



**İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

YÜKSEK LİSANS TEZİ

**YÜZ TANIMA PROBLEMİNE KARMA YÖNTEMLERİN
UYGULANMASI**

**Bilgisayar Müh. Ergün GÜMÜŞ
Bilgisayar Mühendisliği Anabilim Dalı
Bilgisayar Mühendisliği Programı**

**Danışman
Prof. Dr. Ahmet SERTBAŞ**

Haziran, 2008

İSTANBUL



**İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

YÜKSEK LİSANS TEZİ

**YÜZ TANIMA PROBLEMİNE KARMA YÖNTEMLERİN
UYGULANMASI**

**Bilgisayar Müh. Ergün GÜMÜŞ
Bilgisayar Mühendisliği Anabilim Dalı
Bilgisayar Mühendisliği Programı**

**Danışman
Prof. Dr. Ahmet SERTBAŞ**

Haziran, 2008

İSTANBUL

Bu çalışma/...../ 2008 tarihinde ařağıdaki jüri tarafından
Anabilim Dalı programında Yüksek Lisans Tezi olarak kabul
edilmiştir.

Tez Jürisi

Danışman Adı (Danışman)
İstanbul Üniversitesi
Mühendislik Fakültesi

Jüri Adı
Üniversite
Fakülte

Jüri Adı
Üniversite
Fakülte

Jüri Adı
Üniversite
Fakülte

Jüri Adı
Üniversite
Fakülte

Bu alıřma İstanbul Üniversitesi Bilimsel Arařtırma Projeleri Yürütücü Sekreterliđinin
XXXXXXXXXX numaralı projesi ile desteklenmiřtir.

ÖNSÖZ

Tez çalışmalarım boyunca gösterdiği her türlü destek ve yardımdan dolayı çok değerli hocam Prof. Dr. Ahmet SERTBAŞ'a, tez uygulamamda yardımcı olan Yük. Müh. Niyazi Kılıç'a, yüksek lisans öğrenimim boyunca beni destekleyen Türkiye Bilimsel ve Teknolojik Araştırma Kurumu (TÜBİTAK)'a ve her zaman yanımda olan aileme en içten dileklerimle teşekkür ederim.

Haziran, 2008

Ergün GÜMÜŞ

İÇİNDEKİLER

ÖNSÖZ	i
İÇİNDEKİLER	ii
ŞEKİL LİSTESİ	iv
TABLO LİSTESİ	vi
SEMBOL LİSTESİ	vii
ÖZET	ix
SUMMARY	xi
1. GİRİŞ	1
2. GENEL KISIMLAR.....	5
2.1. BİYOMETRİK SİSTEMLERDE ÖRÜNTÜ TANIMA	5
2.2. TEMEL YÜZ TANIMA SİSTEMİ.....	7
2.2.1. Yüz Algılama	8
2.2.2. Ön İşleme	10
2.2.3. Yüz Tanıma.....	11
2.2.3.1. Özellik Temelli Yüz Tanıma.....	12
2.2.3.2. Temel Bileşen Analizi İle Yüz Tanıma	15
2.3. BAŞLICA YÜZ TANIMA YÖNTEMLERİ.....	16
2.3.1. LDA (Doğrusal Diskriminant Analizi).....	17
2.3.2. EBGm (Elastik Salkım Grafiği Eşleştirme)	18
2.3.3. Üç Boyutlu Yüz Tanıma	20
2.3.4. Genetik Algoritmalar İle Yüz Tanıma.....	21
2.3.4.1. Genetik Algoritmalar Hakkında.....	21
2.3.4.2. Genetik Algoritmaların Karakteristik Özellikleri.....	24
2.3.4.3. Genetik Algoritmaların Yüz Tanıma Problemine Uygulanması	25

3. MALZEME VE YÖNTEM	26
3.1. YAPAY SİNİR AĞLARI	26
3.1.1. Yapay Sinir Ağlarının Kökeni	27
3.1.2. Yapay Sinir Ağlarının Gelişimi	28
3.1.3. Yapay Sinir Ağlarının Özellikleri	30
3.1.4. Yapay Sinir Ağlarının Uygulanışı	32
3.1.4.1. Aktivasyon Fonksiyonları	34
3.1.5. Yapay Sinir Ağlarının Sınıflandırılması	35
3.1.5.1. İleri Beslemeli Sinir Ağları	35
3.1.5.2. Geri Beslemeli Sinir Ağları	36
3.1.5.3. Danışmalı Öğrenme Kullanan Sinir Ağları	38
3.1.5.4. Danışmasız Öğrenme Kullanan Sinir Ağları	40
3.1.6. Hatayı Geriye Yansıtma Algoritması	41
3.1.7. Yapay Sinir Ağlarının Yüz Tanıma Problemi İçin Kullanımı	43
3.2. ÖZYÜZLER YÖNTEMİ	44
3.2.1. Özyüzler Yöntemi İçin Çalışma Basamakları	44
3.2.2. Özyüzler Yönteminin Diğer Yöntemlerle Karşılaştırılması	47
3.3. DESTEK VEKTÖR MAKİNELERİ YÖNTEMİ	48
3.3.1. Destek Vektör Makinelerinin Çok Sınıflı Yüz Tanıma Problemine Uygulanışı	51
3.4. UYGULAMA PROGRAMI	52
3.4.1. Özyüzler İle Yüz Tanıma Bölümü	55
3.4.2. PCA-NN İle Yüz Tanıma Bölümü	57
3.4.3. PCA-SVM İle Yüz Tanıma Bölümü	58
4. BULGULAR	59
5. TARTIŞMA VE SONUÇ	67
KAYNAKLAR	71
ÖZGEÇMİŞ	74

ŞEKİL LİSTESİ

Şekil 2.1	: Biyometrik Sistemde Örüntü Tanıma Basamakları	5
Şekil 2.2	: Temel Yüz Tanıma Sistemi	7
Şekil 2.3	: Yüz Algılama Teknikleri	9
Şekil 2.4	: Biçim değiştiren şablonlar yöntemiyle göz ve ağız konumlandırma	14
Şekil 2.5	: Nokta dağılım modeli örneği	14
Şekil 2.6(a,b)	: Bir grup noktanın projeksiyonu	17
Şekil 2.7	: EBGGM yüz ağı modelleri.....	19
Şekil 2.8	: Farklı açılardan alınan pozlarla oluşturulan üç boyutlu yüz modeli	21
Şekil 2.9	: N parametrelili bir genotip.....	21
Şekil 2.10	: K adet parametrenin bir fenotipten geçirilişi.....	22
Şekil 2.11	: Mutasyon işlemi.....	22
Şekil 2.12	: Crossing - Over işlemi.....	23
Şekil 2.13	: Genetik Algoritma Döngüsü.....	23
Şekil 2.14	: Boş değerli gen içeren bir genotip	24
Şekil 2.15	: Evrimsel Arayış yönteminde kullanılan bir genotip	25
Şekil 3.1	: Sinir sistemindeki bilgi akışı	27
Şekil 3.2	: Biyolojik Nöron	28
Şekil 3.3	: McCulloch ve Pitts nöron modeli.	28
Şekil 3.4	: 3 katmanlı yapay sinir ağı örneği.....	32
Şekil 3.5	: Yapay Nöron Modeli.....	33
Şekil 3.6(a-c)	: Aktivasyon Fonksiyonları	34
Şekil 3.7	: İleri beslemeli sinir ağı için blok diyagram.....	35
Şekil 3.8	: Geri beslemeli sinir ağı için blok diyagram.....	37
Şekil 3.9(a,b)	: Yinelene Sinir Ağları	37
Şekil 3.10(a,b)	: Yapay Sinir Ağlarında öğrenme yöntemleri.....	38
Şekil 3.11	: Çok katmanlı sinir ağı için blok diyagram	41
Şekil 3.12	: Hatayı geriye yansıtma algoritması için akış diyagramı	42
Şekil 3.13	: Aranan k. yüz için özellik vektörü temelli sinir ağlarının çalışması.....	43
Şekil 3.14	: Bir grup örnek yüz resmi.....	44
Şekil 3.15	: Sisteme verilen N adet yüz resminden çıkarılan ortalama yüz resmi ...	45
Şekil 3.16	: Eğitim kümesinden elde edilen 7 özyüz.....	46
Şekil 3.17 (a,b)	: Veri gruplarının düzlemlerle ayrılması.....	49
Şekil 3.18 (a)	: Kayıt düzenleme işlemleri için program arayüzü	52
Şekil 3.18 (b)	: Özyüzler yöntemi ile yüz tanıma işlemi için program arayüzü	53
Şekil 3.18 (c)	: PCA-NN yöntemi ile yüz tanıma işlemi için program arayüzü.....	53
Şekil 3.19	: Kullanılan eğitim seti (ORL yüz veritabanı)	55
Şekil 3.20	: Özdeğerlerin Sıralanışı.....	56
Şekil 4.1	: Özyüz Uzayı Boyutu – Tanıma Başarısı İlişkisi.....	60
Şekil 4.2	: Eşik Değeri – Tanıma İlişkisi	61
Şekil 4.3	: Tanıma Oranı – Eğitim Hatası İlişkisi.....	62

