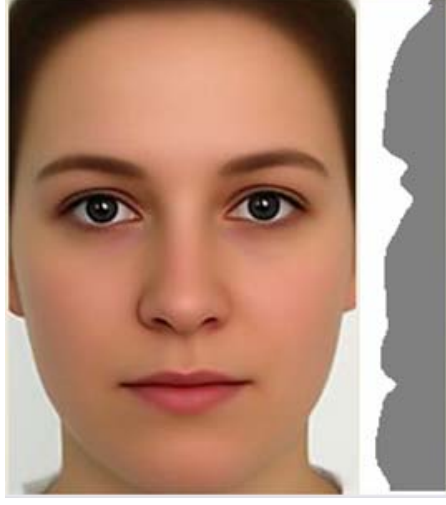


(a)

(b)

Şekil 2.6.Yüz görüntüsü ve yüz görüntüsünün düşey dağılım grafiği

Bulunan düşey dağılım incelendiğinde yüz görüntüsünün üst yarı bölgesinde minimum noktalar göze çarpmaktadır. Bulunan bu iki minimum noktanın üst kısımda kalan tarafı kaşların olduğu yeri, ikinci minimum nokta ise gözün konumunu vermektedir. Gözün konumunu tespit ederken gürültülerden dolayı minimum bulma probleminde yerel minimuma takılmamak için düşey dağılım üzerinde düzgünleştirme işlemi gerçekleştirilir. Dağılım düzgünleştirme işlemi, histogram yumuşatma işleminin yapıldığı gibi yapılmaktadır. Bu işlem bölüm 1.4.6'da verilmiştir. Dağılım düzgünleştirme işlemi sonucu Şekil 2.27'de verilmiştir.



Şekil 2.7. Dağılım düzgünleştirme sonucu

#### 2.4.2. Gözün Bulunması

Düşey dağılım incelendiğinde yüz görüntüsünün üst yarı bölgesinde iki adet yerel minimum nokta göze çarpmaktadır. Bu yerel minimum noktalardan üst tarafta kalan genellikle kaşların, alt taraftaki minimum nokta ise gözlerin konumunu vermektedir.

Göz konumu bulunduktan sonra gözün alt bölgesinde bulunan yüz bölgeleri üst tarafa göre daha değişmezlik göstermektedir. Örneğin alın açıklığı, değişik yüzlerde daha büyük ya da daha küçük olabilmektedir. Ancak yüz görüntüsü üzerinde göz konumundan aşağı kalan kısımda değişik insan yüzleri için çok fazla konum değişikliği olmamaktadır. Zaten göz konumunun bulunmasının amacı da bu değişmezlikten yararlanmaktır.

Bu aşamadan sonra dudağın içinde bulunduğu dudak bölgesinin bulunması amaçlanmaktadır.



itibaren Y/6 kadar uzaklıkta olacaktır. Son durumda bulunan dudak bölgesi Şekil 2.9b’de verilmiştir.

## 2.5. Dudak Tespiti

Otomatik dudak okuma işleminde söylenen kelimenin anlaşılabilirliğinin en fazla olduğu özellik dudağın dış sınırlarıdır. Bunun dışında dudak okuma için gerekli olabilecek özellikler şu şekilde sıralanabilir.

- Dudağın dış sınırları
- Dudağın alt ve üst yükseklikleri
- Dudak eni ve genişliği
- Ağız açıklığı
- Dişlerin görünüp görünmemesi
- Ağız açıklığının geometriksel alanı

Tüm bu özelliklerin bulunabilmesi ve dayanıklı bir dudak okuma sisteminin elde edilmesi için öncelikle dudağın ikili şeklinin çıkarılması gerekmektedir. Bu işleme dudak tespiti (dudak bölütleme) denilmektedir. Ayrıca ağız açıklığının bulunması ve incelenmesi ayrı bir problem olarak ele alınacaktır. Otomatik dudak okuma için önerilen yöntemler bölüm 2.9’da incelenmektedir.

### 2.5.1. Dinamik Eşiklemeye Dayalı Dudak Tespiti

Bu yöntemde bir takım ön görüntü işleme teknikleri kullanarak dinamik eşikleme yapılmıştır. Uygulanan adımlar şu şekildedir.

- Dudak bölgesinin gri seviyeye çevrilmesi
- Dudak bölgesinde ön işlemlerin gerçekleştirilmesi
- Dinamik eşik değerinin elde edilmesi
- Elde edilen eşik değerine göre dudak bölgesinin ikili hale dönüştürülmesi
- Simetrik katlama yönteminin uygulanması
- İkili olarak oluşan bölgelerin yorumlanarak dudak olan bölgeye karar verilmesi
- Zincirleme metodu ile dış sınırların elde edilmesi
- Dış sınırları temsil edecek noktaların elde edilmesi

### *Dudak Bölgesinin Gri Seviye Çevrilmesi*

Görüntüde elde edilen her bir çerçeve aynı zamanda bir resimdir. Bu resim genellikle RGB renk uzayında olmaktadır. Bu renk uzayında R, G ve B bileşenleri kullanılarak görüntünün parlaklık değerleri elde edilmekte ve bu parlaklık değerleri ise gri seviye değerlerini oluşturmaktadır. Gri seviye parlaklık değerine P dersek RGB renk uzayından gri seviye parlaklık değerinin elde edilmesi (64)'deki bağıntıya göre olmaktadır.

$$P=0.11*R + 0.59*G + 0.30*B \quad (64)$$

### *Dudak Bölgesinde Gerçekleştirilen Ön İşlemler*

Dudak bölgesinde işlemlerin daha kolay gerçekleştirilmesi ve ani gürültü değişimlerinden etkilenmeyi en aza indirmek için görüntü üzerinde ortalama alma işlemi uygulanır. Ortalama alma, her bir değer etrafındaki değerlerin ortalamasından oluştuğu esasına dayanır.

Ortalama alma işlemi uygulandıktan sonra oluşan yeni resim biraz daha az keskin renk değişimlerine sahiptir. Şekil 2.10. Ortalama alma işleminin uygulaması 'da gri seviye resmin ortalama alma işleminin uygulanmış hali görülmektedir.



Şekil 2.10. Ortalama alma işleminin uygulaması

### *Dinamik Eşik Değerinin Elde Edilmesi*

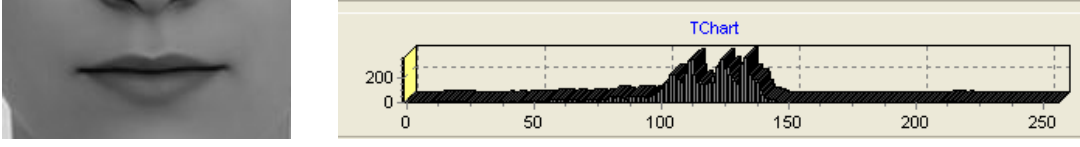
Dinamik eşik değerinin elde edilmesi Otsu algoritmasına göre olmaktadır. Otsu algoritması gri seviye resimdeki parlaklık değerlerinin dağılımından(histogram) yararlanarak görüntünün dağılımını iki bölgeye ayırmaktadır[11].

### *Gri Seviye Resmin Histogramının Çıkarılması Ve Resmin İkiliye Dönüştürülmesi*

Gri seviye resmin histogram değeri her bir parlaklık değerinin tüm görüntüde kaç kez bulunduğu karşılık gelmektedir. Bir başka deyişle histogram dağılımı görüntüdeki her bir parlaklık değerinin frekansına karşılık gelmektedir. Şekil 2.11'de histogram dağılım

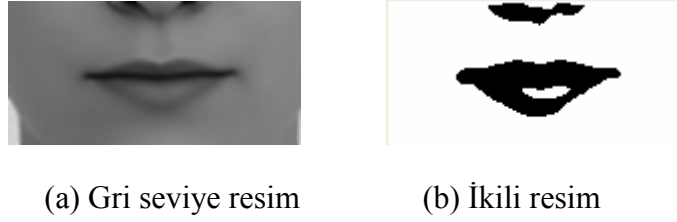


grafığı verilmektedir.



Şekil 2.11. Histogram dağılım grafığı

Otsu algoritması, görüntüdeki piksel değerlerinin dağılımlarına göre bu piksellerin kümelenmesini sağlamaktadır. Ayrıntılı bilgi birinci bölümde verilmişti. Şekil 2.11. Histogram dağılım grafığı'nda bulunan eşik değeri  $t^*=95$ 'tir. Bulunan eşik değerine göre elde edilen ikili resim Şekil 2.12'de verilmiştir.



Şekil 2.12. Elde edilen eşik değerine göre resim ikiliye çevrilmesi

### *Simetrik Katlama*

Şekil 2.12. Elde edilen eşik değerine göre resim ikiliye çevrilmesi'nde elde edilen ikili resimde dudak sınırları net bir şekilde çıkmakta ancak gölge ve benzeri nedenlerden dolayı tam simetrik çıkması gereken dudak, simetrik çıkması gerekirken sınırlarda bazı eksiklikler olmaktadır. Bu eksiklikleri gidermek için dudak üzerinde simetrik katlama gerçekleştirilir.

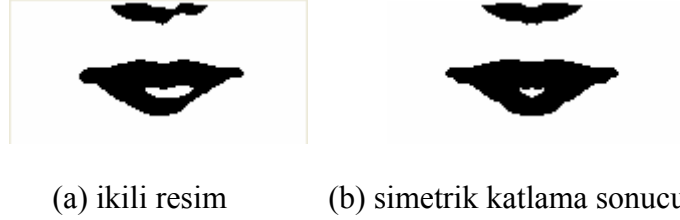
Simetrik katlama yapılabilmesi için öncelikle dudağın simetri ekseninin bulunması gerekir. Basitlik açısından dudağı incelenen kişinin tam düz baktığı varsayılır. Bu durumda simetri eksenini dudak bölgesi görüntüsünün tam ortası olmaktadır. Ancak tam düz bakılmadığı durumlarda görüntünün düşey histogram dağılımı incelenerek simetri eksenine karar verilir.

Simetrik katlama yapılırken görüntünün tam ortadan ikiye katlandığı varsayılır. Ve her bir piksel değeri kendisine karşılık gelen diğer piksel değeri ile *mantıksal veya* işlemine tabi tutulur. Çıkan sonuç her iki pikselin de değeri olur. Uygulanan *mantıksal veya* işlemi

(65)'teki formülde verilmiştir.

$$X_i = OR(X_i, X_{n-i}) \quad (65)$$

Son durumda simetrik katlama sonucu elde edilen ikili görüntü Şekil 2.13'de verilmiştir.



