

İNSAN YÜZ İMGELERİNDEN YÜZ İFADESİ TANIMA

Müh. Mehmet ŞAKAR

Yüksek Lisans Tezi

Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Yrd. Doç. Dr. Ahmet ÇINAR

OCAK-2011

**T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

İNSAN YÜZ İMGELERİNDEN YÜZ İFADESİ TANIMA

YÜSEK LİSANS TEZİ

Müh. Mehmet ŞAKAR

(08129104)

Tezin Enstitüye Verildiği Tarih : 14 Ocak 2011

Tezin Savunulduğu Tarih : 07 Şubat 2011

Tez Danışmanı : Yrd. Doç. Dr. Ahmet ÇINAR (F.Ü)

Diğer Jüri Üyeleri : Doç.Dr. İbrahim TÜRKOĞLU (F.Ü)

Yrd.Doç.Dr. Galip AYDIN (F.Ü)

OCAK-2011

ÖNSÖZ

Yüksek Lisans çalışmalarım süresince değerli bilgi ve engin tecrübesiyle yardımlarını esirgemeyen danışman hocam Sayın Yrd.Doç.Dr. Ahmet ÇINAR'a sonsuz teşekkürlerimi, şükranlarımı ve saygılarımı sunarım.

Sürekli fikirlerinden yararlanma imkânı ve yardım bulduğum Bilgisayar Mühendisliği Anabilim Dalının değerli öğretim üyelerine teşekkür ve minnettarlığımı sunarım.

Mehmet ŞAKAR
ELAZIĞ – 2011

İÇİNDEKİLER

Sayfa No

ÖNSÖZ	I
İÇİNDEKİLER.....	II
ÖZET	IV
SUMMARY	V
ŞEKİLLER LİSTESİ	VI
TABLolar LİSTESİ	IX
KISALTMALAR.....	X
1. GİRİŞ.....	1
1.1. Yüz İfadelerin Oluşumu ve Etkisi	2
1.2. Yüz İfadeleri Analizi Sisteminin Kullanım Alanları	4
1.3. Tezin İçeriği.....	5
1.4. Tezin Amacı	6
2. YÜZ İFADE ANALİZİ.....	7
2.1. Yüz Belirleme.....	8
2.2. Yüz İfadelerine Ait Veri Çıkartılması	8
2.3. Yüz İfade Değişimlerin Analizleri	9
2.4. Literatürde Yapılmış Yüz İfadesi Analiz Çalışmaları	12
2.5. Literatürde Kullanılan Veritabanları	16
2.6. Literatürde Önerilen Sistemler	16
2.7. Biyometrik Sistemler.....	20
2.7.1. Biyometrik Sistem Çeşitleri.....	21
3. GABOR FİLTRELERİN YÜZ İFADE TANIMADA KULLANIMI	25
3.1. Gabor Filtresi.....	25
3.2. Özniteliklerin Çıkartılması	32
3.3. Öznitelik Boyutunun İndirgenmesi	32
3.4. Temel Bileşenler Analizi	33
3.4.1. Temel Bileşenler Analizi Algoritması	35
3.5. Özyüzler (Eigenfaces)	42
3.6. Laplacian(Laplasyen) – Kenar Bulma Yöntemi	44

3.7.	Marr-Hilderth Kenar Belirleme	45
3.8.	Difference of Gaussians (DoG)	49
3.9.	Adaptif Yaklaşım.....	53
3.9.1.	LMS (En Küçük Ortalama Kareler) Algoritması.....	55
3.9.2.	RLS (Özyinelemeli En Küçük Kareler) Algoritması.....	57
3.10.	Görüntü İyileştirme ve Geliştirme İşlemleri	59
3.11.	Sınır Eğrisi Çıkartmak	60
3.12.	Kullanılan Veritabanı.....	62
4.	UYGULAMALAR VE SONUÇLARI.....	65
4.1.	Yüz İfade Tanıma Sistemi	65
4.2.	Gabor Filtresi Uygulaması	68
4.3.	Laplacin of Gaussian (LoG) Uygulaması.....	69
4.4.	Difference of Gaussians (DoG) Uygulaması.....	71
4.5.	Karma Yöntemlerin Uygulaması.....	72
4.5.1.	Birinci Aşama Uygulama Sonuçları	73
4.5.2.	İkinci Aşama Uygulama Sonuçları	75
4.6.	Adaptif Yaklaşım Uygulaması	78
4.7.	Genel Sonuçlar Tablosu.....	81
5.	SONUÇLAR VE TARTIŞMA	82
	KAYNAKLAR.....	83
	EKLER	88
	ÖZGEÇMİŞ.....	93

ÖZET

Yüz ifadeleri, insanlar arası iletişimde anlam bütünlüğünün sağlanması için büyük rol oynayan, sözlü olmayan işaretlerdir. İnsanoğlu yüz ifadelerini kavramada herhangi bir zorluk çekmezken, bu durum makineler için geçerli olmayıp, halen güvenilir ifade tanıma sistemleri üzerinde çeşitli araştırmalar yapılmaktadır. Bu çalışmada, insanın içinde bulunabileceği 6 ifade durumunun (öfke, iğrenme, korku, mutluluk, üzüntü ve şaşkınlık) analizi gerçekleştirilmiştir. Bu amaçla, her bir ifade için alınan görüntülerin öznetelikleri, Gabor, LoG(Laplacian of Gaussians), DoG (Difference of Gaussians) ve geliştirilmiş adaptif Gabor-LoG(Laplacian of Gaussians) kullanılarak çıkartılmıştır. Temel Bileşenler Analiz yöntemi ile boyut indirgenip işlem yapılmış ve Öklid uzaklık hesabı yapılarak sınıflandırma işlemi yapılmıştır. Kullanılan yöntemlerin ifade sınıflandırılması üzerindeki etkileri incelenerek karşılaştırma yapılmıştır. Uygulama sonuçları %90 üzerinde bir doğruluk elde edilmiştir.

Anahtar Kelimeler: Yüz ifade analizi, Adaptif filtreler, LoG, DoG

SUMMARY

Facial Expression Reconition From Human Face Images

Facial expressions are non-verbal signs that play important role to provide complete meaning in human communication. While humans can easily comprehend the facial expressions, it is not valid for the computers, thus the researchers are still working on developing reliable facial expression recognition systems. In this research, the analysis of 6 different human facial expressions (anger, disgust, fear, happiness, sadness and surprise) is performed from human facial images. For this purpose, the features for every facial expression are extracted using the Gabor, LoG(Laplacian of Gaussians), DoG (Difference of Gaussians) and adaptive Laplacian of Gaussians (LoG) filter. Dimension has been reduced to by Principal Component Analysis method and the classification process is made by Euclidean distance calculation. A detailed comparison is made to analyze the effect of the methods and techniques used for expression classification. The application results have been obtained over 90% accuracy.

Key Words: facial expression analysis, adative filters, Laplacian of Gaussian, Difference of Gaussians

ŞEKİLLER LİSTESİ

Sayfa No

Şekil 1.1. Yüz ifadelerinin oluşmasına neden olan kaynaklar.....	2
Şekil 1.2. İfade şiddeti zaman grafiği.....	3
Şekil 2.1. Örüntü tanıma yöntemi.....	7
Şekil 2.2. 6 temel yüz ifadesinde yüz temel bileşenlerinin hareket yönleri.....	12
Şekil 2.3. Altı temel yüz ifadesi örnekleri: kızgınlık, şaşırma, üzüntü, tikslenme, Korku ve mutluluk.....	15
Şekil 2.4. Yüz Hareketleri Kodlama Sistemiyle tanımlanan Hareket Birimleri ve Bileşimleri örneği.....	15
Şekil 2.5. Biyometrik Sistemlerin Genel Çalışma Mekanizması	22
Şekil 3.1. Kompleks gabor fonksiyonunun reel (üst) kısımları.....	28
Şekil 3.2. Kompleks gabor fonksiyonunun sanal (alt) kısımları.....	29
Şekil 3.3. Gabor filtre bankası sistemi.....	29
Şekil 3.4. 5x8 adet gabor filtresi örneği.....	30
Şekil 3.5. 100x100 boyutundaki filtre örnekleri.....	31
Şekil 3.6. 8 dalga boyu (0,45,90, 135, 180, 225, 270, 315), yönelim değeri 0, faz ofset değeri 0,9 ve görüş açısı 0.5 için filtre çıkış örneği.....	31
Şekil 3.7. Ortalama değeri sıfır olarak elde etmek için yazılan (normalizasyon) Matlab kodu....	36
Şekil 3.8. Orijinal veriyi Kartezyen koordinat düzlemine döken kod.....	37
Şekil 3.9. TBA örneğindeki orijinal verinin Kartezyen koordinat sistemindeki izdüşümü.....	37
Şekil 3.10. Kovaryans matrisi hesaplayan kod.....	38
Şekil 3.11. Özvektör ve özdeğerleri hesaplayan kod.....	38
Şekil 3.12. TBA ile her iki özvektör kullanılarak elde edilen son veri kümesinin izdüşümü.....	40
Şekil 3.13. TBA ile boyut azaltıp orijinal veriyi geri elde ettikten sonraki verinin izdüşümü.....	41
Şekil 3.14. AT & T Laboratuvarlarından bazı özyüz imgeleri.....	43

Şekil 3.15. Bir yüze ait vektörün yüz uzayında şekilsel olarak gösterimi.....	43
Şekil 3.16. Laplacian of Gaussianın ilk aşamaları	46
Şekil 3.17. Laplacian of Gaussian filtre yapısı.....	46
Şekil 3.18. Laplacian of Gaussian filtre çekirdeği.....	47
Şekil 3.19. Log of Gaussian (LoG) filtresi.....	48
Şekil 3.20. Log of Gaussian (LoG) filtrenin örnek çıkış imgesi.....	48
Şekil 3.21. DoG filtresi.....	49
Şekil 3.22. DoG çalışma şeması.....	50
Şekil 3.23. DoG filtresinin çıkış imge örneği.....	50
Şekil 3.24. DoG filtre dalgacığı.....	51
Şekil 3.25. Meksika Şapkası dalgacığı.....	51
Şekil 3.26. DoGve Meksika Şapkası dalgacıkları.....	51
Şekil 3.27. LMS algoritmasının çıkış resimleri a) ikili seviyede, b) gri seviyede.....	53
Şekil 3.28. RLS algoritmasının çıkış resimleri a) ikili seviyede, b) gri seviyede.....	54
Şekil 3.29. Adaptif filtrenin blok şeması.....	55
Şekil 3.30. LMS algoritması akış şeması.....	56
Şekil 3.31. RLS algoritması akış şeması.....	58
Şekil 3.32. %20 oranında tuz-biber gürültülü gri seviyeli resim ($gR = \text{imnoise}(\text{resim}, 'salt \& pepper', .2);$).....	59
Şekil 3.33. 5x5 medyan filtre ile gürültüden arındırılmış ($mR = \text{medfilt2}(gR, [5 \ 5]);$).....	60
Şekil 3.34. 7x7 ortalama filtre ile gürültüden arındırılmış ($aR = \text{imfilter}(mR, \text{fspecial}('average', [7 \ 7]));$).....	60
Şekil 3.35. Çeşitli kenar tespit algoritmalarının çıkış resimleri.....	61
Şekil 3.36. JAFFE veritabanından örnek resimler.....	62
Şekil 3.37. Amatör veritabanından örnek resimler.....	63

Şekil 3.38. Kırpılmış resimlerin örneği.....	64
Şekil 4.1. JAFFE veritabanından örnek resimler.....	66
Şekil 4.2. Yüz ifade eksenini.....	66
Şekil 4.3. İfade tanıma uygulamasının akış şeması.....	67
Şekil 4.4. Gabor filtresi uygulaması için örnek giriş resmi.....	68
Şekil 4.5. LoG filtresi uygulaması için örnek giriş resmi.....	70
Şekil 4.6. DoG filtresi uygulaması için örnek giriş resmi.....	71
Şekil 4.7. Birinci aşama için örnek giriş resmi.....	73
Şekil 4.8. İkinci aşama için örnek giriş resmi.....	75
Şekil 4.9. Adaptif LoG uygulaması için örnek giriş resimleri.....	79
Şekil 4.10. Adaptif LoG uygulaması için örnek çıkış imgeleri.....	80

TABLÖLAR LİSTESİ

Sayfa No

Tablo 3.1. TBA’de kullanılacak örneğe ait (a) veri kümesi; (b) veri kümesinin ortalama değer hesabı yapıldıktan sonraki değerleri.....	37
Tablo 3.2. TBA ile iki özvektör kullanılarak dönüştürülmüş veri kümesi değerleri.....	40
Tablo 3.3. TBA’de tek özvektör kullanılarak dönüşümü alınmış veri.....	40
Tablo 4.1. Gabor filtresi uygulama sonuçları.....	69
Tablo 4.2. LoG uygulaması sonuçları.....	70
Tablo 4.3. DoG uygulaması sonuçları.....	72
Tablo 4.4. Birinci aşama uygulamasının ortalama başarı oranları.....	73
Tablo 4.5. Birinci aşama uygulamasının çıkış imge örneği.....	74
Tablo 4.6. Birinci aşama uygulamasının ortalama çalışma süreleri.....	75
Tablo 4.7. İkinci aşama uygulamasının ortalama başarı oranları.....	76
Tablo 4.8. İkinci aşama uygulamasının çıkış imge örneği.....	76
Tablo 4.9. İkinci aşama uygulamasının ortalama çalışma süreleri.....	77
Tablo 4.10. Adaptif yaklaşım uygulama sonuçları.....	80
Tablo 4.11. Tüm uygulamaların sonuçları.....	81

KISALTMALAR

TBA	: Temel Bileşenler Analizi
PCA	: Principal Component Analysis
LoG	: Lablacian of Gaussian
DoG	: Difference of Gaussians
LMS	: Least mean squares-En küçük ortalama kareler
RLS	: Recursive least squares- Özyinelemeli en küçük kareler
FIR	: Finite Impulse Response
IIR	: Infinite Impulse Response
SVM	: Support Vector Machine
JAFFE	: The Japanese Female Facial Expression
AU	: Hareket birimi (Action Unit)
YSA	: Yapay Sinir Ağları
MATLAB	: Matrix Laboratory

SEMBOLLER LİSTESİ

L2	: Benzerlik ölçütü (Euclidean Distance)
L1	: Benzerlik ölçütü (City-Block)
C	: Kovaryans Matrisi
ψ	: Ana Dalgacık
d	: Özdeğer
v	: Özvektör
G	: İzdusum matrisi
y_i	: TBA’da görüntü öznitelik vektörü
R	: TBA’da N görüntüden oluşan $N \times N$ ’lik veri matrisi
R_{ϵ}	: TBA’da merkezîleştirilmiş veri matrisi
$\mu(r)$	: Ortalama vektörü
r^i	: Görüntü vektörü
$\sigma(r)$	: Standart sapma vektörü
σ	: Standart sapma
μ	: Levenberg-Marquardt parametresi
W	: Gabor filtre maskesinin boyutu

1. GİRİŞ

Mimik, bir duygu ve düşüncenin kas, göz, ağız ve yüz hareketleriyle anlatılmasıdır. Mimikler günlük iletişimimizde önemli bir yer tutmaktadır. İnsanlar mimiklerini kullanarak birçok duygu ve düşüncesini karşılarındaki kişiye anlatabilir. Mimikler, ifadeleri kuvvetlendirici anlamlar da taşımaktadır. Yüz ifadelerini inceleyerek bir kişinin mutlu, üzgün, şaşırılmış gibi ruh hallerini tahmin etmek mümkündür.

İnsanlar arasında gerçekleştirilen yüz-yüze iletişim, gelişmiş insan-bilgisayar ara yüzlerinin tasarlanmasında ideal bir model teşkil etmektedir [1–3]. İnsan iletişiminin temel özelliği, iletişim kanallarının çok çeşitli olması ve yapılan iletişimin farklı şekillerde gerçekleşmesidir. Kanal, iletişimin yapıldığı ortamı belirtirken, iletişim şekli, dış dünyadan sinyallerin alınma biçimini ifade etmektedir. İnsanlardaki iletişim kanalları, konuşulan seslerin ve seslere ait vurgulamaların taşındığı işitsel kanal ile yüz ifadeleri ve beden hareketlerinin taşındığı görsel kanaldan oluşmaktadır. İnsanlardaki iletişim yapısı çok gelişmiş olup, herhangi bir iletişim kanalındaki problem, mesajın bir diğer iletişim kanalı tarafından iletilmesi ile telafi edilmektedir.

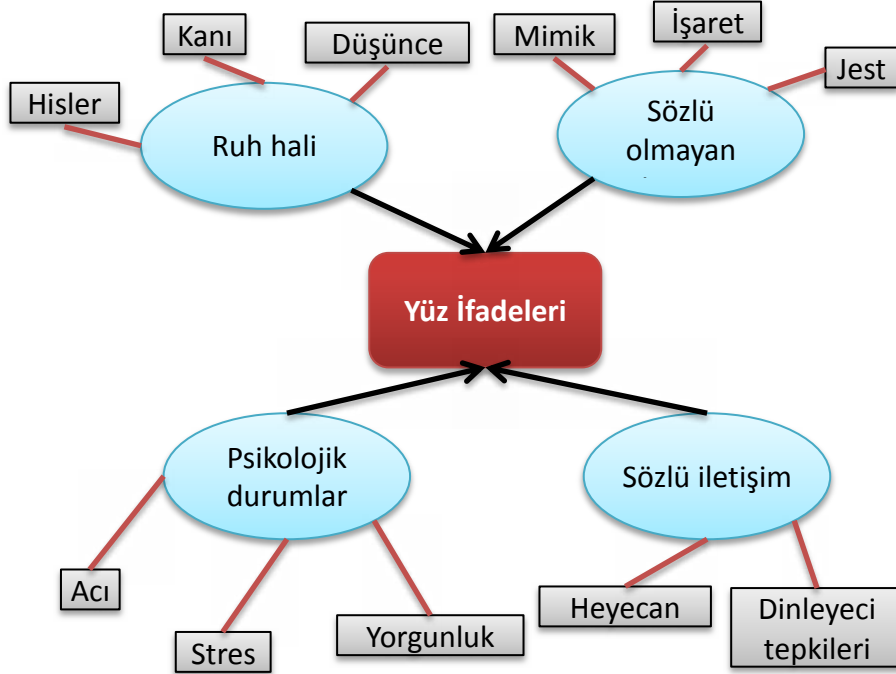
İnsanlar arasındaki iletişimde, yüz yüze iletişim çok büyük rol oynamaktadır. İnsan yüzü, diğer iletişim şekillerinin önemli bir göstergesi ve söz konusu göstergenin kaynağı olan herhangi bir yüz ifadesi de, bir insanın hissi durumunun, anlayış tavrının, karakterinin ve içinde bulunduğu psikolojik durumunun görünen belirtisidir. Yapılan birçok sosyo-psikolojik çalışma, karşılıklı yapılan konuşmalarda yüz ifadelerinin büyük etkisinin olduğunu göstermiştir [4,5]. Mehrabian [6], yüz ifadelerinin karşımızdaki dinleyiciler üzerinde çok büyük etkisinin olduğunu belirtmiştir ki, bu etkilerin yaklaşık %55'i konuşmacının açıklamalarındaki yüz ifadelerinden, %38'i konuşmadaki kelime tonlamalarından ve %7'si de konuşulan kelimelerden kaynaklanmaktadır. Oranlara dikkat edilirse yüz ifadelerinin iletişimde çok önemli bir paya sahip olduğu anlaşılmaktadır.

Bilgisayarların günlük hayatımızda yer edinmesi ile başlayan süreçte, insanları daha iyi anlayan ve insanlara daha iyi hizmet edecek bilgisayarlı sistemler oluşturmayı hedefleyen bilim adamları, insanlar arası iletişimi hedef model kabul etmişlerdir. Bu

doğrultuda yüz ifadeleri analizi ile ilgili çalışmalar 25 yılı aşkın bir süredir devam etmektedir. Bu çalışma ile de yüz ifadelerini analiz ederek kişinin içinde bulunduğu ruh halini tespit eden bir sistem oluşturulması hedeflenmiştir.

1.1. Yüz İfadelerin Oluşumu ve Etkisi

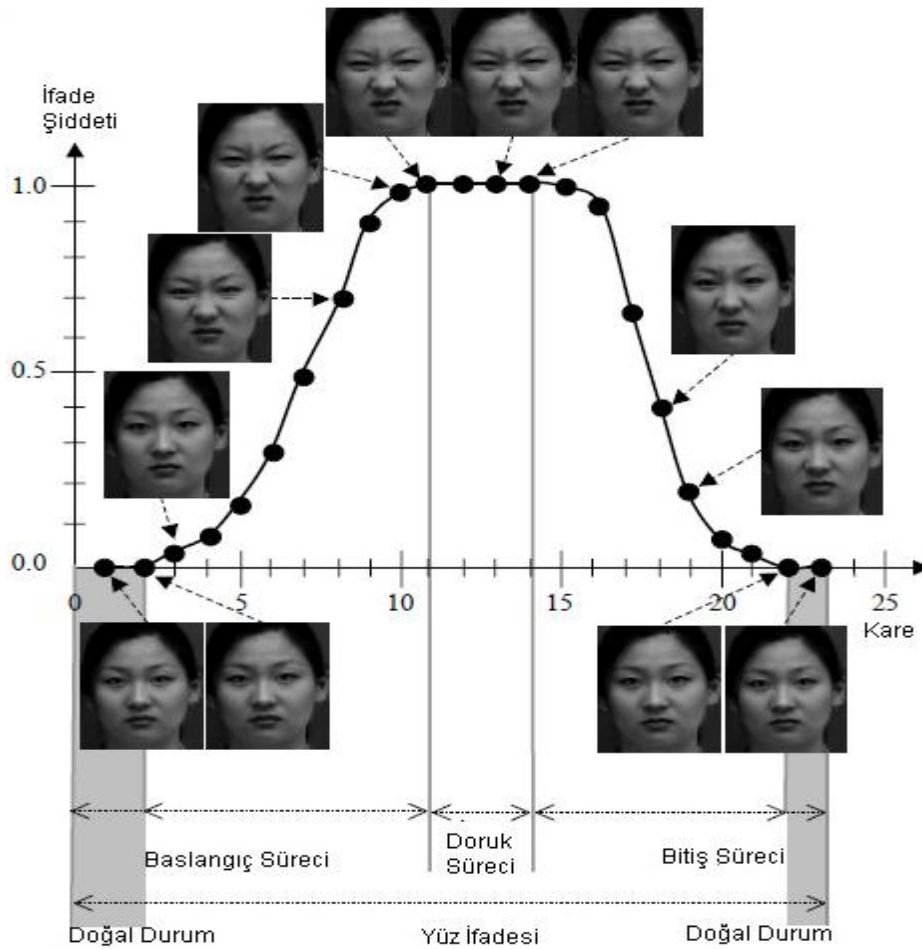
Yüz ifadeleri biçimsel olarak yüzdeki belirli bölgelerin şekil değiştirmesi ile ortaya çıkmaktadır. Özellikle yüz kaslarının kasılarak göz kapakları, kaşlar, burun, dudaklar ve yüz derisi gibi yüz özneliliklerinde geçici değişimler oluşturması sonucu ortaya çıkan, zamana bağımlı ve birkaç saniye süren bu değişimlerin yeri, yoğunluğu ve dinamiğinin ölçülmesi, yüz ifadelerinin tanınarak sınıflandırılması için önem taşıyan niteliklerdir. Bu değişimlerin kişiye, yaşa, ırka, cinsiyete göre miktar ve zaman olarak farklılık göstermesi, problemi çok boyutlu hale getirip zorlaştırmaktadır [7]. Şekil 1.1’de yüz ifadelerinin oluşmasına neden olan kaynaklar gösterilmiştir.



Şekil 1.1. Yüz ifadelerinin oluşmasına neden olan kaynaklar

Sabit görüntülerin ve görüntü dizilerinin kullanılmasıyla birlikte yüz ifade analizi, sadece psikoloji alanında yapılan bir araştırma konusu olmaktan çıkmıştır. Bununla birlikte görüntülerde yüzün tespiti, yüzü takip etme ve yüz tanıma alanındaki ilerlemeler, yüz ifade analizi üzerine yapılan çalışmaların artmasına sebep olmuştur [8].

Fizyolojik, zihinsel, sözlü ya da sözsüz iletişim kaynakları neticesinde yüz kaslarında oluşan hareket ile göz kapakları, kaslar, burun, dudak ve deri dokusu gibi yüz özniteliklerinde deformasyon meydana gelir. Kaslarda kasılma ve gevşeme çoğunlukla 1 saniye sürer, ancak birkaç saniyeden daha uzun süren değişimlerin yanında 250 milisaniyeden daha kısa süren değişimler de gözlenebilir. Şekil 1.2, ifade oluşumu esnasında zamanla ifade şiddetinin değişimini göstermektedir [9].



Şekil 1.2. İfade şiddeti zaman grafiği

Doğal(sakin) durum ile başlayan yüz kasları doruk noktasına ulaşana dek kasılır. Doruk noktası ifadenin tam olarak yüzde yerleştiği andır. Bitiş sürecinde ise yüz kasları tekrar ilk durumuna gelecek şekilde gevşer. Başlangıç-doruk noktası ve doruk noktası-bitiş süreleri ifadelere, kişilerin yas, cinsiyet gibi özelliklerine ve kültürlere göre farklılık göstermektedir. Her bir yüz özneliği farklı deformasyonlar ile farklı ifadelerin oluşmasını sağlar. Örneğin; kasların 40, göz kapaklarının ise 23 ayrı ifade şeklinin olduğu bilinmektedir [10]. Birbirleriyle kombine edildiği, farklı şiddet ve sürelerde olduğu düşünüldüğünde çok sayıda ifade seçeneği ortaya çıkar.

1.2.Yüz İfadeleri Analizi Sisteminin Kullanım Alanları

Özellikle bilgisayar bilimleri ve teknolojileri ile geliştirilen yüz ifadesi analiz sistemleri farklı kullanım alanlarının kapılarını aralamıştır. Bilgisayarların hayatımıza hızla yerleşmesi ile basit ve doğal bilgisayar-insan ara birimleri ihtiyacı hızla arttı. Yüz ifadesi analizi sistemi ile bireyin ihtiyaçlarını ve isteklerini algılayabilen bilgisayarlı sistemler popüler hale geldi. Örneğin klasik klavye, fare arabirimleri yerine kişilerin bakışlarını takip eden, göz, kas ve dudaklardan alınacak komutlar ile çalışan bilgisayarlı sistemler, normal kullanıcılara kolaylık sağlaması yanında engelli kullanıcılar için de güzel bir alternatif oluşturmaktadır. Yüz ifadelerinin duyguları ifade etmenin en doğal ve en başarılı yolu olması sebebiyle, başarılı bir bilgisayar-insan ara yüzünün önemli parçası yüz ifadeleri analizi sistemidir [11, 12].

İnsanlar ile uyumlu şekilde çalışacak insansı robotlar için de yüz ifade analizi vazgeçilmez adımlardan bir tanesidir. İnsan ile konuşan ve emirleri algılayan robot, yüz ile anlatılan ifadeleri de algılamalı ve bu doğrultuda cevaplar vermelidir. Video konferanslarda, ağ tabanlı eğitimlerde ve uzaktan bilgisayarlı özel ders seanslarında konuşmacının uzaktan dinleyicilerin reaksiyonlarını algılayabilmesi başarı oranını artırır. Konuşmacıya dinleyicilerin görüntülerini aktarmak gereksiz veri trafiğine neden olmanın yanında kişilerin özel hayatlarının görüntülenmesine de sebep olacaktır. Bunun yerine dinleyicilerin ifadelerini algılayan ve konuşmacıya ileten sistem sayesinde konuşmacı konferans ya da eğitimin gidişatını kontrol altında tutabilir.

Hava aracı ya da hava trafik kontrolü, uzay uçuşu ya da nükleer tesis gözetleme veya sadece kamyon, otobüs benzeri bir uzun yol aracı kullanımı gibi uzun süre konum değişikliği olmadan yapılan işlerde, dikkatin dağılmasına sebep olan can sıkıntısı, yorgunluk ve stresin otomatik tespiti bu anlamda çok değerli olacaktır. Zamanında fark edilen bu olumsuz durumlar sayesinde önemli kazaların önüne geçilecektir. Hatta otomatik ifade analizi yapan sistemler bir adım öteye gidip bu stres, yorgunluk, can sıkıntısı gibi durumlar meydana geldiğinde sistemlere müdahale edip, alarm verilmesini sağlayabilir.

Yüzden yayılan sinyalleri izleyen ve yorumlayan sistemler, avukatların, güvenlik birimlerinin, yargı organlarının işine yaramaktadır. Potansiyel suçluların korkma, tedirginlik ve benzeri ruh hallerini, yalan söyleyip söylemediklerini anlamalarını sağlamaktadır. Rahatsızlıklarını sözle ifade edemeyen hastalar için, hastanelerde otomatik uyarı sistemleri oluşturulabilir. Hastanın yüz ifadeleri takip edilerek acı çekip çekmediği ve sağlık durumu kontrol edilebilir.

Bir yüz ifadesinin sınıflandırılabilmesi için öncelikle görüntüdeki “yüz” kısmının belirlenmesi gerekmektedir. Görüntüler genellikle kişilerin ön yüzlerinden çekilmiş şekildedir. Buradaki en önemli problem, görüntüdeki yüz bölgesine ait kısmın doğru bir şekilde çıkarılmasıdır. Çünkü yüz bölgesinin konumu ve büyüklüğü her bir görüntü için farklılık gösterebildiğinden, görüntülerden yüz kısmının çıkarılmasında aynı türden bir şablon kullanılması pek mümkün olmamaktadır [13]. Bu nedenle yüz kısımları, çeşitli yüz belirleme teknikleri [14,15] veya el ile yapılan çalışmalar sonucu çıkarılmaktadır. Ancak şuana kadar yüzde yüz güvenilir bir yüz tespit tekniği geliştirilememiştir.