Şekil 4.4	: Artan Poz Sayısı – SVM Kernel Fonksiyonlarının Tanıma Oranları İlişkisi	64
Şekil 4.5	: Kullanılan Yöntemlerin Artan Poz Sayısına Göre Tanıma Oranları	66

TABLO LİSTESİ

Tablo 1.1	: Bazı biyometrik yöntemlerin kıyaslanması.....	3
Tablo 4.1	: Gizli katmanda kullanılan nöron sayısı – erişilen eğitim hatası eşleşmesi.....	63
Tablo 4.2	: Artan poz sayısı için eğitimde kullanılan pozlar	65
Tablo 5.1	: ORL veritabanı için bazı yöntemlerle çıkarılan hata oranları	68

SEMBOL LİSTESİ

Σ	: toplam sembolü
E_{snake}	: aktif hatlar modeli için enerji terimi
$E_{internal}$	: aktif hatlar modeli için iç enerji terimi
$E_{external}$	: aktif hatlar modeli için dış enerji terimi
a	: nokta dağılım modeli için bir nokta
$\bar{a}$	: nokta dağılım modelindeki bir nokta için ortalama özellik
P	: nokta dağılım modelinde kovaryans matrinden edinilen matris
v	: ağırlıklandırma vektörü
S_w	: sınıf içi dağılım matrisi
S_b	: sınıflar arası dağılım matrisi
Γ_i^j	: LDA için j. sınıfın i. örneği
μ_j	: LDA için j. sınıfın ortalaması
Y	: LDA için sınıf sayısı
N_j	: LDA için j. sınıftaki örneklerin sayısı
H	: LDA için özvektör matrisi
α	: evrimsel arayış yöntemi için vektör döndürme açıları, SVM için lagrange katsayısı
o^{k+1}	: nöronun k+1 anında ürettiği çıktı
T	: nöron için tetikleme eşiği
w_j	: nörona gelen j. ağırlık katsayısı
u_j	: nörona gelen j. giriş değeri
σ	: nöronun ürettiği net etki
$\hat{\sigma}$	: gizli katmandaki nöronun ürettiği net etki
$f()$	: fonksiyon, aktivasyon fonksiyonu, yapay sinir ağı için fark üretici
$\hat{f}()$	: gizli katmanda kullanılan aktivasyon fonksiyonu
x	: fonksiyon için operand, SVM için eğitim kümesindeki bir nokta
W	: yapay sinir ağı için ağırlık katsayı matrisi veya özyüzler yönteminde özellik vektörleri matrisi
V	: yapay sinir ağı için ağırlık katsayı matrisi veya özyüzler yönteminde özvektör matrisi
c	: öğrenme katsayısı
$\hat{c}$	: gizli katmanda kullanılan öğrenme katsayısı
g_i	: i. nöron için aktivasyon fonksiyonu
d_i	: i. nöronun üretmesi istenen çıktı
U	: yapay sinir ağına verilen giriş vektörü veya özyüz matrisi
r	: fark üretici fonksiyon kalıbı
E_{max}	: maksimum hata değeri
p	: işlenen giriş vektörünün indisi
Z	: verilen girdi vektörlerinin matrisi
δg	: yapay sinir ağının çıkış katmanındaki hata

δu	: yapay sinir ağının gizli katmanındaki hata
Ψ	: özyüzler yöntemi için ortalama yüz
N	: toplam yüz sayısı
Γ_i	: özyüzler yöntemi için i. resim
Φ_i	: özyüzler yönteminde i. yüzün fark vektörü
C	: özyüzler yöntemi için kovaryans matris
D	: özyüzler yönteminde fark vektörlerinden oluşan matris
Fark(i)	: özyüzler yönteminde aranan yüzün özellik vektörüyle i. yüzün özellik vektörünün farkı
w	: SVM yöntemi için uygun ağırlık vektörü
b	: SVM yönteminde ayırıcı düzlem denklemi katsayısı
$d(x)$	: SVM yönteminde x noktasının ayırıcı düzleme olan mesafesi
$\Phi(x)$	: SVM yöntemi için x noktasını daha yüksek boyutlu bir uzaya genişleten fonksiyon

ÖZET

YÜZ TANIMA PROBLEMİNE KARMA YÖNTEMLERİN UYGULANMASI

İnsan ayırt etme problemi uzun süredir üzerinde çalışılan ve öne sürülen yeni yöntemlerle hâlâ gelişmekte olan bir konudur. Günümüze kadar bu problemin çözümüne yönelik olarak biyolojik özelliklerin ayırt edici yönlerini inceleyen pek çok yöntem öne sürülmüştür. Bu yöntemler günümüzde ihtiyaçların çeşitliliğine göre gümrükler, havaalanları, bankacılık işlemleri, özel güvenlik önlemleri gerektiren kurumlara giriş gibi çeşitli uygulamalarda kullanılmaktadır. İnsan tanıma problemi, genel olarak iris tanıma, parmak izi tanıma, yüz tanıma, el veya yüz damarları ile tanıma şeklinde karşımıza çıkmaktadır. Bununla birlikte, yüz tanıma işlemi, diğer yöntemlere nazaran gereksinim duyduğu veri kümesinin daha kolay elde edebilmesi nedeniyle, iş yerine girişte personel tanıma ve suçluların güvenlik kameralarıyla tespiti gibi konularda yaygınlıkla kullanılmaktadır.

Yüz Tanıma işlemi temel olarak, eldeki giriş verisinden aranacak olan yüz resminin çıkarılması (Yüz Algılama), çıkarılan resmin ortamdan kaynaklanan ve tanımayı güçleştirecek dış etkilerden arındırılması, kalan veri kümesinden tanıma işlemi için bu yüze ait özgün özellik vektörlerinin çıkarılması (Özellik Çıkarma) ve bu özellik vektörlerinin mevcut diğer resimlere ait vektörlerle karşılaştırılması (Sınıflandırma) basamaklarından oluşmaktadır.

Bu çalışmada literatürde iyi bilinen yüz tanıma yöntemlerinden PCA tabanlı Özyüzler Yöntemi, Özellik Çıkarma işleminde uygulanmıştır. Diğer temel basamak olan Sınıflandırma işleminde, 2 farklı sınıflandırıcı olarak Yapay Sinir Ağları (NN) ve Destek Vektör Makineleri (SVM) yöntemleri kullanılmış ve yüz tanıma işleminde karma yöntemlerin tanıma performansına etkileri araştırılmıştır. Karma yöntemlerle elde edilen sonuçlar, Özyüzler yöntemi ile bulunan sonuçlarla karşılaştırılmış ve PCA-SVM karma yönteminin, eğitim kümesindeki artan poz sayısı ile beraber Özyüzler yöntemine göre daha iyi tanıma oranları verdiği görülmüştür. Özyüzler yöntemi için özyüz uzayı boyutunun, 1. karma yöntem (PCA-NN) için yapay sinir ağının saklı katmanındaki nöron sayısı ve eğitim hatasındaki değişimin ve 2. karma yöntem (PCA-SVM) için de sınıflandırmada kullanılan Kernel Fonksiyonlarının tanıma performansına etkileri incelenmiştir.