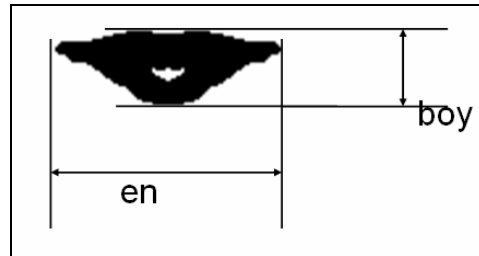
Şekil 2.13. Simetrik katlama işlemi

#### *İkili Olarak Oluşan Bölgelerin Yorumlanarak Dudak Olan Bölgeye Karar Verilmesi*

Şekil 2.13. Simetrik katlama'deki resimde görüldüğü gibi elde edilen ikili resimde gürültü ve gölgelerden dolayı dudaktan başka bölgeler de elde edilmektedir. Bu bölgeler istenmeyen bölgeler olduğundan gürültü olarak isimlendirilir. Bu bölgelerin elenmesi ve geriye dudağın kendisinin kalması işlemine gürültü elemine işlemi de denilmektedir. Gürültü eleme işlemi, ikili resimdeki bölgelerin bulunarak dudak olmayan bölgelerin elenmesi işlemidir. Dudak bölgesi yorumlanırken şu ölçütler dikkate alınmalıdır:

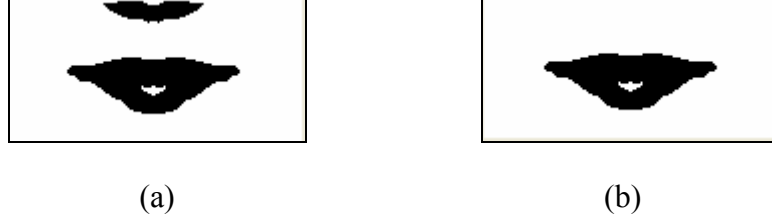
- Dudak şekli dudak bölgesinin büyük bir bölümünü doldurmalıdır. (%10 dan fazlasını)
- Dudak şekli yaklaşık simetrik olmalıdır.
- Dudak şeklinin eni boyundan daha uzun olmalıdır ( $1.2 \leq en/boy \leq 1.9$ ).

Şekil 2.14'te bu oran dudak eni ve boyu gösterilmektedir.



Şekil 2.14. Dudak eni ve boyu

Yukarıdaki ölçütlerin teker teker değerlendirilmesi için elde edilen ikili resim üzerindeki bölgelerin bulunması gerekmektedir. Bu bölgeler tüm resim taranarak bölge büyüme esasına dayanarak bulunur. Şekil 2.15'te bulunan 2 bölgenin alanı, eni, boyu ve benzeri parametreler bulunarak dudak olmayan bölge elenir ve son durumda Şekil 2.15b'deki görüntü elde edilir.



Şekil 2.15.Dudağın elde edilmesi

#### *Zincirleme metodu ile dış sınırların elde edilmesi*

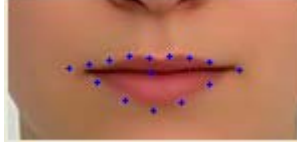
Dudak tespit edildikten sonra bu dudağın dış sınırlarının bulunması gerekmektedir. Dudağın en soldaki piksel değeri bulunduktan sonra zincirleme yöntemi izlenerek dış sınırlar bulunur. Çıkarılan dış sınırlar Şekil 2.16'de verilmiştir.



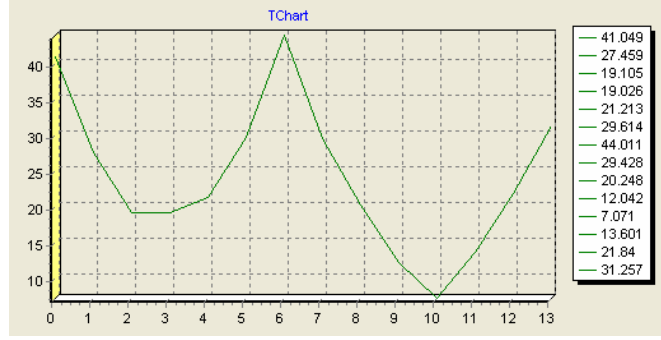
Şekil 2.16.Dudağın dış sınırlarının bulunması

#### *Dış sınırları temsil edecek noktaların bulunması*

Dudak okumada en önemli parametrenin dudağın dış sınırları olduğu çoğu araştırmacı tarafından kabul görmüştür. Bu nedenle bu noktaların elde edilmesi için 14 nokta yöntemi önerilmiştir. Bu yöntemde göre dudak çevresinden alınacak bu 14 nokta dudağın dış sınırlarını temsil edecektir. Ayrıca elde edilecek bu 14 nokta dudağın piksel olarak büyük ya da küçük olmasından etkilenmemektedir. 14 noktanın belli başlı olan 4 tane noktası sağ, sol, alt ve üst noktalarıdır. Geri kalan noktalar ise bu noktaların arasında olan ve ağırlık olarak daha az ağırlığa sahip noktalarıdır.



Şekil 2.17.Dış sınırları temsil edecek noktalar



Şekil 2.18.Elde edilen noktaların ağırlık merkezine olan uzaklıklarının grafiği

Bu noktalardan elde edilecek özellik vektörü bu dudak için ayırt edici bir özellik olarak kabul edilebilir. Bu yüzden bu noktaların ağırlık merkezlerinin bulunması gerekmektedir.

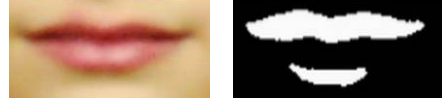
### 2.5.2. Renkler Arasındaki Bağlılıardan Yararlanarak Dudak Tespiti

Bu yaklaşımda görüntünün RGB renk uzayındaki R, G ve B bileşenleri arasında çeşitli bağıntılar yazarak bir pikselin dudağa ait olup olmadığı tespit edilmeye çalışılmıştır.

Dudak ve yüzdeki piksel dağılımları incelendiğinde Şekil 1.12'deki gibi bir dağılım elde edilmiştir. Bu dağılıma göre yüz piksellerinde R değerinin G değerinden ve G değerinin de B değerinden daha büyük olmasına rağmen dudakta sadece R değeri büyük olmakta, buna karşılık G ve B değerleri birbirine yakın olmaktadır. Çeşitli denemelerden sonra R, G ve B arasında (66) bağıntısı önerilmiş ve sonuçlar gösterilmiştir. Burada R kırmızı bileşeni, B mavi bileşeni ve G ise yeşil bileşeni oluşturmaktadır.

$$K = \text{kırmızı} - (\text{mavi} + \text{yeşil}) \quad (66)$$

$K < 0$  ise ilgili bölge yüz,  $K > 0$  ise ilgili bölge dudak olarak yorumlanmaktadır. Bu varsayımdan yola çıkarak, çalışmada Şekil 2.19'deki görüntüler elde edilmiştir.

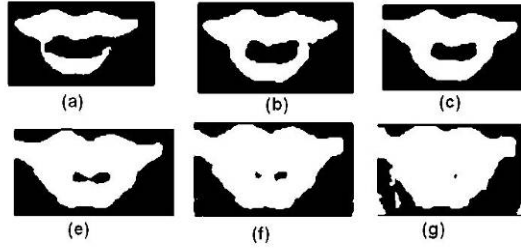


Şekil 2.19. Dudak resmi ve K faktörüne göre dudak şeklinin elde edilmesi

Şekil 2.19’de de görüldüğü gibi dudağın yapısının ortaya çıkmasına rağmen yine de eksiklikler mevcuttur. Örneğin alt dudağın sınırları belirgin olmamakta, buradan dudağın kapalı yapısı elde edilememektedir. Dolayısıyla bu varsayım üzerinde küçük bazı değişikliklerin yapılmasına gerek duyulmaktadır. Dudağın tam olarak ortaya çıkarılabilmesi için yukarıda elde edilen K ölçütünün negatifte kalan kısımlarının pozitifte çekilmesi gerekir. Bu nedenle piksellerin mavi parlaklık değerinin sıfırdan büyük uygun bir katsayı ile çarpılarak yeşil rengin değeriyle toplanması ve kırmızıdan çıkarılması önerilmiştir. Burada yeşil rengin parlaklık değerinin sabit tutulmasının nedeni genelde her pikseldeki yeşilin parlaklık değerinin maviden büyük olmasıdır. Düzenlenen yeni ifadeye göre K değeri,

$$K = [kırmızı - (yeşil + c * mavi)] \quad (67)$$

olmaktadır. Burada c, (0,1) aralığında değerler alan bir sabittir. Düzenlenen K ölçütüne göre elde edilen sonuçlar aşağıda gösterilmiştir.



Şekil 2.20. Farklı c değerlerine göre elde edilen dudak yapıları

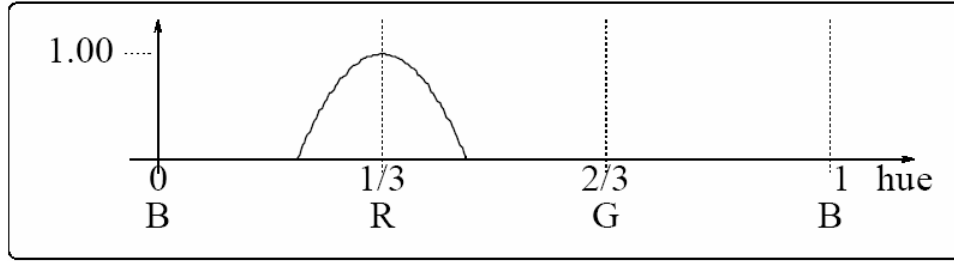
Şekil 2.20 (a)’da  $c = 0.8$ , (b)’de  $c = 0.7$ , (c)’de  $c = 0.6$ , (d)’de  $c = 0.45$ , (e)’de  $c = 0.4$  ve (f)’de  $c = 0.35$  olarak alınmıştır. “c” değerinin 0.45 değerinin altında alınması dudak şeklinin güçlü bir biçimde bozulmasına neden olmaktadır. Bu yüzden en uygun katsayı değerinin 0.45 ile 0.6 arasında olması gerektiği sonucuna varılmıştır.

Dudak ile ten arasındaki bir diğer ayırt edici ölçüt, renklerin birbirleriyle oranıdır. Özellikle yeşil/kırmızı renk oranı dudak yerinin tespitinde oldukça önemli bir etkidir.

Renk oranlarına göre özellik vektörleri çıkarılarak yüzün bölgelerinin yeri büyük ölçüde tespit edilebilmektedir.