1.3. Tezin İçeriği

Bu çalışmada “Yüz İfade Analizi”, Ekman ve Friesen [16] tarafından tanımlanan 6 temel duyguya ait yüz ifadelerinin (“öfke”, “iğrenme”, “korku”, “mutluluk”, “üzüntü” ve “şaşkınlık”) sınıflandırılması çerçevesinde gerçekleştirilmiştir. Bu tezin ikinci bölümünde “Yüz İfade Analizi”nin genel olarak nasıl gerçekleştirildiği anlatılmış, analizin hangi bölümlerden oluştuğu ve bu bölümlere ait bilgiler verilmiş, geçmişten günümüze “Yüz İfade Analizi” üzerine literatürde yapılmış olan çalışmalar derlenmiş ve bu çalışmanın

amacından bahsedilmiştir. Üçüncü bölümde kullanılan metotlar hakkında bilgiler verilmiştir. Gabor filtresi, LoG(Laplacian of Gaussian) filtresi, DoG(Difference of Gaussians), adaptif LoG, yardımcı kenar tespit fonksiyonları, Temel Bileşenler Analizi(TBA) ve kullanılan veritabanı hakkında genel bilgiler sunulmuştur. Sonraki bölümde ise sözü edilen algoritmalar ve teknikler kullanılarak yüz ifadelerinin sınıflandırılmasına yönelik uygulamalar yapılmıştır. Tüm metotların sonuçları ile karma sistemin sonuçları yüz ifade analizlerin üzerindeki etkisi araştırılmış ve bulunan sonuçlar karşılaştırılmıştır.

1.4. Tezin Amacı

Bu çalışmada, 6 farklı yüz ifadesinin sınıflandırılması hedeflenmiştir. İnsanların herhangi bir zorluk çekmeden tanıyabildikleri bu ifadelerin makinelerce olabildiğince doğru ve hızlı bir şekilde sınıflandırılması ancak, hızlı ve güvenilir algoritmalar kullanılarak, makinelerin önceden doğru örneklerle eğitilmesi ile mümkün olacağı inkâr edilemez bir gerçektir.

Bu amaç doğrultusunda, yapılan çalışmada, literatürde sık rastlanılan birçok algoritma kullanılarak, hızlı ve doğru yüz ifade sınıflandırılması için kapsamlı bir inceleme gerçekleştirilmiştir. Çalışmanın temeli, farklı ifadelere ait yüz görüntülerine ilişkin özniteliklerin farklı Gabor dalgacıkları kullanılarak elde edilmesi üzerine kurulmuştur. En iyi ifade analizi sistemine sahip olan insan örneğinden yola çıkıldığında, insanlardaki görme merkezindeki hücrelerin tepki işaretlerinin Gabor dalgacık fonksiyonları ile modellenebilmesi, öznitelik çıkartılmasında Gabor dalgacıklarının kullanılmasının en önemli nedeni olmuştur.

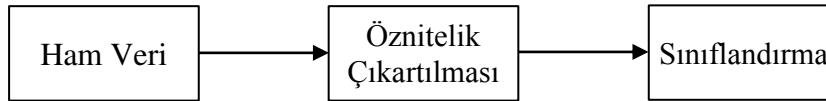
İfade sınıflandırması üzerindeki etkileri bakımından kullanılan algoritmalar ve teknikler arasında kapsamlı bir karşılaştırma yapılmıştır. Ayrıca, yüz ifadelerinin doğru bir şekilde sınıflandırılmasına yönelik çeşitli yaklaşımlar kullanılarak sınıflandırma başarımları arttırılmaya çalışılmıştır.

2. YÜZ İFADE ANALİZİ

İnsanlarda öğrenmenin nasıl gerçekleştiği psikologlar tarafından hala incelenmekte olan bir konudur. Genel olarak öğrenme; geçmiş deneyimler, örnekler ve merakla gelen bir işlemler bütünü olup, kesin olarak açıklanabilmesi pek mümkün değildir. Yüz tanıma örneğini ele alalım. Hemen her gün aile bireylerini ve arkadaşlarımızı yüzlerine bakarak veya fotoğraflarından, saç stillerindeki farklılıklara, makyaj yapıp yapmadıklarına ve duruşlarındaki değişikliklere rağmen hiç zorluk çekmeden tanıyabiliriz. Bunu bilinçsizce ve nasıl gerçekleştiğini açıklayamadan gerçekleştiririz.

Yapay Sinir Ağlarının (YSA) bir altkümesi olan “Makine Öğrenmesi”nin amaçlarından biri de, insanlardaki öğrenmenin nasıl gerçekleştiğini anlamamıza yardımcı olmaktır. Makine öğrenmesi tanımsal olarak, “bir problemin çözümünde örnekler veya geçmiş deneyimlere ait verilerin kullanılarak bilgisayarların programlanmasıdır” şeklinde ifade edilir.

Makine Öğrenme uygulamaları genellikle farklı nesnelerin sınıflandırılmasını amaçlar. Yüz tanıma, ses tanıma, karakter tanıma, hava durumu tahmini, yüz ifade analizi çalışmaları Makine Öğrenme uygulamalarından sadece bir kaçıdır. Farklı nesnelerin sınıflandırılması işlemi, bir başka deyişle Makine Öğrenme problemlerinin çözümü klasik olarak şekil 2.1’de gösterildiği gibi gerçekleştirilmektedir [17,18].



Şekil 2.1. Örüntü tanıma yöntemi

Bir makine öğrenme problemi olarak “Yüz İfade Analizi” karmaşık bir sisteme sahiptir. Cinsiyet, etnik yapı ve yaş farklılıklarının yanı sıra, yüzde olması muhtemel sakal, kozmetik ürünler, gözlükler ve saçlar nedeniyle kişiden kişiye değişen yüz ifadelerini tanıyan, genel ve geçerli bir sistem oluşturmak bu karmaşıklığın temel nedenidir. Ayrıca yüz görünümleri duruş ve ışıktaki değişimlere göre farklılıklar gösterebilir [19].

Pantic ve Rothkrantz [13] yüz ifade analizinde 3 temel problem bulunduğunu belirtmişlerdir: Bir görüntüdeki yüz kısmının belirlenmesi, yüz ifadelerine ait özniteliklerin çıkartılması ve yüz ifadelerinin sınıflandırılması. Pantic ve Rothkrantz [13]'ün işaret ettiği problemler ve makine öğrenme yaklaşımı çerçevesinde, yüz ifade analizi üzerine bir çalışma yapılması için öncelikle, resimlerdeki ifadelerin taşındığı kısım, yani yüz görüntüleri çıkarılır. Ardından yüz görüntülerinde bulunan ifadeleri temsil eden bilgiler (öznitelikler) çıkartılır. Çıkartılan öznitelikler, her bir ifade için sınıflandırılırlar. Ayrıca hızlı ve doğru bir analiz yapılması amacıyla, sınıflandırmadan hemen önce gereksiz öznitelikler atılarak öznitelik boyutu azaltılabilir.

2.1. Yüz Belirleme

Bir yüz ifadesinin sınıflandırılabilmesi için öncelikle görüntüdeki “yüz” kısmının belirlenmesi gerekmektedir. Görüntüler genellikle kişilerin ön yüzlerinden çekilmiş şekildedir. Buradaki en önemli problem, görüntüdeki yüz bölgesine ait kısmın doğru bir şekilde çıkarılmasıdır. Çünkü yüz bölgesinin konumu ve büyüklüğü her bir görüntü için farklılık gösterebildiğinden, görüntülerden yüz kısmının çıkarılmasında aynı türden bir şablon kullanılması pek mümkün olmamaktadır [13]. Bu nedenle yüz kısımları, çeşitli yüz belirleme teknikleri [14,15] veya el ile yapılan çalışmalar sonucu çıkarılmaktadır. Belirgin ve kesin bir yöntem olmadığından resimden yüz bilgisini çıkartma işleme el ile yapılırsa dahi kesin olarak yüzde yüz çıkartılması mümkün olmamaktadır. Sözelimi farklı boyutlarda iki resimdeki yüz resimleri çıkarttığımızda, çıkarılan yüzlerin boyutları da farklı olma ihtimali %99.9'dur ve boyut eşitlenirse dahi matematiksel olarak her bir yüz resmin matris değerleri boyut olarak farklı olacaktır. Bu çalışmada ise uygulamada kullanılan resim setleri yazılım aracılığıyla değil, manuel olarak el ile kırpılarak resimdeki yüz alanı çıkartılmaya çalışılmıştır. Yüz belirleme işlemleri ileriki bölümlerde açıklanmıştır.

2.2. Yüz İfadelerine Ait Veri Çıkartılması

Görüntülerde ifadeleri taşıyan yüz kısımlarının belirlenmesinden sonra her bir ifadeye ait veri, yani öznitelikler çıkartılır. Bu işlem görsel verinin sayısal veriye dönüştürülmesi şeklinde açıklanabilir. Yüz bileşenlerine (ağız, kaşlar, burun) ait olan

noktalardan elde edilen “Geometrik Öznitelikler” ile yüz bölgesindeki değişimlerden (kırıksıklık, izler) kaynaklanan “Görüntüye Dayalı Öznitelikler” olmak üzere 2 farklı tipte öznitelik bulunmaktadır [19].

Öznitelik çıkartılması aynı zamanda öznitelik uzayının boyutunun azaltılmasını da kapsamaktadır. Çünkü birçok örüntü tanıma sisteminde bulunan büyük boyutlardaki veriler, problem çözümünü neredeyse imkânsız hale getirmektedir. Boyut azalması ile verideki esas bilgi kaybolmamalı, bunun yanında boyutu azalan veri, orijinal veriye göre sınıf ayrımı çok daha fazla olan öznitelikleri içermelidir.

2.3. Yüz İfade Değişimlerin Analizleri

İnsan yüzünde ifade değişikliği iki farklı yaklaşım ile analiz edilebilir. Bunlardan birincisi, 5 (ağız, burun, göz, kaş ve çehre) temel yüz bileşeninin 6 farklı yüz ifadesinde gösterdiği şekil değişimi iken ikincisi 5 temel yüz bileşenin 6 farklı yüz ifadelerinde birbirlerine göre nasıl bir konum değişimi gösterdikleridir. Temel yüz bileşenlerinin farklı yüz ifadelerinde gösterdikleri şekil değişimleri aşağıdaki gibi ifade edilebilir.

a) Ağız

Ağız, bazı ifadelerde ilk şekline göre şiddetli bir değişiklik gösterirken bazı ifadelerde hafif değişiklikler gösterir. Mutluluk durumunda, sol ve sağdan iki yana açılarak geniş bir elips halini alır. Elipsin boyutu gülme şiddetine göre değişir. Gülümseme esnasında hafif bir aralık oluşurken, kahkaha atma durumunda geniş bir açılma yaşanır. Üzülme durumunda alt dudak yukarıya doğru hareket ederken, üst dudak kenarları aşağıya yönelir ve ağız hilale yakın bir şekil alır. Şaşırma esnasında alt dudak aşağıya doğru hareket eder ve ağız geniş bir elips oluşturur. Gülmeden farklı olarak sol ve sağ kenarlar açılmadığı için y-ekseni boyunca genişleyen bir elips oluşur. Kızgınlık durumunda sol ve sağ yandan dudak kenarları içe ve yukarıya doğru hareket eder, ağız genişliği azalır. Tiksinme durumunda ağız şeklinde çok şiddetli bir değişim yaşanmaz. Korku anında ise ağız, sol ve sağdan yanlara doğru açılırken sol alt ve sağ alttan da aşağıya doğru yönelir. Korku anında da şaşırma ve gülme durumlarındaki gibi ağız şekli genişler. Ancak genişlik şiddeti şaşırma ve gülmeye oranla daha azdır. Ayrıca gülme anında yukarıya yönelerek genişlerken, korku anında aşağıya doğru yönelme vardır. Mutluluk, şaşırma ve korku

esnasında ağız şekli genişlerken; kızgınlık, üzülmeye esnasında daralır. Tiksime de ise çok belirgin bir şekil değişimi gözlemlenmez.

b) Çehre

Sol kulak yanından çene alt noktası, sağ kulak yanı ve kaşlarla tamamlanan alan yüz çehresini oluşturur. Bazı ifadelerde şiddetli bir değişim yaşanırken bazılarında çehre yerine iç bileşenlerde değişimler yaşanır. Mutluluk aşamasında çehrede çok büyük bir değişiklik gözlemlenmez. Sol ve sağ dudak kıyılarından yanlara doğru hafif bir esneme görülür. Üzüntü ifadesinde kaşlar aşağıya alt çene ise yukarıya doğru hareket eder. Bu hareket neticesinde çehre ilk pozisyonuna göre küçülür. Şaşırma esnasında üzüntünün aksine kaşlar yukarı çene aşağıya doğru yönelir. Çehre y ekseninde boyunca genişleyerek elips şeklini alır. Kızgınlık ifadesinde de üzüntü ifadesine benzer şekilde kaşlar aşağıya doğru yönelir. Ayrıca kaşlar birbirlerine doğru da hareket ederler. Çene üzüntüye nazaran daha az hareketlidir. Çehre kaşların içe yönelmesi sebebiyle küçülür. Tiksime ifadesinde mutluluğa benzer şekilde yüzde şiddetli bir değişim gözlemlenmez. Kaşlar gözlere ve burun bitiş noktasına doğru yönelir ve çehrede hafif bir daralma görülür. Korku ifadesinde kaşlar yukarıya, dudaklar sol, sağ ve aşağıya yönelir. Ancak şaşırma ifadesindeki kadar şiddetli bir hareket yoktur. Çehre az miktarda genişler. Mutluluk ve tiksime esnasında çehrede belirgin bir değişiklik yaşanmazken; şaşırma ve korkuda genişler, üzüntü ve kızgınlık esnasında daralır.

c) Kaşlar

Kaşlar yapıları itibarıyla şekil olarak çok belirgin değişiklik gösteremezler. Daralıp genişleyerek az miktarda değişiklikler gösterirler. Mutluluk ifadesinde çok belirgin bir şekil değişimi görülmez. Üzüntü ifadesinde aşağı doğru yönelip göz kapaklarına yaslanarak ilk hallerine göre inceliyorlar. Kızgınlık ifadesinde kaşlar birbirine ve gözlere yaklaşarak ilk hallerine göre daralırlar. Tiksime esnasında buruna bakan kaş uçları birbirine yaklaşarak sıkışık bir hal alırlar. Şaşırma esnasında kaş yukarıya kalkarken esneme yaparak ilk haline göre genişler. Korku ifadesinde ise kaşlar gözlere yaklaşırken, buruna bakan kaş uçları yukarıya doğru yönelir.

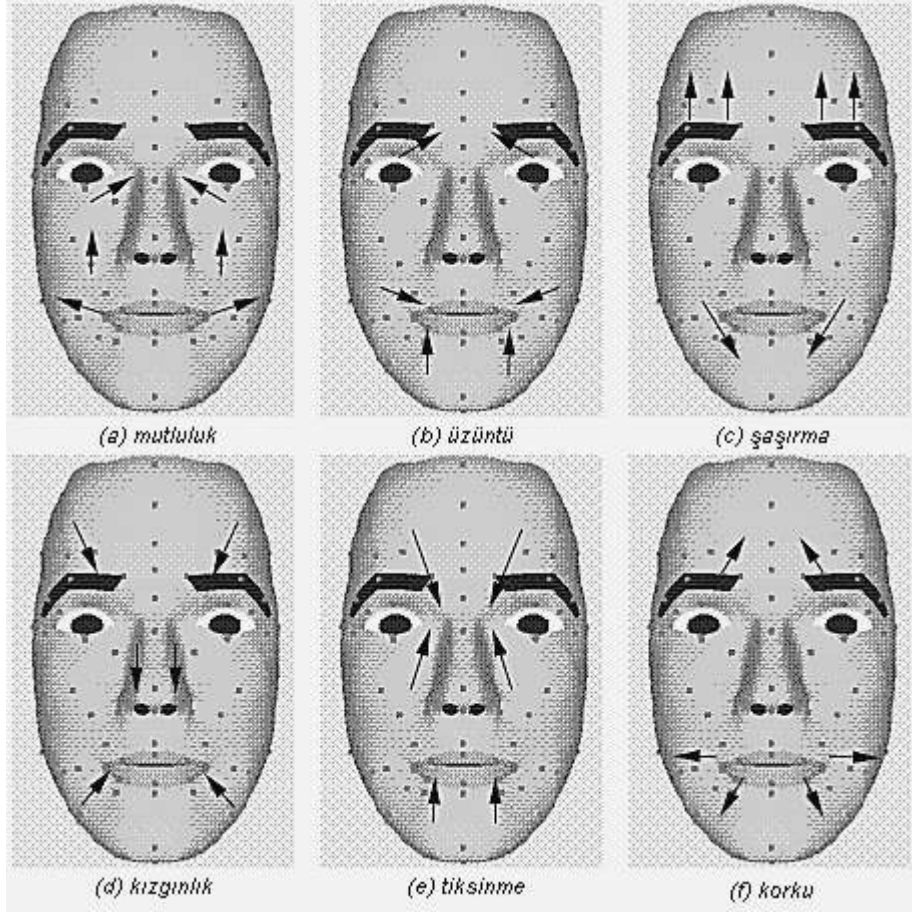
d) Gözler

Gözler de kaşlar gibi yapı itibarıyla şiddetli şekil farkları gösteremezler. Her ikisinde de değişim çok küçük boyutlardadır. Mutluluk esnasında göz belirgin bir değişiklik göstermez. Üzüntü ifadesinde göz y eksenini boyunca daralarak ilk haline göre küçülür. Kızgınlık aşamasında da üzüntüdekine benzer şekilde y eksenini boyunca daralır. Ancak üzüntü ifadesine göre daha şiddetli bir değişim vardır. Tiksiniye ifadesinde de y eksenini boyunca daralma görülür. Şaşırma ifadesinde göz açılarak büyür. Korku ifadesinde de göz açılır ve şekli genişler. Korku ifadesi şaşırma ile benzerlik göstermekle birlikte şiddetleri farklıdır.

e) Burun

5 temel yüz bileşenlerinden sonuncusu olan burun da 6 farklı yüz ifadesinde farklı şekil değişimleri gösterir. Mutluluk ifadesinde burun delikleri çevresindeki alt noktalarından sola ve sağ genişler. Üzüntü ifadesinde burun şeklinde belirgin bir farklılık gözlemlenmez. Kızgınlık ifadesinde burun deliklerinden yanlara doğru burun açılır ancak şiddeti mutluluk kadar değildir. Tiksiniye ifadesinde burun göz uçlarına doğru sıkışarak daralır. Şaşırma ifadesinde burun yukarıya doğru hareket ederken şeklinde belirgin bir farklılık gözlenmez. Korku ifadesinde de kızgınlık ve mutluluk ifadelerine benzer şekilde burun deliklerinden yanlara doğru genişleme görülür. Ancak şiddetleri birbirlerinden farklıdır.

İfade değişikliği analizinde kullanılan ikinci yaklaşım, 5 temel yüz bileşenin farklı yüz ifadelerinde birbirlerine göre nasıl bir konum değişimi gösterdiğidir. Yukarıda ifade edilen birinci yaklaşım içerisinde de görülebileceği gibi bazı yüz bileşenleri bazı ifadelerde birbirlerine benzer şekil değişikliği gözlemlenmezken bazı durumlarda belirgin farklılıklar görülmektedir. Bu durumun meydana getirdiği eksikliği gidermenin çözümü ise yüz bileşenlerinin birbirlerine göre gösterdikleri konum değişimlerinin incelenmesidir. Temel yüz bileşenlerinin farklı ifadelerdeki bağıl pozisyon değişiklikleri şekil 2.2’de bir arada gösterilmektedir [20].



Şekil 2.2. 6 temel yüz ifadesinde yüz temel bileşenlerinin hareket yönleri

2.4. Literatürde Yapılmış Yüz İfadesi Analiz Çalışmaları

Literatür incelemesi aşamasında literatürde yaygın olarak kullanılan işlem süreçleri ve yöntemler, veri tabanları ve önemli çalışmalardan birkaçı gözden geçirilecektir.

1872 yılında Darwin [21]'in insanlarda doğuştan gelen bazı duyguları olduğunu belirtmesi, “Yüz İfade Analizi” üzerine yapılan çalışmaların temelini oluşturmuş ve “Yüz İfade Analizi”, birçok davranış bilimci için popüler bir araştırma konusu olmuştur.

Darwin [21]'in yapmış olduğu çalışmadan yola çıkarak, 1971’ de Ekman ve Friesen [16] ve 1983’te Izard [22] insanlarda 6 temel duygu bulunduğunu ve bu duygusal ifadelerin her birine karşılık gelen yüz ifadelerinin olduğunu belirtmişlerdir. Farklı kültür

ve milletler için geçerli kabul edilmiş olan bu duygular “öfke”, “iğrenme”, “korku”, “mutluluk”, “üzüntü” ve “şaşkınlık” olarak gösterilmiştir.

Önceleri psikoloji alanında bir araştırma konusu olan yüz ifade analizi üzerine yapılan ilk çalışma, 1978’ de Suwa [23] tarafından bir görüntü dizisi üzerinde 20 nokta belirlenerek gerçekleştirilmiştir.

1980’lerde Marcelja [24] ve Daugman [25] tarafından insan görsel sisteminin 2-boyutlu karmaşık Gabor fonksiyonları ile modellenenebildiğinin gösterilmesi, örüntülere ait özniteliklerin çıkartılmasında önemli bir adım olmuştur.

Suwa [23]’nın 1978’de yapmış oldukları çalışmadan sonra, bilgisayar temelli yüz ifade analizi 1990’lara kadar yapılmamıştır. Mase [26] yüz ifadelerinin tanınmasında optik akış tekniğini kullanmıştır.

Black ve Yacoob [27] görüntüdeki hareketlerden yola çıkarak elde ettikleri verileri 6 temel yüz ifadesinin sınıflandırmasında kullanmışlardır.

Rosenblum, Yacoob ve Davis [28] yüz üzerindeki bölgelerde optik akış hesabı yapmışlar ve Radial Basis Fonksiyon Ağı kullanarak ifade sınıflandırmasını gerçekleştirmişlerdir.

Essa ve Pentland [29] bölgesel optik akış metodunu kullanarak ifadeleri sınıflandırmışlardır.

Otsuka [30] 6 farklı yüz ifadesini sınıflandırmak amacıyla Saklı Markov Model (Hidden Markov Model – HMM) temelli sınıflandırıcılar kullanmıştır.

Donato vd. [35] yüz ifadelerinin tanınmasında farklı tiplerdeki öznitelikleri karşılaştırmışlardır.

Martinez [31] yüz görüntülerinin, farklı aydınlık ortamlar, yüz ifadeleri ve görüntüyü kapayan engelleyici durumlar altında tanınması üzerine çalışmıştır. Bu amaçla, Saklı Markov Modelleri kullanılmıştır.

Oliver [32] ağız şeklinin değişiminden kaynaklanan öznitelikleri Saklı Markov Model temelli tanıma sisteminde kullanmışlar ve “ifadesizlik”, “mutluluk” ve “üzüntü” gibi ifadelerin sınıflandırması üzerine çalışmışlardır.

Kotsia ve Pitas [33] görüntü dizilerinden yüz ifade analizini Destek Vektör Makineleri kullanarak gerçekleştirmiştir. Çalışmalarında, dizideki ilk görüntü üzerine yerleştirilen noktalar referans alınıyor ve sonraki görüntülerle olan yer değiştirmeye göre ifade sınıflandırması yapıyordu.

Bartlett [34], Cohn ve Kanade veri setindeki yüz görüntülerinden farklı ifadeler için elde ettikleri Gabor özniteliklerini sınıflandırmak için yeni geliştirdikleri AdaSVM metodunu kullanmışlardır.

Yüz ifadesi analizi sistemlerinde giriş verisi önden ya da yandan alınmış imge ya da imge dizilerinden oluşabilir. Yüz kaslarındaki hareket neticesinde alında, kaslarda, gözlerde, ağız ve çene kaslarında deformasyon meydana gelir. Önden alınan görüntülerde bu deformasyon daha net görülürken, yandan alınan görüntülerde ayrıntılar kaçırılabilir. Literatürde yaygın olarak önden alınan görüntüler üzerinde çalışılmıştır. Mevcut, halka açık veritabanlarında önden alınan görüntüler, yandan alınanlara göre daha fazladır. Bu çalışmada önden alınan imge dizileri giriş verisi olarak kullanılmıştır.

Yüz ifadesi analizi sistemlerinde yorumlanan bilgi tipleri mesaj yorumlama ve işaret yorumlama olmak üzere iki gruba ayrılır:

Mesaj yorumlama yönteminde amaç yüzdeki ifade değişiminden anlam çıkarmaktır. Yüz kaslarındaki deformasyon toplu olarak ele alınır ve kişinin o anki duygusal ve zihinsel hali ile ilgili anlam çıkartılır. Araştırmacılar tarafından yaygın olarak Ekman [36] tarafından tanımlanan evrensel altı yüz ifadesi kullanılmaktadır. Şekil 2.3'te gösterilen

kızgınlık, şaşırma, üzüntü, tiksınme, korku, mutluluk ifadeleri evrensel kabul edilen ifadelerdir.



Şekil 2.3. Altı temel yüz ifadesi örnekleri: kızgınlık, şaşırma, üzüntü, tiksınme, korku, mutluluk

İşaret yorumlama yöntemindeki amaç ise yüzdeki kaslarının deformasyonunu tespit etmektir. Yüz üzerinde belirli kasların hareketleri incelenir, amaç hareketin olup olmadığını, yönü ve şiddetini tespit etmektir. Belirlenen hareketlerin gruplanarak yorumlanması ve anlam çıkarılması gerekmektedir. Ekman ve Freisen [37] tarafından oluşturulan ve Ekman ve arkadaşları [38] tarafından revize edilen Yüz Hareketleri Kodlama Sistemi (*Facial Action Coding System*), işaret yorumlama yöntemini kullanan araştırmacıların yaygın olarak kullandıkları yapıdır. Tanımlanan her bir değişim Hareket Birimi (*Action Unit, AU*) olarak adlandırılmıştır. Yüzde oluşan değişimler tek bir AU ile ifade edilebileceği gibi bileşimler halinde de ifade edilebilir [39].



Şekil 2.4. Yüz Hareketleri Kodlama Sistemiyle tanımlanan Hareket Birimleri ve bileşimleri örneği

2.5. Literatürde Kullanılan Veritabanları

Otomatik yüz ifadesi analizi sistemi geliştirebilmek için gerekli en temel ihtiyaçlardan birisi etiketlenmiş veridir. Doğal ifadeleri görüntülemek ve toplamak oldukça zor bir süreçtir, ayrıca görüntülense bile etiketleme zaman alan, hata olasılığı çok yüksek bir süreçtir. Literatürde gerçekleşen pek çok çalışma özel olarak hazırlanmış imge ve/veya imge dizisi veri tabanlarını kullanır. Cohn-Kanade yüz ifadeleri veri tabanı en yaygın olarak kullanılandır [40]. Bunun dışında Yin ve arkadaşları tarafından hazırlanan BU-3DFE veri tabanı 6 temel yüz ifadesi için 3D veri sunmaktadır [41]. Güneş ve Piccardi tarafından hazırlanan veri tabanı yüz ifadelerinin yanı sıra vücut hareketlerini de içerir [42]. MMI veri tabanı hem imge hem de imge dizisi görüntüleri içeren önden/yandan alınmış kayıtları ile geniş bir kaynaktır.

Cohn-Kanade veri tabanında yaşları 18-30 arası değişen yaklaşık 100 kişiden alınan görüntülerden oluşmaktadır. Veri tabanında yer alan insanların %65'i kadın; %15'i Afrikalı-Amerikalı, %3'ü Asya ya da Latin kökenlidir. Veri tabanı ile yaşanan en temel sıkıntılar, görüntülerin doruk noktasında sonlanması ve bazı görüntülerde sol altta tarih/saat bilgisinin yer almasıdır.

MMI veri tabanında yaşları 21-45 arasında değişen 18, 9-13 arasında değişen 11 kişi bulunmaktadır. Katılan insanlardan %48'i kadın; %66'sı Kafkasyalı, %30'u Asyalı, %4'ü ise Afrikalıdır. Veri tabanında 1395 oturum işaret yorumlama yöntemi için hareket birimleri ile etiketlenmiş veri bulurken, 197 oturum mesaj yorumlama yöntemi için 6 temel ifade ile etiketlenmiştir. Ayrıca MMI veri tabanında 65 adet doğal yüz ifadesi yer almaktadır. Halka açık veri tabanları içerisinde doğal görüntüsü olan tek veritabanıdır.

2.6. Literatürde Önerilen Sistemler

Yüz ifadesi analizi üzerine 20 yılı aşkın süredir çalışmalar yapan bilim adamları pek çok yaklaşım geliştirmişlerdir. Yüz edinim, yüz öznitelik çıkarma ve sınıflandırma adımlarının her birinde önerilen yöntemlere göre başarı oranının farklılık göstermektedir. Yüz öznitelik çıkarma aşamasında gerçekleştirilen çalışmaları seçilen yöntemlere göre geometrik özellik tabanlı [43-46], görünüm tabanlı [47-49] ve karma [50-53] olmak üzere üç grupta

toplayabiliriz. Ayrıca 3D model tabanlı yaklaşımlar kullanılarak başarılı sistemler geliştirilmiştir. [43, 46, 54]

Barlett ve arkadaşları geometrik tabanlı yöntemlerin görünüm tabanlı yöntemlerden daha başarılı olduğunu bildirmektedirler [39]. Tabi ki bu seçilen sınıflandırma yöntemi ve/veya öznelikleri giriş olarak hazırlayan otomatik öğrenme yaklaşımına göre değişir. Yakın tarihte önerilen çalışmalar geometrik yöntemlerin görünüm tabanlı yöntemlerden daha başarılı olduklarını gösteren örneklerdir [55,56]. Bununla birlikte 3D model tabanlı yaklaşımlar karmaşık yapıları ve yoğun hesaplama maliyetlerine karşın literatürde en iyi sonucu veren yöntemleri oluşturmaktadır. Yüz ifadesiyle ilgili yapılan çalışmaları inceleyen pek çok çalışma yapılmıştır [39, 57-59].