Tezde, yüz tanıma işlemi için geliştirilen uygulama programında sisteme verilen ham yüz resmi gri tona çevrilerek yüz algılama işlemi gerçekleştirilmiştir. Bu işlemlerden sonra çıkarılan yüz resmi üzerinde kullanıcı seçimine bağlı olarak sadece Özyüzlerle ya da diğer karma yöntemlerle (PCA-NN veya PCA-SVM) yüz tanıma işlemi uygulanmaktadır.

Bu alıřmanın ilk blmnde, biyometrik sistemler ve yz tanıma hakkında genel bilgiler; ikinci blmde, yz tanıma konusunda literatrde genel kabul grmř teknikler; nc blmde bu alıřmaya konu olan zyzler, Yapay Sinir Aėları ve Destek Vektr Makineleri yntemleri; drdnc blmde, yz tanıma iřlemi iin geliřtirilen uygulama programı ile elde edilen sonular; beřinci blmde ise, varılan sonuların yorumu ve ileriye ynelik alıřmalar konusunda bilgi verilmektedir.

SUMMARY

APPLICATION OF HYBRID TECHNIQUES TO FACE RECOGNITION PROBLEM

Human distinguishing problem is a long term study which is still developing with newly proposed methods. Till present day many methods that use distinguishing aspects of biological features are stated in order to solve this problem. Today some of these methods are used at customs, airports, banking operations and places that require security measures according to variety of their necessities. Generally, human recognition problem is processed with methods like iris recognition, fingerprint recognition, face recognition and hand/face vein recognition. By the way, in comparison to other stated methods face recognition method can gain necessary input data set easier, allowing it to be commonly used in subjects like personnel or criminal identification with security cameras.

In general definition face recognition process is a union of sub-processes. These sub-processes are; extraction of face image from raw input image which is going to be searched (Face Detection), removing the physical noise which hardens recognition from extracted image, calculation of face image's deterministic features (Feature Extraction) and comparison of calculated features with deterministic features of existing face data set (Classification).

In this study, Eigenfaces method which is based on PCA and one of well known face recognition methods in literature, is used in Feature Extraction. At following basic step, Classification, Neural Networks (NN) and Support Vector Machines (SVM) are used as 2 different classification methods and effects of using these hybrid techniques in face recognition process are examined. Results of hybrid techniques are compared with results of Eigenfaces method and due to increasing number of poses in training set better recognition rates with PCA-SVM hybrid technique are obtained against Eigenfaces technique. For Eigenfaces method difference in eigenspace dimension, for first hybrid technique (PCA-NN) difference in number of neurons in hidden layer and difference in training error and for second hybrid technique (PCA-SVM) difference of used Kernel Functions are studied with their effects on recognition performance.

In prepared application software, the input raw face image is converted into grayscale format and processed in face detection step. After these steps, user can choose to use only Eigenfaces method or other hybrid techniques (PCA-NN or PCA-SVM) for face recognition.

In first part of this study, general information about biometrical systems and face recognition are given. In second part commonly agreed methods in literature for face recognition, in third part information about this study's main subjects Eigenfaces, Artificial Neural Networks and Support Vector Machines, in fourth part results that are obtained with application software, in fifth part comments on results and consideration about future work are explained.

1. GİRİŞ

İnsan ayırt etmede kullanılan biyometri anlamını eski Yunancadaki “bios” (hayat) ve “metron” (ölçüm) kelimelerinden alır ve temel olarak incelenen ögenin fiziksel özelliklerinin ölçümü yöntemiyle diğer örneklerden ayırt edilebilirliğine dayanır.

Başlarda yöntem ormancılığı da içeren çeşitli konularda toplulukların nicel özelliklerinin toplanması, sentezi, analizi ve işlenmesi için kullanılmıştır ve bir biyoloji bilimi dalı olarak bu konuda yapılan araştırmalar nesiller boyunca temel olarak “biyolojik istatistik” çıkarımı çalışmaları olarak görülmüştür [1].

Biyometri, 19. yüzyılda kriminoloji araştırmalarında insanların fiziksel ve karakteristik özelliklerinin suça eğilimlerine etkisini belirlemek amacıyla yapılan araştırmalar ile yeni bir boyut kazanmıştır. Yapılan araştırmalar sonucunda insanların biyolojik özelliklerini ölçmek maksadıyla pek çok ölçüm aleti üretilmiştir. Elde edilen ölçüm yöntemleri içerisinde parmak izi karşılaştırma yöntemi genel kabul görmüş ve günümüze kadar yaygın kullanım alanları bularak gelmiştir. Bu yöneme dayanak olarak da her insanın parmak izinin sadece kendisine ait olduğu ve dünyada başka bir benzerinin olmadığı gösterilmiştir.

İnsanlarda biyometrik özellikler temelde iki ana sınıfta incelenmektedir. Bunlar fizyolojik ve davranışsal özelliklerdir. Fizyolojik özellikler vücuttaki uzuvların şekil ve ölçülerine dayanır ve en çok bilinen örnekleri parmak izi tanıma, yüz tanıma, el geometrisi ile tanıma, el ve yüz damarlarının şekilleri ile tanıma, iris tanıma ve DNA tanımadır. Davranışsal özellikler ise insanın zamanla gelişen davranışlarından yola çıkarak elde edilen ayırt edici nitelikler olup başlıca örnekleri el yazısı ve imza ile tanıma, ses ile tanıma ve tuş vuruşları ile tanıma yöntemidir. Bu yöntemlerden tuş vuruşları ile tanıma yöntemi telgraf döneminde operatörlerin vuruş hızlarıyla birbirlerini tanımasında, 2. dünya savaşında mors kodu ile haberleşen askerlerin dost ile düşmanı ayırt etmesinde kullanılmıştır [2].

Günümüzde biyometrik sistemler farklı türleriyle, farklı ülkelerde çeşitli alanlarda (gümrükler, havaalanları, interaktif bankacılık ve ATM işlemleri, özel güvenlik önlemleri gerektiren kurumlara giriş, güvenlik kameraları ile zanlı/suçlu tespiti vb.) kullanılmaktadır.

Avustralya, 2001 yılı Temmuz ayında kurulan ve kâr amacı gütmeyen bir kuruluş olan Biyometrik Enstitüsü ile ortak bir çalışmaya girerek “Biyometrik Gizlilik Kodu” (Biometrics Privacy Code) uygulamasına geçmiş olup ülkeye gelen turistleri bu uygulama ile biyometrik doğrulamaya tabi tutmayı planlamaktadır. Japonya, bankacılık işlemleri için ATM cihazlarında avuç içi damar doğrulama sistemine geçmiştir. Almanya, içinde sahibinin sayısal yüz resmi ve her elinden birer parmak izi bulunan bir yonganın kullanıldığı “ePass” adında pasaportlar üretmeyi kararlaştırmıştır. İsrail hükümeti, ülkesine çalışmaya gelen işçileri tanımak için kontrol noktalarında içinde bu işçilerin parmak izi, yüz ve el geometrisi bilgileri kayıtlı bulunan Smart Card’lar kullanmaktadır. 11 Eylül 2001 tarihinde yaşananlardan sonra Amerikan hükümeti havaalanlarında yüz tanıma sistemine geçmeye başlamıştır [1]. Türkiye’de de suçluların eşkâlinin çıkarılması ve aranması maksadıyla mevcut olan MOBESE (Mobil Elektronik Sistem Entegrasyonu) sistemine yüz tanıma modüllerinin eklenmesi planlanmıştır [3].