### 2.5.3. HSI Renk Uzayı Kullanarak Dudak Tespiti

Bu bölümde HSI renk uzayına dayalı dudak tespiti üzerinde durulmuştur. İnsan dudağı incelendiğinde gri seviye renk bilgisinin çok fazla anlam ifade etmediği anlaşılmıştır. Örneğin aydınlatma ve diğer problemler dudak sınırlarının tespiti için dezavantaj olmaktadır. Oysa dudak üzerindeki renk bilgisi dudak için daha fazla anlam ifade etmektedir. Bu amaçla HSI renk uzayında renk bileşenini ifade eden Hue değerinden faydalanılmıştır. Çeşitli insan dudakları incelendiğinde dudak üzerinde kalan piksellerin Hue değerleri kırmızı etrafında yoğunlaşmaktadır. O halde bir pikselin dudağa ait olup olmadığını kırmızı değere yakınlığı ile anlaşılabilir. Şekil 2.21’de verilen grafik dudak bölgesine ait pikselleri filtrelemek için kullanılmıştır.



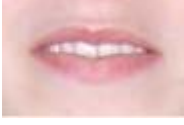







Şekil 2.21. Dudağa ait piksellerin belirlenmesi için kullanılan filtre

Şekil 2.21’de görülen filtreyi uygulamak için resim üzerinde (68)’deki fonksiyon uygulanır.

$$f(h) = \begin{cases} 1 - \frac{(h - h_0)^2}{w^2} & , |h - h_0| \leq w \\ 0 & , \text{diğer durumlarda} \end{cases} \quad (68)$$

Burada  $h$ , her bir pikselin hue değerini,  $h_0$  ise uygulanan filtrenin merkezini (Şekil 2.21’de görülmektedir),  $w$  ise uygulanan filtrenin genişliğini belirler. Şekil 2.21’teki filtre için  $h_0$  değeri  $1/3$  ve  $w$  değeri  $1/8$  olarak alınmıştır[16]. Aşağıdaki tabloda çeşitli resimler alınan  $h_0$  değerleri ve sonuçlar gösterilmiştir.

Tablo 2.1. Farklı  $h_0$  değerlerine göre elde edilen dudak tespiti sonuçları

Orijinal Resim	Alınan $H_0$ değeri	Sonuç
	1/3	
	1/4	
	1/5	
	1/3	

#### 2.5.4. Renk Bileşenlerinin Güçlendirilerek Dudağın Tespit Edilmesi

Dudak tespiti için önceki bölümlerde bahsedilen yaklaşımların her birinin kendine göre bazı dezavantajları olmaktadır. Ancak hiç biri tam olarak dudak tespiti gerçekleştirememektedir. Gerçekleştirirse dahi dudak okuma sistemine uyarlarken ortaya çeşitli problemler çıkıyordu. Yalnız dinamik eşilemeye dayalı yöntem ile renk bileşenleri arasındaki bağıntılardan yararlanan yöntem tek başına işe yaramamaktadır. Bu yüzden bu iki yöntemin karşımı bir yöntem takip edilmiştir. İzlenen işlem adımları şu şekildedir.

Bu yöntemde önce renk bileşenleri belli bir dönüşümden geçirilerek dudak üzerindeki pikseller güçlendirilir. Daha sonra güçlendirilmiş dudak üzerinde dinamik eşikleme (Otsu algoritması) uygulanarak ikili dudak elde edilmiş olur.

Renkli olarak verilen bir resim üzerinde dudağa ait pikselleri güçlendirmek için (69)'daki dönüşüm uygulanır[17].

$$I = \frac{2G - R - 0.5B}{4} \quad (69)$$

Burada R, G ve B değerleri [0,255] arasında değerler aldığıında I değeri ise [-96,128] arasında değer alır. I değerleri üzerinde dinamik eşikleme uygulayabilmek için bu aralık [0,255] arasına gelecek şekilde ayarlanır.

Yukarıda bahsedilen yöntem ile dudak tespiti algoritması aşağıda verilmiştir.

- Görüntüden (66)'daki formüle göre I değerlerini elde edilir.
- I değerlerini [0-255] arasına ötelenir
- Dinamik eşikleme ile görüntü ikili şekle çevrilir.
- İkili resimde oluşabilecek gürültüler dudak şekli yorumlaması gerçekleştirilerek elenir.
- Elde edilen ikili dudak resminden özellik vektörü oluşturulur.

Bir önceki paragrafta 1 nolu işlemin nasıl gerçekleştirildiği açıklanmıştı. 2-4 nolu adımlar ise bölüm 2.5'te ayrıntısıyla verilmiştir. Elde edilen ikili dudak resminden özellik vektörünün çıkarılması ise sonraki bölümde ayrıca anlatılacaktır.

## 2.6. Özellik Çıkarma

Otomatik dudak okuma sistemleri için en önemli işlem, tespit edilen dudağın özellik vektörünün çıkarılmasıdır. Çalışmada özellik vektörü için aşağıdaki özelliklere dayalı özellik vektörü kullanılmıştır.

- alt ve üst dudak konumları
- sağ ve sol dudak konumları
- ağız açıklığı
- alt ve üst dişlerin görünüp görünmediği
- dudak şeklinin alanı
- dudak şeklinin ağırlık merkezi
- dudak şeklinin en ve boyu
- dudak şekli üzerinde dudağın dış sınırlarını temsil eden 14 noktanın koordinatları

Bu yaklaşımda öncelikle dudağın ikili resmi elde edilmektedir. Daha sonra yukarıdaki özelliklerin her biri teker teker bulunarak bir özellik vektörü oluşturulmaktadır.

### 2.6.1. Alt ve Üst Dudak Koordinatlarının Bulunması

Elde edilen ikili dudak şeklinin en üstte kalan noktası üst dudak olarak ve en altta kalan koordinatı da alt dudak koordinatı olarak belirlenir. Bu işlem için sırasıyla en üst ve en alt noktadan başlayarak arama yapılması ve ilk bulunan noktaların alt dudak ve üst dudak koordinatı olarak alınması gerekmektedir. Alt ve üst dudakların dudak okuma işlemindeki anlamları ağız hareketinin takibini kolaylaştırmaktır. Ayrıca ağız açıklığının



bulunmasında da kullanılmaktadır.

### **2.6.2. Sağ ve Sol Dudak Koordinatlarının Bulunması**

Sağ ve sol dudak koordinatlarının bulunması alt ve üst dudak koordinatların bulunmasına benzemektedir. Tek fark arama işleminin sırasıyla sağ ve sol taraftan başlanarak yapılmasıdır. Sağ ve sol dudakların dudak okumada oynadığı faktör, ağzın konuşma esnasında gergin olup olmadığının karar verilmesi aşamasıdır.

### **2.6.3. Ağız Açıklığının Bulunması**

Ağız açıklığının bulunması işlemi, otomatik dudak okuma esnasında önemli olan bir diğer parametredir. Ağız açıklığının bulunması için öncelikle alt ve üst dudak kalınlıklarının bulunması gerekmektedir. Alt ve üst dudak kalınlıkları dudak okuma esnasında konuşmaya başlarken sakın dudak olacağı düşüncesiyle ilk çerçeveden elde edilmektedir. Bir başka deyişle otomatik dudak okuma sisteminde konuşmaya başlamadan önce ilk alınan dudağın sakın dudak olması beklenmektedir. Bu şekilde kişinin ilk dudak görüntüsünden dudağın şekli öğrenilmektedir.

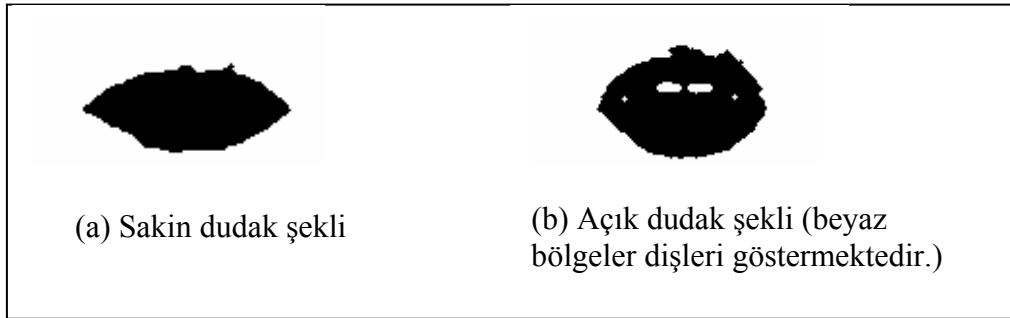
Dudak okuma esnasında ilk alınan çerçeve sakın dudak olacağı için ağız açıklığı sıfır (0) olarak alınmaktadır. Daha sonraki çerçeveler için ağız açıklığının bulunması işlemi dudağın boyundan(yüksekliğinden) alt ve üst dudak kalınlıklarının çıkarılması ile elde edilir. Ancak konuşma esnasında alt ve üst dudakların esnek olmasından dolayı alt ve üst dudakların kalınlıkları değişebilmektedir. Bu yüzden ağız açıklığı bulunurken önceden belirlenen bir eşik değeri kadar alt ve üst dudak kalınlıklarının değişebileceği dikkate alınmaktadır. Şekil 2.22’de ağız açıklığı ile ilgili yapılan işlemler verilmektedir.



Şekil 2.22. Alt ve üst dudak kalınlıkları

#### 2.6.4. Alt ve Üst Dişlerin Görünüp görünmediğinin İncelenmesi

Otomatik dudak okuma için önemli diğer bir parametre ise dişlerin görünüp görünmediği bilgisidir. Dudak tespiti esnasında elde edilen ikili dudak görüntüsü üzerinde dişlerin de belirgin olduğu Şekil 2.23'ten de görüldüğü gibi anlaşılmaktadır.



Şekil 2.23. Elde edilen ikili dudak görüntüsü

Bu ikili dudak şekli üzerinde birçok beyaz bölge bulunmaktadır. Amaç diş olma olasılığı en yüksek olan bölgenin tespit edilmesidir. Bu yüzden diş ile ilgili aşağıdaki işlemler yapılarak diş için en uygun bölge tespit edilir.

- İkili dudak bölgesi üzerinde beyaz bölgeler tespit edilir.
- Bu bölgeler ayrı ayrı değerlendirilerek en büyük alana sahip bölge tespit edilir.
- Bu bölgenin dudağın alt ya da üst tarafında olmasına göre alt ve üst dişlerin görünüp görünmediğine karar verilir.

### 2.6.5. Dudağın Alanının Bulunması

Dudağın alanının Bulunması işlemi dudak şekli üzerinde tüm siyah piksellerin sayılması ile gerçekleştirilmektedir. Bulunan sonuç piksel değeri olarak saklanır.

### 2.6.6. Dudağın Eninin ve Boyunun Bulunması

Dudağın eninin ve boyunun bulunması işlemi, alt üst sağ ve sol dudak koordinatları bulunduktan sonra (70) ve (71)'deki bağıntılarla hesaplanmaktadır.

$$en=sağ\_x-sol\_x \quad (70)$$

$$boy=alt\_y-üst\_y \quad (71)$$

### 2.6.7. Dudağın Dış Sınırlarının Bulunması

Otomatik Dudak okumada önemli olan diğer bir parametre dudağın dış sınırlarıdır. Bu sınırların hareketi dudak okuma için önemli olmakla birlikte aynı zamanda dudağın hareketlerinin izlenmesinde de önemli rol oynarlar.