Pantic ve Rothkrantz [44], statik önden ve/veya yandan alınmış imgeler ile çalışmışlar ve geometrik özellik tabanlı bir yaklaşım izlemişlerdir. Yüzde belirlenen 19 nirengi noktası ile çalışmışlar, kural tabanlı bir yaklaşım ile 23 farklı AU ve farklı bileşimlerini %86 başarı oranı ile tanımışlardır.

Sebe ve arkadaşları [45], model-tabanlı takip yöntemiyle 6 temel yüz ifadesi tanıma hedeflemişlerdir. İlk karede göz ve ağız çevresi gibi öznelıklar elle işaretlenmiş ardından Bézier kıvrımları ile model çıkarılmıştır. Piecewise Bézier Volume Deformation (PBVD) algoritması kullanılarak her bir karede takip gerçekleştirmişlerdir. Bayes Ağı sınıflandırıcısı ve Karar ağacı yöntemlerini kullanarak gerçekleştirilen sınıflandırmada Cohn-Kanade veri tabanı ile 6 temel ifade için %91,3 oranında başarı elde etmişlerdir.

Pantic ve Patras [50], yandan çekilmiş görüntüler ile 27 farklı AU ve bileşenlerini tanımayı hedeflemişlerdir. Belirlenen 15 nirengi noktası parçacık süzgeci yöntemi ile (*particle filtering*) takip edilmiş, kural tabanlı yöntem kullanılarak sınıflandırılmıştır. MMI veri tabanı kullanılarak %87 oranında başarı elde edilmiştir.

Valstar ve Pantic [56], geometrik tabanlı yaklaşımla elle hiçbir müdahaleye gerek duymayan otomatik AU tanıma sistemi ve zaman analizi yapmayı hedeflemişlerdir. Gabor dalgacıklarını kullanan GentleBoost algoritması ile 20 nirengi nokta tespit etmişlerdir. Noktaların takibini *Particle Filtering with Factorized Likelihoods* algoritmasıyla

gerçekleştirmişler, sınıflandırma için SVM kullanmışlardır. Cohn- Kanade veri tabanı ile 15 farklı AU ve bileşimlerinin tespitinde %90.2 başarı elde etmişlerdir.

Black ve Yacoob [60], yüz temel bileşenlerinden ağız ve göz üzerinde parametrelendirilmiş hareket modelleri üzerinden takip gerçekleştirmişlerdir. Hareketi küçük bir grup parametre ile ifade ederek hareket değişimlerini kural tabanlı yöntemlerle incelemiş 6 temel yüz ifadesi için %88 oranında başarı elde etmişlerdir.

Yeasin ve arkadaşları [61], 6 temel yüz ifadesini tanımak için zaman-uzamsal bir yaklaşım geliştirmişlerdir. Hareket vektörleri ile iki aşamalı sınıflandırma gerçekleştirilmiştir. İlk aşamada lineer sınıflandırıcı ikinci aşamada ise Saklı Markov Model kullanmışlardır. Cohn-Kanade veri tabanı ile %90.9 oranında başarı elde etmişlerdir.

Anderson ve McOwan [48], 6 temel yüz ifadesini analiz eden gerçek zamanlı bir sistem oluşturmayı hedeflemişlerdir. Yüz bütün olarak uzamsal oran şablonu ile ifade edilmiş ve optik akış algoritması ile takip edilmiştir. Destek Vektör Makinesi (Support Vector Machine - SVM) sınıflandırıcısı kullanılmış, %81.82 oranında başarı elde edilmiştir.

Whitehill ve Omlin [62], Haar dönüşümü ve Adaboost hızlandırıcı sınıflandırıcı kullanarak AU tanıma sistemi oluşturmayı hedeflemişlerdir. Cohn-Kanade veri tabanı kullanılarak %91 oranında başarı elde etmişlerdir.

Guo ve Dyer [47], statik imge üzerinde elle 34 nirengi noktası belirlemişler ve bu noktaların çevresinde Gabor dalgacıklarını kullanarak öznelik çıkarmışlardır. Bayes Ağı, SVM ve AdaBoost sınıflandırıcıları ile %60-%90 oranında başarılar elde etmişlerdir.

Lyons [49], 2D Gabor dalgacıkları ile öznelik çıkarımı gerçekleştirmiş, sınıflandırmada da Lineer Fark Sınıflayıcısı (*Linear Discriminant Classifier*) kullanarak tek bir statik imgede 6 temel yüz ifadesi tanımayı hedeflemişlerdir. %75 başarı oranı elde etmişlerdir.

Tian ve arkadaşları [50], yüzde yer alan kalıcı özniteliklerin şekli (ağız, göz vb. şekil değişiklikleri) ve ifade değişimi ile oluşan geçici öznitelikleri (kırışıklıklar, esnemeler vb.) inceleyerek karma bir sistem oluşturmuşlardır. Geçici öznitelikleri Canny kenar bulma algoritması kullanarak tespit etmişlerdir. Öznitelik vektörleri yapay sinir ağları kullanılarak sınıflandırılmış, 16 farklı AU %93 başarı oranı ile tespit edilmiştir.

Zhang ve Ji [51], yüz geometrisini belirten 26 nirengi noktası ile geometrik değişimleri ve kırışıklıklar, esnemeler gibi geçici özellikler ile görünümle alakalı değişimleri göz önüne alarak karma yaklaşım önermişlerdir. Yüz takibi için, kızılötesi (Infrared, IR) aydınlatma kameraları kullanarak gözbebeği takibini referans olarak +/-30 derece kafa hareketinde çalışmışlardır. Göz bebeği takibinde Kalman Süzgeci ile birleştirilmiş Mean-Shift algoritması ile takip gerçekleştirilmiştir.

Görünüm tabanlı değişikliklerin analizinde Tian ve arkadaşlarının [50] çalışmalarına benzer bir yaklaşım izlemişlerdir. Dinamik Bayes Ağları kullanılarak sınıflandırma gerçekleştirmişlerdir. Byun [53], statik imge üzerinde yüz temel bileşenleri geometrik şekli ve bölümlenmiş yüz renk istatistiği kullanarak karma yaklaşım geliştirmişlerdir.

Cohen [52], zamansal ipuçları kullanarak imge dizileri üzerinde 6 temel ifade analizi gerçekleştirmeyi hedeflemişlerdir. Sebe ve arkadaşları [45] tarafından önerilen çalışmaya benzer bir çalışma gerçekleştirmişlerdir. İlk karede işaretlenen nirengi noktalar baz alınarak Bézier kıvrımları ile 3D model oluşturulmuştur. Model Piecewise Bézier Volume Deformation (PBVD) algoritması ile takip edilmiş, sınıflandırma için Saklı Markov Modeli kullanılmıştır. Kişi bağımsız testlerde %80 başarı elde edilirken, kişi bağımlı testlerde %88 başarı elde edilmiştir.

Gokturk [46], geometrik tabanlı bir yaklaşımla yüz üzerinde 19 nirengi noktası belirlemişler ve bu 19 nokta kullanarak 3D model oluşturmuşlardır. 3D model her karede takip edilerek şekil ve duruş değişimlerini incelemişler ve SVM kullanarak sınıflandırmışlardır. Doğal, ağız açık, ağız kapalı ve kaslar kalkık durumlarını tespit etmek için önerilen çalışmada %90 oranında başarı elde etmişlerdir.

Essa ve Pentland [54], geometri, fiziksel ve hareket temelli yüz yapısını oluşturan dinamik 3D modeller ile birleştirilmiş en uygun optik akış tahmini (*optimal estimation optical flow*) yöntemi ile yüz hareketlerini incelemişlerdir. Yüz ifade tahmini yerine yüz kaslarının hareketini incelemeyi hedeflemişler ve %98 oranında bir başarı elde etmişlerdir.

Kotsia ve Pitas [43], geometrik temelli bir yaklaşım kullanarak yüze yerleştirdikleri Candide piramitleri yerleştirmişlerdir. Geometrik deformasyonu incelemişler, SVM ile sınıflandırmışlardır. 6 temel yüz ifadesini %99 oranında bir başarı tanımışlardır.

2.7. Biyometrik Sistemler

Biyometrik sistemlerin basit halleri ile binlerce yıl önceden beri kullanıldığı bilinmektedir. Yakın zamanda ise araştırmacıların insanların fiziksel özellikleri ve karakteristiklerin suça eğilimleri ile bir ilgisinin olup olmadığını araştırmaları biyometri alanına ilgiyi arttırmıştır. Günümüzde biyometrik incelemelerin boyutu, çeşitliliği ve kullanım alanları artmıştır. Bu sayede de pek çok yeni biyometrik kimlik doğrulama sistemi yerini almıştır.

Biyometrik sistemlerin uygulama alanları günümüzde oldukça çeşitlidir. Özellikle havaalanları giriş ve çıkış işlemleri, kredi kartı uygulamaları, kriminal amaçlı teşhis ve tespit uygulamaları, sigorta şirketleri, ağ ve veri güvenliği, sosyal güvenlik, vergi süreçleri gibi kamu hizmetleri, e-ticaret, elektronik imza uygulamaları, internet bankacılığı, ATM'ler, çağrı merkezleri, personel takibi, hasta takibi bu gibi sosyal sistemlerde kullanılmalarının yanında artık, bilgisayarlar, pda olarak adlandırılan el bilgisayarları, cep telefonları ve ev kilit sistemlerinde de kullanılmaktadırlar [91]. Örneğin parmak izi, iris veya yüz tanıma sistemini barındıran bir bilgisayar, kimliğini doğrulayamayan kullanıcıların bilgileri açmasına ve işlem yapmasına izin vermemektedir.

Biyometri uygulayıcılarının genel amacı kişilerin kimliklerini doğrulayabilmeleri için, akıllarında tutmaları gereken herhangi bir bilgi ya da yanlarında taşımak, kaybetmemek ya da unutmamak zorunda oldukları kart, anahtar gibi araçların yerine; kopyalanması ya da taklit edilmesi imkansız olan özelliklerini kullanmalarını sağlamaktır.

Biyometrik sistemlerde, kimlik belirleme işlemi, kişilerin fiziksel ya da davranışsal özelliğine dayanarak gerçekleştirildiği için başkasına devredilmesi, unutulması ya da kaybedilmesi durumu söz konusu değildir. Diğer yöntemlere göre çok daha az riske sahiptir. Ancak biyometrik sistemlerin oluşturulabilmesi için bazı standart ölçüler kullanılmalıdır. Biyometrik ölçüler olarak adlandırılan bu ölçülerin şifrelerde kullanımı için INCITS (International Committee for Information Technology Standards-Uluslararası Bilgi Teknolojileri Standartları Komitesi) tarafından oluşturulmuş uluslararası bir standart mevcuttur [95].

2.7.1. Biyometrik Sistem Çeşitleri

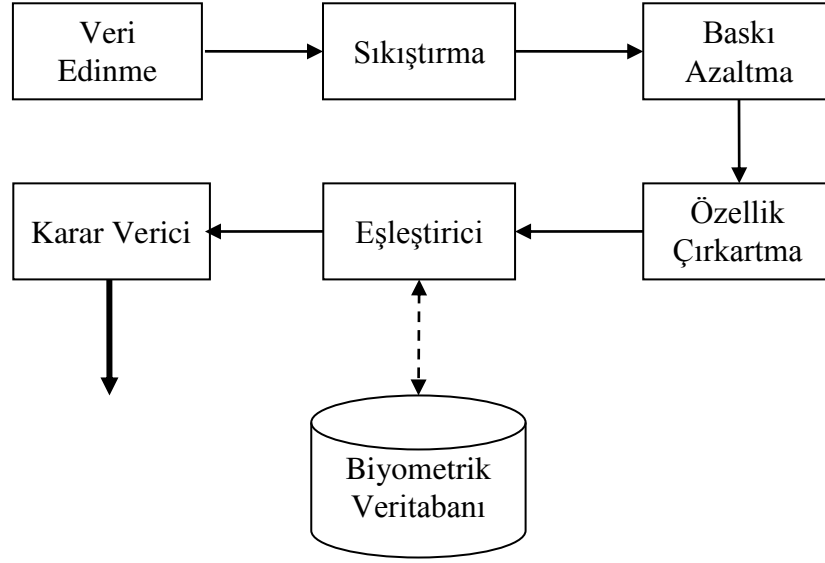
Günümüzdeki mevcut biyometrik tanıma sistemleri şunlardır:

A. Fizyolojik özellikler [92]:

- Parmak izi
- Retina
- DNA
- Damar
- Yüz
- El Geometrisi
- Ses
- Yüz Termogramı
- İris

B. Davranışsal özellikler:

- İmza Atımı
- Yürüyüş
- Tuş Vuruşu
- Konuşma



Şekil 2.5. Biyometrik Sistemlerin Genel Çalışma Mekanizması

a) Parmak izi

Kullanılan biyometrik sistemlerin belki de en önemlisi polis merkezlerinde, pasaport ve vize başvurularında (İngiltere 2007 yılından beri vize başvurularında, başvuran kişiden biyometrik veriler almaktadır) kullanılan parmak izi sistemleridir.

Parmak izi en fazla kullanılan, taklit edilemez bir biyometrik bilgidir. İlk kullanılmaya başlandığı yıllardan bu yana gerek yazılım gerekse donanım alanında parmak izi sistemlerinde önemli ilerlemeler kaydedilmiştir [93]. Bir otomatik parmak izi tanıma sisteminde (OPTS) parmak izi tanıma genellikle parmak izinde bulunan özellik noktalarının ve bunlara ait parametrelerin karşılaştırılması esasına dayanır [94].

Bu sistemlerin en önemli dezavantajı, parmak izinin taklit edilmesi durumunda sistemin yanılabilmesidir. Diğer bir dezavantaj bazı kişilerin pek çok sebepten ötürü (organ eksikliği, yanma, deri hastalıkları) parmak izlerinin bulunmamasıdır. Parmak izi taklit problemi, parmak izinin alındığı parmağın canlılığını test edecek gelişmiş sensörlerin kullanılması ile giderilebilecekken parmak izinin bulunmaması probleminin çözümü bulunmadığından bu sistem bu tip kişilerde uygulanamaz.

b) DNA

Kişinin saç, tırnak, deri parçası, kan, sperm veya herhangi diğer bir biyolojik materyali ele alınarak hücre içerisinde bulunan DNA moleküllerindeki dizilim incelenir. Özellikle emniyet güçleri tarafından cinayet mahallinde kalan biyolojik materyaller incelenerek katillere ulaşılması veya babalık davalarının sonuçlanması işlemlerinde kullanılmaktadır.

Doğruluğu çok yüksek bir yöntem olmasına rağmen maalesef pek çok dezavantaja da sahiptir. DNA'nın elde edileceği biyolojik dokunun kirlenmesi gibi durumlarda örnek kalitesi düşeceğinden analiz yapmak zorlaşır. Diğer dezavantajları işlemin 24 saat gibi bir sürede gerçekleştirilme zorunluluğu ve yüksek maliyet olarak sayılabilir.

c) El Geometrisi

Kişinin elinin veya kullanılan sisteme göre iki parmağının geometrik yapısı analiz edilir. Söz konusu yöntemde belirleyici özellikler parmakların uzunluğu, genişliği ve büküm noktalarıdır. Özellikle Amerika'da havaalanları ve nükleer güç istasyonlarında kullanılır. El geometrisi de diğer biyometrik yöntemler gibi doğruluk oranı yüksek bir yöntemdir. Ancak büyük ve ağır okuma cihazları nedeniyle maliyet ve kullanım açısından, resmin alınma süresinin uzun oluşu nedeniyle hız açısından dezavantajlara sahiptir. Bunun dışında elde bulunan yüzük gibi aksesuarlar, yara bandı gibi maddeler sebebiyle ya da yaralanma ve parmakların kaybedilmesi, gut veya kireçlenme gibi bir takım hastalıklar nedeniyle elin tanınması zorlaşır. Çocuklarda ve el ve ayakların çok hızlı büyüdüğü hastalıklara sahip olan kişilerde ise bu sistem kullanılamamaktadır.

d) Yüz

Biyometrik teknolojide devrim sayılabilecek buluşlardan bir tanesi olan yüz tanıma sistemleri gelişen birçok teknolojide olduğu gibi ilk kez askeriyede kullanılmıştır. İleri teknoloji silahlarının yönetimi için, özellikle ABD'de sıkça kullanılan bu sistemler bunun dışında, caddelere yerleştirilen güvenlik kameraları ile caddelerin izlenmesi ve aranmakta olan bir suçlunun bu şekilde yakalanması gibi uygulamalarda da kendilerine yer edinmişlerdir. Özellikle son 10 yıldır uygulama alanlarının artması nedeniyle yüzlerin otomatik olarak tanınması popüler bir konu haline gelmiştir.

e) İris

Yaklaşık 30 senedir kullanılan iri tanıma sistemlerinin çıkış noktası, kişinin sahip olduğu iris şeklinin ömrü boyunca değişmemesi ve diğer biyometrik sistemlere göre gözün daha az deforme olacak ve dış etkenlerden daha az zarar görecektir bir yapıya sahip olmasıdır. Uykusuzluk, gözyaşı, hastalıklar iris yapısını etkilemekle beraber diğer yöntemlerdeki kadar bariz bir etkilenme söz konusu değildir. Elbette ki bu yöntem gözü olmayan, gözleri görmeyen, Nistagmus hastalığına sahip (gözleri titreyen) veya irisleri olmayan kişilerde uygulanamaz. Ancak bu kişiler dışında havaalanı gibi kimlik doğrulamanın mutlak surette önemli olduğu yerlerde oldukça yüksek bir doğruluk oranı ile uygulanabilmektedir. Genel olarak parmak izi tanımaya benzetilen bu sistemin, parmak izine göre en önemli avantajı, parmak izi kullanılan biyometrik sistemlerde 60 veya 70 karşılaştırma noktası bulunurken, iris taramada karşılaştırma için yaklaşık 200 referans noktası kullanılmasıdır.

f) İmza

Bir kişinin, herhangi bir yazının altına söz konusu bu yazıyı yazdığını, okuduğunu veya onayladığını belirtmek için her zaman aynı biçimde yazdığı ad veya işaretler olarak tanımlanabilen imza kişiler tarafından yaşamları boyunca pek çok kez kullanılmaktadır. Özellikle hukuksal açıdan büyük yaptırımlarının bulunması ve taklit edilmesi sonucunda kişiyi borç altına sokabilmesi, tüm malvarlığını başka bir kimseye bağışlamasına sebep olması, işlemediği suçların üzerine kalmasına neden olması gibi sebeplerle hayatı önem taşımaktadır. Dolayısıyla kimlik doğrulamasında belki de en sık kullanılan yöntem olan imzanın gerçekten o kişi tarafından atılıp atılmadığının belirlenmesi önemli bir sorun olarak karşımıza çıkmaktadır. Bu sebeple kullanılan imza tanıma sistemlerinde imzayı tanımak için iki tip bilgi kullanılmaktadır. Bunlardan ilki imzalama süresi, hızı, ivmesi, kalemin basım şiddeti, kalemin gibi kişinin imzalama işlemi ile ilgili özellikler, diğeri ise bir desen olarak imzaya ait özelliklerdir. Bir imzayı taklit eden herhangi bir kişi desen olarak imzayı taklit edebilse bile imza atış şeklini (süre, ivme, kalemi yerden kaldırma miktarı vs.) tekrarlaması güçtür. İmza tanıma sistemlerinin dezavantajları, sistemin kullanıcının hızını, imza atma davranışını vs. öğrenebilmesi için uygun sayıda örneğe ihtiyaç duyması ve imza atımının kullanıcının o anki ruh haline, özellikle de acelesi olup olmadığına bağlı olarak değişmesidir.

3. GABOR FİLTRELERİN YÜZ İFADE TANIMADA KULLANIMI

3.1. Gabor Filtresi

1946 yılında Dennis Gabor [63] sinyallerdeki karakteristik bilgilerin elde edilebilmesi amacıyla, sinyale uygulanacak bir fonksiyonun zaman ve frekans eksenlerindeki bant genişliğinin, Fourier analizi belirsizlik ilkesine göre minimum olması gerektiğini belirtmiştir. Gabor [63], bu fonksiyonun bir Gauss sinyali ile modüle edilmiş karmaşık üstel bir sinyal ile elde edilebileceğini göstermiş ve bir boyutlu Gabor fonksiyonunu üretmiştir.

Memeli canlılardaki görüntü sisteminin Gabor filtreleri ile modellenebileceğini gösteren Marcelja [64], görüntü sisteminde bulunan hücrelerdeki alıcı alanların bir görüntü karşısında verdikleri sinyallerin çift veya tek simetrik yapıda olduğu belirtmiştir. Ayrıca, memeli canlılardaki görüntü merkezinde bulunan hücreler üzerine yapılan fizyolojik çalışmalar bu hücrelerin farklı yönelim ve uzaysal frekanslarda sinyal verdiklerini göstermiştir. Bu bilgiler ışığında Daugman [65], görüntü merkezindeki alıcı hücrelerin verdikleri tepki sinyallerinin teorik olarak 2 boyutlu karmaşık Gabor fonksiyonları şeklinde ifade edilebileceğini göstermiştir.

Daugman'ın görüntü sistemini 2 boyutlu karmaşık Gabor fonksiyonları ile matematiksel olarak göstermesi, Gabor filtrelerinin doku bölütlemesi [66], parmak izi tanıma [67], yüz tanıma [68] ve iris tanıma [69] gibi birçok bilgisayarlı görüntü ve örüntü tanıma uygulamasında öznitelik çıkartılmasında yaygın olarak kullanılmasına imkan sağlamıştır. Bunun yanı sıra Hubel ve Wiesel'in [70] görüntü sistemindeki alıcı alanların birer kenar detektörü işlevi gördüklerini göstermesi, Gabor filtrelerinin de iyi birer kenar detektörü olduklarını belirtmektedir. Bu özellik sayesinde, yüz ifadelerine ait öznitelikler çıkartılırken göz, ağız, kaşlar, burun gibi ifadeyi temsil eden önemli noktaların belirlenmesi gerçekleşmektedir.

Gabor filtresi, impuls cevabı, bir Gauss fonksiyonuyla harmonik bir fonksiyonun çarpılmasıyla tanımlanan lineer bir filtredir. Konvolüsyon işleminin çarpma özelliğinden dolayı, Gabor filtresinin impuls cevabının Fourier dönüşümü, harmonik fonksiyonun

Fourier dönüşümü ile Gauss fonksiyonunun Fourier dönüşümünün konvolüsyonudur [71]. Buradan, memeli canlılarda görme sisteminin modellendiği ve Daugman tarafından 2 boyutlu hale getirilen karmaşık Gabor dalgacık fonksiyonu $\psi_{\mu,\nu}(z)$

$$\psi_{\mu,\nu}(z) = \frac{\|k_{\mu,\nu}\|^2}{\sigma^2} e^{-\frac{\|k_{\mu,\nu}\|^2 z^2}{2\sigma^2}} \left[e^{ik_{\mu,\nu}z} - e^{-\frac{\sigma^2}{2}} \right] \quad (3.1)$$

olarak tanımlanır. 3.1 eşitliği ile verilen 2 boyutlu karmaşık Gabor fonksiyonda $\frac{\|k_{\mu,\nu}\|^2}{\sigma^2}$ terimi görüntülerdeki frekansa bağımlı güç spektrumunun azalmasını telafi ederken(46), $e^{-\frac{\|k_{\mu,\nu}\|^2 z^2}{2\sigma^2}}$ terimi Gauss dalgasını, $e^{ik_{\mu,\nu}z}$ terimi de karmaşık üstel sinyali ifade etmektedir. $e^{-\frac{\sigma^2}{2}}$ terimi ise Gabor filtresinin görüntüdeki farklı ışık dağılımlarına karşı etkilenmeyecek bir yapıda olduğunu göstermektedir. 3.1 fonksiyonunda $z=(x,y)$, yatay eksen x ve dikey eksen y üzerinde bulunulan noktayı belirtir. μ ve ν parametreleri sırasıyla, Gabor kernelinin yönelimi ve ölçeğini belirtirken, Gauss penceresinin standart sapmasını belirten σ parametresi, Gauss pencere genişliğinin dalga boyuna oranı şeklinde tanımlanır. Dalga vektörü $k_{\mu,\nu}$ ise;

$$k_{\mu,\nu} = k_{\nu} e^{i\phi_{\mu}}, \quad k_{\nu} = \frac{k_{max}}{f^{\nu}} \quad \text{ve} \quad \phi_{\mu} = \frac{\pi\mu}{n} \quad (3.2)$$

biçiminde tanımlanır. k_{max} maksimum frekansı, f^{ν} kerneller arasındaki uzaysal frekansı ve n de yönelim sayısını ifade eder. Bu çalışmada $\nu \in \{0,1,2\}$ ve $\mu \in \{0,1,2,3\}$ olmak üzere 3 farklı frekans ve 4 farklı yönelim ile toplam 12 farklı Gabor dalgacığı yüz görüntülerinden öznitelik çıkartmak amacıyla kullanılmıştır. Gabor dalgacıklarının diğer parametreleri ise $\sigma = \pi$, $k_{max} = \frac{\pi}{2}$ ve $f = \sqrt{2}$ olarak seçilmiştir. Ancak seçilen bu değerler ile kalmayıp parametreleri değişik değerler ile deneyerek işlemlerin başarı durumları karşılaştırılmıştır. Şekil 3.1’de uzay bölgesinde karmaşık Gabor fonksiyonunun sanal ve gerçek kısımları gösterilmektedir.

Gabor filtresinin bir diğer önemli parametresi de yüz görüntülerine uygulanacak olan Gabor filtre maskesinin boyutudur. Filtre maskesinin boyutu Gabor dalgacıklarını belirtecek kadar büyük olmalı, fakat işlemsel zorluklar yaşanmaması adına çok fazla da büyük olmamalıdır. Dunn [72]'ne göre Gabor dalgacığının boyutu, Gauss fonksiyonunun yayılma alanının 6 katı olarak belirlenmiştir. Gauss fonksiyonunun yayılma alanı $\frac{\sigma}{\|k_{max}\|}$ olmak üzere Gabor filtre maskesinin boyutu W ;

$$W = \frac{6\sigma}{\|k_{max}\|} + 1 \quad (3.3)$$

olarak belirlenir. Bu çalışmada, $k_{max} = \frac{\pi}{2}$, $f = \sqrt{2}$ ve Gauss fonksiyonunun standart sapması $\sigma = \pi$ olarak belirlendiğinden Gabor filtre maskesinin boyutu;

$$W = \frac{6\sigma}{\|k_{max}\|} + 1 = 12\sqrt{2}^v + 1 \quad (3.4)$$

olacaktır. Bu durumda, $v \in \{0,1,2\}$ olmak üzere 3 farklı frekans için Gabor filtre maske boyutları 13x13, 18x18 ve 25x25 olup, bu çalışmada 25x25 maske boyutu kullanılmıştır.

Farklı yönelimli hedef görüntü ile katlanarak ve bu görüntülerin ortalamaları alınarak hedef görüntüdeki farklı açılı bileşenleri ortaya çıkarabilen harmonik bir fonksiyon ile Gaussian bir fonksiyonunun çarpımından oluşan lineer bir Gabor filtresi, $G(x, y, \lambda, \theta, \psi, \sigma, \gamma)$ gibi denklemindeki parametreler; λ kosinüsün dalga boyu faktörünü, θ Gabor fonksiyonunun yönünü, ψ fazın ofset değerini ve γ ise uzamsal görüş açısını belirtir. Bu filtre ile;

- ✓ kenar bulmada,
- ✓ basit ve karmaşık hücreleri simüle etmede,
- ✓ görüntüde özellik çıkarımında,
- ✓ kontur tespit etmede

ve bazı görsel algı etkilerini açıklamada kullanılabilir.

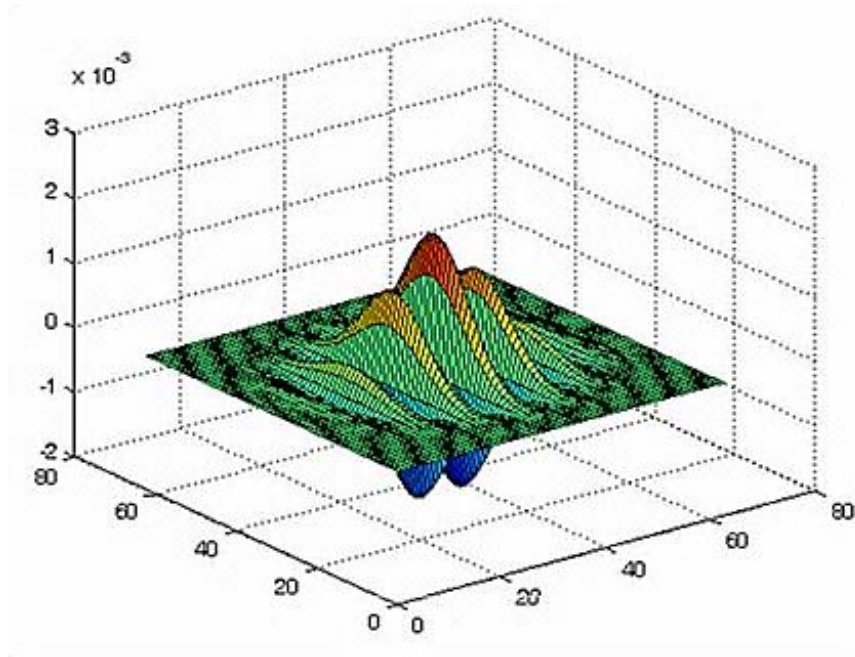
Gabor filtresi, görüntü işleme ve bilgisayarla görme alanında özellikle öznelilik çıkarımında (feature extraction) yaygın olarak kullanılan bir tekniktir. Matematiksel olarak 2 boyutlu karmaşık Gabor fonksiyonu;

$$G(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left[-\frac{1}{2}\left(\frac{\tilde{x}^2}{\sigma_x^2} + \frac{\tilde{y}^2}{\sigma_y^2}\right)\right] \exp[j2\pi f\tilde{x}]$$

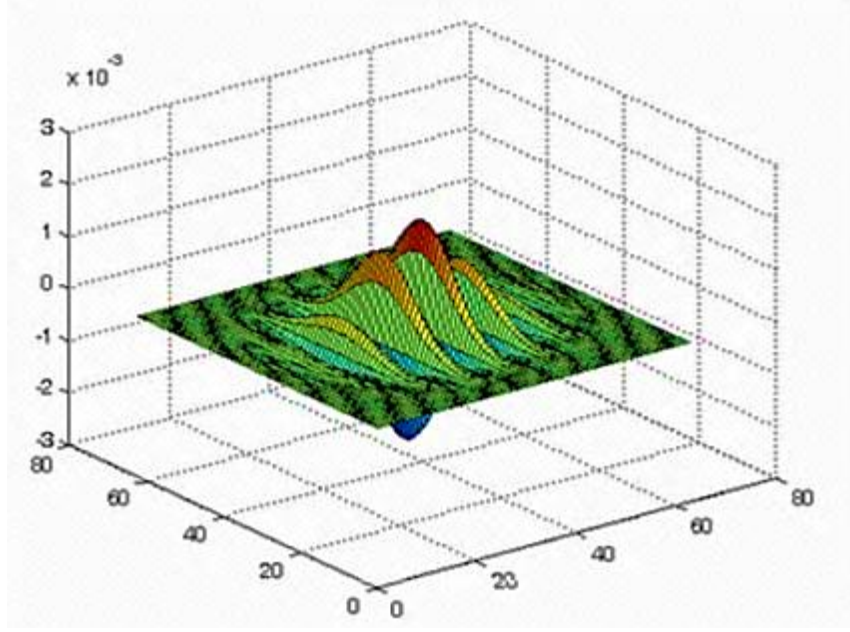
$$\tilde{x} = x\cos\theta + y\sin\theta$$

$$\tilde{y} = -x\sin\theta + y\cos\theta$$
(3.5)

biçiminde ifade edilir. Burada σ_x ve σ_y sırasıyla x ve y eksenleri boyunca Gauss zarfının standart sapmalarıdır. θ ($\theta \in [0, \pi)$), Gabor filtresinin yönelimini belirtir ve f sinusoidal dalganın frekansıdır. Şekil 3.1’de uzay bölgesindeki karmaşık Gabor fonksiyonunun reel ve sanal kısımları görülmektedir.