Biyometrik yöntemler “Eşsizlik” (ölçülen biyolojik özelliğin bir örneği diğerinden ne kadar iyi ayırabildiği), “Kalıcılık” (ölçülen biyolojik özelliğin yaşlanmaya karşı ne kadar iyi karşı koyabildiği), “Toplanabilirlik” (Karşılaştırılacak biyolojik özelliğin ne kadar kolay ölçülebildiği), “Performans” (Kullanılan tekniğin doğruluk, hız ve güvenilirliği), “Kabul Edilebilirlik” (Kullanılan teknolojinin onaylanabilirlik derecesi) gibi başlıklar altında birbirleriyle kıyaslanabilir. Buna göre bazı biyometrik yöntemlerin bu kıstaslara göre karşılaştırmalı sonuçları [4] Tablo 1.1’de verilmiştir.

Sayılan biyometrik sistemler içinden bu çalışmaya konu olarak seçilen yöntem “Yüz Tanıma” yöntemidir. Bu seçimin amacı yüz tanıma yönteminde kullanılan tekniklerin genel kabul görmüş olması ve özellikle bu yöntemde kullanılacak olan veri kümesinin diğer yöntemlere nazaran daha kolay elde edilebilmesidir.

Tablo 1.1 : Bazı biyometrik yöntemlerin kıyaslanması

Biyometrik Yöntem	Eşsizlik	Kalııcılık	Toplanabilirlik	Performans	Kabul Edilebilirlik
Yüz	Düşük	Orta	Yüksek	Düşük	Yüksek
Parmak İzi	Yüksek	Yüksek	Orta	Yüksek	Orta
El Geometrisi	Orta	Orta	Yüksek	Orta	Orta
El Damarları	Orta	Orta	Orta	Orta	Orta
İris	Yüksek	Yüksek	Orta	Yüksek	Düşük
Retina Tarama	Yüksek	Orta	Düşük	Yüksek	Düşük
İmza	Düşük	Düşük	Yüksek	Düşük	Yüksek
Ses	Düşük	Düşük	Orta	Düşük	Yüksek
Yüz Termografi	Yüksek	Düşük	Yüksek	Orta	Yüksek
DNA	Yüksek	Yüksek	Düşük	Yüksek	Düşük

Sosyal hayatımızda yüz, duygu ve kimlik belirleme için bizim birincil dikkat noktamızdır. Fiziksel görünümünden zekâ ve karakter gibi noktaların çıkarılması şüpheli yöntemlerken yüz tanıma kayda değer bir başarıdır.

Yüz Tanıma işlemi temel olarak eldeki giriş verisinden aranacak olan yüz resminin çıkarılması (Yüz Algılama), çıkarılan resminin ortamdaki kaynaklanan ve tanımayı güçleştirecek dış etkilere (ışık kaynağının şiddeti ve yönü, arka plan görüntüleri, yüz protezleri, görüntüdeki bulanıklık vb.) arındırılması ve kalan veri kümesinden tanıma işlemi için bu yüze ait özgün özellik vektörlerinin çıkarılması basamaklarından oluşur. Bu özellik vektörlerinin mevcut diğer resimlere ait vektörlerle kıyaslanması ile de yüz tanıma işlemi gerçekleştirilmiş olur.

Bir insan ayırt etme problemi olarak yüz tanıma tarihsel açıdan baktığımızda 1888 yılında Sir Francis Galton tarafından insanların fiziksel özelliklerine göre sınıflandırılmasına dair bir çaba sarf edildiğini görürüz [5]. Yakın zamanda ise 1980'li yıllar yüz tanıma konusunda yapılan çalışmaların büyük ölçüde durgun olduğu yıllarken 1990'lı yıllar gerçek zamanlı yüz tanıma için gereken donanımın elde edilmesi ve sivil/askeri alanlarda bu konu için başlatılan ARGE çalışmalarına verilen desteğin artması nedeniyle bu konuya duyulan ilginin katlanarak arttığı yıllardır.

Yüz tanıma problemi, görüntü işleme konusunda çalışan uzmanları bugüne kadar en çok uğraştıran örüntü tanıma problemidir. Bir insanın birkaç kez gördüğü bir yüzü aradan geçen zamana, yaşlanmaya bağlı olarak yüzdeki değişimlere, farklı ışıklandırma koşullarına ve değişik pozlara rağmen daha sonra tekrar tanıyabilmesi bu yeteneğin bilgisayarlar da kazandırılabilceğı fikrine yol açmıştır. Böylece mevcut bilgisayar görüntü işleme tekniklerinin ışığında çeşitli yöntemler geliştirilmiştir. Ancak geliştirilen bu yöntemlere rağmen insan beyninin yüz tanıma problemini nasıl çözdüğü tam olarak açıklanamadığından bu problem için insan temelli bir bilgisayar modeli kurmak oldukça zordur ve ne yazık ki günümüzde bu konu için tasarlanan bilgisayar sistemleri henüz insanın gösterdiği başarıyı gösterememektedir.

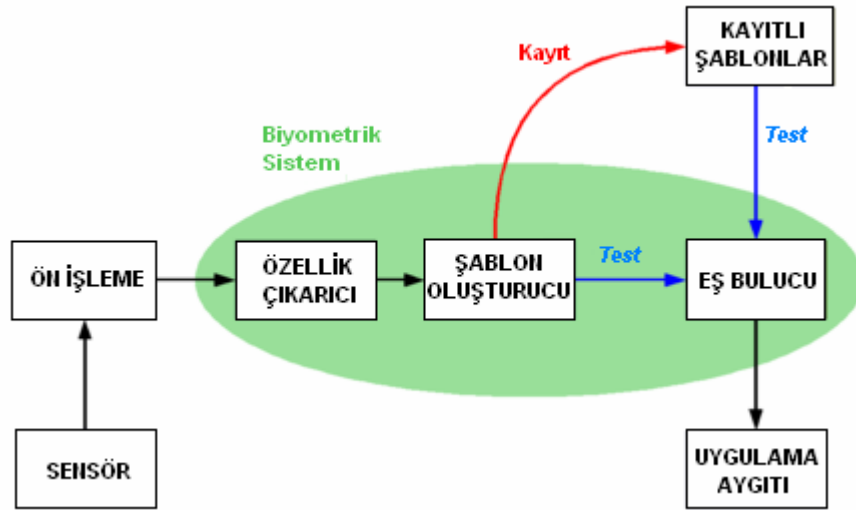
Tezin; ikinci bölümde, yüz tanıma konusunda literatürde genel kabul görmüş teknikler; üçüncü bölümde bu çalışmaya konu olan Özyüzler, Yapay Sinir Ağları ve Destek Vektör Makineleri yöntemleri; dördüncü bölümde, yüz tanıma işlemi için geliştirilen uygulama programı ile elde edilen sonuçlar; beşinci bölümde ise, yüz tanıma karma yöntemlerin performansına ilişkin sonuçlar ve ileriye yönelik çalışmalar verilmiştir.

2. GENEL KISIMLAR

2.1. BİYOMETRİK SİSTEMLERDE ÖRÜNTÜ TANIMA

Örüntü işleme, tanım olarak veri içindeki desenleri tanımak amacıyla yapılan çalışma ve tasarlanan sistemler bütünüdür. Bu araştırma konusu diskriminant analizi, özellik çıkarımı, hata tahmini, kümeleme çözümlemesi, dilbilgisel çıkarım ve hecelere ayırma gibi alt disiplinlerden oluşur. Bu konunun günümüzdeki uygulamaları görüntü analizi, karakter tanıma, ses analizi, insan ayırt etme ve üretim bantlarının denetlenmesi gibi alanlarda ortaya çıkmaktadır.