Dudağın dış sınırlarının bulunması işlemi daha önce anlatılan kontur bulma işlemi sonucunda bulunmaktadır. Bu noktaların sayısı farklı dudak şekilleri için farklı sayıda olabilmektedir. Bu farklılığın dezavantajı sistemler arası uyumsuzluk olarak düşünülebilir. Bu dezavantajdan sakınmak için iki farklı yol izlenebilir. Bu yollardan birincisi elde edilen tüm dudak şekillerinin belli bir boyuta indirgenmesidir. Bu şekilde elde edilen dış sınır noktaları belir bir sayıda olacak ve standardizasyon sağlanacaktır. Ancak bu yöntemde bir çok nokta alınıp işleneceği için zaman kayıpları da artmaktadır. İkinci yöntem olarak ise dudağın sınırlarındaki tüm noktaların alınması yerine dudağı temsil edecek belli başlı noktaların alınmasıdır. Dudağı temsil eden noktalar için 14 nokta yöntemi önerilmektedir. Alınacak 14 nokta Şekil 2.24'te gösterilmektedir.



Şekil 2.24. Dudak üzerinde elde edilen 14 nokta

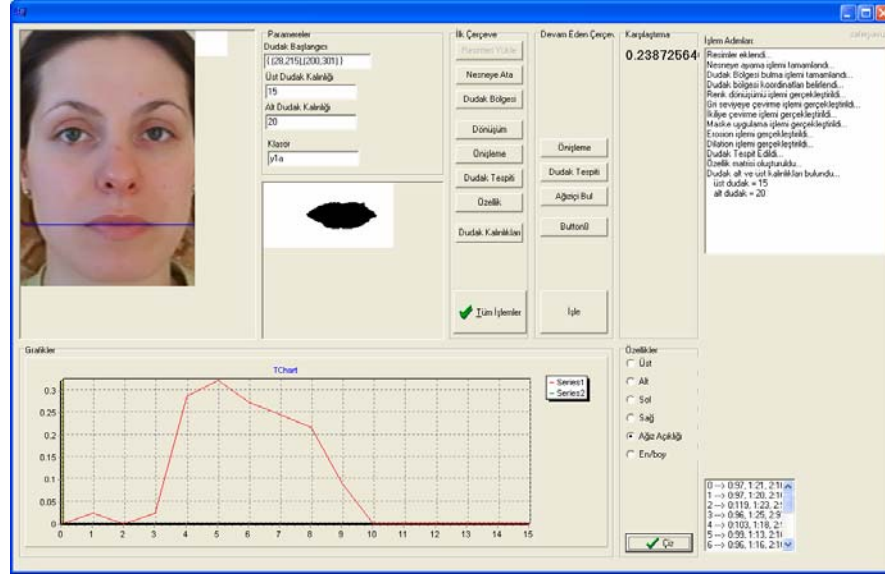
### 2.6.8. Oluşturulan Özellik Matrisi

Bilgisayarlı Dudak okuma için tasarlanan çalışma şeması giriş bölümünde verilmişti. Çalışma şemasında önemli aşamalardan biri özellik çıkarmadır. Özellik çıkarma için dikkate alınan hususlar şu şekildedir:

- Görüntüler ayrıık olarak alınmış hecelerden oluşmaktadır.
- Hecelerin her biri ortalama 60-120 çerçeveye sahiptir.
- Bu çerçeveler bir heceyi temsil etmektedirler.
- Heceleri oluşturan çerçeveleri 16 çerçeveye karşılık gelecek şekilde normalize edilmektedir.
- Her bir satır bir çerçevenin özellik vektörü olacak şekilde 16 satıra sahip bir özellik matrisi oluşturulur.
- Özellik matrisinin her bir satırı bir çerçeve için çıkarılan özellikleri, her bir sütunu ise ayrı ayrı çıkarılan özellikleri tutar.
- Alınan her bir hece için önce ilk çerçeve resmin özellik vektörü çıkarılır.
- İlk çerçeve resmin özelliklerinden dudağın sakin halinin özellikleri oluşturulmuş olur. (Heceler söylenirken ilk çerçeve resminin sakin olduğu kabul edilir.)
- Daha sonra gelen 15 çerçeve resim için de aynı özellik vektörleri çıkarılarak son

durumda o hecenin özellik matrisi oluşturulmuş olur.

Şekil 2.25'te özellik çıkarılması sonucu oluşan sistemin ekran görüntüsü verilmektedir.



Şekil 2.25. Sistemin örnek ekran görüntüsü

### 2.6.9. Özellik Matrisindeki Sütunların Anlamları

Bilgisayarlı dudak okuma işlemi için çıkarılan özellik vektöründeki sütunların anlamları Tablo 2.2'de verilmiştir.

Tablo 2.2. Özellik matrisindeki sütunların anlamları

Sütun indis numarası	Anlamı
0-1	Üst dudak koordinatları
2-3	Alt dudak koordinatları
4-5	Sol dudak koordinatları
6-7	Sağ dudak koordinatları
8	Ağız açıklığı
9	Üst dişlerin görünme oranı
10	Alt dişlerin görünme oranı
12	Dudak şeklinin eni
13	Dudak şeklinin boyu (yüksekliği)
16-43	Dudak sınırlarını temsil edecek 14 noktanın koordinatları (her bir nokta için 2 değer tutulur)
44-57	14 noktanın ağırlık merkezine olan uzaklığı (her bir nokta için bir değer tutuluyor)
58-100	Diğer parametreler

## 2.7. Dudak Bölgesi İzleme

### 2.7.1. Giriş

Çalışmada otomatik dudak okuma için, dudak bölgesinin takibi gerçekleştirilmiştir. Bu işlem nesne takibi algoritmaları ile gerçekleştirilmektedir. Her defasında yeniden dudak bölgesi bulma işleminin gerçekleştirilmemesi için ilk çerçeve resimde bulunan dudak bölgesi hareket izleme algoritmaları ile takip edilerek işlemlerin bundan sonraki kısımlarının sadece bu bölge ile sınırlı kalmaları sağlanmıştır. Böylece otomatik dudak okuma sisteminin performansının artırılması amaçlanmaktadır.

Çalışmada dudak bölgesinin izlenmesi için çekirdek tabanlı nesne izleme (Kernel Based Object Tracking) yöntemi uygulanmıştır[18].

Dudak bölgesi koordinatlarının elde edilmesi işlemi dudak bölgesinin bulunması ile gerçekleştirilmektedir. Dudak bölgesinin sol üst koordinatları izlenecek bölgenin koordinat başlangıcını vermektedir.

### 2.7.2. İzlenecek Dudak Bölgesinin Temsil Edilmesi

İzlenecek dudak bölgesi dikdörtgen bir bölge olduğundan bu bölgedeki pikseller için bir olasılık yoğunluk fonksiyonu belirlenmektedir. Olasılık yoğunluk fonksiyonunu belirlerken her bir parlaklıktaki pikselin olasılık yoğunluk fonksiyonunu belirlemek gerekir. Örneğin 0 parlaklıklı piksellerin bu bölgedeki olasılığının bulunması ya da 14 parlaklık değerli piksellerin olasılığının bulunması gibi. Bu şekilde 0 dan 255 e kadar tüm piksellerin olasılık değerleri bulunmaktadır. Bu değerler sonunda 256 elemanlı bir vektör oluşturulur. Bu vektör seçilen dikdörtgen alan içerisindeki dudak bölgesinin temsil eden değerlerden oluşur.

$$q_u = C \sum_{i=1}^n k(|x_i|^2) \delta(b(x_i) - u) \quad (72)$$

$$C = \frac{1}{\sum_{i=1}^n k(|x_i|^2)} \quad (73)$$

$i=1...n$  ve  $u=0...255$  olmak üzere

Burada  $q$  vektörü yukarıda bahsedilen ve piksellerin olasılıklarını ifade eden vektördür. Bu vektör izlenecek olan dudak bölgesini temsil eden vektör olarak kabul edilmektedir.

Yukarıda kullanılan  $k(x)$  ve  $\delta(x)$  fonksiyonları (74) ve (75)'de verilmiştir.

$$k(x) = \begin{cases} \frac{3}{4}(1-x^2) & , 0 \leq x \leq 1 \\ 0 & , \text{aksi durumlarda} \end{cases} \quad (74)$$

$$\delta(x) = \begin{cases} 1 & , x = 0 \\ 0 & , \text{aksi durumlarda} \end{cases} \quad (75)$$

olarak alınmıştır. Toplam sembolünde kullanılan  $n$  değeri izlenecek bölgedeki piksel sayısını ifade etmektedir. Uygulamada dudak bölgesi iki boyutlu olduğundan bu değer ( $n=en \times boy$ ) olarak düşünülmektedir.

### 2.7.3. Dudak Bölgesi Takibinin Gerçeklenmesi

Seçilen alan içerisindeki nesneyi  $q$  vektörü ile temsil ettikten sonra ilerleyen çerçevelerdeki bu görüntünün hangi yöne doğru hareket ettiğinin kestirilmesi gerekir. Bunun için bir sonraki çerçevedeki aynı bölge için de  $q$  vektörüne benzer bir  $p$  olasılık vektörü hesaplanmaktadır. Dolayısıyla  $q$  vektörü ilk seçilen dudak bölgesini,  $p$  vektörü ise

bir sonraki görüntüdeki dudak bölgesi adayını temsil edilmektedir. P vektörü de tıpkı q vektöründe olduğu gibi (69) ve (70)'deki formüllerden yararlanılarak hesaplanır.