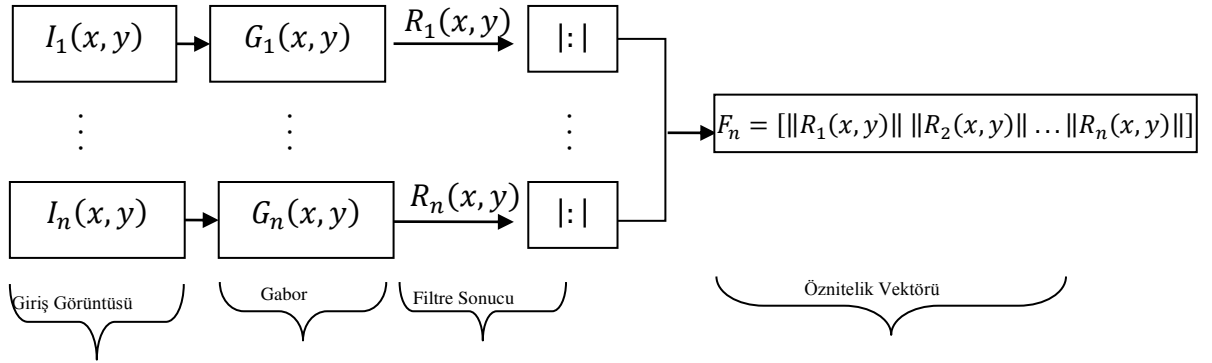


Şekil 3.1. Kompleks gabor fonksiyonunun reel (üst) kısımları



Şekil 3.2. Kompleks gabor fonksiyonunun sanal (alt) kısımları

Öznitelik vektörü şekil 3.3'ten görüleceği üzere filtrelenmiş görüntülerin genliğinden elde edilmiştir.



Şekil 3.3. Gabor filtre bankası sistemi

$$R_n(x, y) = I_n(x, y) * G_n(x, y) \quad (3.6)$$

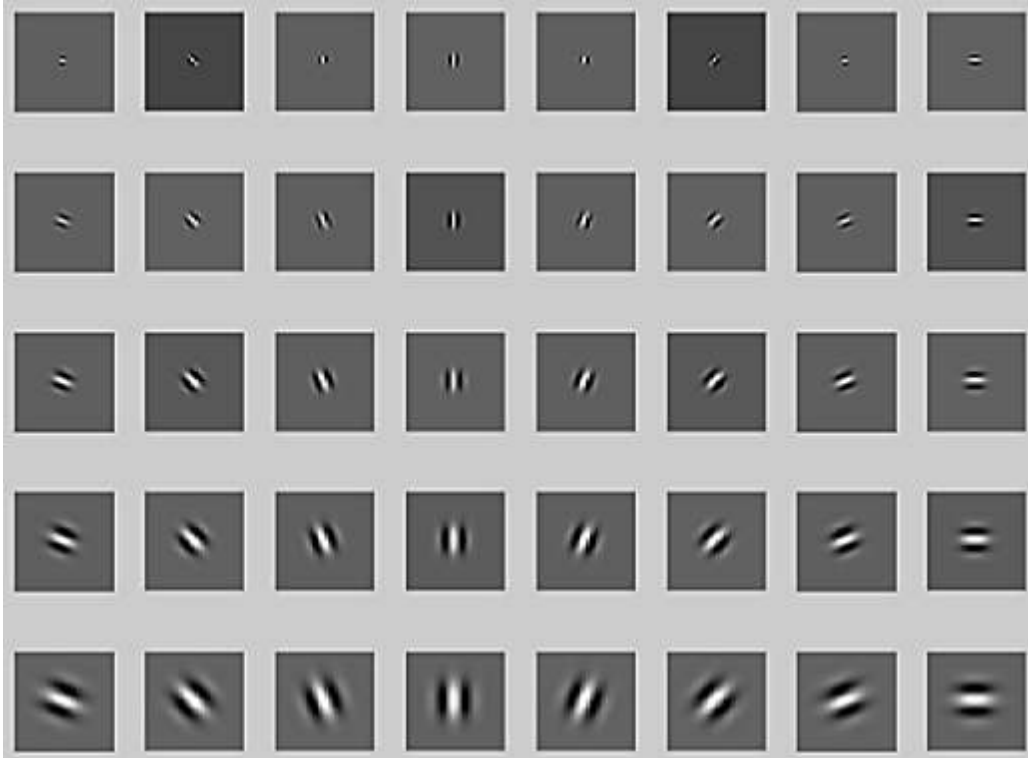
şeklindeki konvolüsyonudur. Burada “*” iki boyutlu konvolüsyonu belirtir.

Herhangi bir giriş görüntüsü $I_n(x, y)$ ile her karmaşık Gabor fonksiyonu $G_n(x, y)$ 'nin konvolüsyonundan elde edilen sonuç olan $R_n(x, y)$, reel ve sanal olmak üzere iki kısımdan oluşur. O halde öznitelik vektörünün elde edilmesi için bu sonucun genliği $\|R_n(x, y)\|$;

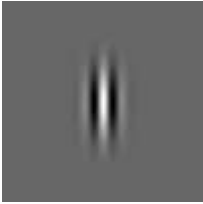
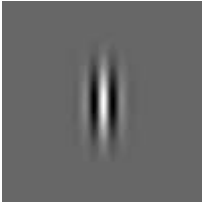
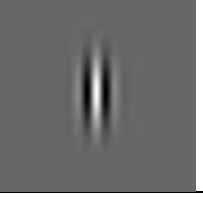
$$\|R_n(x, y)\| = \sqrt{\Re^2\{R_n(x, y)\} + \Im^2\{R_n(x, y)\}} \quad (3.7)$$

olarak elde edilir.

Değişik açı ve frekanslar kullanılarak değişik niteliklere sahip özniteliklerin elde edilebilmesi amacıyla Gabor filtrelerinden oluşan bir filtre bankası oluşturulabilir.



Şekil 3.4. 5x8 adet gabor filtresi örneği

Dalga Boyu	  	soldan sağa dalga boyu değeri sırasıyla 5, 10 ve 15. Diğer parametreler ise: yönelim 0, faz ofset 0 ve görüş açısı 0.5		
Yönelim	  	soldan sağa yönelim değeri sırasıyla 0, 45 ve 90. Diğer parametreler ise: dalga boyu 10, faz ofset 0 ve görüş açısı 0.5		
faz ofset	   	0,180,-90,90		
görüş açısı	 	0,5 ve 1		

Şekil 3.5. 100x100 boyutundaki filtre örnekleri



Gabor giriş resmi



Gabor çıkış resmi

Şekil 3.6. 8 dalga boyu (0,45,90, 135, 180, 225, 270, 315), yönelim değeri 0, faz ofset değeri 0,9 ve görüş açısı 0.5 için filtre çıkış örneği

3.2. Özniteliklerin Çıkartılması

Filtrelenen görüntülerden özniteliklerin çıkartılması için literatürde birçok yöntem kullanılmaktadır. Clausi ve Jernigan [73]'a göre filtre çıkışlarından özniteliklerin elde edilmesinde kullanılan bu yöntemlerden bazıları;

- ✓ Büyüklük cevabını kullanma
- ✓ Uzaysal düzeltme uygulanması
- ✓ Sadece reel bileşenlerin kullanılması
- ✓ Lineer olmayan sigmoidal fonksiyonun kullanılması
- ✓ Piksel komşuluğu bilgisini kullanma
- ✓ Tam dalga doğrultma uygulanması

olarak sıralanmıştır. Yapılan bu tezde öznitelik vektörü filtrelenmiş görüntülerin büyüklük cevaplarından elde edilmiştir. $z=(x, y)$ olmak üzere, herhangi bir giriş görüntüsü $I(z)$ ile herhangi bir kompleks Gabor fonksiyonu $\psi_{\mu,\nu}(z)$ 'nin konvolüsyonundan elde edilen sonuç olan $O_{\mu,\nu}(z)$, reel ve sanal olmak üzere iki kısımdan oluşur. O halde öznitelik vektörünün elde edilmesi için bu sonucun büyüklüğü $\|O_{\mu,\nu}(z)\|$;

$$\|O_{\mu,\nu}(Z)\| = \sqrt{\Re^2\{O_{\mu,\nu}(Z)\} + \Im^2\{O_{\mu,\nu}(Z)\}} \quad (3.8)$$

olarak elde edilir. $\Re^2\{O_{\mu,\nu}(Z)\}$ terimi öznitelik vektörünün reel kısmı, $\Im^2\{O_{\mu,\nu}(Z)\}$ terimi ise öznitelik vektörünün sanal kısmını ifade eder.

3.3. Öznitelik Boyutunun İndirgenmesi

Yüz görüntülerine Gabor filtrelerinin uygulanması sonucu çok sayıda öznitelik elde edilmektedir. Bu durum veri analizi, makine öğrenmesi, görüntü tanıma, metin sınıflandırma, web madenciliği ve biyolojik gen analizi uygulamalarında karşılaşılan en büyük problemlerden biridir. Bu nedenle, yapılan analize ilişkin etkili ve doğru sonuçlar alınması amacıyla veri boyutunun indirgenmesi çoğu zaman yararlı olmaktadır.

Birçok makine öğrenme ve örüntü tanıma uygulamalarında kullanılan algoritmalar literatürde “boyutun laneti” [74] olarak bilinen probleme karşı hassas olduklarından, öznitelik sayısının artması kullanılan öğrenme algoritmasının performansının azalmasına neden olmaktadır. Zira Gabor filtreleme sonucu elde edilen özniteliklerin tamamı yüz ifade analizi için gerekli olmayıp ifade sınıflandırması için önemi ve yararı olmayan bilgiler içermektedirler. Bu nedenle, yapılan çalışmalarda çoğu zaman büyük boyutlu veri kümelerinden çeşitli boyut indirgeme teknikleri kullanılarak daha küçük boyutta azaltılmış yeni veri kümeleri oluşturulmuştur. Sınıflandırma üzerine yapılan çalışmalarda boyutu indirgenmiş verilerin kullanılması ile daha doğru ve anlaşılabilir sonuçlar elde edilmiştir [75]. Boyut indirgeme yapılmasının nedenleri ve avantajları;

- ✓ Boyutu indirgenmiş verideki öznitelikler daha fazla ayırt edici olabilmektedirler.
- ✓ Birçok öğrenme algoritması için eğitim ve sınıflandırma zamanı boyutun indirgenmesiyle önemli ölçüde azalır.
- ✓ Önemsiz ve yararı olmayan öznitelikler sınıflandırma performansını olumsuz yönde etkilemektedirler.
- ✓ Kullanılan algoritmalar daha hızlı çalışır ve sistem belleğinin gereksiz yere kullanılması önlenmiş olur.

biçiminde sıralanabilir [76].

3.4. Temel Bileşenler Analizi

İstatistiksel bir yöntem olan Temel Bileşenler Analizi’nin (TBA) amacı boyut indirgemesi yapmaktır. TBA, orijinal t vektörün varyans yapısını daha az sayıda ve bu t vektörün doğrusal bileşenleri olan yeni vektörlerle ifade etme yöntemidir. Aralarında korelasyon (ilişki) bulunan t tane vektörün açıkladığı yapıyı, aralarında korelasyon bulunmayan ve sayıca orijinal vektör sayısından daha az sayıda ($k \leq t$) orijinal değişkenlerin doğrusal bileşenleri olan vektörlerle ifade etme yöntemi Temel Bileşenler Analizi olarak tanımlanır.

N tane görüntü, $[x_1, x_2, \dots, x_t]$ olmak üzere t boyutlu Gabor öznitelik vektörü ile gösterilmiş olsun. Temel Bileşenler Analizi [77,78] $k \ll t$ olmak üzere orijinal t boyutlu öznitelik uzayını k boyutlu öznitelik alt uzayına dönüştüren lineer bir dönüşüm bulmak için kullanılmaktadır. Yeni öznitelik vektörü y_i ;

$$y_i = W^T X_i, \quad (i = 1, 2, \dots, t) \quad (3.9)$$

biçiminde tanımlanır. Burada W aradaki lineer dönüşümü göstermekte olup, bu lineer dönüşüm $[x_1, x_2, \dots, x_t]^T$ in istatistiksel olarak elde edilen kovaryans matrisinin özdeğer ve özvektörlerinin bulunması ile elde edilir. Buna göre kovaryans matrisinin özdeğer ve özvektör çiftleri $(\lambda_1, e_1), (\lambda_2, e_2), \dots, (\lambda_t, e_t)$ olmak üzere i . temel bileşen;

$$y_i = W^T X_i = e_{1i}x_{1i} + e_{2i}x_{2i} + \dots + e_{ti}x_{ti} \quad (3.10)$$

biçiminde elde edilir. i . temel bileşenin varyansı ise;

$$Var(y_i) = \lambda_i \quad (3.11)$$

olur. Burada $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_t \geq 0$ dır. Temel bileşenler birbirleriyle ilişkisizdir ve varyansları kovaryans matrisinin özdeğerlerine eşittir.

t tane temel bileşen oluşturulduktan sonra, verilerin kaç tane temel bileşen ile ifade edilmesi gerektiği en önemli konudur. Bununla ilgili olarak literatürde çeşitli kurallar mevcuttur. Ancak bunlardan en yaygın kullanılanı “Genel varyansın en az %67’sini açıklayan sayıda temel bileşen” seçilmesidir. Buna göre, ilk k tane özdeğerin toplamı, tüm özdeğerlerin toplamına oranlandığında elde edilen %67 oranı, orijinal veri setinin temsil edilebilmesi için ilk k tane temel bileşenin kullanılmasının yeterli olduğunu ifade etmektedir [79].

Yüksek boyutlu görüntülerde öznitelik vektörlerini bulup çıkarmak zor olduğundan TBA bu noktada fazla bilgi kaybına neden olmadan boyut azaltmak için oldukça yardımcı olur. TBA bu yapısından ötürü görüntü sıkıştırma uygulamalarında da kullanılır.

TBA tekniğinin 3 faydası vardır: Giriş değerlerini birbirine dik yaparak, aralarındaki korelasyonu engeller. Dik bileşenleri (temel bileşenleri), en büyük varyasyona sahip olan ilk gelecek şekilde sıraya sokar. Veri kümesindeki en düşük varyasyona sahip bileşenleri ise eler.

3.4.1. Temel Bileşenler Analizi Algoritması

S , m adet elemana ait bir veri kümesi olsun;

$$S = \{\Gamma_1, \Gamma_2, \Gamma_3, \dots, \Gamma_m\} \quad (3.12)$$

S kümesine TBA uygulamak için,

1. Önişlem olarak kümeyi oluşturan verilerin ortalaması hesaplanır.

$$\Psi = \frac{1}{m} \sum_{n=1}^m \Gamma_n \quad (3.13)$$

2. S kümesi ile ortalama görüntü Ψ arasındaki fark kümesi hesaplanır.

$$\Phi_i = \Gamma_i - \Psi \quad (3.14)$$

3. Kovaryans matrisi C hesaplanır.

$$C = \frac{1}{m} \sum_{n=1}^m \Phi_n \Phi_n^T = AA^T, \quad A = \{\Phi_1, \Phi_2, \Phi_3, \dots, \Phi_n\} \quad (3.15)$$

4. Kovaryans matrisi C kullanılarak özdeğerler ve özvektörler v_l, u_l aşağıdaki gibi hesaplanır.

$$u_l = \sum_{k=1}^m v_{lk} \Phi_k \quad l = 1, 2, \dots, m \quad (3.16)$$

5. Özdeğerler en yüksek değerlikli olandan düşük olana göre yeniden sıralanır. En yüksek R özdeğere karşılık gelen özvektörler seçilir ve S veri matrisi bu

özvektörler matrisi üzerine izdüşürülerek temel bileşen analizi ile boyut azaltma işlemi gerçekleştirilmiş olunur. U , yüksek özdeğerlere gelen özvektörler matrisini (Eşitlik 3.17) ve S veri matrisini (Eşitlik 3.12) göstermek üzere W dönüşüm matrisi Eşitlik 3.18'deki gibi hesaplanır.

$$U = [u_1, u_2, u_3, \dots, u_r] \quad (3.17)$$

$$W = U^T S \quad (3.18)$$

TBA'ı daha iyi anlayabilmek için tablo 3.1a'da örnek olarak bir veri kümesi verilmiştir. Yukarıda verilen TBA algoritması kullanılarak bu veri kümesi üzerinden işlemlerin nasıl yapıldığı 5 adımda basitçe açıklanmıştır.

Adım 1: TBA'in daha iyi çalışabilmesi için, her veri kümesinden, o kümenin kendi ortalama değerini çıkarmamız gerekir (normalizasyon). Yani aşağıdaki her x değerinden $\bar{x}$ ortalama değerinin; her y değerinden de $\bar{y}$ ortalama değerinin çıkarılması gerekir. Buradaki amaç ortalama değeri sıfır olan bir küme elde etmektir (Şekil 3.7). Sonuç tablo 3.1b'deki gibidir.

$$\bar{x} = \frac{1}{N} \sum_{i=1}^{10} x_i \quad \text{ve} \quad \bar{y} = \frac{1}{N} \sum_{i=1}^{10} y_i \quad (3.19)$$

(N , her veri kümesindeki eleman sayısıdır.)

```
%Normalizasyon
x=[2.5 0.5 2.2 1.9 3.1 2.3 2.0 1.0 1.5 1.1]; %x data seti
y=[2.4 0.7 2.9 2.2 3.0 2.7 1.6 1.1 1.6 0.9]; %y data seti

X=A-mean(x);
Y=B-mean(y);
```

Şekil 3.7. Ortalama değeri sıfır olarak elde etmek için yazılan (normalizasyon) Matlab kodu

Tablo 3.1. TBA’de kullanılacak örneğe ait (a) veri kümesi; (b) veri kümesinin ortalama değeri hesap yapıldıktan sonraki değerleri

X	Y
3	2,4
1	0,7
2	2,9
2	2,2
3	3
2	2,7
2	1,6
1	1,1
2	1,6
1	0,9

(a) Veri kümesi

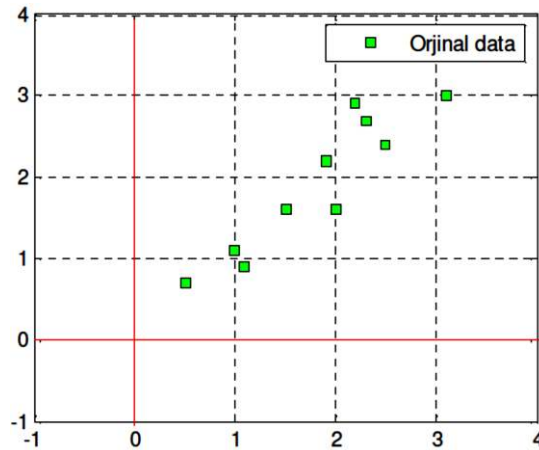
X	Y
0,69	0,49
-1,31	-1,21
0,39	0,99
0,09	0,29
1,29	1,09
0,49	0,79
0,19	-0,31
-0,81	-0,81
-0,31	-0,31
-0,71	-1,01

(b) Ortalama değeri sıfırlanmış veri

Orijinal verinin izdüşümü şekil 3.8’de gösterildiği gibidir.

```
%orjinal verinin projeksiyonu
plot(x,y,'rs','LineWidth',1,'MarkerEdgeColor','k','MarkerFaceColor',
,'g','MarkerSize',5);
grid on;
```

Şekil 3.8. Orijinal veriyi Kartezyen koordinat düzlemine döken kod



Şekil 3.9. TBA örneğindeki orijinal verinin Kartezyen koordinat sistemindeki izdüşümü

Adım 2: Normalize edilmiş verinin kovaryans matrisi C hesaplanır. (Şekil 3.10)

$$C = \begin{bmatrix} 0.6166 & 0.6154 \\ 0.6154 & 0.7166 \end{bmatrix} \quad (3.20)$$

```
%Kovaryans matrisi hesapla  
covariance=cov(X,Y);
```

Şekil 3.10. Kovaryans matrisi hesaplayan kod

C kovaryans matrisindeki diyagonal olmayan elemanlar pozitif olduğundan x ve y veri kümesinin birbiriyle doğru orantılı olduğu sonucuna varılabilir.

Adım 3: Kovaryans matrisi kare bir matris olduğundan özvektör (eigenvector) ve özdeğerleri (eigenvalue) hesaplanabilir. Bu hesap oldukça önemlidir. Zira örnekteki verinin biçim için önemli ve kullanılabilir olan yanını ortaya koymaktadır. (Şekil 3.11)

```
%Özdeğer ve özvektörleri hesapla  
[v d]=eig(covariance) % v= özvektörler, d= özdeğerler
```

Şekil 3.11. Özvektör ve özdeğerleri hesaplayan kod

$$v = \begin{bmatrix} -0.7352 & 0.6779 \\ 0.6779 & 0.7352 \end{bmatrix}, \quad d = \begin{bmatrix} 0.0491 \\ 1.2840 \end{bmatrix} \quad (3.21)$$

Özvektörler v ve özdeğerler d ile gösterilmiştir. Elde edilen bu özvektörler ile veri kümeleri karakterize edilmeye başlanmış olup her iki veri kümesinin (x ve y) birbiriyle ne kadar ilişkili olduğu ortaya koyulmaktadır.

Elde edilen özdeğer matrisindeki en yüksek değerli bileşen, temel bileşendir. Yani her iki veri kümesi arasındaki en önemli ilişkidir. Genelde izlenilen yöntem özvektörlerin hesaplanmasının ardından bu değerlerin azalan sırada dizilmesidir. Bu sayede en önemli olandan en az önemli olan bileşene doğru bir sıralama elde edilmiş olur. Bu dizilime göre istenilirse sifıra yakın veya ihmal edilebilecek daha az önemli olan bileşenler elenerek boyut azaltılmış olunur. Yani n boyutlu bir veri kümesinin, n adet özvektörü ve n adet özdeğeri olacaktır. Bizim için önemli olan p adet özdeğer ve özvektörü seçtiğimizde, sonuç kümemiz de p boyutuna sahip olacaktır.

Örneğe devam edilirse, sırada öznitelik vektörünü oluşturmak vardır. Öznitelik vektörü, aslında seçilen özvektörlerden oluşturulan yeni vektör matrisidir. Sadece 2 adet özdeğeri olduğundan, sadece 2 adet tercih seçeneği bulunmaktadır: Öznitelik vektörünü bütün özvektörleri kullanarak oluşturmak ya da sadece küçük ve daha az önemli olan kolonu kullanarak oluşturmak. Bu seçimlerin sonuçları ilerleyen adımlarda açıklanıyor olacaktır.

Adım 4: Bir önceki adımda kullanılacak bileşenler (özvektörleri) seçilip öznitelik vektörü oluşturulduktan sonra seçilen vektör matrisinin tersi (transpoz) alınarak sol tarafta normalize edilmiş veri kümesiyle çarpılır.

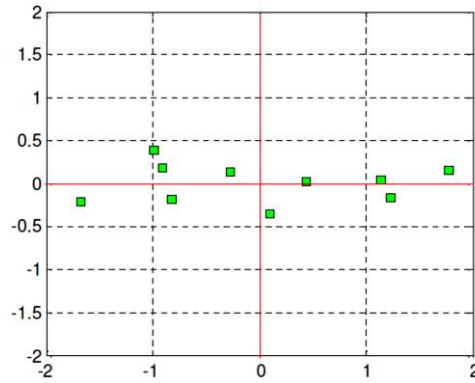
$$Z = u \times z \quad (3.22)$$

u , kolon halinde olan özvektörlerin en önemliden daha az önemli olana göre sıralandıktan sonra transpoze edilerek satır haline getirilmiş halidir. z , birinci adımda normalize edilmiş verinin transpoze edilerek satır haline getirilmiş halidir. Z , elde edilen yeni veridir. Son dönüşüm, her iki öznitelik vektörüyle birlikte hesaplandığında sonuç çizelge 1.2'deki, gibi, izdüşümü ise şekil 1.8'deki gibi olmaktadır.

Tablo 3.2. TBA ile iki özvektör kullanılarak dönüştürülmüş veri kümesi değerleri

X	Y
-0.828	-0.1751
1.7776	0.1429
-0.9922	0.3844
-0.2742	0.1304
-1.6758	-0.2095
-0.9129	0.1753
0.0991	-0.3498
1.1446	0.0464
0.438	0.0178
1.2238	-0.1627

Uygulanabilecek diğer senaryoda ise en büyük özdeğere sahip olan özvektör alınır (yani boyut azaltma uygulanmış olunur) ve aynı hesap yapılır. Bu durumda elde edilen yeni veri kümesi tablo 3.3'te gösterilmiştir.



Şekil 3.12. TBA ile her iki özvektör kullanılarak elde edilen son veri kümesinin izdüşümü

Tablo 3.3. TBA'de tek özvektör kullanılarak dönüşümü alınmış veri

X
0.828
1.7776
-0.9922
-0.2742
-1.6758
-0.9129
0.0991
1.1446
0.438
1.2238

Adım 5: TBA kullanılan algoritmalarda dönüştürülmüş veriden orijinal veriye geri dönülmesi başşı başına bir sorundur. Eğer öznitelik vektörü hesaplamasında en önemli özdeğerlere sahip olan özvektörler değil de tamamı alınırsa herhangi bir sorun yaşamadan, uygulanan adımlar tersine uygulanarak orijinal veri yeniden elde edilebilir. Ancak, önemli özvektörler alınıp boyut küçültülerek hesaba devam edildiyse ki genelde böyle yapılır, orijinal verinin tam olarak geri elde edilmesi pek mümkün değildir. Çünkü ihmal edip elenen özvektörlere karşılık gelen veri kaybolmuştur.

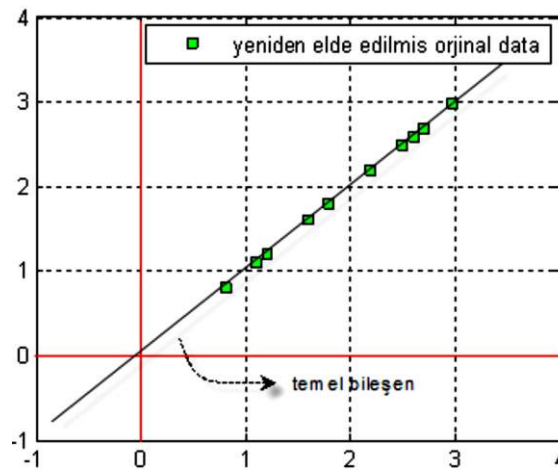
$$Z = u \times z \quad (3.23)$$

$$z = u^T \times Z \quad (3.24)$$

Orijinal veriyi tam olarak elde etmek için normalizasyon için çıkarılan ortalama değerinin de geri ilave edilmesi gerekir.

$$Z_{orj} = (u^T \times Z) + m \quad (3.25)$$

Z_{orj} , orijinal verinin satır formundaki halini, m ortalama değeri göstermektedir. Bu hesap boyut azaltmayla elde edilen vektör kullanılarak yapıldığında geri elde edilen orijinal verinin izdüşümü şekil 3.13'te gösterildiği gibi olmaktadır.



Şekil 3.13. TBA ile boyut azaltıp orijinal veriyi geri elde ettikten sonraki verinin izdüşümü

Bu noktaya dek basit bir örnek üzerinden gidilerek TBA'nın temelinde yatan mantık ve uygulama biçimiyle ilgili genel bir bilgi verilmiştir. Önerilen modelde TBA boyut azaltma ve izdüşüm matrisini oluşturmakta kullanılmaktadır.