Biyometrik sistemlerde örüntü tanıma dendiğinde akla insan ayırt etme probleminin çözümüne yönelik olarak tasarlanan uygulamalar gelmektedir. Basit bir biyometrik sistemdeki örüntü tanıma basamakları Şekil 2.1’de gösterilmiştir.



Şekil 2.1 : Biyometrik Sistemde Örüntü Tanıma Basamakları

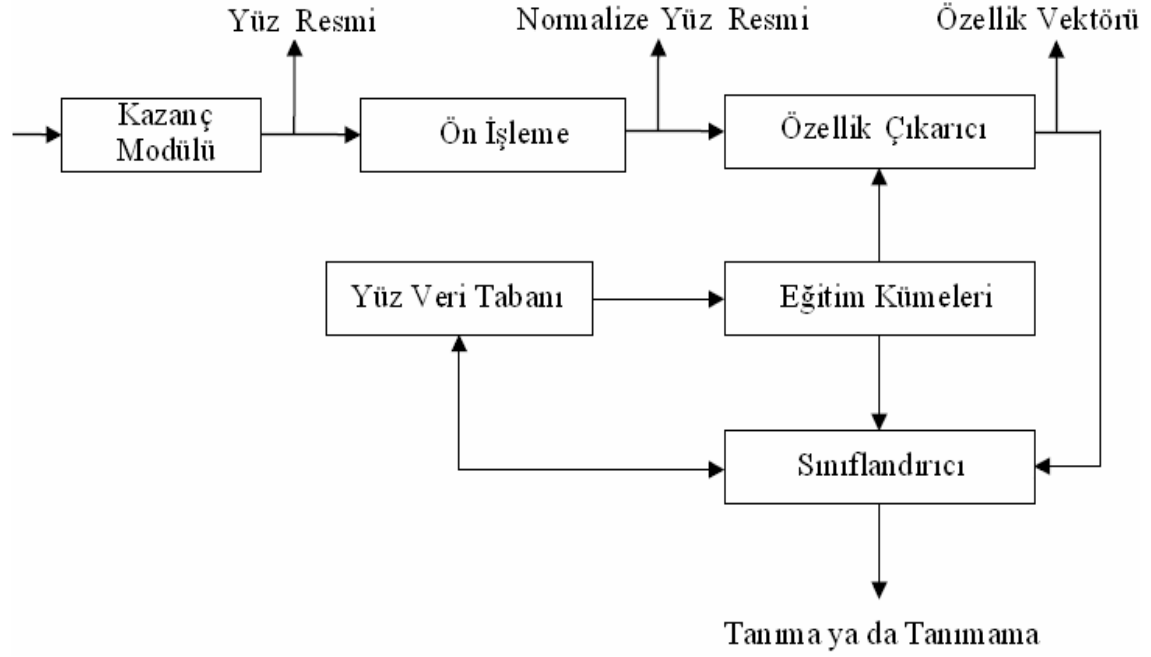
Bu sistemde yerine getirilen temel işlemler kayıt etme ve test etmedir. Kayıt etme işlemi sırasında sisteme verilen giriş verisinden elde edilen ve sadece o veriye özgün özellik kümesi biyometrik veri olarak saklanır. Test işlemi esnasında ise sisteme verilen

girdinin özellik kümesi ile sistemde kayıtlı girdilerin özellik kümeleri karşılaştırılır. Bu türde bir işlemin yapıldığı sistemlerin “güvenilir” olarak nitelenebilmesi için kaydetme ve kayıt çekme işlemlerinin güvenli olması gerekmektedir. Bu sistemin bileşenleri ve görevleri şu şekildedir :

- a. Sensör : Gerçek dünya ile tanıma sistemi arasındaki geçiş noktası olan ve görevi gerekli olan girdiyi elde etmek olan aygıttır. Uygulamada sensör genellikle bir resim kazanım aygıtıdır ama sistemin tasarım amacına göre değişiklik gösterebilir.
- b. Ön İşleme : Bu blok gerekli tüm ön işleme işlemlerini yerine getirir. Görevi sensörden alınan girdideki gürültüyü (görseller için bulanıklık, ışık patlaması, arka plan vb.) temizlemek ve girdiyi normalize formata sokmaktır.
- c. Özellik Çıkarıcı : Bu blok gereken özellik kümesini girdiden en verimli şekilde çıkarması gereken kısımdır. Burada amaç, girdiden kendisine özgün ayırt edici veriyi mümkün olduğunca çok ama kıyaslanacağı diğer kayıtlı örneklerle karıştırılmasına neden olmayacak kadar da az çıkartabilmektir.
- d. Şablon Oluşturucu : Bu blokta özellik çıkarıcı modülden alınan girdiye özgün veri, sayılardan oluşan bir vektör yardımıyla söz konusu girdiyi tanıtacak bir şablon haline getirilir. Şablon girdiyi tanıtan karakteristik özelliklerin yeterli tanımayla imkân verecek kadarlık kısmının saklandığı kapsüldür.
- e. Eş Bulucu : Eğer bir kaydetme işlemi yapılmakta ise çıkarılan şablonlar “Kayıtlı Şablonlar” bloğunda saklanır ancak bir karşılaştırma işlemi yapılacaksa Eş Bulucu ünitesi devreye girer. Bu ünite mevcut şablonlarla girdi şablonu arasındaki mesafeyi seçilen algoritma (Hamming distance, Euclidian distance vb.) ile hesaplayarak girdi şablonunun kayıtlı her şablondan farkını ortaya çıkarır. Ortaya çıkan fark değerleri kullanıcı seçimine göre değişen bir eşik değeriyle kıyaslanarak girdiye olan farkı kabul edilebilir sınırlar içinde kayıtlı şablon tanınan örüntü olarak seçilir.

2.2. TEMEL YÜZ TANIMA SİSTEMİ

Şekil 2.2’de biyometrik örüntü tanıma esasına dayanan bir temel yüz tanıma sisteminin şeması verilmiştir.



Şekil 2.2 : Temel Yüz Tanıma Sistemi

Bu sistemde Kazanç Modülü giriş resmini sisteme alan sensördür. Buradan alınan resim Ön İşleme basamağından geçirilip arka plan etkisi ve gürültüden temizlendikten sonra Özellik Çıkarıcı modülüne verilerek girdiyi ayırt etmekte kullanılacak özellik vektörü elde edilir. Eğer bir kayıt ekleme işlemi yapılmaktaysa bu vektör Eğitim Kümeleri modülünde sistemin tasarımında kullanılan yüz tanıma tekniğine göre, kayıtlı diğer yüz resimlerine ait özellik vektörleriyle beraber saklanır. Ancak bir tanıma işlemi yapılacaksa çıkarılan özellik vektörü eğitim kümelerindeki her vektörle Sınıflandırıcı isimli eş bulma modülünde önceden seçilen bir kıyaslama algoritmasına göre karşılaştırılır. Aranana en yakın (genellikle önceden belirlenen bir eşik değerinin sınırları içinde kalan) özellik vektörünün hesaplandığı yüz resmi Yüz Veri Tabanı modülünden çekilerek kullanıcıya “tanınan yüz” olarak sunulur. Ancak çıkarılan özellik vektörü eğitim kümelerinde kayıtlı özellik vektörlerine önceden belirlenen eşik değeri dahilinde benzerlik göstermiyorsa Sınıflandırıcı modülü tarafından kullanıcıya girdi resminin tanınmadığı bildirilir.

İyi bir yüz tanıma sistemi için önemli üç basamak vardır. Bunlar :

- 1) Yüz Algılama (Face Detection)
- 2) Ön İşleme (Pre-Processing)
- 3) Yüz Tanıma (Face Recognition)

basamaklarıdır.