P vektörünün hesaplanmasından sonra nesnenin hangi yöne doğru kaydığının bulunması gerekir. Bunun için hem yatay hem de düşey yönde kayma miktarının bulunması gerekir. (73)'deki bağıntı hem yatay yönde hem de düşey yönde merkezin yeni koordinatlarını vermektedir. Bağıntının genel mantığı merkezin en yüksek olasılıkla ne yöne kaydığının belirlenmesidir.

$$Y = \frac{\sum_{i=1}^n x_i w_i}{\sum_{i=1}^n w_i} \quad (76)$$

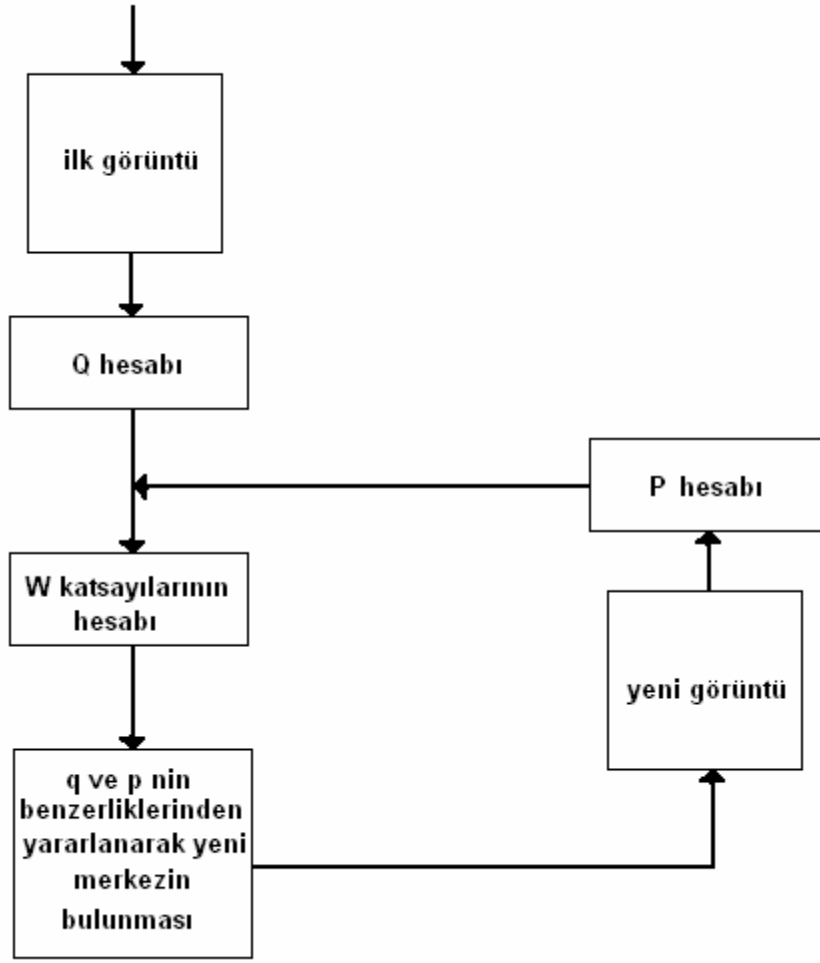
(76)'daki  $W_i$  ağırlık katsayısı ise (77)'deki bağıntı ile bulunmaktadır.

$$w_i = \sqrt{\frac{q_u}{p_u}} \delta(b(x_i) - u) \quad (77)$$

$u=0,1, \dots,255$  olmak üzere

Eğer merkezin yeni koordinatları ile eski koordinatları arasındaki uzaklık 1 birimi geçerse o zaman dudak bölgesi hareket etti yorumu yapılarak merkez koordinatları yenileri ile değiştirilir. Genel olarak kullanılan yöntemin blok şeması Şekil 2.26'da gösterilmiştir.

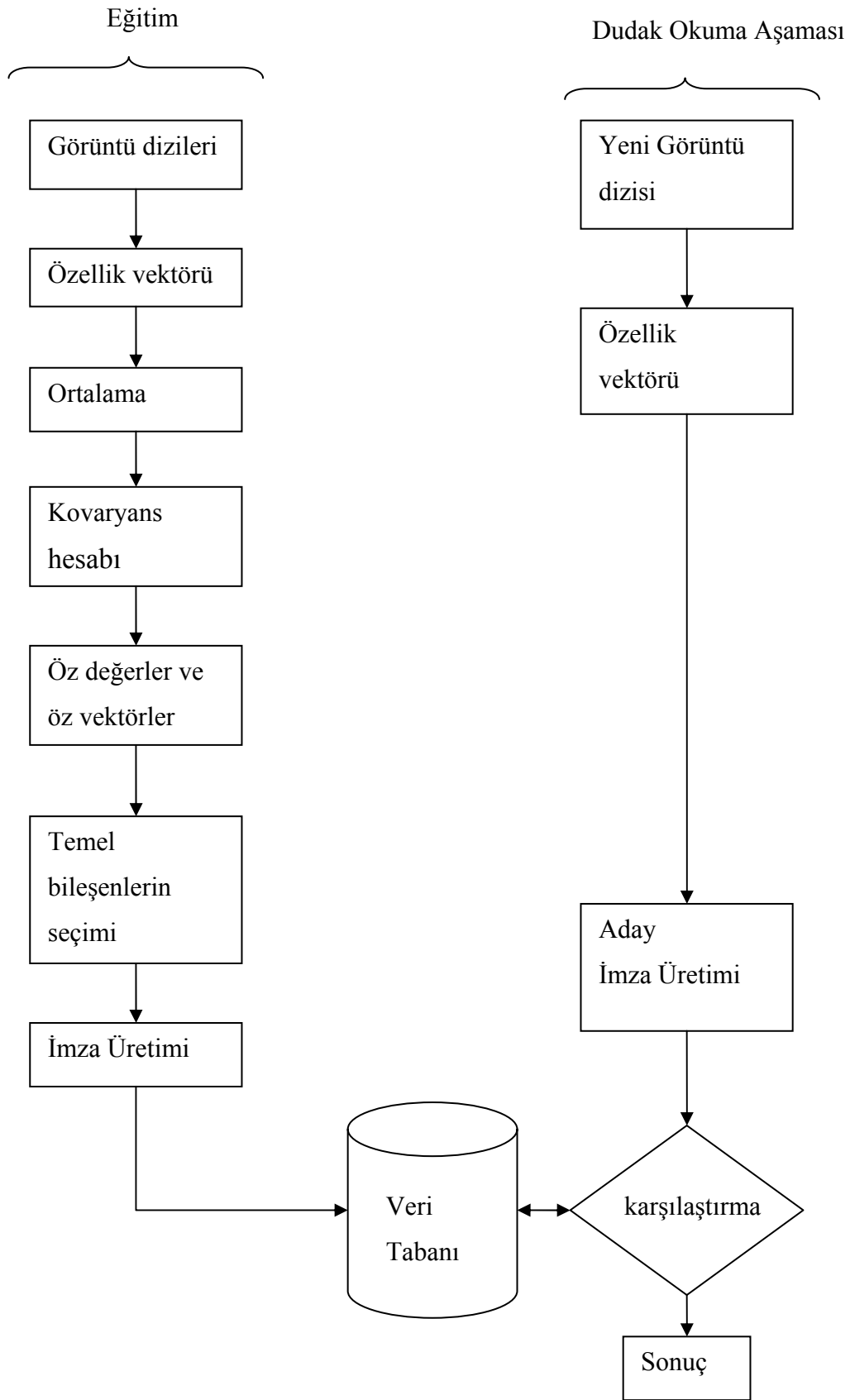




Şekil 2.26. Kullanılan algoritmanın blok şeması

## 2.8. Temel Bileşen Analizi (PCA) Algoritması

Temel bileşen analizi (PCA) algoritması ile, özellikleri çıkarılan dudak görüntülerinin eğitilmesi ve yeni gelen dudak görüntülerinin eğitim kümesi ile karşılaştırılarak tanınması gerekir. Örnek çalışma şeması Şekil 2.27’de verilmiştir.



Şekil 2.27. PCA yönteminin çalışma şeması

## 2.9. PCA'nın Bilgisayarlı Dudak Okumaya Uygulanışı

PCA yönteminin çalışma şeması yukarıdaki Şekil 2.27'de gösterilmiştir. PCA yöntemi iki aşamadan oluşmaktadır:

- Eğitim aşaması
- Dudak okuma aşaması

### 2.9.1. Eğitim Aşaması

Eğitim aşaması, eğitim için oluşturulan veri kümelerinin PCA yöntemi ile danışmansız olarak öğretilerek veri tabanına kaydedilmesi aşamasıdır. Bu aşamanın ayrıntıları aşağıda verilmiştir.

#### *Görüntü Dizileri*

Görüntü dizileri olarak dudak okuma sisteminin girişi olan ve daha önceden alınmış dudak görüntüleri düşünülmüştür. Bu görüntü dizileri  $k$  çerçeve resme indirgenmiştir. Yani her bir söylenen hecenin olduğu çerçeve sayısı ne olursa olsun özellik çıkarma için bu sayı  $k$  sayısına indirgenmiştir. Burada  $k=16$  olarak alınmıştır. Ayrıca sisteme sunulan görüntü dizileri ayrı olarak alınmış hecelerden oluşmaktadır. Yani görüntü dizisinde bir tane hece bulunduğu varsayılmaktadır.

#### *Özellik Vektörü*

Giriş bilgisi olan görüntü dizilerinden dudak okuma için kullanılacak özellik vektörü bu aşamada oluşturulmuştur. Özellik vektörü çıkarma işlemi daha önceki bölümlerde açıklanmıştır.

Bu aşamada özellik vektörünün PCA yöntemine uygulanması için özellik vektöründe yeniden düzenlemelere gidilmesi gerekir. Daha önce de anlatıldığı gibi dudak okumada eğitim için kullanılacak tek bir örneğin özellik vektörü  $n$  satır ve  $m$  sütundan oluşan bir matris idi. PCA yönteminde ise tüm örnekler tek bir matriste toplanacağından her satıra bir örnek karşılık gelmektedir. Bu durumda her bir örneğin özellik matrisi satır vektörüne dönüştürülerek bu şekilde PCA için kullanılacak eğitim matrisi oluşturulmuş olur. Bu durum (78)'de gösterilmektedir.

$$p.\text{örneğ.}\bar{o} \text{ özellik matrisi} = \begin{vmatrix} 1.\text{resmin özellik vektörü} \\ 2.\text{resmin özellik vektörü} \\ \dots \\ 16.\text{resmin özellik vektörü} \end{vmatrix} \quad (78)$$

$$PCA \text{ için kullanılan } k \text{ özellik matrisi} = \begin{vmatrix} 1.\text{örneğrn özellik matrisi} \\ 2.\text{örneğrn özellik matrisi} \\ \dots \\ l.\text{örneğrn özellik matrisi} \end{vmatrix} \quad (79)$$

(79)'da görüldüğü gibi PCA için kullanılacak özellik matrisinde her bir satır vektörü dudak okuma için kullanılan örneklere karşılık gelmektedir. Şekilde kullanılan  $l$  sayısı eğitimde kullanılacak örnek sayısını göstermektedir.

#### *Ortalama Alma*

Bir önceki adımda eğitim için kullanılacak veri kümeleri elde edilmiş oldu. Bu veri kümelerinin kendi aralarında ortalamaları alınarak işlemlere devam edilir. Ortalama alma işlemi birinci bölümde anlatılmıştır.