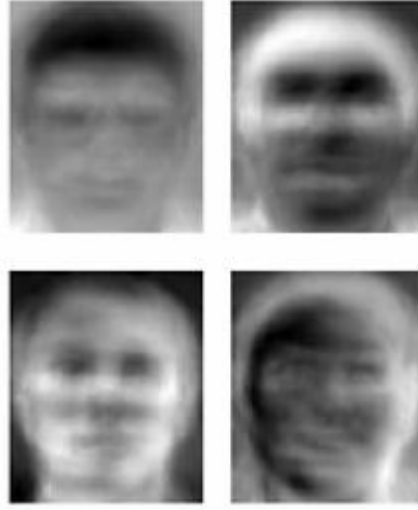
3.5. Özyüzler (Eigenfaces)

Yüz ifade tanıma için bir eigenvektör kümesi kullanan metottur. Bu eigenvektörleri olası insan yüzlerinin ihtimal dağılımının kovaryans matrislerinden türetilmişlerdir.

Layman'e göre, özyüzler "Yüz içeriğinin standartlaştırılmış" kümesinden ibarettir. Ve yüzün birden fazla resminin istatistiksel analizinden türetilmişlerdir. Herhangi bir insan yüzü bu standart yüzlerden birisi olabilir varsayımı yapılmıştır. Yaklaşım şu şekildedir: Bir insanın yüzü %10 birinci yüzden, %24 ikinci yüzden, %X üçüncü yüzden v.s. şeklinde yüzün tamamı için bir küme oluşturulur. Bu şekilde yüz tanıma için saklanan yüzler gerçekte fotoğraf makinesi ile çekilen yüzden daha az yer kaplayarak saklanabilmektedir.

Özyüzler tekniğinde çalışabilmek için önce büyük bir özyüz veritabanının oluşturulması gerekmektedir. Bunun için önce pek çok yüzün fotoğrafları çekilir, sonra bu yüz fotoğraflarının ağız ve göz seviyeleri standart hale getirilir, sonra yüzün resim içindeki hizalanması hepsi için aynı yapılır ve son olarak da hepsi aynı piksel boyutuna getirilir. Sonra bu resimlerin piksel değerlerini içeren eigenvektörleri üretilir. Bu eigenvektörleri ile en sonunda kovaryans matrisi çıkarılmaktadır.

Ortaya çıkan özyüzlerin tam olarak ortalaması alınabilirse ortalama bir insanın, yani herkesin benzeyebileceği bir insanın, gri tonlamada resmi çıkarılabilir.



Şekil 3.14. AT & T Laboratuvarlarından bazı özyüz imgeleri

Özyüzler metodu, Temel Bileşen Analizi (Principal Component Analysis) dönüşümleri yaparak yüzleri modellediği bir uzaya taşır. Bu dönüşümlere aynı zamanda Karhunen-Loeve dönüşümleri de denilmektedir. TBA insan yüzleri arasındaki değişkenliği belirlemektedir. Örneğin birbirinden farklı iki yüze bakıldığında aslında nelerin değişik olduğunu matematiksel olarak belirlemek için TBA kullanılır. TBA belirleme işlemini yüzleri veya yüz parçalarını kategorize ederek yapmamaktadır. Yani burnun büyüklüğü şu kadar, yüzün genişliği bu kadar şeklinde bir grupta yapılarak özellikleri belirlemez. Bunun yerine pek çok yüz incelenerek insanların yüzleri arasındaki farklılığı oluşturan değişkenler belirlenmektedir. İşte bu değişkenlere eigenfaces – özyüzler denilir. Herhangi bir insanın yüzünün diğer yüzlerle benzeme oranı esastır ve bir yüz başka yüzler cinsinden ifade edilir. Aşağıdaki şekil bunu örneklemektedir.

$$\begin{aligned}
 \text{Yüz} &= 4,0719 * \text{Eigenface 1} + 0,1874 * \text{Eigenface 2} + 0,7253 * \text{Eigenface 3} \\
 &+ 0,0392 * \text{Eigenface 4} + 0,1725 * \text{Eigenface 5} + \dots
 \end{aligned}$$

Şekil 3.15. Bir yüze ait vektörün yüz uzayında şekilsel olarak gösterimi

3.6. Laplacian(Laplasyen) – Kenar Bulma Yöntemi

İki değişkenli görüntü fonksiyonu $f(x,y)$ için basit bir isotropic türev operatörü Laplacian [80];

$$\Delta^2 f = \frac{\delta^2 f}{\delta x^2} + \frac{\delta^2 f}{\delta y^2} \quad (3.26)$$

denklemleri verilir. x -yönünde kısmi ikinci türevi göz önüne alınırsa,

$$\frac{\delta^2 f}{\delta x^2} = f(x+1, y) + f(x-1, y) - 2f(x, y) \quad (3.27)$$

y -yönünde kısmi ikinci türevi göz önüne alınırsa,

$$\frac{\delta^2 f}{\delta y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y) \quad (3.28)$$

olur. İki boyutlu Laplacian operatörünün sayısal uygulaması yukarıda verilen iki bileşenin toplanmasıyla elde edilir:

$$\Delta^2 f = f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) - 4f(x, y) \quad (3.29)$$

Laplacian, bir türev operatörü olduğuna göre görüntüdeki grilik seviyesi süreksizliklerin altını çizerek, grilik seviyesinin çok yavaş değiştiği bölümlerin ise önemini azaltacaktır. Dolayısıyla Laplacian operatörü bir görüntüye uygulandığında çıkışta kenar çizgileri ve diğer süreksizlik arz eden noktaların grimsi, bir özelliğe sahip olmayan geri kalan kısımların ise siyah arka plan olarak kaldığı bir görüntü elde edilir.

3.7. Marr-Hilderth Kenar Belirleme

Bu yöntem David Marr ve Ellen Hildreth [80] tarafından yayınlanan bir makalede geliştirilmiştir.

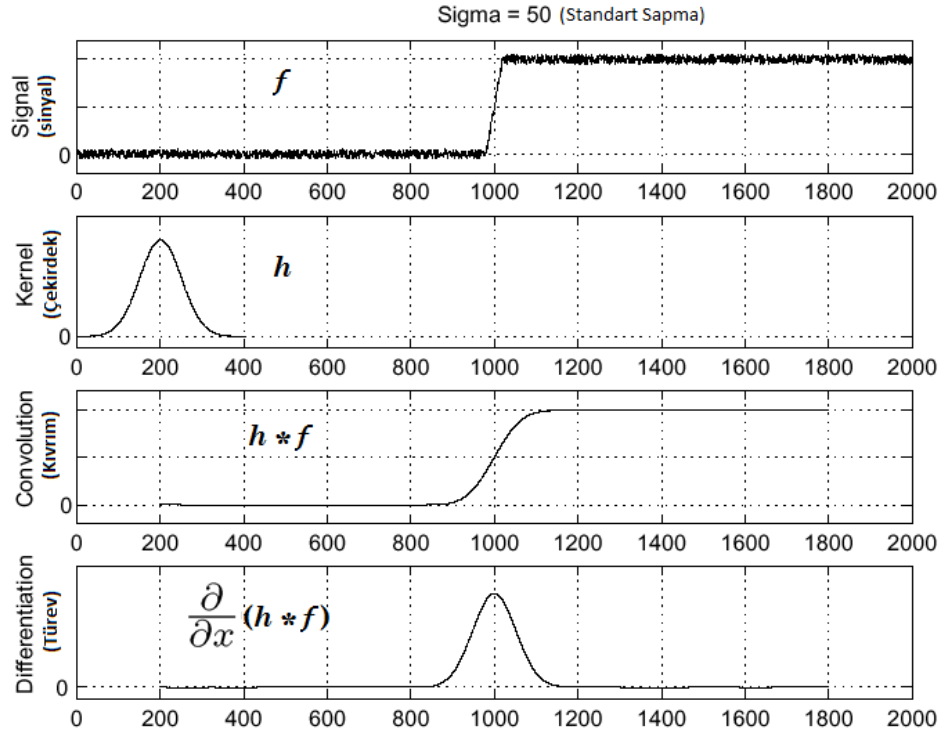
Marr-Hilderth yöntemi LoG (Laplacian of Gaussian) operatörünün sıfır kesişlerine dayanmaktadır. Sıfır kesişi bir fonksiyonun sıfır eksenini kestiği yerdir. Bu yöntemde sıfır kesişleri boyunca kenarları bulmak için Laplacian önışlem olarak uygulanan yumuşatma operatörüyle birleştirilmektedir. Yumuşatma operatörü,

$$h(r) = -e^{-\frac{r^2}{2\sigma^2}} \quad (3.30)$$

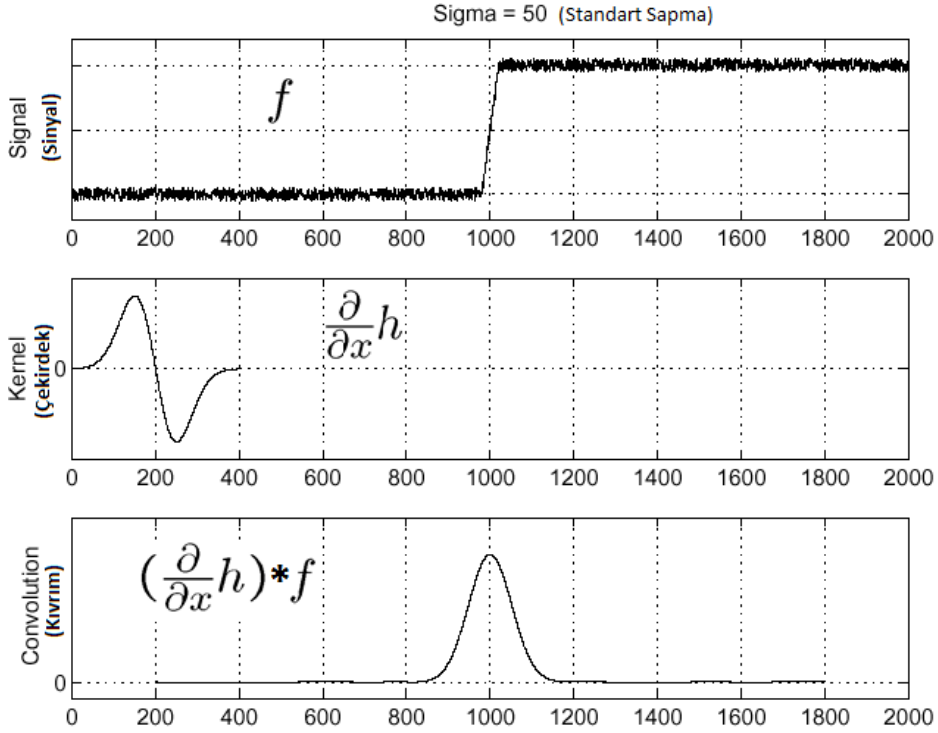
denklemleri verilmektedir. $r^2 = x^2 + y^2$ 'dir ve σ da standart sapmadır. $h(r)$ fonksiyonun r 'ye göre ikinci türevi, yani h 'nin Laplacian'ı;

$$\Delta^2 h(r) = -\left[\frac{r^2 - \sigma^2}{\sigma^4}\right] e^{-\frac{r^2}{2\sigma^2}} \quad (3.31)$$

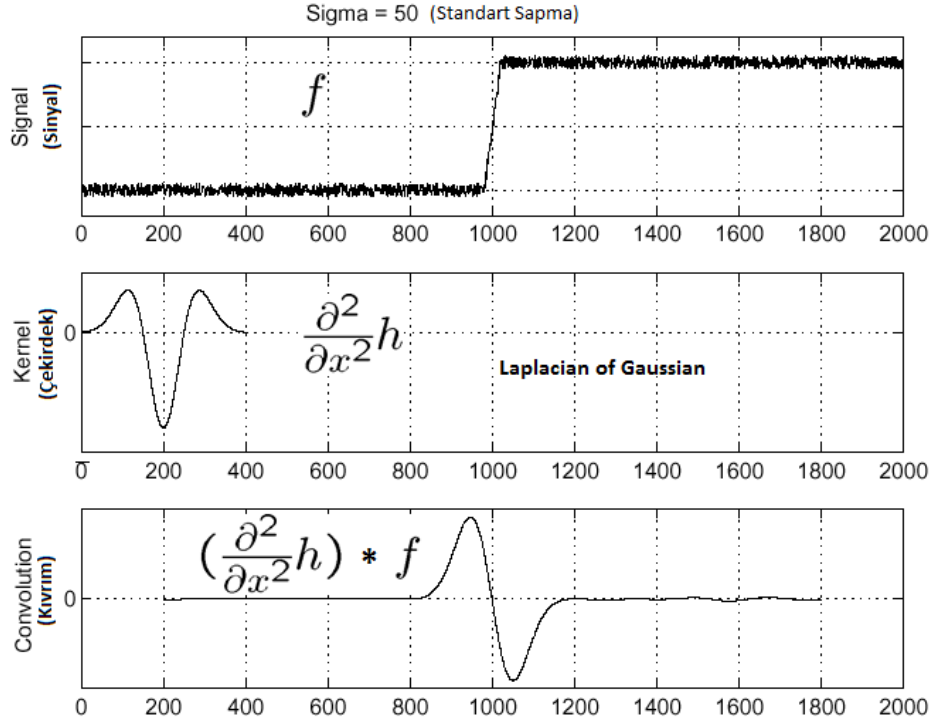
Bu fonksiyon genellikle LoG (Laplacian of Gaussian) olarak adlandırılmaktadır. Laplacian operatörünün en büyük zaaflarından birisi gürültüye karşı çok duyarlı olmasıdır. Diğer bir nokta ise herbir kenar için iki değer üretmesidir. Bu istenmeyen bir özelliktir. Gürültünün meydana getireceği bozulmaları engellemek için Laplacian'dan önce görüntüye Gaussian operatörü uygulanır. Bu operatör aslında bir alçak geçiren filtredir ve görüntünün bulanıklaşmasına sebep olur. Ardından Laplacian uygulanarak sıfır kesişleri tespit edilir.



Şekil 3.16. Laplacian of Gaussianın ilk aşamaları



Şekil 3.17. Laplacian of Gaussian filtre çekirdeği



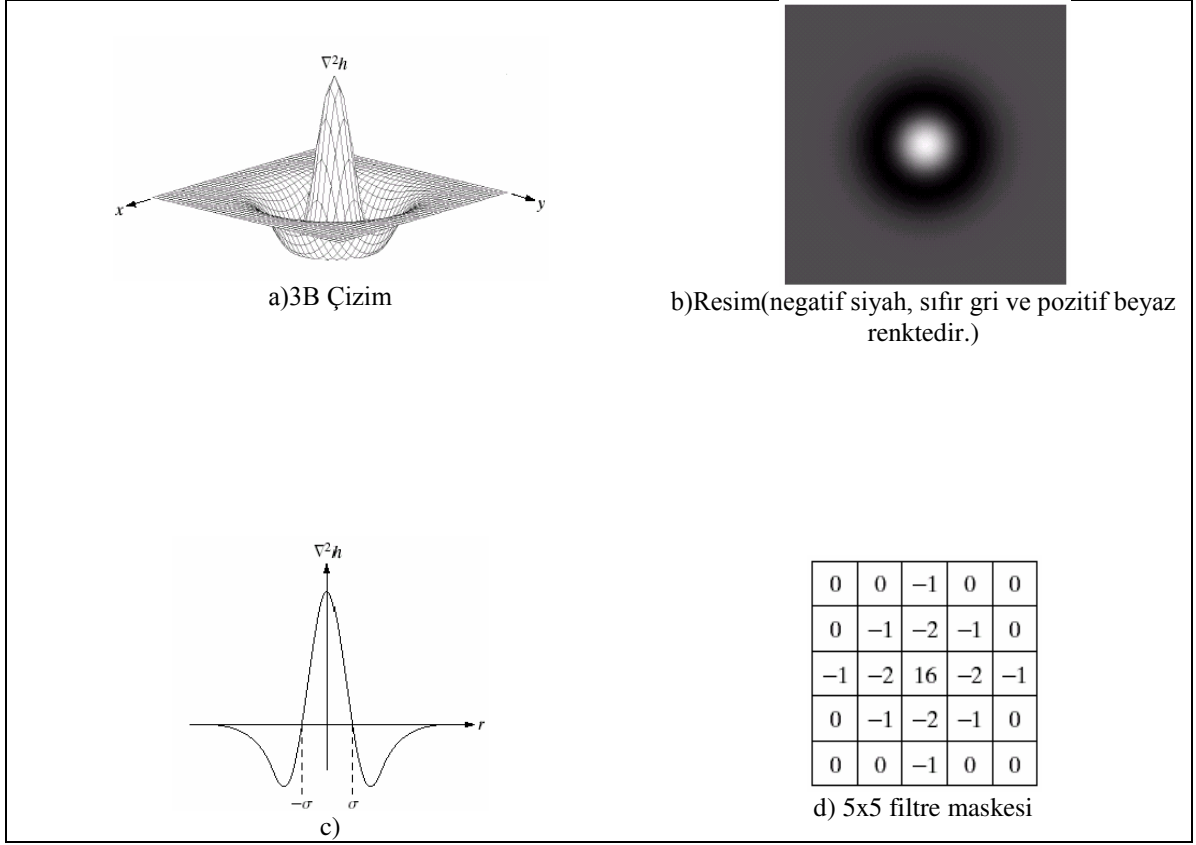
Şekil 3.18. Laplacian of Gaussian filtre yapısı

Şekil 3.16'daki f sapma fonksiyonunu göstermektedir. Siyahtan beyaza yumuşak bir geçiş vardır. İdeal bir kenar aslında basamak fonksiyonudur ancak görüntüyü elde eden sistemler ve örnekleme işlemelerinden dolayı kenarlar daha çok rampa fonksiyonlarına benzemektedir. h Gaussian fonksiyonunu göstermektedir. $h*f$ konvolüsyonu sonucunda gürültü yok edilmiştir. Sonucun birinci dereceden türevi alındığında piksel değerlerinin sabit kaldığı yerlerde değişim sıfır olduğu için sonuç sıfır olmuş, değişimin olduğu yerde (kenar) ise sonuç sıfırdan farklı olmuştur.

Şekil 3.18'de ise kenar tam olarak sıfır kesişinin olduğu yerdedir (3. grafik). Şekil 3.18, 2. grafikte LoG fonksiyonu gösterilmektedir. LoG benzerliğinden dolayı bazen Meksika şapkası fonksiyonu olarak da adlandırılmaktadır. Marr-Hilderth yöntemiyle f görüntüsünün kenarları belirlenmek istenirse adımları şöyle,

- LoG filtresi ($\Delta^2 h(r)$) belirlenir,
- LoG*f hesaplanır,
- sonuç görüntü eşik değerinden geçirilir

-kenarları belirlemek için elde edilen görüntüye yüksek geçiren filtre (HPF) uygulanır.



Şekil 3.19. Log of Gaussian (LoG) filtresi

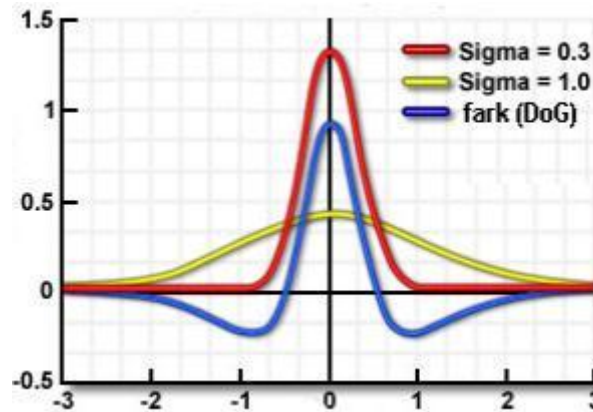


Şekil 3.20. Log of Gaussian (LoG) filtrenin örnek çıkış imgesi

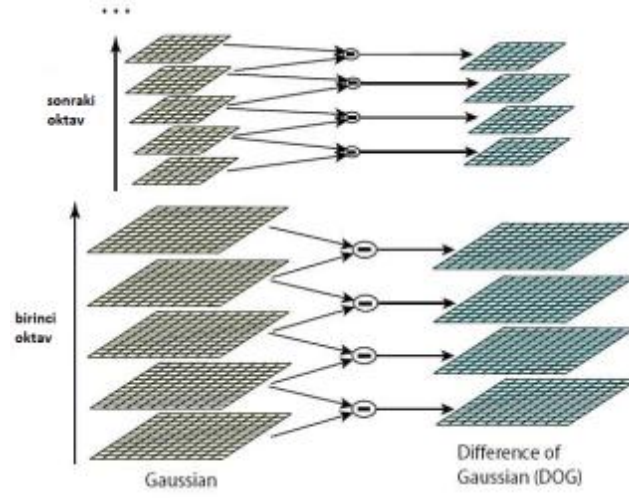
Özet olarak, kenarlar bir görüntüdeki yüksek frekansların zaman uzayındaki karşılığıdır. Nasıl ki frekans tanım olarak saniyedeki çevrim sayısı ise görüntüdeki ani değişimlerin olduğu yerler frekans domeninde yüksek frekans bölgesine denk gelmektedir. -Salınımın az- Değişimin az olduğu yerde ise frekans düşüktür. Türev olarak bir nevi bu değişimin hızını, ikinci türev olarak da ivmesini hesaplamaktadır. Eğer bir kenar yoksa düzgün bir bölgeyse türev sonucu sıfır olur, eğer bir kenar, çizgi ya da gürültü noktası varsa türev sıfırdan farklı sonuç verir.

3.8. Difference of Gaussians (DoG)

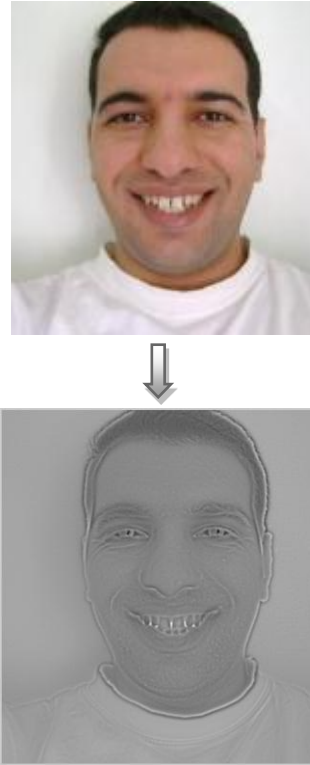
DoG, bulanık görüntüden başka bir bulanık görüntünün farkı alınarak yeni görüntü hesaplayan gri tonlamalı bir görüntü iyileştirme algoritmasıdır. Bulanık görüntüler, orijinal görüntüden farklı standart sapmalara sahip Gauss bulanıklaştırma kullanarak elde edilir. Bir resmi bulanıklaştırma yalnızca yüksek frekanslı uzaysal bilgiyi bastıran bir Gauss çekirdeği kullanılır. Yeni bir görüntü çıkarma işlemi, farklı değerdeki iki standart sapma kullanılarak Gauss ile bulanıklaştırılmış iki resmin farkı alınmasıyla yapılır. Fark alma işlemi ise iki bulanık resmin piksellerindeki sıfır geçişleri yapılır ve sonuç olarak sıfır geçiş yoğunluğu resmin kenar veya özellik olduğu alanlarda belirgin olarak artar. Aslında bir yerde, difference of Gaussians (DoG) orijinal gri tonlu resimdeki uzaysal frekansların çok azı haricindeki frekansları eleyen bant geçiren bir filtreye benzer [81].



Şekil 3.21. DoG filtresi

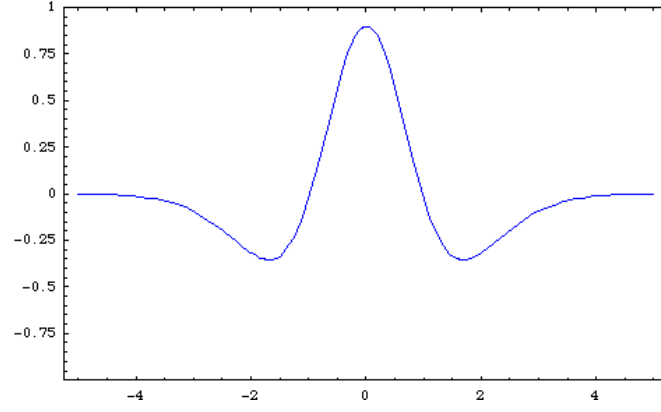


Şekil 3.22. DoG çalışma şeması

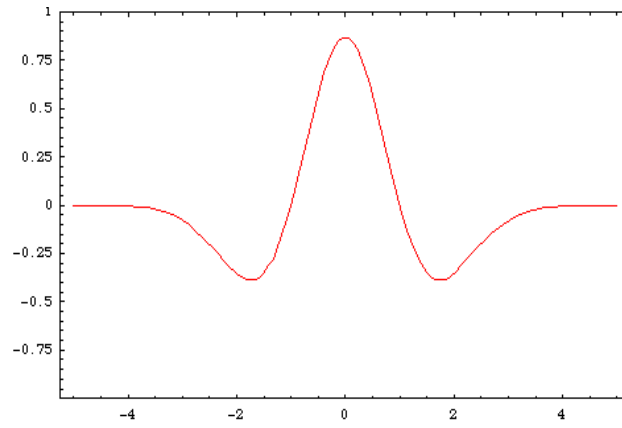


Şekil 3.23. DoG filtresinin çıkış imge örneği

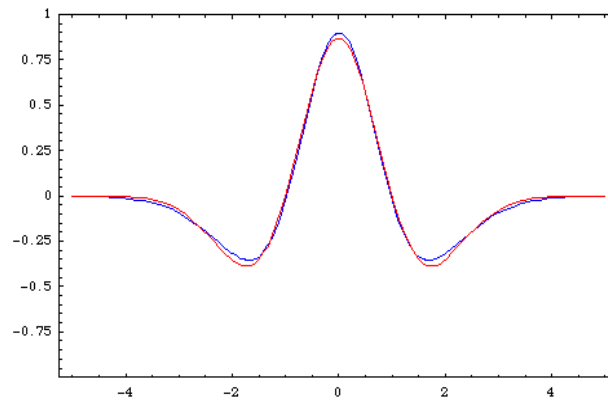
DoG filtresi, literatürde kimi yerde Meksika Şapkası olarak da bilinir. Dalgacık yapısı Meksika Şapkası dalgacığına çok benzemektedir.



Şekil 3.24. DoG filtre dalgacığı



Şekil 3.25. Meksika Şapkası dalgacığı



Şekil 3.26. DoG ve Meksika Şapkası dalgacıkları

LoG (Laplace of Gaussian) filtresine benzer, özel standart sapma σ_1 değerine sahip Gauss kıvrımıyla birinci resim aşağıdaki denklem ile düzleştirilmektedir.

$$G_{\sigma_1}(x, y) = \frac{1}{\sqrt{2\pi\sigma_1^2}} e^{\left[-\frac{x^2+y^2}{2\sigma_1^2}\right]} \quad (3.32)$$

düzleştirilmiş resmin konvolüsyonu ise:

$$g_1(x, y) = G_{\sigma_1}(x, y) * f(x, y) \quad (3.33)$$

Farklı σ_2 standart sapma ile ikinci düzleştirilmiş resmi elde etmek için ise;

$$G_{\sigma_2}(x, y) = \frac{1}{\sqrt{2\pi\sigma_2^2}} e^{\left[-\frac{x^2+y^2}{2\sigma_2^2}\right]} \quad \text{ise,} \quad (3.34)$$

$$g_2(x, y) = G_{\sigma_2}(x, y) * f(x, y)$$

Farklı σ standart sapmayla ile bulanıklaştırılmış iki resmin farkını alarak resimdeki kenarlar tespit edilebilir.

$$\begin{aligned} g_1(x, y) - g_2(x, y) &= G_{\sigma_1} * f(x, y) - G_{\sigma_2} * f(x, y) \\ &= (G_{\sigma_1} - G_{\sigma_2}) * f(x, y) \\ &= DoG * f(x, y) \end{aligned} \quad (3.35)$$

Şeklinde olur. DoG operatörü veya sarmalama çekirdeği ise şöyledir:

$$DoG \triangleq G_{\sigma_1} - G_{\sigma_2} = \frac{1}{\sqrt{2\pi}} \left[\frac{e^{-(x^2+y^2)/2\sigma_1^2}}{\sigma_1} - \frac{e^{-(x^2+y^2)/2\sigma_2^2}}{\sigma_2} \right] \quad (3.36)$$

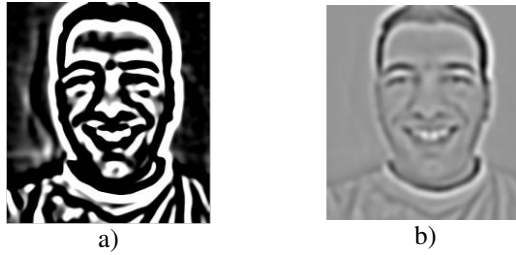
Şeklinde olur.

3.9. Adaptif Yaklaşım

Sistemlerin modellenmesinde kullanılan klasik teknikler, modelin yapısının ve bazı istatistikî değerlerin (model derecesi, giriş ve gürültünün dağılımı v.b.) bilinmesi durumunda iyi çözüm sunar. Bu bilgilerin elde edilemediği durumlarda performansta düşme yaşanmaktadır.

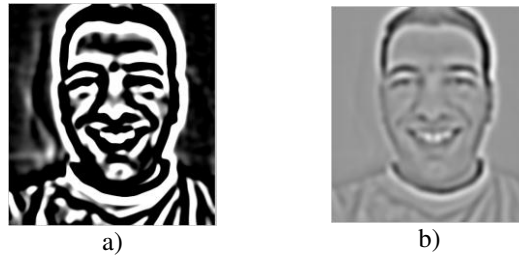
Adaptif sistemler çevre şartlarına göre kendi kendisini en iyiye doğru kanalize eden sistemlerdir. Bu sistemler en çok haberleşme ve kontrol sistemlerinde uygulanmaktadır. Adaptif sistemlerin en önemli özelliği zamanla değişen sistemlere rahatlıkla uygulanabilmesi ve yeni durumlara göre kendi kendini ayarlayabilmesidir. Doğrusal sistemlerde belirli girişlere karşı çıkışın istenen şekilde olması, diğer tür girişlerin uygulandığı durumlarda veya kontrol edilen sistemin zamanla çevre şartlarından etkilenerek özelliklerinin değişmesi durumunda sistemlerin kararsız davranması adaptif sistemlerde daha az gözlenmektedir. Bu şekilde sistemden beklenen ile elde edilen çıkış arasındaki fark hata adını almakta ve adaptif sisteme giriş olarak verilmektedir. Bu farkın sıfır olması, sistemin arzu edilen şekilde çalışması anlamına gelir. Adaptif modelleme, sistem parametrelerini, hatayı sıfır yapacak şekilde ayarlamak için kullanılır [82-85].