2.2.1. Yüz Algılama

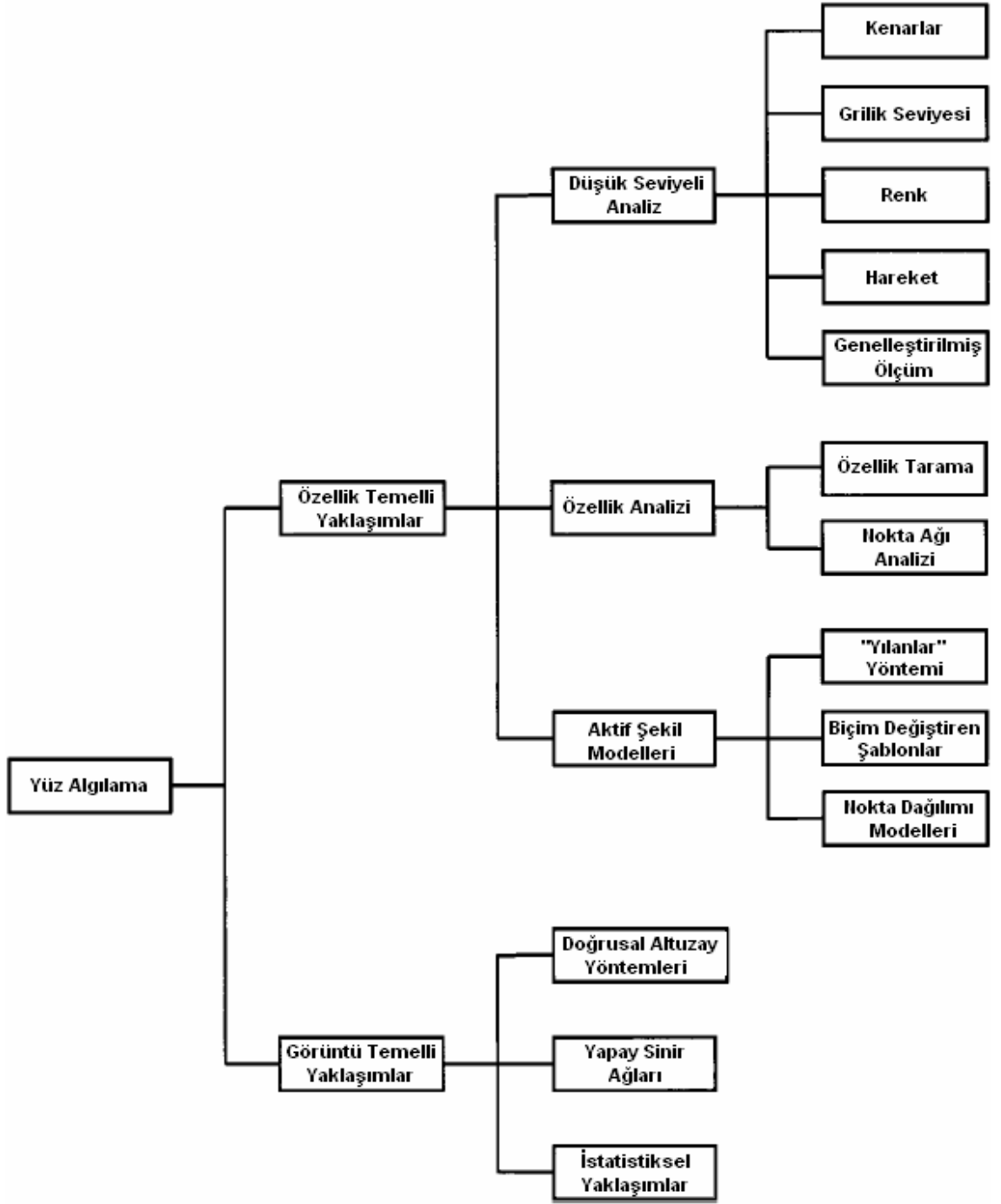
Yüz algılama, yüz tanıma sistemleri için ilk aşamada gerekli olan ve amacı bir resimdeki yüzü konumlandırıp arka plan verisinden kurtarmak olan bir işlemdir. Diğer bir tanım olarak yüz algılama işlemini özel bir tür nesne tanıma işlemi olarak da isimlendirebiliriz.

Günümüzde kullanılan yüz algılama teknikleri kenar belirlemeye dayalı yöntemlerden gelişmiş örüntü tanıma işlemine dayalı yöntemlere kadar çeşitlilik göstermektedir. Bu yöntemlerin tümü “Özellik Temelli” ve “Görüntü Temelli” yöntemler olarak iki sınıfa ayrılabilir [6].

Özellik temelli yöntemler, yüz algılama işleminde bilgiye dayalı analize bağlı olarak çıkarılan düşük seviyeli özelliklerden yararlanan klasik tespit yöntemleridir. Bu yöntemlerde tipik olarak deri rengi ve yüz geometrisi gibi özelliklerin mesafe, açı ve alan ölçümüne göre irdelenmesiyle yüz bulma işlemi yapılır.

Görüntü temelli yöntemler ise örüntü tanıma alanındaki son gelişmeler ile ön plana çıkmıştır. Yüzlerin görüntü temelli algılanmasında yüzdeki geometrik özelliklerden faydalanılmaz, bunun yerine yüz resmi piksel değerlerinin tamamını içeren sayı vektörleriyle ifade edilir. Daha sonra bu vektörler yapay sinir ağlarındaki gibi eğitim algoritmalarıyla işlenir.

Özellik temelli ve Görüntü temelli yaklaşımların literatürde geçen örnekleri Şekil 2.3’de verilmiştir.



Şekil 2.3 : Yüz Algılama Teknikleri

Bilgisayar görüntü işleme teknikleri içerisinde sıklıkla başvurulanan kenar belirleme yöntemi yüz algılama problemine yüz ait karakteristik özelliklerin çıkarılması için uygulanmıştır. Kenar belirleme yöntemi aynı zamanda gözlük gibi yüz protezlerinin algılamasında da kullanılabilir [7]. Sıklıkla kullanılan kenar belirleme operatörleri Sobel, Kirsch, Robinson, Laplace ve Canny operatörleridir.

Yüz algılama probleminde gri bölgelerin tespiti de işe yarar bir yöntemdir. Yüzümüzdeki kaş, göz bebeği ve dudak gibi kısımlar diğer bölgelere göre daha koyu bir renk tonuna sahiptirler ve bu nedenle yüz algılama işlemi için kullanılabilirler. Bu yöntem, yüzümüzdeki bölgelerin birbirlerine göre konumlarını içeren yüz şablonları ile kullanıldığında göz gibi hem yüz algılamaya hem de yüzün eğimini belirlemeye yardımcı olabilecek organlar tespit edilebilir [8].

Hareketli görüntülerde nesne algılama tekniği de yüz algılama probleminde kullanılabilir. Arka planda hiç ya da ihmal edilebilecek kadar az bir değişimin olduğu videolarda ardışık iki frame arasında yapılan incelemelerde vücudun hareketli kısımları ayırt edilebilir. Bu hareketin belirlendiği bölgede bir yüzün bulunup bulunmadığını algılamak için sınırları çıkarılan hareketli bölgeye elips uydurma tekniği uygulanabilir.

Yüz algılamada kullanılan bir diğer yöntem de Yapay Sinir Ağları'dır. Bu yöntemde yapay sinir ağı içinde yüz bulunana ve bulunmayan girdilerle eğitilip daha sonraki test girdileri için denenebilir. Yapay sinir ağları yönteminde sistemin başarısı giriş ve gizli katmanlardaki nöron sayılarına bağlıdır. Bu nöronların sayısındaki artış girdi verilerini daha iyi ayırt etmeye ancak sistem eğitim süresinin de uzamasına neden olur. Bu nedenle bu tür sistemlere verilecek yüz resimlerinde ölçekleme yöntemine gidilmektedir. Bu şekilde yapılan ilk çalışmalardan birinde [9] tasarlanan sinir ağı resimde 20x20 boyutundaki (400 piksel) pencerelerden aldığı resmi gizli katmanındaki 26 nöronla işlemiştir. Taranan alanda önceden seçilen bir eşik değerinden fazla sayıda yüz algılandığında o bölgede bir yüzün var olduğu kabul edilmiştir.