#### *Kovaryans Hesabı*

Ortalama sonucu oluşan matrisi  $A$  olarak adlandırırsak kovaryans matrisi (80)'de verilen formülle hesaplanır.

$$C = \frac{A^T \cdot A}{(n-1)} \quad (80)$$

Kovaryans hesabı ile ilgili gerekli değer hesaplamalar birinci bölümde verilmiştir.

#### *Öz Değer ve Öz Vektör Hesabı*

Bir önceki bölümde elde edilen kovaryans matrisin öz değer ve öz vektörleri hesaplanır. Bu işlem adımları da birinci bölümde verilmiştir. Öz değer ve öz vektörler elde edildikten sonra bu değer ve vektörlerin sıralanması işlemi gerçekleştirilir. Sıralama işlemi öz değerlere göre ve büyükten küçüğe doğru yapılır. Aynı şekilde öz vektörler de sıralama sırasında karşılıklı olarak yer değiştirir. Öz değerlerden en büyük değere sahip olan değer, eğitim kümesinin temel bileşeni olarak kabul edilir.

### *Temel Bileşenlerin Seçimi*

Öz değerler dizisi incelendiğinde en büyük elemanlar dizinin başlangıcında bulunur. Dizinin sonuna doğru olan öz değerler giderek küçülmekte hatta sıfıra yaklaşmaktadır. Dizinin sonunda olan küçük değerli öz değerler atılarak boyut azaltılır ve böylece daha az işlemle PCA yöntemi gerçekleştirilir. Atılan öz değerlere karşılık gelen öz vektörler de elenir. Böylece boyut azaltılmış olur.

Boyut azaltılması yöntemi özellikle resim sıkıştırma için kullanılan bir tekniktir. Küçük olan değerlerin atılması işlemi resimde kayıplı sıkıştırmaya karşılık gelmektedir. Ancak atılan değerlerin çok küçük olması resim sıkıştırma için çok fark edilmeyen bir durumdur. Otomatik dudak okuma sisteminde kullanılan özelliklerin sayısı PCA yöntemindeki boyuta karşılık geldiğinden boyut küçüktür. Bu durumda sıfıra yakın değerlerin elenmesi işlemi de isteğe bağlıdır.

### *İmza Üretimi*

Temel bileşenlerin elde edilmesi aşamasında küçük değerli öz değerlere karşılık gelen öz vektörler elenmiştir. Geriye kalan öz vektörlerden oluşan matris U transformasyon matrisi olarak adlandırılır. Orijinal değerlerden oluşan özellik matrisindeki her bir satır bir örneğe karşılık geliyordu. Bu satırdan oluşan vektör ile U transformasyon matrisi çarpılarak o örneğin imzası elde edilir. (81)'de imza üretimi gösterilmektedir.

$$S_k = I_k x U = \begin{vmatrix} i_{k1} & i_{k2} & \dots & i_{kl} \end{vmatrix} x \begin{vmatrix} u_{11} & u_{12} & \dots & u_{1n'} \\ u_{21} & u_{22} & \dots & u_{2n'} \\ \dots & & & \dots \\ u_{l1} & u_{l2} & \dots & u_{ln'} \end{vmatrix} = \begin{vmatrix} S_{k1} \\ S_{k2} \\ \dots \\ S_{kn'} \end{vmatrix} \quad (81)$$

Burada I, l boyutlu ve transformasyon matrisinin her bir satırından oluşan bir vektör, U ise  $lxn'$  boyutlu transformasyon matrisi ve S de son durumda elde edilecek imza vektörüdür. Burada l değeri örnek sayısını, n' değeri ise temel bileşenlerin seçimi aşamasında yapılan boyut azaltılması sonucu elde edilen yeni boyuttur. Burada  $k=1,2,\dots,l$  ye kadar değişen indis olarak ele alınmaktadır.

### *Üretilen İmzaların Veri Tabanına Kaydedilmesi*

Bu aşama otomatik dudak okuma için kullanılan PCA yönteminin son aşamasıdır. İmza üretimi kısmında eğitim kümesindeki her bir örnek için üretilen imzalar bu kısımda

tek tek veri tabanına kaydedilir. Ayrıca tanıma aşamasında kullanabilmek için kullanılacak olan U transformasyon matrisi de veri tabanına kaydedilerek eğitim aşaması sona erdirilir.

### 2.9.2. Dudak Okuma Aşaması

Bu aşamada daha önceden sisteme öğretilen hecelerin tanınması gerçekleştirilecektir. Bu işlem için öncelikle yeni gelen görüntü dizisinden özellik vektörü çıkarılır. Özellik vektörü çıkarma işlemi eğitim aşamasındaki özellik vektörü çıkarma işlemi ile aynıdır.

#### *Aday İmza Üretimi*

Çıkarılan özellik vektöründeki her bir satır vektörü, daha önce veri tabanına kaydedilen U transformasyon matrisi ile çarpılır ve aday imza üretilir. Aday imza ( $S'$ ) üretii (82)'de verilmiştir.

$$S' = I'xU = \begin{vmatrix} i'_1 & i'_2 & \dots & i'_l \end{vmatrix} \times \begin{vmatrix} u_{11} & u_{12} & \dots & u_{1n'} \\ u_{21} & u_{22} & \dots & u_{2n'} \\ \dots & \dots & \dots & \dots \\ u_{l1} & u_{l2} & \dots & u_{ln'} \end{vmatrix} = \begin{vmatrix} s'_1 \\ s'_2 \\ \dots \\ s'_{kn} \end{vmatrix} \quad (82)$$

#### *Karşılaştırma ve Sonuç*

Aday imzalar üretildikten sonra veri tabanındaki kayıtlı imzalarla tek tek karşılaştırılır. Karşılaştırma yapılırken en küçük kareler (RMSE) yöntemi izlenir. Karşılaştırma işlemi sonunda aday imzanın en küçük hatayı veren imza ile aynı olduğu kanaatine varılır. Karşılaştırma için kullanılan örnek kod aşağıdaki gibidir (C++).

```
for(int j=0;j<n;j++)
{
    x1=aday_imza[j]-imza[j];
    hata[i]+=x1*x1;
}
// hata dizisi küçükten büyüğe doğru sıralanır
for(int i=0;i<or_say;i++)
{
    for(int j=i;j<or_say;j++)
```

```
{
  if(hata[i]>hata[j])
  {
    stemp=hrf[i];
    hrf[i]=hrf[j];
    hrf[j]=stemp;

    dtemp=hata[i];
    hata[i]=hata[j];
    hata[j]=dtemp;
  }
}
```

### **3. SONUÇLAR VE ÖNERİLER**

#### **3.1. Yüz Tespiti ile İlgili Sonuçlar**

Otomatik dudak okuma için yüz tespiti işlemi bölüm 2.2’de detaylı olarak incelenmiştir. Bu bölümde yüz tespiti için incelenen yöntemlerden elde edilen sonuçlar tartışılmaktadır.

##### **3.1.1. Sabit Arka Plana Göre Yüz Tespiti**

Bu yöntemde dinamik eşikleme yöntemi kullanılmıştır. Görüntüde tek bir yüz olduğu kabul edilmiştir. Yüz tespiti başarımını etkileyen sebepler şunlardır:

- Arka planın sabit (koyu renkli) olup olmaması.
- Görüntüde konuşan kişinin giysilerinin ten rengine yakın olup olmaması.
- Düzgün olmayan ışık dağılımı ve bundan kaynaklanan gölgeler ve diğer aydınlatma problemleri

Tablo 3.1’de sabit arka plan göre elde edilen yüzler verilmiştir.



Tablo 3.1. Sabit arka plana göre elde edilen yüzler

Orijinal Görüntü	Bulunan Yüz	Başarımın Sebepleri
		Konuşanın elbisesi açık renkli olmasına rağmen çene altındaki gölge yüzün doğru şekilde tespit edilmesine sebep olmuştur.
		Konuşanın elbise rengi ve arka plan rengi koyu olduğu için yüz tespiti doğru bir şekilde gerçekleşmiştir.
		Konuşanın elbise rengi ve arka plan rengi koyu olduğu için yüz tespiti doğru bir şekilde gerçekleşmiştir.
		Konuşanın çene altında herhangi bir gölge olmadığından yüz tespitinde çene de yüz olarak algılanmıştır.









### 3.1.2. YCbCr Renk Uzayı ile Ten Renginin Belirlenmesi

Bu yöntemde işlemi bir pikselin ten rengi olup olmadığını belirlemektedir. Bu durumda piksel rengini etkileyen etkenler yüz tespiti işlemini de etkilemektedir. Bu etkenler:

- Yakın çekim sonucu piksellerin renginin değişmesi
- Aydınlatma problemlerinden ve gölgelerden dolayı ten üzerindeki piksellerin renginin değişmesi
- Konuşanın ten rengine yakın giysi giymesi
- Ten rengi tespitinde kullanılan sabitler oluşturulurken kullanılan eğitim kümesindeki ten renklerinin nasıl olduğu. (Örneğin eğitim kümesi ağırlıklı olarak açık ten rengine sahip insanlardan oluşursa yüz tespiti işlemi de açık ten renkli insanlarda daha başarılı olacaktır)

Tablo 3.2’de bu yöntemle elde edilen bazı sonuçlar verilmiştir.

Tablo 3.2. YCbCr renk uzayı kullanılarak yüz tespiti

Orijinal Görüntü	Bulunan Yüz	Başarımın Sebepleri
		Konuşan kişi eğitim kümesindeki örneklerle benzememektedir.
		Konuşanın giydiği elbise ten rengine benzemektedir.
		Ten rengi sorunsuz bir şekilde tespit edilmiştir.
		Ten rengi sorunsuz bir şekilde tespit edilmiştir.

### 3.2. Dudak Bölgesi Bulma

Dudak Bölgesi Bulma işlemi bölüm 2.4'te anlatılmıştı. Bahsedilen yöntemle dudak bölgesi bulma işlemini 199 erkek yüzü ve 427 bayan yüzü üzerinde denenmiş ve Tablo 3.3'teki sonuçlar elde edilmiştir.

Tablo 3.3. Dudak bölgesi tespitinde elde edilen başarı yüzdeleri

	Örnek Sayısı	Doğru Bulunan Dudak Bölgesi Sayısı	Başarı Yüzdesi
Erkek	199	184	92.46
Bayan	427	399	93.44
Toplam	626	583	93.13

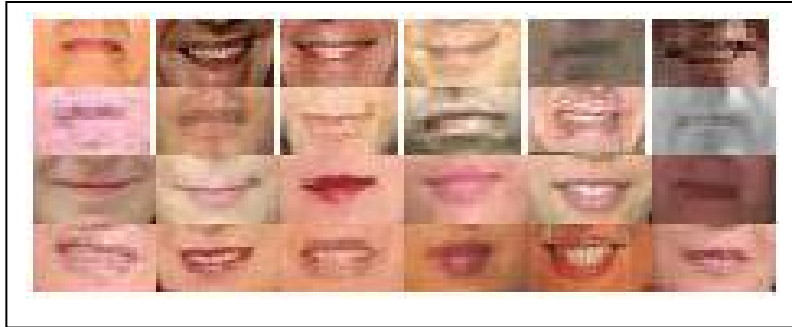
Dudak bölgesi tespiti ilk olarak gözün tespit edilmesine bağlıdır. Sonuçlarda elde edilen başarısız örneklerde gözün yanlış tespit edilmesi sonucu hatalı sonuçlar bulunmuştur ve bulunan dudak bölgesinde dudak resmi yarım ya da bir kısmı eksik çıkmıştır.