Adaptif algoritmalar iki gruba ayrılır. Birinci grup, LMS (Least mean squares-En küçük ortalama kareler) algoritmalarına dayalı olan algoritmalar. LMS algoritması, bir azaltım (gradient) arama algoritması kullanarak sistem hatasının karesinin ortalamasını minimize eder ve hesap karmaşasının az olmasından ve gürbüzlüğünden dolayı çok popülerdir. Fakat LMS algoritmasının yakınsama oranı sisteme ve giriş istatistiklerine bağlıdır. Sistem parametrelerinin tahminindeki düşük yakınsama oranından dolayı LMS algoritması her zaman tatmin edici çözümler vermemektedir.



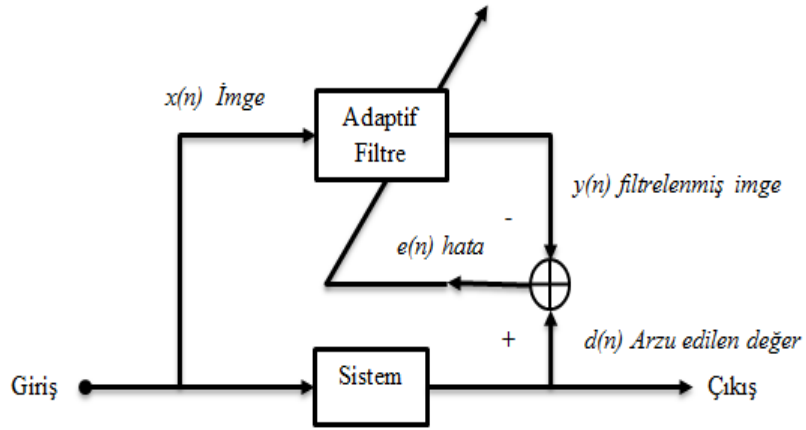
Şekil 3.27. LMS algoritmasının çıkış resimleri a) ikili seviyede, b) gri seviyede

İkinci grup, hatanın karesinin deterministik bir toplamını minimuma indiren RLS (Recursive least squares- Özyinelemeli en küçük kareler) algoritmasına dayanır. RLS algoritması, LMS algoritmasından daha yüksek performans göstermektedir fakat hesaplama karmaşıklığı fazladır. Levinson-Durbin algoritmasının bir uzantısı olarak yorumlanabilen kafes yapılı sistem çok sayıda avantajlara sahiptir ve işaret işleme gibi geniş uygulamalarda kullanılır [86].



Şekil 3.28. RLS algoritmasının çıkış resimleri a) ikili seviyede, b) gri seviyede

Sayısal filtrelerin, Sonlu Darbe Cevabı (Finite Impulse Response, FIR) ve Sonsuz Darbe Cevabı (Infinite Impulse Response, IIR) gibi özel tipleri vardır. Adaptif filtrelerin gürültü temizleme, doğrusal kestirim, adaptif sinyal iyileştirme ve adaptif kontrol gibi birçok uygulama alanları vardır. Genel bir adaptif filtre yapısı şekil 3.28’de görüldüğü gibidir. FIR filtrenin çıkışı $y(n)$, hata sinyalini $e(n)$ oluşturmak için giriş sinyali $d(n)$ ile karşılaştırılır. Hata sinyali $e(n)$, giriş sinyali $d(n)$ ile çıkış sinyali $y(n)$ arasındaki farkı temsil etmektedir. Daha sonra hata sinyali $e(n)$, filtrenin ağırlıklarını güncelleştirmek için adaptif kontrol algoritmasına giriş olarak uygulanmaktadır. $e(n)$ ’in uygulama süreci, hata sinyali çok küçük değerlere düşürülünceye kadar devam etmektedir [87].



Şekil 3.29. Adaptif filtrenin blok şeması

3.9.1. LMS (En Küçük Ortalama Kareler) Algoritması

En küçük ortalama kareler (LMS) algoritması Widrow & Hoff tarafından sunulmuştur. En yaygın kullanıma sahip, kör olmayan algoritmalar sınıfındadır. Optimum ağırlık vektörlerini hesaplamak için referans işaretine, yani istenen işaret bilgisine ihtiyaç vardır. LMS algoritmasında matris ters alma veya ilinti matrislerini hesabı gerekmediğinden basit bir algoritmadır. LMS algoritması aşağıdaki denklemler ile ifade edilir:

Çıkış:

$$y(n) = w^T \cdot x(n) \quad (3.37)$$

Hata:

$$e(n) = d(n) - y(n) \quad (3.38)$$

Ağırlık:

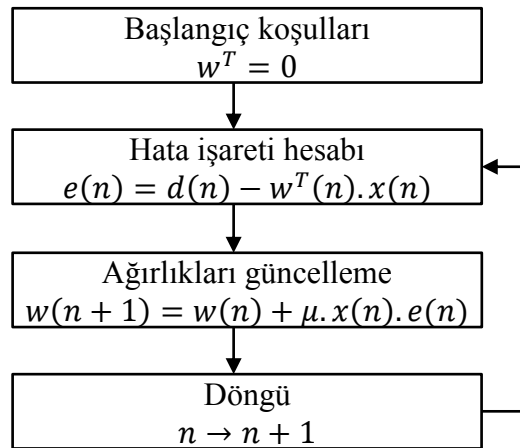
$$w(n+1) = w(n) + \mu \cdot x(n) \cdot e(n) \quad (3.39)$$

Algoritmada kullanılan μ adım büyüklüğü parametresi $w(n)$ ağırlıklarının yakınsamasını kontrol eder. Bu algoritma sürekli olup her örneklenen giriş işaretinden

sonra ağırlıklar güncellenir. μ adım büyüklüğü parametresinin değeri öz ilinti matrisinin (R_{xx}) öznitelik yayılımına bağlıdır. Öznitelik yayılımı arttıkça algoritmanın yakınsamasını etkileyen adım büyüklüğü parametresinin değeri azalır. Öznitelik yayılımının artması girişin artması ile mümkündür. Ayrıca öznitelik yayılımı sisteme gelen işaretlerin gücüne de bağlıdır. Adım büyüklüğü parametresi aşağıdaki aralıkta seçilebilir:

$$0 \leq \mu \leq \frac{2}{\lambda_{max}} \quad (3.40)$$

Yukarıdaki eşitlikte (3.40), μ adım büyüklüğü bir noktada birleştirme oranını belirler. Öz ilinti matrisinden elde edilecek öznitelik değerlerinin en büyüğü (λ_{max}) kullanılarak μ adım büyüklüğü parametresi sınırlandırılır [88,89]. LMS algoritmasının başında ağırlık vektörleri rastgele seçilir ya da sifıra eşit alınır. Ancak ağırlık vektörleri daha uygun bir değer seçilerek algoritma başlatılırsa yakınsama daha hızlı olur.



Şekil 3.30. LMS algoritması akış şeması [90].

3.9.2. RLS (Özyinelemeli En Küçük Kareler) Algoritması

Adaptif algoritmalarından biri olan LMS algoritmasında, adaptasyon parametresi, adım büyüklüğü μ sabit alındığından dolayı çalışma bölgesi sınırlanmaktadır. Bu sınırlamayı azaltmak için yapılan çalışmalar sonucunda geliştirilen özyinelemeli en küçük kareler (Recursive Least Squares, RLS) algoritması, LMS'ye göre daha geniş bir çalışma bölgesi sağlamaktadır. Bu algoritma sürekli bir adaptif yaklaşım olup, ağırlıklar her örneklenen verinin ardından güncellenir. İşaret ortamıyla ilişkili istatistik durağan ise, bu sürekli adaptasyon işlemi uygun sonuçlar vermektedir. Ayrıca normal çalışma koşullarında RLS algoritmasının yakınsaması, LMS'ye göre daha yüksektir.

RLS algoritması aşağıdaki eşitliklerle verilen “Matrix Inversion Formula” denklemleri temel alınarak oluşturulmuştur. Korelasyon matrisi yaklaşımı için ağırlıklandırılmış toplamları kullanır, matrisin tersi yinelemeli bir şekilde elde edilir ve ağırlıklar tekrar güncellenir. RLS algoritması aşağıdaki denklemler ile ifade edilir:

Kazanç:

$$k(n) = \frac{\lambda^{-1} \cdot R^{-1}(n-1) \cdot x(n)}{1 + \lambda^{-1} \cdot x^T \cdot R^{-1}(n-1) \cdot x(n)} \quad (3.41)$$

$R(n)$ matrisinin özyinelemeli ifadesi:

$$R^{-1}(n) = \lambda^{-1} \cdot R^{-1}(n-1) - \lambda^{-1} \cdot x^T \cdot R^{-1}(n-1) \quad (3.42)$$

Hata:

$$e(n) = d(n) - w^T(n-1) \cdot x(n) \quad (3.43)$$

Ağırlık:

$$w(n) = w(n-1) + k(n) \cdot e(n) \quad (3.44)$$

Burada (3.41) ve (3.42) eşitliklerinde geçen λ unutma faktörüdür ve (3.45)'daki değerler arasında seçilebilir. Durağan (stationary) sinyaller için yapılan çalışmalarda

unutma faktörü genelde 1'e yakın seçilir. Seçilecek sinyaller durağan yapıda değilse λ değeri en yüksek başarımla elde edilecek şekilde deneysel olarak belirlenir.

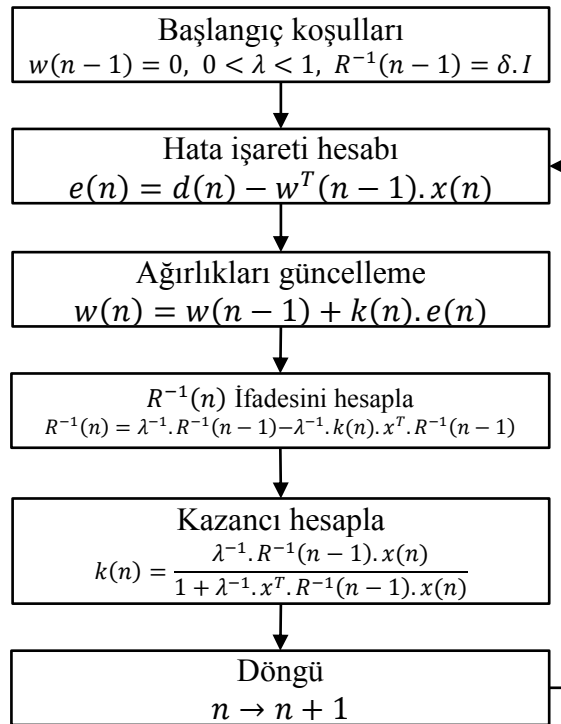
$$0 < \lambda < 1 \quad (3.45)$$

Başlangıç koşullarında, ağırlık katsayılarının ve korelasyon matrisinin ilk değerlerinin seçimi algoritmanın yakınsamasını etkiler. Korelasyon matrisinin ilk değerlerinin belirlenmesinde (3.46)'daki eşitlik kullanılır.

$$R^{-1}(n-1) = \delta \cdot I \quad (3.46)$$

Son eşitlikte belirtilen I , köşegen elemanları 1 olan köşegen matrisidir. δ ise sinyalin varyansı ile ilişkili bir sabittir. Genellikle bu sabit uygulamalarda eşitlik (3.47)'deki gibi seçilir.

$$\delta > 100 \cdot \sigma_x^2 \quad (3.47)$$



Şekil 3.31. RLS algoritması akış şeması [90].

3.10. Görüntü İyileştirme ve Geliştirme İşlemleri

Görüntü iyileştirmek için çeşitli filtreler kullanılarak gürültüden arındırılması, görüntü parlaklığının ayarlanması ile gerçekleştirilir. Aynı zamanda görüntüye gerekli durumlarda çeşitli algoritmalarla gürültüler eklenebilir.

Gaussian filtre için diğerlerinden farklı olarak *sigma* parametresinin de belirtilmesi gerekir. Medyan filtre maske matrisi içerisinde kalan değerlerin medyanının ortadaki elemana yerleştirilmesi ile uygulanır. Ortalama filtre ise maske içerisinde kalan değerlerin ortalamasının maske içerisindeki orta elemana yerleştirilmesi ile gerçekleşir.

Adaptif filtrede ise maske içerisindeki değerlerin varyansının yerleştirilmesi ile uygulanır. Aslında bir fourier dönüşümü olan Gaussian filtre Gauss çan eğrisi formu ile sonsuz bir transfer fonksiyonunun kapalı bir alan olan maske matrisi içerisinde uygulanması ile elde edilir.

Medyan filtre için *medfilt2* fonksiyonuna, adaptif filtre için *wiener2* fonksiyonuna görüntü matrisi ve maske boyutu değerleri gönderilir. Ortalama ve gaussian filtreleme için *fspecial* fonksiyonu çağırılır. Uygulanacak filtrenin türü de maske boyutu ve gaussian filtre için *sigma* değeri ile bir parametre olarak bu fonksiyona gönderilir. *Fspecial* fonksiyonunun geri dönüş değeri bir filtre matrisidir. Bu filtre *imfilter* fonksiyonu kullanılarak görüntüye uygulanır.



Şekil 3.32. %20 oranında tuz-biber gürültülü gri seviyeli resim (gR = imnoise(resim,'salt & pepper',.2);)



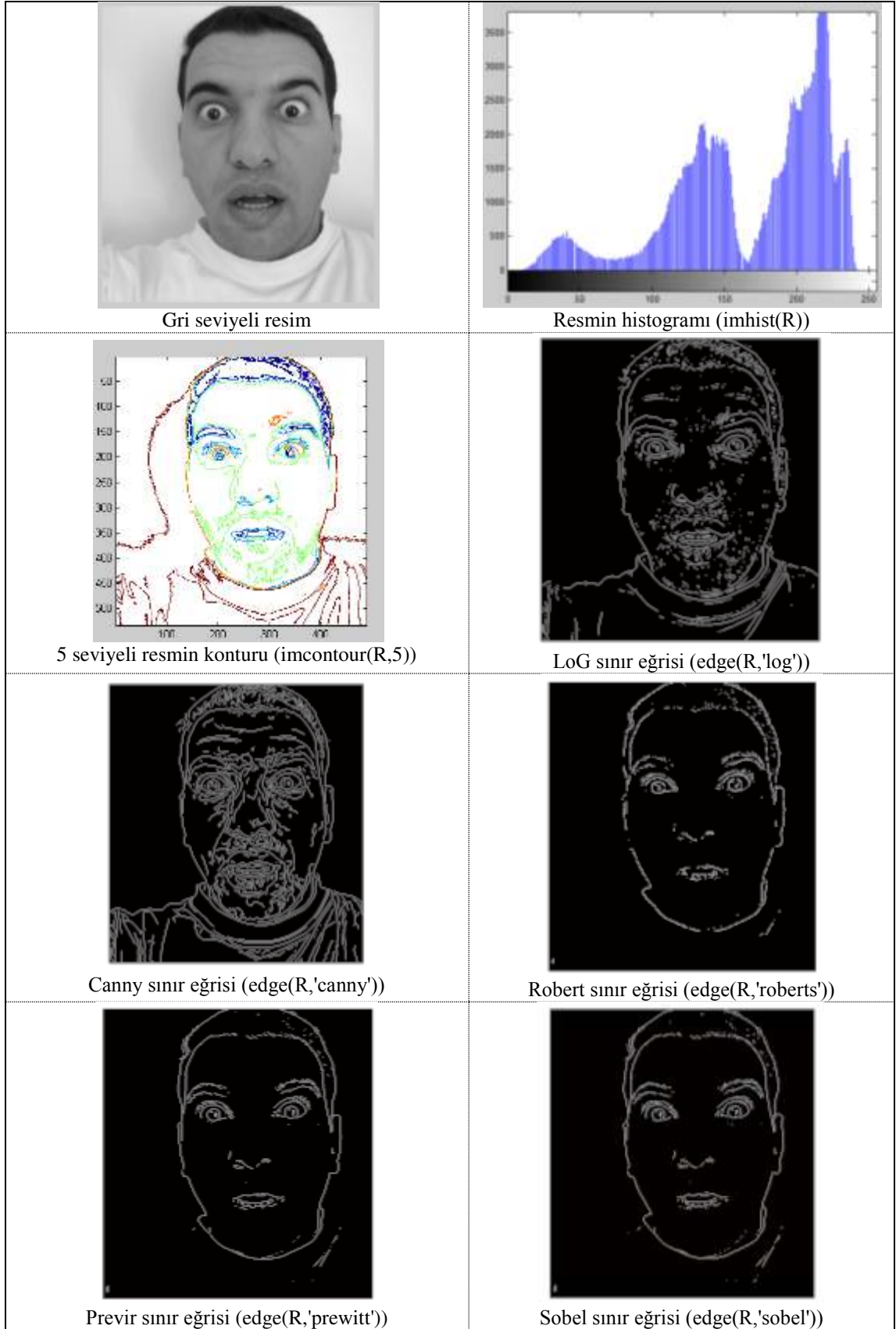
Şekil 3.33. 5x5 medyan filtre ile gürültüden arındırılmış ($mR = \text{medfilt2}(gR,[5\ 5]);$)



Şekil 3.34. 7x7 ortalama filtre ile gürültüden arındırılmış ($aR = \text{imfilter}(mR,\text{fspecial('average',[7\ 7])});$)

3.11. Sınır Eğrisi Çıkartmak

Görüntünün sınır eğrisi yerel süreklilik gösteren ve çevresine göre belirginleşmiş bölgelerin değişim noktalarından oluşur. Görüntü sınırı 5 farklı algoritma kullanılarak çıkartılabilmektedir. Bunlar; Sobel, Prewitt, Roberts, Log (Laplacian Of Gaussian) ve Canny algoritmalarıdır. Sınır grafiği siyah zemin üzerine beyaz olarak çizilir. Sınır çıkartma işlemleri Matlab varsayılan ayarları olarak yatay ve dikey yönde görüntüye göre belirlenen otomatik eşik değeri ile yapılır. Görüntü matrisi, kullanılacak yöntem *edge* fonksiyonuna parametre olarak gönderilir. Geri dönüş değeri olarak sınır görüntüsü 0 ve 1'lerden oluşan bir matris olarak elde edilir.



Şekil 3.35. Çeşitli kenar tespit algoritmalarının çıkış resimleri

3.12. Kullanılan Veritabanı

Uluslararası kabul görmüş ve yüz ifadesi tanıma uygulamalarında kullanılmak üzere oluşturulmuş olan “The Japanese Female Facial Expression” (JAFFE) veritabanı kullanılmıştır. JAFFE, renkli resmi olmayan gri seviyeli 256x256 boyutundaki vesikalık resimlerden oluşmaktadır. Veritabanı, 10(on) Japon bayanın ön cepheden çekilmiş olup her kişinin sakin, mutlu, sinirli, tiksinişmiş, şaşırılmış, üzgün ve korkmuş olmak üzere 7(yedi) adet yüz ifadesini barındırmaktadır. Her yüz ifadesi için de 3 veya 4 adet farklı seviyelerde (örneğin az tiksinişmiş, orta tiksinişmiş ve çok tiksinişmiş gibi) çekilmiş resim olmak üzere toplam 213 adet yüz ifadesi resmi ihtiva etmektedir. Tüm veritabanının ifade dağılımı ise şöyle; korkmuş 32, mutlu 31, sakin 30, şaşırılmış 30, sinirli 30, üzgün 31 ve tiksinişmiş ifade sayısı ise 29 adettir.



Şekil 3.36. JAFFE veritabanından örnek resimler

Diğer veritabanı ise amatör olarak çekimi yapılmış resim seti olup her ifadeyi (7 adet ifade) barındırmaktadır.



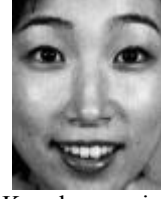
Şekil 3.37. Amatör veritabanından örnek resimler

Test için ise 2 farklı resim seti kullanıldı. Birincisi JAFFE veritabanından rastgele 20 adet resim seçilmiş olup seçilen resimler eğitim setinden çıkarılarak 213 adet resimden kalan 193 adet resim sistemin eğitilmesinde kullanıldı. İkincisi ise amatör olarak çekilmiş 25 adet resimlerden oluşmuştur. Ancak ikincisinde hem eğitimdeki 213 adet JAFFE veritabanındaki resimler ile testte kullanılan 25 adet amatör resimleri resmin arka planındaki değişik gürültülerden kurtulmak amacıyla yüz hatları korunarak kırılmıştır.

Amatör resimlerin testlerinde yaşanan sorunları azaltmak için orijinal resmin arka planındaki değişik çeşitli nesne ve gölgelerin oluşturduğu farklı yoğunlukları kaldırmak için yüzün bulunduğu alanda kırılma işlemleri yapıldı. Aşağıdaki şekilde görüldüğü üzere arka planının çoğu kaldırılmış durumdadır. Böylece kırılmış resim ile orijinal resimden daha yüksek başarı sağlanmış oldu. Aynı işlemler test resimleri üzerinde de yapıldı.



Orijinal resim



Kırpılmış resim



Orijinal resim



Kırpılmış resim

Şekil 3.38. Kırpılmış resimlerin örneği

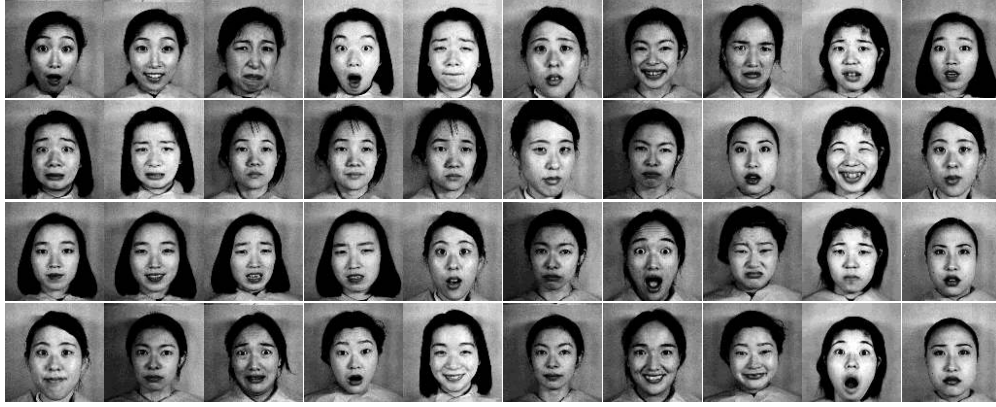
4. UYGULAMALAR VE SONUÇLARI

4.1. Yüz İfade Tanıma Sistemi

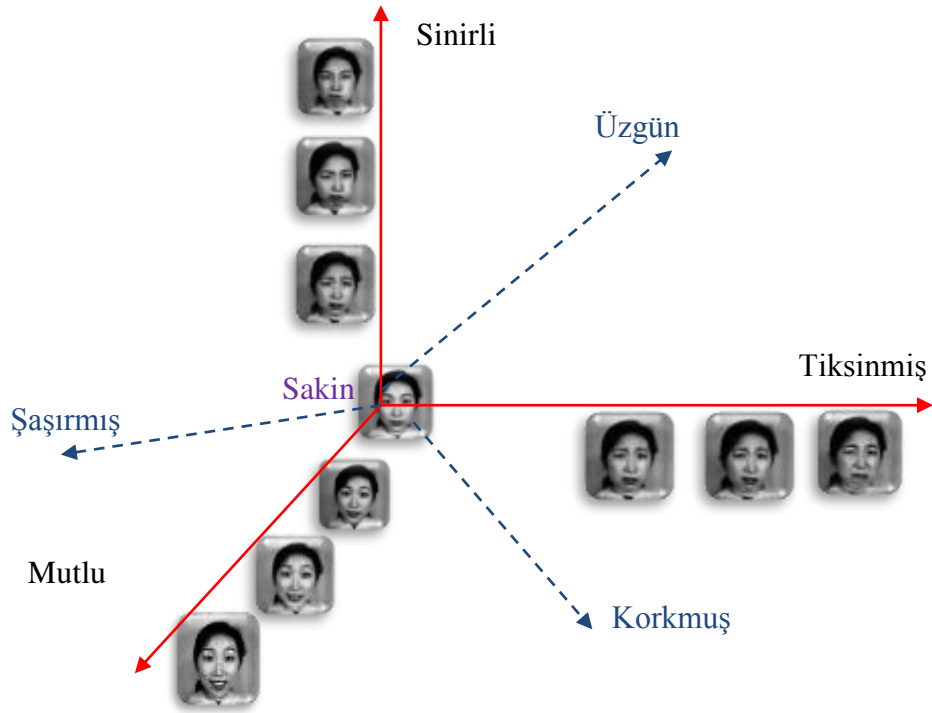
Bu çalışmada, yüz ifadelerini sınıflandırmak için özyüzler kullanılmaktadır. Yüz ifadesi bazı sağduyulu (öfke, mutluluk, iğrenme veya üzüntü gibi) sınıflara sınıflandırılabilceğı kabul edilmiştir. Eğer hiçbir ifade yoksa, ifade temel olarak “sakin”(nötr) kabul edilmektedir. Sakin (nötr) ifadeden farklılık seviyesi ile belirli bir ifade yoğunluęu belirlenebilir. Bu yolla yüz ifadelerinin temsil edilmesinin birçok avantajı vardır. Birincisi, iki bilginin kullanımıyla birçok tür ifadeyi temsil ettirilebilir; bir ifadeye ait sınıf ve ifade yoğunluęu. Üç boyutlu bir eksenle orijinin “sakin”(nötr) ifadeyi temsil etmektedir ve yoğunluęa baęlı olarak her eksen belirli bir ifadeye sınıflandırılır. İkincisi, iki veya daha fazla ifadelerin karışımıyla bir ifade belirlenebilir. Örneęin ifade, %60 mutlu, %20 sakin %10 sinirli ve %10 iğrenmiş gibi yüzdeliklerle de ifade edilebilir. Ancak bu uygulamada karışımlar ifade edilmeyip sadece bir ifade belirtilmektedir.

Veritabanı 2 adet klasörden oluşmaktadır; bir tanesinde eğitim için dięeri ise test için kullanılmaktadır. Eğitim ve test klasörlerindeki resim sayısına baęlı olarak uygulamanın hızını direkt olarak etkilemektedir. Ancak eğitim resimlerinin çeşitlilięi sistemin başarımı açısından çok önemlidir.

Bu çalışmada yüz ifadesi olarak; sakin, mutlu, sinirli, tiksiniş, şaşırış, üzün ve korkmuş olmak üzere 7 (yedi) adet ifade üzerinde çalışılmıştır. Bu tez çalışmasının amacı, yüz görüntüleri kullanılarak imge özelliklerinin durumlarının deęerlendirilmesi ve her yüz durumunun bir ifade ile ilişkilendirilmesidir. Bu sayede özellikle iletişim olanakları sınırlı felçli veya konuşamayan hastaların temel isteklerini, basit yüz ifadeleri aracılığıyla sisteme iletmesi ve isteklerin otomatik olarak işlenerek ilgili kişiye bildirilmesi hedeflenmiştir.



Şekil 4.1. JAFFE veritabanından örnek resimler



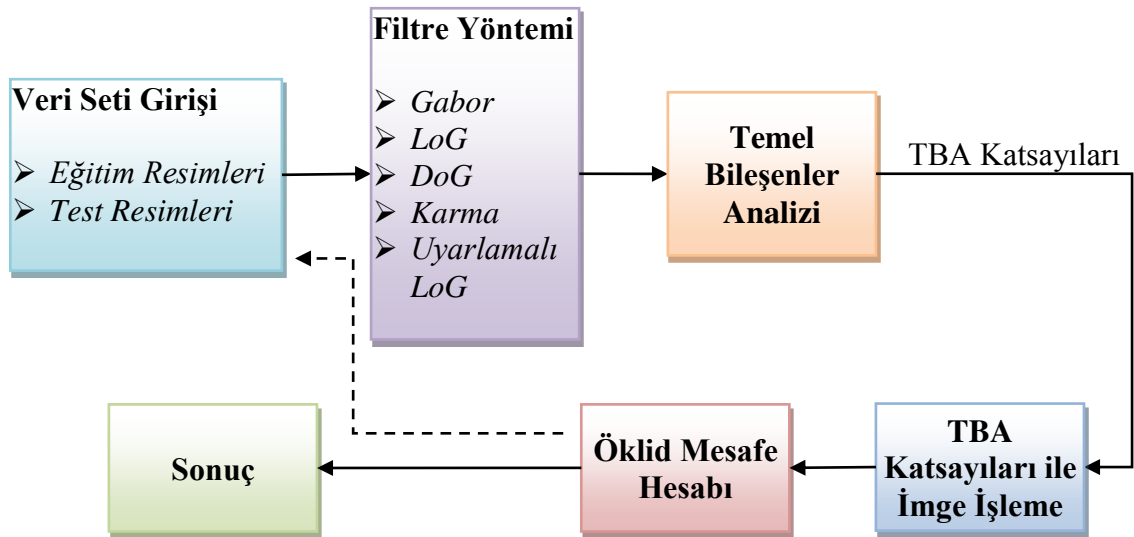
Şekil 4.2. Yüz ifade eksen

Mimik tanıma için sistem özet olarak aşağıdaki adımlardan oluşmaktadır:

- 1) Öncelikle ifade tanımlamada kullanılacak eğitim resimlerin giriş işlemleri yapılmaktadır.

- 2) Her bir eğitim resmi bir filtre yöntemi (Gabor, LoG, DoG, Gabor-LoG-DoG karışımı ve uyarlamalı LoG) ile işlenmektedir.
- 3) Tüm eğitim resimlerin ortalama değerleri alınır.
- 4) Ortalaması alınan eğitim resimleri Temel Bileşen Analizi ile analizi yapılır.
- 5) Temel Bileşen Analizin katsayıları test aşaması için hazırlanır.
- 6) 1-4 adımları test resimleri için uygulanır.
- 7) L2 normuna göre Öklid uzaklık mesafe hesabı yapılır.
- 8) Her bir test resmin “sakin” yüz ifadesinden minimum mesafedeki uzaklık değeri bulunur.
- 9) En kısa mesafe olan resmin yüz ifadesi test resmi için atanır ve çıkışa verilir.
- 10) Her bir test resmi için yüz ifadesi bulunan sonuç dosyaya yazılır.