2.2.2. Ön İşleme

Bir yüz tanıma sisteminde ön işleme basamağı, sensör tarafından alınan giriş resmindeki arka planının ve sensörden kaynaklanabilecek gürültünün etkilerinin mümkün olduğunca aza indirildiği basamaktır. Bu aşamada yapılabilecek işlemler şunlardır [10]:

- a. Resim Boyutu Normalizasyonu : Yüz resminin tanıma sisteminin kabul edeceği boyutta yeniden ölçeklenmesi işlemidir. Genellikle yüz maskesinin kullanılmadığı uygulamalarda bu yöntem kullanılır.

- b. Histogram Eşitlemesi : Yüzdeki ayrıntıların daha iyi çıkarılabilmesi için resimdeki ışığın şiddetinin yeniden düzenlenmesi işlemidir.
- c. Medyan Filtreleme : Kameradan kaynaklanan gürültüyü yüzdeki ayırt edici özellikleri kaybetmeden çıkarma işlemidir.
- d. Yüksek Geçişli Filtreleme : Yüz hatlarına dayalı özellik çıkarıcı modüller kenar belirleme işlemine ihtiyaç duyabilirler. Bu aşamada yüksek geçişli filtreleme, yüzdeki sınırları belirginleştirerek kenar belirleme işleminin performansını artırabilir.
- e. Arka Plan Verisinin Kaldırılması : Resimde sadece yüz verisiyle ilgilenildiği durumlarda arka plan etkisinin kaldırılması işlemidir. Ancak bu işlemi yapabilecek ön işleme modülünün resimdeki yüz sınırlarını belirleyebilmesi gerekmektedir.
- f. Döndürme Normalizasyonu : Yüz tanıma sistemleri için üzerinde çalışması en kolay veri cepheaden çekilen ve başın dik durduğu resimlerdir. Ancak giriş verisinin bu koşulu sağlamadığı durumlarda tanıma performansını artırmak için yüz, dönüşüm matrisleri yardımıyla istenilen konuma gelinceye kadar Z ekseninde etrafında döndürülebilir. Bu aşamada yüzün eğimini bulmak için genellikle gözlerin konumundan faydalanılmaktadır.
- g. Aydınlatma Normalizasyonu : Farklı aydınlatma koşulları altında çekilen resimler Temel Bileşen Analizi (Principal Component Analysis) gibi yüz verisindeki tüm pikselleri kullanarak tanıma işlemini gerçekleştiren yöntemler için sorun oluşturabilir. Bu gibi yöntemlerde ideal olan yüzün her bölgesinin aynı şiddette aydınlatılmasıdır. Bu nedenle tanıma işleminden önce bu basamak uygulanabilir.

2.2.3. Yüz Tanıma

Yüzün algılanması ve yüz resminde tanıma performansını iyileştirmek için yapılan ön işlemlerden sonra yüz tanıma basamağına geçilebilir. Bu noktada iki yaklaşım seçilebilir. Bunlar “Özellik Temelli Yüz Tanıma” ve “Temel Bileşen Analizi İle Yüz Tanıma” yaklaşımlarıdır.

2.2.3.1. Özellik Temelli Yüz Tanıma

Özellik Temelli Yüz Tanıma yöntemi yüzdeki organların özelliklerinin ve bu organların birbirlerine göre konumlarının esas alındığı bir yaklaşımdır. Bu yaklaşım için kullanılabilecek olan etkin özellikler şu şekildedir :

- a. Birincil Özellikler : Bunlar yüz üzerindeki organların birbirlerinden bağımsız olarak sahip oldukları nicel özelliklerdir. Burun genişliği, ağız yayı uzunluğu gibi.
- b. İkincil Özellikler : Birincil özelliklerin de kullanılmasıyla elde edilen özelliklerdir. Bu özellikler yüzdeki organların birbirleri ile ilişkilerini inceler. Gözlerin ve ağzın birbirlerine olan uzaklıkları gibi.
- c. Üçüncül Özellikler : İfade ve yaşlanma gibi işlenmesi daha zor olan özelliklerdir. Bu özelliklerle ilgili bilgi birincil ve ikincil özellikler kullanılarak elde edilemez. Güvenilir bir yüz tanıma için değişken olmayan özellikler gerektiğinden bu sınıfa giren özellikler genelde tanıma işlemlerinde kullanılmazlar.

Özellik temelli yüz tanıma yönteminde esas olarak ikincil özellikler kullanılarak yüzdeki organların bulundukları bölgeler bir eşik değerinin yardımıyla kabaca belirlenir ve sonrasında birincil özelliklerin yardımıyla ayırt etme işlemine geçilir. Bu amaçla kullanılabilecek yöntemler şu şekildedir :

- 1) Aktif Hatlar Modeli (Yılanlar) : Yüzdeki kaş ve burun gibi bölgelerin uzunlukları değişken olduğundan bu bölgelerin çıkarılmasında Aktif Hatlar Modeli kullanılır [11][12]. Bu yöntemde aradığımız bölgenin burun olduğunu varsayalım. Burnun sınırlarını bulabilmek için burun çevresindeki bir bölgeden tarama başlatılır. Sonrasında yılan burun kenarlarına yerleşerek burun şeklini ortaya çıkarır. Bir yılan hattının değişimi bir enerji fonksiyonunun (E_{snake}) minimizasyonu ile sağlanır.

$$E_{snake} = E_{internal} + E_{external} \quad (2.1)$$

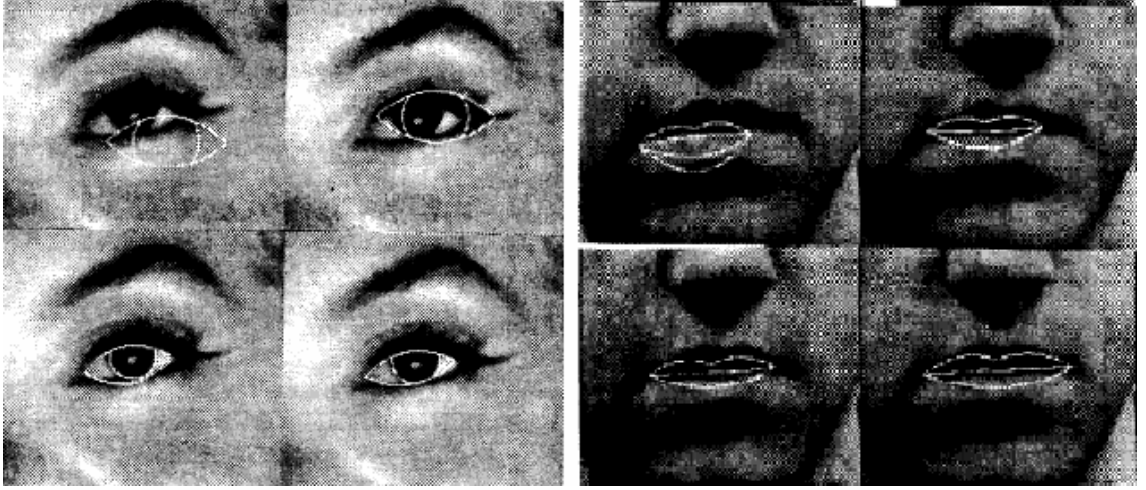
E_{internal} , yılan'ın kendisine has özelliklerine dayanan bir iç enerji fonksiyonudur. Bir yılan'ın tipik evrimi kısalmak ya da uzamak üzerine kuruludur. E_{external} yılan hattının sahip olduğu iç enerjiye zıt yönde etkiyerek 2.1 denklemini denge durumuna getirir ve burnun hatlarının belirlenmesine yardımcı olur.