Gözün tespiti doğru olarak gerçekleştirildiğinde dudak bölgesinin tespiti için başarı %100 olmaktadır.

Şekil 3.1’de kullanılan yüz resimlerinden bazıları gösterilmiş ve Şekil 3.2’de tespit edilen dudaklar gösterilmiştir.



Şekil 3.1. Dudak bölgesi bulmak için kullanılan örnek yüzler

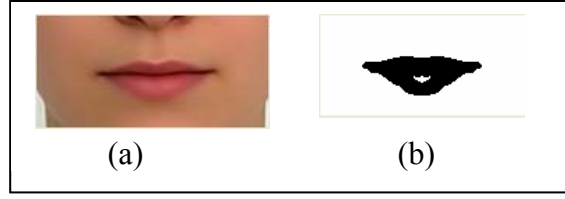


Şekil 3.2. Örnek yüzlerden elde edilen dudak bölgeleri

### 3.3. Dudak Tespiti Sonuçları

Dudak tespitinde önceki bölümlerde de anlatıldığı gibi 4 farklı yöntem kullanılmıştır. Bu yöntemlerin birbirlerine göre avantajları ve dezavantajları bulunmaktadır.

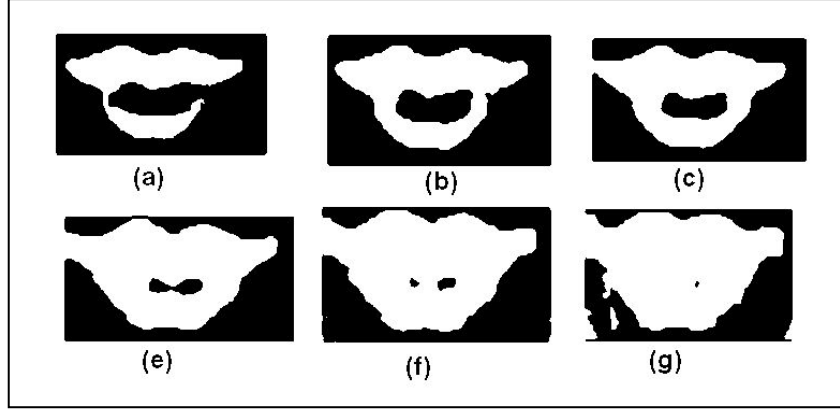
İlk yöntemde sadece dinamik eşiklemeyle dayalı bir dudak tespiti üzerinde durulmuştur. Dinamik eşikleme işleminden önce görüntü üzerinde bir takım ön işlemler yapılarak dinamik eşiklemenin sonucunda tespit edilen dudak görüntüsü iyileştirilmeye çalışılmıştır. Öte yandan ikili dudak resmi elde edilirken ilgili dudak bölgesinde dudağa ait olmayan bölgelerin de tespit edilmesi söz konusu olmaktadır. Bu bölgelerin elenmesinde kullanılan yaklaşım da yine önceki bölümde verilmiştir. Şekil 3.3'te birinci yönteme ilişkin tespit edilen dudak şekilleri görülmektedir.



Şekil 3.3. Dudak tespiti sonuçları

Sonuç olarak ilk yöntemin dudak tespitinde istenilen sonuçları vermesi her durumda mümkün olmamaktadır. Aydınlatma problemleri ve diğer faktörlerden dolayı bu yöntemin çalışması ve elde edilen sonuçlar değişmektedir. Bu yöntemin kullanılması ancak bir takım ön işlemlerden sonra mümkün olmaktadır.

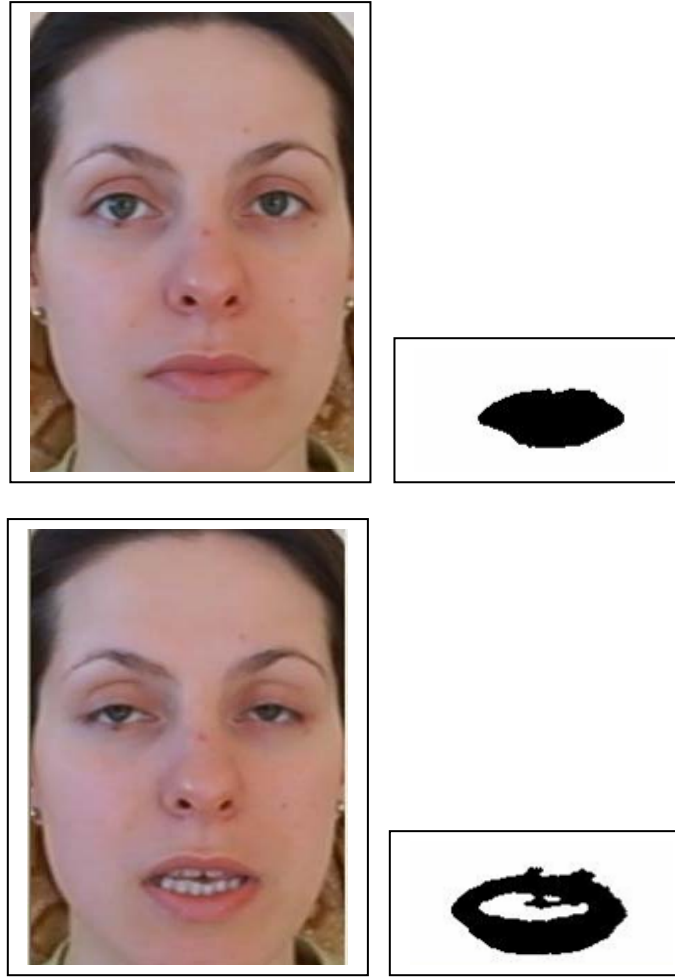
İkinci yöntemde ise renkler arasında bazı bağıntılar kullanılarak dudak tespit edilmektedir. İzlenen yöntem daha önceki bölümde anlatılmıştı. Bu formülün ve elde edilen sonuçların güvenilirliği resimden resme değişmektedir. Şekil 3.4'te farklı c katsayılarına göre elde edilen dudak şekilleri gösterilmektedir. Uygun c katsayısı ancak belli sayıda dudak resimlerinin eğitilerek belirlenmesi sonucunda elde edilebilir. Ancak bu işlem büyük veri tabanı ve çok miktarda işlem gerektirmektedir.



Şekil 3.4. Farklı katsayıların uygulanması ve elde edilen dudak şekilleri

Üçüncü yöntemde ise farklı bir renk uzayı seçilmiştir. Kullanılan renk uzayı HSI renk uzayıdır. Bu yöntemle elde edilen dudak şekilleri görüntünün yakın ya da uzak çekim olmasına göre değişmektedir. HSI renk uzayı ile dudak tespiti yönteminde elde edilen sonuçlar  $h_0$  ve  $w_0$  değerlerine bağlıdır. Çalışmada kullanılan değerler ortalama dudak için düşünülmüştür. Ayrıca her dudak şekli için kullanılacak  $h_0$  değerleri farklı olmaktadır. Ancak her durumda kullanılan bu değerler doğru sonuç vermemektedir. Her dudak için hangi  $h_0$  değerinin uygun olacağı ayrıca bir çalışma konusudur.

Dördüncü yöntemde kullanılan işlem adımları ilk yönteme benzemektedir. Farklı olarak dinamik eşiklemeden önce RGB renk değerlerinin dudak üzerindeki renklere göre güçlendirilmesi için yeni bir dönüşüm yapılmaktadır. Bu dönüşüm sonunda elde edilen dudak şekilleri Şekil 3.5'te verilmiştir.



Şekil 3.5. Örnek dudak tespiti sonuçları

### 3.4. Dudak Okuma Sonuçları

Bilgisayarlı Dudak Okuma işlemi için gerekli özellik vektörü ikinci bölümde elde edilmişti. Elde edilen özellik vektörleri kullanılarak sistem eğitilmiş ve dudak okuma işlemi gerçekleştirilmiştir.

Sistem için kullanılan eğitim ve test kümesi ayırık olarak çekilmiş görüntülerden oluşmaktadır. Bilgisayarlı Dudak Okuma sistemi için sesli harfler (a, e, ı, i, o, ö, u, ü) ve rakamlar (bir, iki, üç, dört, beş, altı, yedi, sekiz, dokuz, on) kullanılmıştır. Detaylar Tablo3.4'te verilmiştir.



Tablo 3.4. Sistemin detayları

Sesli Harfler (a, e, ı, i, o, ö, u, ü)	
Eğitim kümesi	24 tane (8X3)
Test kümesi	32 tane (karıştırılmış)
Rakamlar (bir, iki, üç, dört, beş, altı, yedi, sekiz, dokuz, on)	
Eğitim kümesi	20 tane (10X2)
Test Kümesi	25 tane (karıştırılmış)

Tablo 3.4'te verilen eğitim kümesi ile sistem eğitilmiş ve test kümesi ile test edilmiştir. Sesli harfler için elde edilen sonuçlar Tablo 3.5'te, rakamlar için elde edilen sonuçlar ise Tablo 3.6'da verilmiştir.

Tablo 3.5. Sesli harfler için dudak okuma sonuçları

Sesli harfler	Tanıma Sonuçları	
	1. en yakın	2. En yakın
A	I (%60)	İ (%20)
E	I (%80)	E (%20)
İ	İ (%80)	I (%20)
I	I (%80)	İ (%10)
O	U (%60)	I (%20)
Ö	I (%60)	Ö (%20)
U	Ü (%20)	I (%60)
Ü	U (%40)	Ö (%20)

Tablo 3.6. Rakamlar için dudak okuma sonuçları

Rakamlar	Tanıma Sonuçları	
	1. en yakın	2. en yakın
Bir	Yedi (%67)	Beş (%33)
İki	Sekiz (%50)	Yedi (%50)
Üç	Üç (%50)	Yedi (%50)
Dört	Dört (%50)	Yedi (%50)
Beş	Beş (%67)	Yedi (%33)
Altı	Altı (%50)	Yedi (%50)
Yedi	Yedi (%80)	İki (%20)
Sekiz	Yedi (%100)	-
Dokuz	Yedi (%67)	Dokuz (%33)
On	On (%80)	Yedi (%20)

Tablo 3.5'te görüldüğü gibi sesli harflerde I ve İ harflerinin büyük oranda tanındığı ve e ve ö harflerinin ise %20 oranında tanındığı sonucu elde edilmiştir. Buradan ortalama bir tanıma oranı hesaplanırsa sesli harfler için %25 oranında bir tanıma oranı elde edilmiş olur. Ayrıca (ı-i), (o-ö), (u-ü) seslilerinin benzerliği (görsel uzayda aynı olduğu) varsayılırsa bu durumda elde edilen tanıma %50 oranında olmaktadır.