Yukarıdaki adımların şematik olarak gösterimi aşağıdaki akış şemasındadır.



Şekil 4.3. İfade tanıma uygulamasının akış şeması

Bu çalışma kapsamında her birinde farklı yöntem kullanılan 5(beş) farklı uygulama yapılmıştır. Çalışılan tüm uygulamalarda bölüm 3.12’de belirtilen The Japanese Female Facial Expression” (JAFPE) veritabanı kullanılmıştır. JAFPE veritabanından rastgele 20 adet resim seçilmiş olup seçilen resimler eğitim setinden çıkarılarak 213 adet resimden

kalan 193 adet resim sistemin eğitilmesinde kullanıldı. Tüm veritabanının ifade dağılımı ise şöyle; korkmuş 32, mutlu 31, sakin 30, şaşırmış 30, sinirli 30, üzgün 31 ve tiksiniş ifade sayısı ise 29 adettir. Diğer veritabanı ise amatör olarak çekimi yapılmış resim seti olup her ifadeyi (7 adet ifade) barındırmaktadır. Amatör resimlerin testlerinde yaşanan sorunları azaltmak için orijinal resmin arka planındaki değişik çeşitli nesne ve gölgelerin oluşturduğu farklı yoğunlukları kaldırmak için yüzün bulunduğu alan kırılma işlemleri yapıldı. Böylece kırılmış resim ile orijinal resimden daha yüksek başarı sağlanmış oldu. Aynı işlemler test resimleri üzerinde de yapıldı.

4.2. Gabor Filtresi Uygulaması

Bu çalışmada, insanın içinde bulunabileceği 6 ifade durumunun (öfke, iğrenme, korku, mutluluk, üzüntü ve şaşkınlık) analizi gerçekleştirilmiştir. Bu amaçla, her bir ifade için alınan görüntülerin öznitelikleri Gabor filtreleri kullanılarak çıkartılmıştır.

Hem eğitim setin hem de test setin resimleri boyut üzerinde değişiklik yapılmamış olup her bir resmin orijinal boyutu 256x256 kullanıldı. Eğitim setindeki her bir resim tek tek alınıp Gabor filtresinden geçilerek öznitelik vektörüne aktarıldı ve sonuçta eğitim için 256x193 boyutlu, test için ise 256x20 boyutlu vektör elde edilmiş oldu. Temel Bileşenler Analizi ile analizi yapıldıktan sonra Öklid uzaklığı (L2 normu) ile mesafe hesabı yapılarak sınıflandırma işlemi tamamlanır. Her bir test resmi otomatik olarak dosyaya ifadesiyle beraber yazdırma işlemi yapıldı. Sistemin tüm işlemleri otomatik olarak çalışmaktadır.

Uygulamaların kullanıldığı bilgisayarın konfigürasyon bilgisi ise şöyle: Core2 Duo 2.2 GHz işlemci, 3 GB ram, 320 GB HDD ve 32 bit Windows 7 ultimate işletim sistemi özellikli dizüstü bilgisayar.



Şekil 4.4. Gabor filtresi uygulaması için örnek giriş resmi

Tablo 4.1. Gabor filtresi uygulaması sonuçları

	<i>Gabor</i>	<i>Gabor + Keskinlik</i>	<i>Gabor + Ortalama</i>	<i>Gabor + Keskinlik + Ortalama</i>
<i>Başarı oranı (%)</i>	90	90	90	90
<i>Çalışma süresi (sn)</i>	41	40	43	41
<i>Çıkış imge örneği</i>				

Sonuç olarak Gabor filtre tabanlı sistemin yüz ifadesinin analiz başarı oranı tek başına olsa bile, ek yöntem veya metot olmasa da aynı değere ulaşabilmektedir. Her ne kadar keskinlik filtresi ile kullanıldığında çalışma süresi en az olsa da aradaki farkın kayda değer alınabilecek düzeyde değildir.

4.3. Laplacin of Gaussian (LoG) Uygulaması

Bu çalışmada, insanın içinde bulunabileceği 6 ifade durumunun (öfke, iğrenme, korku, mutluluk, üzüntü ve şaşkınlık) analizi gerçekleştirilmiştir. Bu amaçla, her bir ifade için alınan görüntülerin öznitelikleri geliştirilmiş Gabor (LoG) kullanılarak çıkartılmıştır. Temel Bileşenler Analiz yöntemi ile boyut indirgenip analizi yapılmış ve Öklid uzaklık hesabı yapılarak sınıflandırma işlemi yapılmıştır. Kullanılan yöntemlerin ifade sınıflandırılması üzerindeki etkileri incelenerek karşılaştırma yapılmıştır.

Hem eğitim setin hem de test setin resimleri boyut üzerinde değişiklik yapılmamış olup her bir resmin orijinal boyutu 256x256 kullanıldı. Eğitim setindeki her bir resim tek tek alınıp geliştirilmiş Gabor (LoG) filtresinden geçilerek öznitelik vektörüne aktarıldı ve sonuçta eğitim için 256x193 boyutlu, test için ise 256x20 boyutlu vektör elde edilmiş oldu. Temel Bileşenler Analizi ile analizi yapıldıktan sonra Öklid uzaklığı (L2 normu) ile mesafe hesabı yapılarak sınıflandırma işlemi tamamlanır. Her bir test resmi otomatik olarak

dosyaya ifadesiyle beraber yazdırma işlemi yapıldı. Sistemin tüm işlemleri otomatik olarak çalışmaktadır.

Uygulamaların kullanıldığı bilgisayarın konfigürasyon bilgisi ise şöyle: Core2 Duo 2.2 GHz işlemci, 3 GB ram, 320 GB HDD ve 32 bit Windows 7 ultimate işletim sistemi özellikli dizüstü bilgisayar.



Şekil 4.5. LoG uygulaması için örnek giriş resmi

Tablo 4.2. LoG uygulaması sonuçları

	<i>LoG</i>	<i>LoG + Keskinlik</i>	<i>LoG + Ortalama</i>	<i>LoG + Keskinlik + Ortalama</i>
<i>Başarı oranı (%)</i>	85	85	85	85
<i>Çalışma süresi (sn)</i>	34	26	30	27
<i>Çıkış imge örneği</i>				

Sonuç olarak geliştirilmiş Gabor (LoG) filtre tabanlı sistemin yüz ifadesinin analiz başarı oranı tek başına olsa bile, ek yöntem veya metot olmasa da aynı değere ulaşabilmektedir. Her ne kadar keskinlik filtresi ile kullanıldığında çalışma süresi en az olsa da aradaki farkın kayda değer alınabilecek düzeyde değildir.

4.4. Difference of Gaussians (DoG) Uygulaması

Bu çalışmada, insanın içinde bulunabileceği 6 ifade durumunun (öfke, iğrenme, korku, mutluluk, üzüntü ve şaşkınlık) analizi gerçekleştirilmiştir. Bu amaçla, her bir ifade için alınan görüntülerin öznitelikleri geliştirilmiş Gabor (DoG) kullanılarak çıkartılmıştır. Temel Bileşenler Analiz yöntemi ile boyut indirgenip analizi yapılmış ve Öklid uzaklık hesabı yapılarak sınıflandırma işlemi yapılmıştır. Kullanılan yöntemlerin ifade sınıflandırılması üzerindeki etkileri incelenerek karşılaştırma yapılmıştır.

Hem eğitim setin hem de test setin resimleri boyut üzerinde değişiklik yapılmamış olup her bir resmin orijinal boyutu 256x256 kullanıldı. Eğitim setindeki her bir resim tek tek alınıp geliştirilmiş Gabor (DoG) filtresinden geçilerek öznitelik vektörüne aktarıldı ve sonuçta eğitim için 256x193 boyutlu, test için ise 256x20 boyutlu vektör elde edilmiş oldu. Temel Bileşenler Analizi ile analizi yapıldıktan sonra Öklid uzaklığı (L2 normu) ile mesafe hesabı yapılarak sınıflandırma işlemi tamamlanır. Her bir test resmi otomatik olarak dosyaya ifadesiyle beraber yazdırma işlemi yapıldı. Sistemin tüm işlemleri otomatik olarak çalışmaktadır.

Uygulamaların kullanıldığı bilgisayarın konfigürasyon bilgisi ise şöyle: Core2 Duo 2.2 GHz işlemci, 3 GB ram, 320 GB HDD ve 32 bit Windows 7 ultimate işletim sistemi özellikli dizüstü bilgisayar.



Şekil 4.6. DoG uygulaması için örnek giriş resmi

Tablo 4.3. DoG uygulaması sonuçları

	<i>DoG</i>	<i>DoG + Keskinlik</i>	<i>DoG + Ortalama</i>	<i>DoG + Keskinlik + Ortalama</i>
<i>Başarı oranı (%)</i>	75	50	80	70
<i>Çalışma süresi (sn)</i>	39	40	40	41
<i>Çıkış imge örneği</i>				

Sonuç olarak geliştirilmiş Gabor (DoG) filtre tabanlı sistemin yüz ifadesinin analiz başarı oranı tek başına olsa bile en az çalışma zamanıyla yüksek bir başarı elde edebilmektedir. Ortalama kenar tespit fonksiyonu ile kullanıldığında, keskinlik kenar tespit fonksiyonu ile kullanıldığından daha iyi bir başarı elde edilmiştir.

4.5. Karma Yöntemlerin Uygulaması

Bu çalışmada, insanın içinde bulunabileceği 6 ifade durumunun (öfke, iğrenme, korku, mutluluk, üzüntü ve şaşkınlık) analizi gerçekleştirilmiştir. Bu amaçla, her bir ifade için alınan görüntülerin öznetelikleri Gabor dalgacıkları kullanılarak çıkartılmıştır. Gabor, LoG_Gabor ve DoG_Gabor beraber kullanılarak sistemin başarımının ölçümü için ayrıntılı testler yapılmış ve yüksek başarımla güvenilir olarak çalıştığı görülmüştür. Kullanılan yöntemlerin ifade sınıflandırılması üzerindeki etkileri incelenerek detaylı bir karşılaştırma yapılmıştır.

Uygulama 2 aşamada gerçekleştirilmiştir. Her aşamasında ise çeşitli filtreleme fonksiyonu kullanılarak 24 adet uygulama gerçekleştirilmiş olup uygulamalar başarı oranı ve süreleri bakımından karşılaştırılmıştır. Sonuçların değerlendirilmesi neticesinde LoG_Gabor ile Gabor filtrelerinin beraber kullanıldığı uygulamalarda yüksek başarı oranı sağlamış olduğu gözlenmiştir.

Uygulamaların kullanıldığı bilgisayarın konfigürasyon bilgisi ise şöyle: Core2 Duo 2.2 GHz işlemci, 3 GB ram, 320 GB HDD ve 32 bit Windows 7 işletim sistemi özellikli dizüstü bilgisayar.

4.5.1. Birinci Aşama Uygulama Sonuçları

Bu aşamada 193 adet resimden oluşan JAFFE veri setini eğitim olarak kullanıldı ve test için ise yine eğitim setinde olmayıp daha önce rastgele seçilen 20 adet JAFFE veritabanına ait resimlerdir.

Aşağıdaki 3 tabloda da görüldüğü üzere; ilk tabloda fonksiyonların ortalama başarı oranları, ikinci tabloda her bir uygulamanın örnek çıkış imgesi ve üçüncü tabloda ise ortalama çalışma süreleri bulunmaktadır. Gabor filtresiyle belirli başarı oranını yakalamakla birlikte LoG ile kullanıldığında başarı oranını bariz bir şekilde arttığı gözlenmektedir. Buna ek olarak çalışma süresi ise Gabor filtresinin tek başında kullanıldığından daha az bir sürede çalıştığı görülmektedir.



Şekil 4.7. Birinci aşama için örnek giriş resmi

Tablo 4.4. Birinci aşama uygulamasının ortalama başarı oranları

Ana Fonksiyonlar		Yardımcı Filtreler (%)		
		<i>Keskinlik</i>	<i>Ortalama</i>	<i>Keskinlik+ortalama</i>
<i>Gabor</i>	85	85	85	85
<i>LoG</i>	80	80	85	80
<i>LoG + Gabor</i>	88	86	88	90
<i>DoG</i>	75	80	75	80
<i>DoG + LoG</i>	75	70	75	70
<i>DoG + LoG + Gabor</i>	80	85	80	85

Tablo 4.5. Birinci aşama uygulamasının çıkış imge örneği

Ana Fonksiyonlar		Yardımcı Filtreler		
		<i>Keskinlik</i>	<i>Ortalama</i>	<i>Keskinlik+ Ortalama</i>
<i>Gabor</i>				
<i>LoG</i>				
<i>LoG + Gabor</i>				
<i>DoG</i>				
<i>DoG + LoG</i>				
<i>DoG + LoG + Gabor</i>				

Tablo 4.6. Birinci aşama uygulamasının ortalama çalışma süreleri

Ana Fonksiyonlar		Yardımcı Filtreler (sn)		
		<i>Keskinlik</i>	<i>Ortalama</i>	<i>Keskinlik+ortalama</i>
<i>Gabor</i>	80	73	67	66
<i>LoG</i>	34	39	36	31
<i>LoG + Gabor</i>	58	60	60	64
<i>DoG</i>	25	28	31	29
<i>DoG + LoG</i>	39	40	40	43
<i>DoG + LoG + Gabor</i>	68	64	65	64

4.5.2. İkinci Aşama Uygulama Sonuçları

İkinci aşamada ise JAFFE veritabanındaki resimlerin tamamı olan 213 adet resimler eğitim setinde kullanıldı. Test setinde ise rastgele seçilmiş 25 adet amatör resimlerden oluşmaktadır. Ancak bu aşamada ek olarak hem eğitim hem de test setindeki tüm resimler ön işleme alınıp yüz hatlarının kırılması işlemi gerçekleştirilmiştir.

Bu aşamadaki başarı oranı ise birinci aşamada olduğu gibi LoG ve Gabor'un beraber kullanıldığı uygulamarda başarının daha yüksek olduğu gözlemlenmektedir. Fakat işlem süresi bakımından birinci aşamadaki gibi olmamakla beraber iyi düzeyde olduğu söylenebilir.

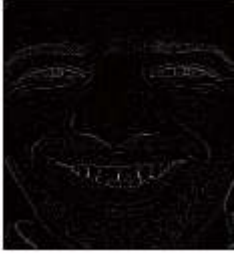


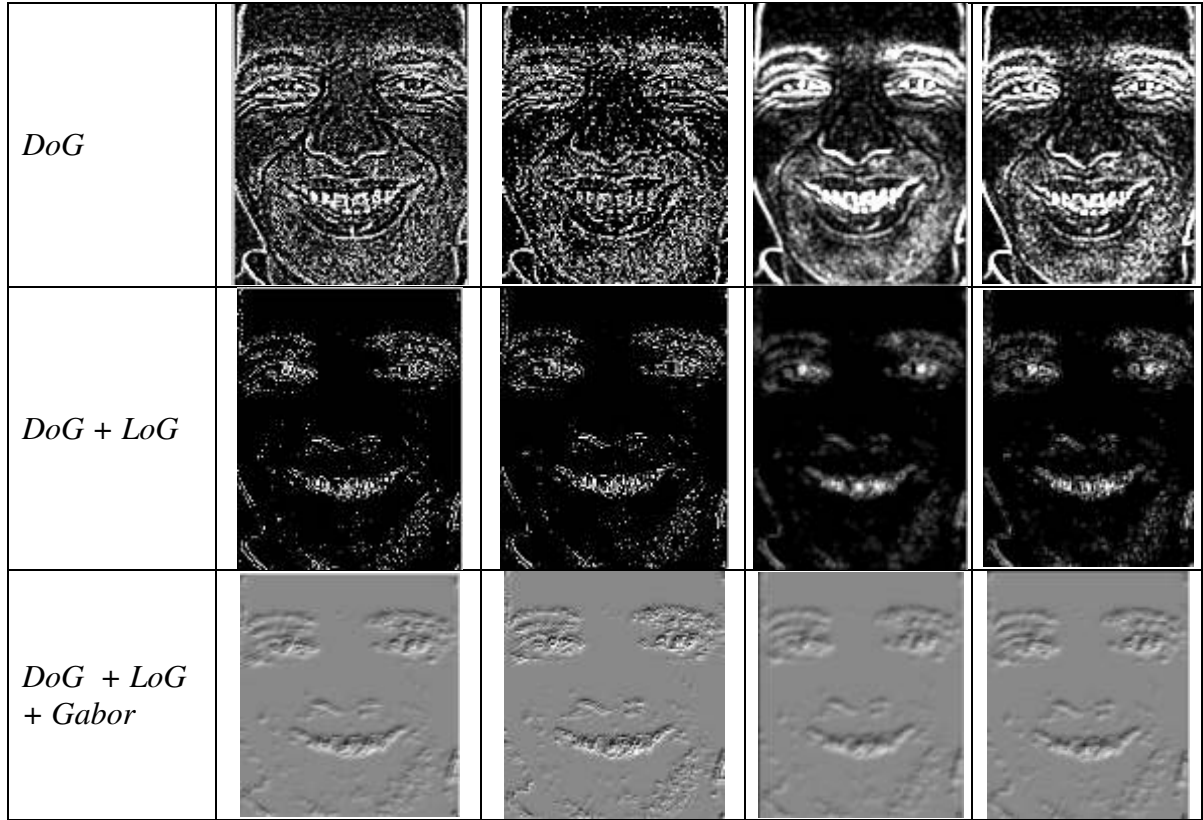
Şekil 4.8. İkinci aşama için örnek giriş resmi

Tablo 4.7. İkinci aşama uygulamasının ortalama başarı oranları

Ana Fonksiyonlar		Yardımcı Filtreler (%)		
		<i>Keskinlik</i>	<i>Ortalama</i>	<i>Keskinlik+ortalama</i>
<i>Gabor</i>	48	44	48	52
<i>LoG</i>	20	26	25	32
<i>LoG + Gabor</i>	84	80	84	88
<i>DoG</i>	24	20	32	20
<i>DoG + LoG</i>	24	24	16	16
<i>DoG + LoG + Gabor</i>	80	85	80	12

Tablo 4.8. İkinci aşama uygulamasının çıkış imge örneği

Ana Fonksiyonlar		Yardımcı Filtreler		
		<i>Keskinlik</i>	<i>Ortalama</i>	<i>Keskinlik+Ort.</i>
<i>Gabor</i>				
<i>LoG</i>				
<i>LoG + Gabor</i>				

Tablo 4.8'in devamı**Tablo 4.9.** İkinci aşama uygulamasının ortalama çalışma süreleri

Ana Fonksiyonlar		Yardımcı Filtreler (sn)		
		<i>Keskinlik</i>	<i>Ortalama</i>	<i>Keskinlik+ortalama</i>
<i>Gabor</i>	20	23	22	24
<i>LoG</i>	10	10	10	10
<i>LoG + Gabor</i>	21	24	27	22
<i>DoG</i>	12	12	12	13
<i>DoG + LoG</i>	13	12	12	15
<i>DoG + LoG + Gabor</i>	68	64	65	28

Sonuç olarak LoG_Gabor filtre tabanlı yüz ifadesinin analiz başarı oranlarının daha yüksek olduğu ve çalışma süresinin ortalama değerde olduğu anlaşılmıştır. Veri setinin daha büyük olduğu sistemlerde düşünüldüğünde LoG_Gabor tabanlı sistemin daha optimal bir çözüm olacaktır.

4.6. Adaptif Yaklaşım Uygulaması

Bu çalışmada, insanın içinde bulunabileceği 6 ifade durumunun (öfke, iğrenme, korku, mutluluk, üzüntü ve şaşkınlık) analizi gerçekleştirilmiştir. Bu amaçla, her bir ifade için alınan görüntülerin öznitelikleri, geliştirmiş adaptif Gabor-LoG (Laplacian of Gaussians) kullanılarak çıkartılmıştır. Temel Bileşenler Analiz yöntemi ile boyut indirgenip işlem yapılmış ve Öklid uzaklık hesabı yapılarak sınıflandırma işlemi yapılmıştır. Kullanılan yöntemlerin ifade sınıflandırılması üzerindeki etkileri incelenerek karşılaştırma yapılmıştır. Uygulama sonuçları %90 gibi bir doğruluk elde edilmiştir.

Programın bütün kısımları MATLAB™ paket programıyla yazılmıştır. Veritabanındaki resimlerin, kullanılacak metottan önce hangi parametreler ile çalışacağı ön kayıt programında gerçekleştirilmiştir. Ardından her bir resmin Laplacian of Gaussiuan (LoG)'ın imgesi, LMS ve RLS metotlarına göre katsayılarını hesaplayan ifade tanıma programı oluşturulmuştur. Daha sonra herhangi bir kişiye ait bir resim girişinde, bu ifadenin ne olduğunu her bir metot için bulan ana program yazılmıştır. Karar aşamasında Euclid mesafesi formüllerinden yararlanılmıştır. Euclid mesafesi sonucunda bulunan minimum değeri sağlayan ifade, sonuç olarak atanmış ve bu şekilde ifade tanıma uygulaması gerçekleştirilmiştir.

Gerçekleştirilen çalışma sonucunda, giriş parametrelerinin sabit tutulduğu uygulamada en yüksek başarı Laplacian of Gaussiuan filtresi ile elde edilmiş ve başarı oranı %80 çıkmıştır. LMS ve RLS algoritmalarının başarı düzeyleri daha düşük gözlenmiştir. Daha sonraki uygulamada, giriş parametrelerinin değişken olarak seçilebilmesi ile başarı düzeylerinin belirgin oranda değiştiği belirlenmiştir. LMS ve RLS metotları ile doğru giriş parametreleri seçildiğinde başarı oranının %90'a çıkabileceği görülmüştür.

Buradan gerçekleştirilen yüz ifade tanıma uygulaması için şu sonuçlar elde edilebilir:

✓ Giriş parametreleri adaptif yöntemin başarımını ciddi oranda belirlemektedir.

✓ LMS metodu başlangıçta ihtiyaç duyulan μ adım büyüklüğü parametresinin tespitinin zor olması nedeniyle RLS yöntemine göre dışarıya daha fazla bağımlıdır.

✓ RLS metodu LMS metoduna göre daha fazla hesapsal karmaşıklığa sahip olması nedeniyle işlem süresi olarak daha yavaş kalmaktadır.

Adaptif yaklaşımlarda giriş parametrelerinin karmaşıklığı ve sonuca etkisi nedeniyle başarımlar düşmektedir. Ancak son dönemde geliştirilen tekniklerle, bu parametrelerin sistem üzerindeki etkisi azaltılmaya çalışılmaktadır. Bununla birlikte doğru giriş parametreleri seçildiği takdirde çok hızlı ve yüksek bir başarımlar elde edilebilmesi, adaptif metotların hâlâ popüler olmasında etkindir.

Hem eğitim setin (193) hem de test setin (20) resimleri boyut üzerinde değişiklik yapılmamış olup her bir resmin orijinal boyutu (256x256) kullanıldı. Eğitim setindeki her bir resim tek tek alınıp geliştirilmiş Gabor (LoG) filtresinden geçilerek öznitelik vektörüne aktarıldı ve sonuçta eğitim için 256x193 boyutlu, test için ise 256x20 boyutlu vektör elde edilmiş oldu. Temel Bileşenler Analizi ile analizi yapıldıktan sonra Öklid uzaklığı (L2 normu) ile mesafe hesabı yapılarak sınıflandırma işlemi tamamlanır. Her bir test resmi otomatik olarak dosyaya ifadesiyle beraber yazdırma işlemi yapıldı. Sistemin tüm işlemleri otomatik olarak çalışmaktadır.

Uygulamalarının kullanıldığı bilgisayarın konfigürasyon bilgisi ise şöyle: Core2 Duo 2.2 GHz işlemci, 3 GB ram, 320 GB HDD ve 32 bit Windows 7 ultimate işletim sistemi özellikli dizüstü bilgisayar.



Şekil 4.9. Adaptif LoG uygulaması için örnek giriş resimleri



Şekil 4.10. Adaptif LoG uygulaması için örnek çıkış imgeleri

Tablo 4.10. Adaptif yaklaşım uygulama sonuçları

	<i>LMS</i>	<i>RLS</i>	<i>LMS+ RLS</i>
<i>Başarı oranı (%)</i>	90	90	90
<i>Çalışma süresi (sn)</i>	341	444	803
<i>Çıkış imge örneği (uint8 türünde)</i>			

Sonuç olarak geliştirilmiş Gabor (LoG) filtre tabanlı adaptif sistemin yüz ifadesinin analizdeki başarı oranı her iki grup için LMS ve RLS katsayıları ile aynı değer ölçüldü. Ancak RLS katsayılarının LMS katsayılarından daha karmaşık olduğundan LMS çalışma süresinden daha uzun bir sürede işlemi gerçekleştirilmiştir. Hem LMS hem de RLS katsayılarının beraber kullanıldığı durumda ise başarı oranı değişmemekte ancak çalışma süresi bakımından her iki grubun ayrı ayrı çalışma zamanlarının toplamından bile fazla olduğu görülmektedir. Son söz olarak bu sistem (adaptif LoG) için LMS'nin en uygun olduğu testler sonucu göstermektedir.

4.7. Genel Sonuçlar Tablosu

Tablo 4.11. Tüm uygulamaların sonuçları

Gabor Filtresi Uygulaması Sonuçları				
	<i>Gabor</i>	<i>Gabor + Keskinlik</i>	<i>Gabor + Ortalama</i>	<i>Gabor + Keskinlik + Ortalama</i>
<i>Başarı oranı (%)</i>	90	90	90	90
<i>Çalışma süresi (sn)</i>	41	40	43	41
Laplacian of Gaussian Uygulaması Sonuçları				
	<i>LoG</i>	<i>LoG + Keskinlik</i>	<i>LoG + Ortalama</i>	<i>LoG + Keskinlik + Ortalama</i>
<i>Başarı oranı (%)</i>	85	85	85	85
<i>Çalışma süresi (sn)</i>	34	26	30	27
Difference of Gaussians Uygulaması Sonuçları				
	<i>DoG</i>	<i>DoG + Keskinlik</i>	<i>DoG + Ortalama</i>	<i>DoG + Keskinlik + Ortalama</i>
<i>Başarı oranı (%)</i>	75	50	80	70
<i>Çalışma süresi (sn)</i>	39	40	40	41
Birinci Aşama Uygulama Sonuçları (JAFTE Veritabanı)				
Ana Fonksiyonlar		Yardımcı Filtreler (%)		
		<i>Keskinlik</i>	<i>Ortalama</i>	<i>Keskinlik+ortalama</i>
<i>Gabor</i>	85	85	85	85
<i>LoG</i>	80	80	85	80
<i>LoG + Gabor</i>	88	86	88	90
<i>DoG</i>	75	80	75	80
<i>DoG + LoG</i>	75	70	75	70
<i>DoG + LoG + Gabor</i>	80	85	80	85
İkinci Aşama Uygulama Sonuçları (Amatör Veritabanı)				
Ana Fonksiyonlar		Yardımcı Filtreler (%)		
		<i>Keskinlik</i>	<i>Ortalama</i>	<i>Keskinlik+ortalama</i>
<i>Gabor</i>	48	44	48	52
<i>LoG</i>	20	26	25	32
<i>LoG + Gabor</i>	84	80	84	88
<i>DoG</i>	24	20	32	20
<i>DoG + LoG</i>	24	24	16	16
<i>DoG + LoG + Gabor</i>	80	85	80	12
Adaptif Yaklaşım Uygulama sonuçları				
	<i>LMS</i>	<i>RLS</i>	<i>LMS+ RLS</i>	
<i>Başarı oranı (%)</i>	90	90	90	
<i>Çalışma süresi (sn)</i>	341	444	803	

5. SONUÇLAR VE TARTIŞMA

Günümüz güvenlik uygulamalarında, kişi sorgulamalarında, konuşamayan hasta tedavilerinde ve birçok uygulamalarda kendine yer bulan yüz ifade tanıma disiplini birçok farklı teknikle gerçekleştirilebilmektedir. Bu çalışmada seçilen yöntemler, bu alanda kullanılan tekniklerin temelini oluşturmaktadır. Adaptif yaklaşımlarda giriş parametrelerinin karmaşıklığı ve sonuca etkisi nedeniyle başarımlar düşmektedir. Ancak son dönemde geliştirilen tekniklerle, bu parametrelerin sistem üzerindeki etkisi azaltılmaya çalışılmaktadır. Bununla birlikte doğru giriş parametreleri seçildiği takdirde çok hızlı ve yüksek bir başarımla elde edilebilmesi adaptif metodların hâlâ popüler olmasında etkindir.

Şuana kadar genel olarak %100 başarımla dünya nüfusu yaklaşık 6 milyar insanın yüz ifadesini tanıyabilen bir sistem bulunmamaktadır, mümkün olmamıştır. Geliştirilen her küçük ölçekli yüz ifade analiz sisteminde kullanılan veritabanının kısıtlı olması, yeteri kadar mimik örneğinin alınamamasından dolayı gerekli hedef ideal sistemi elde etmek için şartlar sağlanamamıştır. 6 milyar insanın %75'i kadar mimik örneği alınsa dahi, şimdilik hiçbir sistemin çalışma performansı hiç iç açıcı bir durumda değildir. İdeal sistem günümüzde dünyada yaşayan ve yaşayacak tüm insanların yüz ifadesini tanıyabilmesi, aynı zamanda bu işi saniyeler seviyesinde yapabilmesi gerekmektedir. Bu durum göz önüne alındığında, çalışılan bu sistem sadece temel parçacık denebilecek kadar küçüktür.

Adaptif sistemler genel olarak başarı yüzdesi yüksektir ancak, çalışma süreleri bakımından dezavantaj bir durumu bulunmaktadır. Eğer çalışma süresini düşürebilecek bir yöntem geliştirilirse sistem performansı açısından optimum çözüme yaklaşmış olacaktır. Kaldı ki bu çalışmada sadece tek kare video, yani resimler üzerinde çalışılmıştır, canlı video ile düşünüldüğünde çok zaman alacağı apaçıktır. Gelecekte elbette bilgisayar donanım bakımından yüksek performanslı olacak ancak sistemin zamansal başarımları açısından çok yetersizdir.

KAYNAKLAR

- [1]. **V. Bruce**, Advanced Robotics, 8, 341(1993).
- [2]. **F. Hara and H. Kobayashi**, Advanced Robotics, 11, 585(1997).
- [3]. **A. Takeuchi and K. Nagao**, Proceedings of the INTERCHI '93 conference on Human factors in computing systems, 1, 187(1993).
- [4]. **E. Boyle, A.H. Anderson, and A. Newlands**, Language and Speech, 37, 1(1994).
- [5]. **G.M. Stephenson, K. Ayling and D.R. Rutter, Britain J.** Social Clinical Psychology, 15, 113(1976).
- [6]. **A. Mehrabian**, Psychology Today, 2, 53(1968).
- [7]. **C. C. Chibelushi and F. Bourel**, Facial Expression Recognition: A Brief Tutorial Overview, School of Computing, Staffordshire University, Staffordshire, 2002.
- [8]. **T. Güneş ve E. Polat**, Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi, 24, 7(2009).
- [9]. **Lien, J.J.**, 1998. "Automatic Recognition of Facial Expressions Using Hidden Markov Models and Estimation Intensity", *PhD Thesis*, The Robotics Institute Carnegie Mellon University Pittsburg Pennsylvania.
- [10]. **Kurt, B.**, 2007 "Bilgisayar ile Psikolojik Durum Değerlendirilmesi", *Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü*, Trabzon.
- [11]. **Pantic, M., Rothkrantz, L. J. M.**, 2003. Toward an Affect-Sensitive Multimodal Human-Computer Interaction. Proceedings of the IEEE, *Spec. Issue on Human-Computer Multimodal Interface*, Vol. 91, No. 9, pp.1370-1390.
- [12]. **Maat, L. & Pantic, M.** 2006. Gaze-X: Adaptive affective multimodal interface for single-user office scenarios, *Proc. ACM Int'l Conf. Multimodal Interfaces*, pp.171- 178.
- [13]. **M. Pantic, and L. J. M. Rothkrantz**, IEEE Trans. on PAMI, 22, 1424(2000).
- [14]. **P. Viola and M. Jones**. Robust real-time object detection, Technical Report CRL 20001/01, Cambridge Research Laboratory, Cambridge Inc., 2001.
- [15]. **M. Yang, D. Kriegman, and N. Ahuja**, IEEE Trans. on Pattern Analysis and Machine Intell., 24, 34(2002).
- [16]. **P. Ekman and W. V. Friesen**, Journal of Personality and Social Psychology, 17, 124(1971).
- [17]. **R. Schapire**, Foundations of Machine Learning, Lecture Notes COS 511, 2003.
- [18]. **N. J. Nilsson**, Introduction to Machine Learning, Robotics Laboratory, Department of Computer Science, Stanford University, 2005.
- [19]. **Z. L. Stan and K. J. Anil**, Handbook of Face Recognition, Springer Inc., New York, 2005.
- [20]. **Essa, I. A.**, 1995. Analysis, Interpretation and Synthesis of Facial Expressions, *Massachusetts Institute Of Technology*.
- [21]. **C. Darwin**, The Expression of the Emotions in Man and Animals, J. Murray Inc., London, 1872.
- [22]. **C. Izard, L. Dougherty, and E. A. Hembree**. A System for Identifying Affect Expressions by Holistic Judgments, In Unpublished Manuscript, University of Delaware, 1983.
- [23]. **M. Suwa, N. Sugie, and K. Fujimora**, Proc. Int'l Joint Conf. Pattern Recognition, 41, 408(1978).
- [24]. **S. Marcelja**, Journal of the Optical Society of America, 70, 1297(1980).

- [25]. **J. G. Daugman**, Journal of the Optical Society of America, 2, 1160(1985).
- [26]. **K. Mase**, IEICE Transactions, 74, 3474(1991).
- [27]. **M.J. Black and Y. Yacoob**, Fifth International Conference on Computer Vision (ICCV'95), Cambridge, 1, 374(1995).
- [28]. **M. Rosenblum, Y. Yacoob, and L. S. Davis**, IEEE Transactions on Neural Network, 7, 1121(1996).
- [29]. **I.A. Essa and A.P. Pentland**, IEEE Transactions on Pattern Analysis and Machine Intelligence, 19, 757(1997).
- [30]. **T. Otsuka, and J. Ohya**, IEEE Proc. International Conf. on FG, 1, 442(1998).
- [31]. **A. Martinez**, In *IEEE Workshop on Content-based Access of Images and Video Libraries*, 1, 35(1999).
- [32]. **N. Oliver, A. Pentland, and F. B'érard**, Pattern Recognition, 33, 1369(2000).
- [33]. **I. Kotsia and I. Pitas**, IEEE International Conference on Image Processing, 2, 966(2005).
- [34]. **G. Littlewort, M. S. Bartlett, I. Fasel, J. Susskind and J. Movellan**, Image and Vision Computing, 24, 615(2006).
- [35]. **G. Donato, M. S. Bartlett, J. C. Hager, P. Ekman, and T. J. Sejnowski**, IEEE Transaction on Pattern Analysis and Machine Intelligence, 21, 974(1999).
- [36]. **Ekman, P.** 1992. Facial expressions of emotion: An old controversy and new findings, *Philosophical Transactions of the Royal Society of London*, B(335):63–69.
- [37]. **Ekman, P. & Friesen, W.V.**, 1978. Facial Action Coding System, *Consulting Psychologist Press*, Palo Alto, USA.
- [38]. **Ekman, P., Friesen, W.V. & Hager, J.C.**, 2002. Facial Action Coding System, A Human Face, Salt Lake City, USA.
- [39]. **Pantic, M., Bartlett, M.S.**, 2007, Machine Analysis of Facial Expressions, Face Recognition, K. Delac and M. Grgic, eds., pp. 377-416, *I-Tech Education and Publishing*.
- [40]. **Kanade, T., Cohn, J., Tian, Y.**, 2000, “Comprehensive Database for Facial Expression Analysis,” *Proc. IEEE Int’l Conf. Face and Gesture Recognition (AFGR ’00)*, pp. 46-53.
- [41]. **Yin, L. , Wei, X., Sun, Y., Wang, J. , Rosato, M.J.**, 2006. A 3D Facial Expression Database for Facial Behavior Research,” *Proc. IEEE Int’l Conf. Automatic Face and Gesture Recognition (AFGR ’06)*, pp. 211-216.
- [42]. **Gunes, H., Piccardi, M.**, 2006 “A Bimodal Face and Body Gesture Database for Automatic Analysis of Human Nonverbal Affective Behavior,” *Proc. 18th Int’l Conf. Pattern Recognition (ICPR ’06)*, vol. 1, pp. 1148-1153.
- [43]. **Kotsia, I., Pitas, I.**, 2007. Facial Expression Recognition in Image Sequences Using Geometric Deformation Features and Support Vector Machines, *IEEE Trans. Image Processing*, vol. 16, no. 1, pp. 172-187, 2007.
- [44]. **Pantic, M., Rothkrantz, L. J. M.**, 2004. Facial Action Recognition for Facial Expression Analysis from Static Face Images, *IEEE Trans. Systems, Man, and Cybernetics Part B*, vol. 34, no. 3, pp. 1449-1461.
- [45]. **Sebe, N., Lew, M.S., Cohen, I., Sun, Y., Gevers, T., Huang, T.S.**, 2004 “Authentic Facial Expression Analysis,” *Proc. IEEE Int’l Conf. Automatic Face and Gesture Recognition (AFGR)*.
- [46]. **Gokturk, S.B., Bouguet, J.Y., Tomasi, C., Girod, B.** 2002. Model-based face tracking for view independent facial expression recognition, *Proc. IEEE Int’l Conf. Face and Gesture Recognition*, pp. 272-278.

- [47]. Guo, G., Dyer, C.R., 2005 Learning from Examples in the Small Sample Case: Face Expression Recognition, *IEEE Trans. Systems, Man, and Cybernetics Part B*, vol. 35, no. 3, pp. 477-88.
- [48]. Anderson, K., McOwan, P.W., 2006. "A Real-Time Automated System for Recognition of Human Facial Expressions," *IEEE Trans. Systems, Man, and Cybernetics Part B*, vol. 36, no. 1, pp. 96-105.
- [49]. Lyons, M., Budynek, J. ve Akamastu, S., 1999, Automatic Classification of Single Facial Images, *IEEE Trans. Pattern Analysis and Machine Intelligence*, 21,1357-1362.
- [50]. Tian, Y.L., Kanade, T. & Cohn, J.F. 2001. Recognizing action units for facial expression analysis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 23, No. 2, pp. 97-115.
- [51]. Zhang,Y., Ji, Q., 2005. Active and Dynamic Information Fusion for Facial Expression Understanding from Image Sequences, *IEEE Trans.Pattern Analysis and Machine Intelligence*, Vol. 27, no. 5, pp. 699-714.
- [52]. Cohen, I., Sebe, N., Garg, A., Chen, L.S. & Huang, T.S., 2003. Facial expression recognition from video sequences – temporal and static modelling. *Computer Vision and Image Understanding*, Vol. 91, pp.160-187.
- [53]. Byun, K.S., Park, C.H., Sim, K.B., 2004."Emotion Recognition from Facial Expression Using Hybrid-Feature Extraction", *SICE Annual Conference in Sapporo*.
- [54]. Essa, I. & Pentland, A., 1997. Coding, analysis, interpretation, and recognition of facial expressions. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7, pp. 757- 63.
- [55]. Pantic, M., Patras, I., 2006. Dynamics of Facial Expression: Recognition of Facial Actions and Their Temporal Segments Form Face Profile Image Sequences, *IEEE Trans. Systems, Man, and Cybernetics Part B*, Vol. 36, no. 2, pp. 433-449.
- [56]. Valstar, M.F., Pantic, M. 2006. Fully automatic facial action unit detection and temporal analysis, *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, Vol. 3, p. 149.
- [57]. Pantic, M., Rothkrantz, L. J. M., 2000. Automatic Analysis of Facial Expressions: The State of the Art, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.22, No.12.
- [58]. Tian, Y.L., Kanade, T., Cohn, J.F., 2005 "Facial Expression Analysis," *Handbook of Face Recognition*, S.Z. Li and A.K. Jain, eds., pp. 247-276, Springer.
- [59]. Zeng, Z., Pantic, M., Roisman, G. I. , Huang, T. S., 2009 "A Survey of Affect Recognition Methods: Audio, Visual, and Spontaneous Expressions", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.31, No.1.
- [60]. Black,M.J., Yacoob, Y., 1997. "Recognizing Facial Expressions in Image Sequences Using Local Parameterized Models of Image Motion", *International Journal of Computer Vision*, 25(1):23-48.
- [61]. Yeasin, M., Bullot, B., Sharma, R., 2006. Recognition of Facial Expressions and Measurement of Levels of Interest from Video, *IEEE Trans. Multimedia*, Vol. 8, no. 3, pp. 500-507.
- [62]. Whitehill, J., Omlin, C.W., 2006. Haar Features for FACS AU Recognition, *Proc. IEEE Int'l Conf. Automatic Face and Gesture Recognition (AFGR '06)*, pp. 217-222.
- [63]. D. Gabor, J. Inst.Electr. Eng., 93, 429(1946).

- [64]. **S. Marcelja**, Journal of the Optical Society of America, 70, 1297(1980).
- [65]. **J. G. Daugman**, Journal of the Optical Society of America, 2, 1160(1985).
- [66]. **A. Jain, and F. Farrokhnia**, Pattern Recogn., 24, 1167(1991).
- [67]. **Y. Hammamoto**, Intelligent Biometric Techniques in Fingerprint and Face Recognition (L.C. Jain, U. Halici et al., Eds.), CRC Press, 1999.
- [68]. **C. Liu, and H. Wechsler**, IEEE Int. Conf. Computer Vision, 2, 270(2001).
- [69]. **J. G. Daugman**, Biometrics: Personal Identification in Networked Society (A.K. Jain, R. Bolle and S. Pankanti, Eds.), Kluwer Academic Publishers, 1998.
- [70]. **D. H. Hubel, and T. N. Wiesel**, Royal Soc. B (London), 198, 1(1978).
- [71]. **World Wide Web**, http://en.wikipedia.org/wiki/Gabor_filter. E.T. 09.08.2010
- [72]. **D. Dunn, W. E. Higgins, and J. Wakeley**, IEEE Transactions on Pattern Analysis and Machine Intelligence, 16, 130(1994).
- [73]. **D. Clausi, and M. Jernigan**, *Pattern Recognition*, 33, 1835(2000).
- [74]. **A. K. Jain, R. P. W. Duin, and J. Mao**, IEEE Trans. Pattern Analysis and Machine Intelligence, 22, 4(2000).
- [75]. **R. Gilad-Bachrach, A. Navot, N. Tishby**, 21st International Conference on Machine Learning (ICML04), 69, 43(2004).
- [76]. **P. Cunningham, M. Cord, P. Cunningham (eds)**. Machine Learning Techniques for Multimedia: Case Studies on Organization and Retrieval, Springer Inc., Berlin, 2008.
- [77]. **A. M. Martinez, and A. C. Kak**, IEEE Trans. Pattern Analysis and Machine Intelligence, 23, 228(2001).
- [78]. **R. O. Duda, P. E. Hart, and D. G. Stork**, Pattern Classification, Wiley, New York, 2001.
- [79]. **K. Özdamar**, Paket Programlar İle İstatistiksel Veri Analizi (Çok Değişkenli Analizler)-2, Kaan Kitabevi, 1999.
- [80]. **Marr D., and Hildreth E.**, Theory of Edge Detection, Proc. R. Soc. London, B 207, 187-217, 1980
- [81]. **John C. Russ**, Materials Science and Engineering Department, North Carolina State University, Raleigh, North Carolina, 27695.
- [82]. **Boroujeng, B. F.**, Adaptive Filters: Theory and Applications, John Wiley & Sons.
- [83]. **Peters, S.D., and Antoniu, A.**, "A Self-Tuning NLMS Adaptive Filter Using Parallel Adaptation," IEEE Trans. On Circuits & Systems-II: Analog and Digital Signal Processing, 44(1), 11-21.
- [84]. **Qiu, W., Chang, C., Liu, W., Poon, P., Hu, Lam, Y. F., Hamernik, K. R. P. Wei, G., Chan, H.Y.**, "Real-Time Data-Reusing Adaptive Learning of a Radial Basis Function Network for Tracking Evoked Potentials," IEEE Transactions on Biomedical Engineering, 53, 226-237, 2006.
- [85]. **Haykin, S.**, Adaptive Filter Theory, Prentice-Hall, NJ.
- [86]. **Rabiner, L. R. and Shafer, R. W.**, September 1978, *Digital Processing of Speech Signals*, Prentice Hall Inc., pp. 396-444.
- [87]. **Manolak, S. D. G., Ingle, V. K. and Kogon, S. M.**, April 2005, *Statistical and Adaptive Signal Processing*, McGraw Hill, pp. 499-573.
- [88]. **Mihıslavljevic, M.**, May 2002, *Delayed Error and Update Scheduling of LMS Algorithms*, ICASSP 2002 International Conference on Acoustic Speech and Signal Processing, pp. 1409-1412.
- [89]. **Özbek, F. and Ertürk, S.**, June 2003, *Adaptif LMS Algoritması ile Polinom Filtresinin Görüntü Stabilizasyonuna Uygulanması*, 11. Sinyal İşleme ve Uygulamaları Kurultayı, pp. 426-429.

- [90]. Uçan, O. N. and Osman, O., November 2006, *Haberleşme Teorisi ve Mühendislik Uygulamaları*, Nobel Yayın Dağıtım, pp. 161-170.
- [91]. M.YOZGAT, “Bilgisayarda Parmak İzi Tanıma”, Gazi Üniversitesi Elektronik ve Bilgisayar Bölümü, Yüksek Lisans Tezi
- [92]. Halici U.; Jain L. C.; Hayashi, I.; Lee, S.B.; Tsutsui T., *Intelligent Biometric Techniques in Fingerprint and Face Recognition*, CRC press, USA, 1999.
- [93]. E. Sönmez, N. Özbek, Ö. Özbek, *Avuç İzi ve Parmak İzine Dayalı Bir Biyometrik Tanıma Sistemi*.
- [94]. Ş. Sağıroğlu, N. Özkaya, *Otomatik Parmak izi Tanıma Sistemlerinde Kullanılan Önışlemler İçin Yeni Yaklaşımlar*, Gazi Üniv. Müh. Mim. Fak. Der. Cilt 21, No 1, 11-19, 2006.
- [95]. R. Şamlı, M. Yüksel, 2009, “**Biyometrik Güvenlik Sistemleri**”, Akademik Bilişim’09 - XI. Akademik Bilişim Konferansı Bildirileri 11-13 Şubat 2009 Harran Üniversitesi, Şanlıurfa.

EKLER

Yüz ifade tanıma sisteminin anaprogramı

```
tic
maxDeney=1; % yapılacak deney sayısı
EsikDeger=100; % deneyin eşiklendiği değer

if(~exist('DeneySayisi','var'))
    DeneySayisi=1; % Deneye başlangıç
    baslangic=1;
end;

egitimKlasorYolu='JAFFE'; %193 adet orijinal jaffe
ifadeDosyasi='JAFFE/ifadeler2.txt';
testKlasorYolu='jaffe_test2\'; %20 adet orijinal jaffe
dosyaturu='.tiff';

testResimSayisi=size(dir([testKlasorYolu,'*',dosyaturu]),1);
fid=fopen(ifadeDosyasi);
resimIfade=textscan(fid,'%s %s','whitespace',''); fclose(fid);

for i=1:length(resimIfade{1,1})
    if (strcmp(lower(resimIfade{1,2}{i,1}),'sakin'))
        SakinResimler=[SakinResimler,i];end; end
if (isempty(SakinResimler))
    disp('HATA: sakin ifadesi eğitimde bulunamadı!'); return; end
testResimlerYapi=dir(testKlasorYolu);resimSayisi=length(resimIfade{1,1});

if (testResimSayisi==0)
    disp('Hatalı Test Klasörü'); return; end

for i = 1:resimSayisi
    EgitimResimler{i,1}=strcat(egitimKlasorYolu,'\resimIfade{1,1}(i);end

for k=1:testResimSayisi
    TestResimler{k,1}=[testKlasorYolu,icerik(k,1).name];end;
    resimBoyut = [256,256];

%Eğitim verilerini yükleme ve İşleme
img_Egitim = zeros(resimBoyut(1)*resimBoyut(2),resimSayisi);

for i = 1:resimSayisi
    fprintf('# %d.Deney İçin %d.Eğitim Resmi  Yükleniyor...
    (%s)',DeneySayisi,i,cell2mat(EgitimResimler{i,1}));
```

```

EgitimResmi=cell2mat(EgitimResimler{i,1});
EgitimResmi1=ResimOku(EgitimResmi,resimBoyut);
[EgitimResmi_ DeneyFonksiyonu]=ResmiFiltrele(EgitimResmi1,DeneySayisi);
img_Egitim(:,i) = EgitimResmi_(:); disp(' yüklendi. '); end

disp('Temel Bileşen Analizin Katsayıları Hesaplanıyor.. ');
ortalamaResim = mean(img_Egitim,2);
img_Egitim = (img_Egitim - ortalamaResim*ones(1,resimSayisi));

[C,S,L]=princomp(img_Egitim,'econ'); %temel bileşen analiz
OzBoyut = [1:33]; C = C(:,OzBoyut); % Çıkış değeri C alınmaktadır

%Yüz Uzayına Test Resimleri Yükleme
img_Test = zeros(resimBoyut(1)*resimBoyut(2),testResimSayisi);
for i = 1:testResimSayisi
    fprintf('# %d.Deney İçin %d.Test Resmi Yükleniyor... (%s)
',DeneySayisi,i,TestResimler{i,1});
    TestResmi=ResimOku(TestResimler{i,1},resimBoyut);
    [TestResmi_ DeneyFonksiyonu]=ResmiFiltrele(TestResmi,DeneySayisi);
    img_Test(:,i) = TestResmi_(:); disp(' yüklendi. '); end

ortalamaResim = mean(img_Test,2);
img_Test = (img_Test - ortalamaResim*ones(1,testResimSayisi));
PlanlananTest = img_Test*C; % Çıkış değeri PlanlananTest alınmaktadır

% Herbir test resmin ortalama resminden uzaklık hesabı
ortalamaSakin = mean(S(SakinResimler,OzBoyut)',2);

for digermesafe = 1:testResimSayisi
    testResmi = PlanlananTest(digermesafe,:);
    OklidUzaklik(digermesafe)=metrik(testResmi',ortalamaSakin,2); end

digerUzaklik = zeros(testResimSayisi,resimSayisi);
for digermesafe = 1:testResimSayisi
    testResmi = PlanlananTest(digermesafe,:);
    for i = 1:resimSayisi
        digerUzaklik(digermesafe,i)=metrik(testResmi',S(i,OzBoyut)',2);end; end
    [Min_Uzak,Min_Uzak_poz] = min(digerUzaklik,[],2);

% Sonuçlandır ve Sonuçları Dosyaya Yaz.
sonucdosyasi = fopen('Sonuç.txt','w');
dogru=0;
for i = 1:testResimSayisi
    b = find(TestResimler{i,1}=='\n');Test_Resmi =
    TestResimler{i,1}(b(end)+1:end);SakindenUzaklik = OklidUzaklik(i);
    EniyiOrtusme = cell2mat(resimIfade{1,1}(Min_Uzak_poz(i)));
    Duygu = cell2mat(resimIfade{1,2}(Min_Uzak_poz(i)));
    dogruMu="";
    if (strcmp(Test_Resmi(4:5),'NE'))

```

```

if(strcmp(Duygu,'sakin'))
    dogru=dogru+1; dogruMu=' Doğru Tanındı';
else
    dogruMu="; end
else
    if (strcmp(Test_Resmi(4:5),'AN'))
        if(strcmp(Duygu,'Sinirli'))
            dogru=dogru+1; dogruMu=' Doğru Tanındı';
        else
            dogruMu="; end
        else
            if (strcmp(Test_Resmi(4:5),'DI'))
                if(strcmp(Duygu,'Tiksinmiş'))
                    dogru=dogru+1; dogruMu=' Doğru Tanındı';
                else
                    dogruMu=";end
                else
                    if (strcmp(Test_Resmi(4:5),'FE'))
                        if(strcmp(Duygu,'Korkmuş'))
                            dogru=dogru+1; dogruMu=' Doğru Tanındı';
                        else
                            dogruMu=";end
                        else
                            if (strcmp(Test_Resmi(4:5),'HA'))
                                if(strcmp(Duygu,'Mutlu'))
                                    dogru=dogru+1; dogruMu=' Doğru Tanındı';
                                else
                                    dogruMu=";end
                                else
                                    if (strcmp(Test_Resmi(4:5),'SA'))
                                        if(strcmp(Duygu,'Üzgün'))
                                            dogru=dogru+1;dogruMu=' Doğru Tanındı';
                                        else
                                            dogruMu="; end
                                        else
                                            if (strcmp(Test_Resmi(4:5),'SU'))
                                                if(strcmp(Duygu,'Şaşırılmış'))
                                                    dogru=dogru+1; dogruMu=' Doğru Tanındı';
                                                else
                                                    dogruMu="; end;end;end;end;end;end; end; end

fprintf(sonucdosyasi,'%s --> %s %s\r\n',Test_Resmi,Duygu,dogruMu);end

fprintf(sonucdosyasi,'Doğru Tanınan Toplam Resim Sayısı:%d \r\n',dogru);
fprintf(sonucdosyasi,'Yalnız Tanınan Toplam Resim Sayısı:%d \r\n',testResimSayisi-
dogru);fclose(sonucdosyasi);
sure(DeneySayisi)=toc;

```

% Başarı Oranı kontrol edilmesi ve Deneylerin Tekrarı

```
BasariOrani=dogru/testResimSayisi*100;DeneySonucu(DeneySayisi)=BasariOrani;
KullanilanFonksiyon{DeneySayisi}=DeneyFonksiyonu;
fprintf('Başarı Oranı: %0.0f', BasariOrani);disp('%');disp(bosluk);
```

```
if(~exist('indeks','var'))
KontrolEt; % Başarı oranlarının kontrol edilmesi
end
```

%Filtreleme işlemleri

```
function [filtrelenmisResim DeneyFonksiyonu]=ResmiFiltrele(Resim,Hangisi)
```

```
if(Hangisi==1)
    DeneyFonksiyonu='Yalnızca Gabor; Resim=gabor(Resim); end;
```

%Metrik degerinin hesaplandığı fonksiyondur

```
function mesafe=metrik(I1,I2,p)
T=0;s1=size(I1);s2=size(I2);
if isequal(s1,s2)
s=s1;
for n=1:s(1,1)
for m=1:s(1,2)
T=T+(abs(I1(n,m)-I2(n,m)))^p;end;end;mesafe=(T)^(1/p);end;
```

%gabor fonksiyonu

```
function cikis=gabor(I)
[G,GABOUT]=Filtrele(I,0.05,0.025,0,0);clear I;I=GABOUT; clear GABOUT;
cikis=uint8(I);end
```

```
function [G,GABOUT]=Filtrele(I,S,F,W,P)
if isa(I,'double')==1
    I=double(I);end; size=fix(1.5/S); k=0.5;

for x=-size:size
    for y=-size:size
        G(size+x+1,size+y+1)=k*exp(-pi*S^2*(x*x+y*y))*...
            (exp(j*(2*pi*F*(x*cos(W)+y*sin(W))+P))-exp(-pi*(F/S)^2+j*P));
    end;end; GABOUT=conv2(I,double(G),'same'); end
```

% Başarı oranlarının kontrol edilmesi

```
if (DeneySonucu(DeneySayisi)<EsikDeger)
    if(DeneySayisi<maxDeney)
        Temizle; % Bazı Değişkenlerin Temizlenmesi
        DeneySayisi=DeneySayisi+1; % Deney sayılarının arttırılması
        IfadeTanima_Karsilastirma;%Ana Sayfa
    else
        DeneySonucDosyasi = fopen('DeneylerSonucu.txt','w');
        for ind=baslangic:maxDeney;% length(DeneySonucu)
            fprintf(DeneySonucDosyasi,'%d. Deney Sonucu: %0.0f; Fonksiyon: %s;
            \n',ind,DeneySonucu(ind),KullanilanFonksiyon{ind});
```

```

fprintf('%d.Deney Sonucu: %0.0f; Fonksiyon: %s
\\n',ind,DeneySonucu(ind),KullanilanFonksiyon{ind}); end
fclose(DeneySonucDosyasi);[Ddeg Dind]=max(DeneySonucu);
disp(' '); fprintf('En Yüksek Sonuç: yüzde %0.0f, Deney: %d \\n ',Ddeg,Dind);
disp(' '); sy=1;
for ind=baslangic:maxDeney;% length(DeneySonucu)
fprintf('%d.Deney Süresi: %f sn\\n',ind,sure(ind));
sure_(sy)=sure(ind); sy=sy+1;
end
[z in]=min(sure_);
disp(' '); fprintf('En düşük zaman: %f sn, Deney: %d \\n ',z,in); sonuc=1; end;
else
disp('Gereken Başarı Eşiği Yakalandı. ');
DeneySonucDosyasi = fopen('DeneylerSonucu.txt','w');
for ind=baslangic:maxDeney;% length(DeneySonucu)
fprintf(DeneySonucDosyasi,'%d. Deney Sonucu: %0.0f; Fonksiyon: %s;
\\n\\n',ind,DeneySonucu(ind),KullanilanFonksiyon{ind});
fprintf('%d.Deney Sonucu: %0.0f; Fonksiyon: %s
\\n',ind,DeneySonucu(ind),KullanilanFonksiyon{ind});
end
fclose(DeneySonucDosyasi); [Ddeg Dind]=max(DeneySonucu);
disp(' '); fprintf('En Yüksek Sonuç: yüzde %0.0f, Deney: %d \\n ',Ddeg,Dind);
disp(' ');
sy=1;
for ind=baslangic:maxDeney;% length(DeneySonucu)
fprintf('%d.Deney Süresi: %f sn\\n',ind,sure(ind));
sure_(sy)=sure(ind); sy=sy+1;
end
[z in]=min(sure); disp(' '); fprintf('En düşük zaman: %f sn, Deney: %d \\n ',z,in);
sonuc=1; return; end

```

% Temizle fonksiyonu

```

clear EgitimResimler ortalamaResim img_Egitim TestResimler img_Test
clear ortalamaSakin testResimSayisi Duygu dogru BasariOrani

```

% resmi okuyan fonksiyondur

```

function ResiM=ResimOku(ResiM,resimBoyut)
ResiM=imread(ResiM);
ResiM=imresize(ResiM,resimBoyut);
try
ResiM=rgb2gray(ResiM);
catch
end

if isa(ResiM,'double')~=1
ResiM=double(ResiM);
end; end

```


ÖZGEÇMİŞ

Mehmet ŞAKAR, 13.06.1980 tarihinde Şanlıurfa’da doğdu, ilk ve orta öğrenimini Şanlıurfa, lise öğrenimi ise İzmir’de tamamladı. 2004 yılında Çanakkale Onsekiz Mart Üniversitesi, Mühendislik-Mimarlık Fakültesi, Bilgisayar Mühendisliği bölümünden mezun oldu. 2005 yılında kısa dönem askerlik hizmetini yaptıktan sonra, Ekim 2005 tarihinde Devlet Su İşleri Diyarbakır Bölgesinde işe başladı ve hale devam ediyor. 2008 yılında Fırat Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Anabilimdalı, yazılım bölümünde yüksek lisans öğrenimine başlamış olup, halen devam etmektedir. Evli ve iki kız çocuğu var.