Tablo 3.6'da verilen sonuçlar ise rakamlara aittir. Bu sonuçlara göre üç, dört, beş, altı, yedi, ve on rakamlarının büyük oranda tanındığı görülmüştür. Ortalama bir tanıma oranı hesaplandığında %59 civarında bir başarı elde edilmektedir. Rakamlar üzerinde elde edilen başarının daha fazla olmasının temel nedeninin hece sayısının fazla olduğu düşünülmektedir. Ayrıca sesli harf benzerliklerinden dolayı (iki, yedi, sekiz) rakamları ile (dört, dokuz, on) rakamlarının benzerliği düşünülürse %78 oranında bir başarı elde edilmektedir.

Yukarıda verilen sesli harfler ve rakamların bilgisayar tarafından otomatik tanıma oranları verilmiştir. Bu tanıma oranları insanlar üzerinde de test edilerek bir sonuç elde edilmiştir. 20 farklı kişiye 2 sesli harf (o, u) ve 2 rakam (on, dört) verilmiş ve bu görüntülerin ne söylediğinin bulunması istenmiştir. Kişilere sesli harfler için sesli harf olduğu ve rakamlar için de rakam olduğu söylenmiştir. Yani durum uzayı bilinmektedir.

Elde edilen sonuçlar şu şekildedir.

O seslisi için;

- %48 o, (15/32)
- %23 ö, (7/32)
- %23 u, (7/32)
- %6 diğer harfler, (3/32)

U seslisi için;

- %50 u, (12/24)
- %25 ü, (6/24)
- %25 diğer harfler, (6/24)

On sayısı için;

- %56 on, (17/30)
- %17 dokuz, (5/30)
- %3 dört, (1/30)
- %24 diğer, (7/30)

Dört sayısı için;

- %70 dört, (14/20)
- %15 dokuz, (3/20)
- %15 diğer, (3/20)

Görüldüğü gibi kişiler üzerinde yapılan testlerde dudak okuma için hiçbir ses bilgisi olmadan yalnızca görüntünün yorumlanmasında insan başarısı da % 100 olmamaktadır. Tablo 3.7’de insana göre bilgisayarın dudak okumadaki başarısı ve insan-bilgisayar karşılaştırması verilmektedir.

Tablo 3.7. Dudak okuma için insan-bilgisayar karşılaştırması

Hece	İnsan	Bilgisayar	Bilg./İnsan
O	48	12	51
U	50	30	60
On	56	50	91
Dört	70	25	35
	Ortalama		~%60

İnsan-bilgisayar karşılaştırmasında seçilen örnekler özellikle tanınma oranı düşük olan hecelerdir. Bu düşük tanıma oranına rağmen dudak okuma işleminde bilgisayarın insana göre ortalama % 60 civarında bir tanıma gerçekleştirildiği görülmüştür. Seçilen örnekler özellikle tanıma başarısı yüksek olan örnekler olduğunda bu durumda bilgisayarın dudak okumadaki performansı insan performansına yakın olacaktır.

### 3.5. Dudak Okumayı Etkilen Faktörler ve Öneriler

Bilgisayarlı dudak okuma işleminde dudak okumayı bazı faktörler etkilemektedir. Bunlar eğitim kümesindeki özelliklerin sırası, konuşan kişinin her defasında farklı şekilde konuşması, eğitim kümesindeki eleman sayısı (örnek sayısı) ve modelleme için kullanılan özelliklerdir. Bu faktörlerin etkisi minimuma indirgendikçe dudak okumadaki başarı daha da artacaktır.

Buraya kadar sesli harfler ve rakamlar için konuşan bağımlı (speaker dependent) bir dudak okuma sistemi geliştirilmiştir. Devam eden çalışmalarda sistem daha da geliştirilerek konuşan bağımsız (speaker independent) sistemler geliştirilebilir. Ayrıca tam bir dudak okuma işleminin gerçekleştirilmesi için sadece sesli harfler ya da rakamlar üzerinde değil Türkçede en fazla kullanılan hecelerin de sisteme öğretilip daha geniş çaplı bir dudak okuma sistemi geliştirilebilir. Bunun için eğitim ve test için daha fazla örneğe ihtiyaç doğacaktır.

Hece tabanlı bir dudak okuma sistemi daha da geliştirilip kelime tabanlı bir sistem geliştirilebilir. Böylece kelimelerin okunması ve kelimelerden de cümlelerin anlaşılması işlemi gerçekleştirilebilir. Tüm bu işlemler hiçbir ses bilgisine ihtiyaç duymadan gerçekleştirilmektedir. Geliştirilen bilgisayarlı dudak okuma sistemi ses bilgisi ile

birleřtirilerek daha etkili bir insan-bilgisayar etkileřimi gerekleřtirilebilir. Boyece sesin tanınmasında karřılařılan gurlt ve diđer problemlerin etkisi en aza indirgenmiř olur.

#### 4. KAYNAKLAR

1. Nabiyev, V., “Yapay Zeka-Problemler-Yöntemler-Algoritmalar”, Seçkin Kitabevi, 2. Baskı, Ankara, 2005.
2. Nabiyev, V. ve Yavuz, Z., 2005. “Vücut Dilinin Bilgisayarda Yorumlanması ve Bilgisayarlı Dudak Okuma”, 4. Uluslar arası İleri Teknolojiler Sempozyumu, sayfa 232-238, Eylül 2005, Konya.
3. Bregler, C. ve Konig, Y., “Eigenlips for Robust Speech Recognition”, Proceeding of IEEE International Conference on Acoustics, 1994.
4. Matthews, I., Cootes T., Bangham, J., Cox, S. ve Harvey, R., “Extraction of Visual Features for Lipreading”. Pattern Analysis and Machine Intelligence, IEEE Transaction, 2002, 198-213.
5. Petajan, E., “Automatic Lipreading to Enhance Speech Recognition”, Proc. Global Telecommunications Conference, Atlanta, 1984, 265-272.
6. Rogozan, A. ve Deléglise P., “Visible Speech Modelling and Hybrid Hidden Markov Models / Neural Networks Based Learning for Lipreading”. IEEE Computer Society, France 1998.
7. Xie, L. Cai X. Fu Z., Zhou R. ve Jiang D., “A Robust Hierarchical Lip Tracking Approach for Lipreading and Audio Visual Speech Recognition”, Proceedings of the Third International Conference on Machine Learning and Cybernetics, Shanghai, China, 2004, 3620-3624.
8. Wang S., Lau, H., Leng S. ve Yan H., “A Real Time Automatic Lipreading System”, ISCAS 2004, Sayı 2, Sayfa 101-104, 2004, Hong Kong.
9. Umbaugh, S. E., “Computer Vision and Image Processing”, Printice Hall PTR, June, USA, 1999.
10. Zhang, J., Wang, L., Niu D. ve Zhan Y., “Research and Implementation of a Real Time Approach to Lip Detection in Video Sequences”. Proceedings of the Third International Conference on Machine Learning and Cybernetics, China, 2003, 2795-2799.
11. Otsu, N., A Threshold Selection Method from Gray-Level Histograms. IEEE Transactions on System Man. and Cybernetics, 1979, 62-66.
12. <http://csnet.otago.ac.nz>, 19 Mart 2007.
13. Vezhnevets, V., Sazonov, V. ve Andreeva, A., “A Survey on Pixel-Based Skin Color Detection Techniques”. Proc. Graphicon, Moscow , Russia, 2003, 85-92.

14. Sandeep, K., ve Rajagopalan A., “Human Face Detection in Cluttered Color Images Using Skin Color and Edge Information”. Indian Institute of Technology, India, 2002.
15. Hsu, R., Abdel-Mottaleb, M. ve Jain, A.K., “Face Detection in Color Images”, Pattern Analysis and Machine Intelligence, 2002, 24(5), 696-706.
16. Coianiz, T., Torresani, L. ve Caprile, B., “2D Deformable Models for Visual Speech Analysis”, Proceedings of the 28. Asilomar Conference on Signals, Systems and Computers, 1996, 578-582.
17. Saeed, U., Matta, F. ve Dugelay, J., “Person Recognition based on Head and Mouth Dynamics”, IEEE International Workshop on Multimedia Signal Processing, Canada, 2006, 3-6.
18. Comaniciu, D., Ramesh V. ve Peter Meer, P., “Kernel-Based Object Tracking” Pattern Analysis and Machine Intelligence, 2003, 25(5), 564-577.
19. Sum, K., Lau W., Leung S., Alan W. Liew ve Tse K., “A New Optimization Procedure for Extracting the Point-Based Lip Contour Using Active Shape Model”, IEEE Computer Society, Hong Kong, 2001.
20. John, F., “İşaretler ve simgeler ansiklopedisi”, 1997/çeviri, Moskova.
21. “Yetişkinler İçin İşaret Dili Kılavuzu”, T. C. Milli Eğitim Bakanlığı Özel Eğitim Müd., Ankara, 1995.
22. Zaytseva, G.L., “Daktiloloji. Jest konuşması”, Moskova, 1991.
23. <http://www.deafblind.com>, 23 Ekim 2006.
24. Lindsay, I. S., “A Tutorial on Principal Components Analysis”, USA, 2002.
25. <http://homepages.inf.ed.ac.uk/rbf/HIPR2/morops.htm>, 12 Mart 2007.

## ÖZGEÇMİŞ

Zafer YAVUZ, 1981’de Trabzon’un Maçka ilçesinde doğdu. İlkokulu Trabzon 24 Şubat İlkokulunda, ortaokulu Trabzon Cumhuriyet Ortaokulunda ve liseyi Trabzon Fatih Lisesinin Yabancı Dil Ağırlıklı bölümünde tamamladı. 1999 yılında Karadeniz Teknik Üniversitesi Bilgisayar Mühendisliği Bölümüne başladı ve bu bölümden 2004 yılında bölüm birincisi ve fakülte ikincisi olarak mezun oldu. Aynı yıl Karadeniz Teknik Üniversitesi Bilgisayar Mühendisliği Bölümünde yüksek lisansa başladı. Halen bu bölümde uzman kadrosu ile görev yapmaktadır. İyi derecede İngilizce bilmektedir.