- 2) Biçim Değiştiren Şablonlar : Biçim değiştiren şablonlar yüzden görüntüsü çıkarılmak istenen organın şekli hakkında önceden bilgi edinilmesiyle çalışır. Her biri yüzün bir organını simgeleyen bu şablonlar boyutları ve diğer geometrik değerleri değiştirilerek kendini organ modeline uydurabilir. Yüzdeki göz ve ağız gibi bölgeler bu yöntem ile belirlenebilir. Bu bölgelerdeki düşük ışıktandırma kenar belirleme işlemini sorunlu hale getirir. Bu nedenle Yuille [13] aktif hatlar modelini bir adım daha ilerleterek göz bölgesinin evrensel bilgisini kullanma yoluna gitmiştir. Çalışmasında şekil değiştirebilen bir göz şablonunu 11 parametre ile ifade etmiş ve göz bölgesinin yakınında başlatılan bir göz şablonunun şekil değiştirerek uygun sınırlara ulaştığını göstermiştir. Bu göz şablonu vadi (E_v), kenar (E_e), doruk (E_p) ve resim parlaklığına (E_i) bağlı olan 2.2 denklemi ile şekil değiştirmektedir.

$$E = E_v + E_e + E_p + E_i + E_{\text{internal}} \quad (2.2)$$

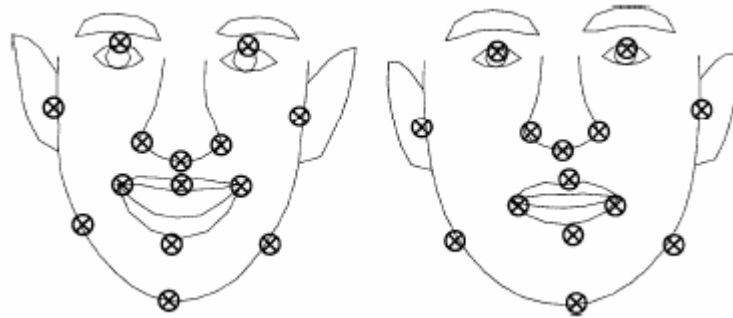
Biçim değiştiren şablonlar yönteminde dikkat edilmesi gereken bir çok ana öge vardır.Örneğin bir şablonun evrimi başlatıldığı nokta ile yakından ilişkilidir. Ağız bölgesinin üstüne yerleştirilen bir şablon burun bölgesine, göz bölgesinin üstüne yerleştirilen bir şablon da kaş bölgesine doğru genişleyebilir. Bu yöntem sıralı işlemeli bir evrim algoritmasına dayalı olduğundan işleme süresi uzundur. Ayrıca bahsi geçen enerji terimlerinin ağırlıklandırılması da deneme yanılma yöntemine dayanır. Bu zorlukların aşılması için bazı değişiklikler de yapılmıştır. Göz bölgesini diğer bölgelerden daha iyi ayırmak için Welsh [14] göz şablonunu yerleştirirken gözdeki beyaz alanlarla ilişkili yeni parametreler kullanmıştır. Chow [15] da iris bölgesini tespit ederek göz konumlandırma yoluna gitmiştir. İşleme zamanından tasarruf için de daha basit şablonlar kullanılmıştır.

Şekil 2.4'de biçim değiştiren şablonlar yöntemiyle ağız ve göz bölgesinin konumlandırılması [13] gösterilmiştir.



Şekil 2.4 : Biçim değiştiren şablonlar yöntemiyle göz ve ağız konumlandırma

- 3) Nokta Dağılımı Modelleri : Nokta dağılım modelleri şekil tabanlı istatistiksel yöntemlerin bir formudur. Bu yöntemde tarama yapılacak alanın sınırları önceden işaretlenmiş noktalar ile belirlenir. Bu noktaların dağılımı farklı boyut ve pozlardan oluşan bir eğitim kümesi üzerinden belirlenir. Temel Bileşen Analizi yöntemi ile doğrusal esnek bir model hazırlanır. Bu model eğitim setindeki tüm resimlerin özelliklerinin ortalamasından elde edilen noktalardan oluşur. Bu noktalar beraberlerinde gösterebilecekleri değişimin limiti hakkında da bilgiye sahiptir. Şekil 2.5’de Lanitis’in çalışmasında [16] geçen bir nokta dağılım modeli örneği görülmektedir.



Şekil 2.5 : Nokta dağılım modeli örneği

$$a = \bar{a} + Pv$$

(2.3)

2.3 denkleminde a , dağılım modelindeki bir noktayı, $\bar{a}$, bu nokta için eğitim kümesindeki ortalama özelliği, P , eğitim kümesinden edinilen kovaryans matristen çıkarılan yüksek değerli vektörlerden oluşan bir matrisi ve v de bir ağırlıklandırma vektörünü ifade etmektedir.

Bu yöntem kullandığı noktalar ile yüzdeki göz, kaş, ağız, burun gibi varlığı değişmez organları konumlandırmaktadır. Bir nokta dağılımı modelini bir yüze oturtmak için öncelikle $\bar{a}$ değerlerinden oluşan ortalama dağılım maskesi resmin üzerine yerleştirilir. Sonra bu maskedeki her nokta gri seviyeli tarama yöntemiyle resimde eşi olan bölgeye taşınır. Bu işlem sırasında maskenin, sadece eğitim kümesinden edinilen bilgiyle tutarlı bir şekilde biçim değiştirmesine izin verilir.

Günümüzde nokta dağılımı modelleri yaklaşımı yüz verisinden ayırt edici kodların çıkarılması (Hashing), yüz algılama işlemi için istatistiksel verilerin hesaplanması veya yüzdeki ifadenin yorumlanması gibi konularda kullanılabilmektedir.

2.2.3.2. Temel Bileşen Analizi İle Yüz Tanıma

Yüz tanımda kullanılan ikinci yöntem ise “Temel Bileşen Analizi” (TBA) adıyla anılan ve yaygınlıkla kullanılan bir alt uzay yöntemidir [17]. Temel bileşen analizi yöntemi çok boyutlu veri kümelerini kolay analiz etmek için daha düşük boyutlu veri kümelerine indirme işlemidir. Bu yaklaşımda yüzü en az ama öz bilgi ile ifade etmek esastır. Bu yöntemde yüzdeki organlar birbirlerinden bağımsız olarak düşünülmemektedir.

Başlarda Sirovich ve Kirby [18] insan yüzlerini o zaman kadar önerilen yöntemlerden daha etkili bir şekilde ifade edebilmek için temel bileşen analizi yöntemini kullanmıştır. Yöntem farklı yüzlerden oluşan bir yüz setinden elde edilen kovaryans matrisi kullanarak mevcut yüzlerin dağılımını özvektörler ile ifade etmekteydi. Bu kümedeki her yüz uygun ağırlıklandırma katsayıları ile en büyük özvektörlerin doğrusal birleşimi ile ifade edilebilmekteydi. Turk ve Pentland [17] işi bir adım ileri götürerek temel

bileşen analizini yüz tanıma işlemi için kullanmışlardır. Çalışmalarında her resim için hesaplanan özvektörlerden elde edilen özyüzlerin ağırlıklarının yüzleri ayırt etmeye yarayan özgün değerler olduğunu göstermişlerdir. Özyüzleri kullanarak yeniden oluşturulan yüz resimlerinin orijinal resimle arasındaki farkı “yüz uzayına olan mesafe” (distance-from-face-space) DFFS olarak isimlendirmişlerdir. Bu çalışmaları, takip eden araştırmalara sadece yüz tanıma konusunda değil aynı zamanda hesaplanan DFFS değeri ile yüz algılama ve yüz organlarının özelliklerinin çıkarılması [19] (özgöz, özburun, özağz gibi) konularında da ışık tutmuştur.

Bu tez çalışması kapsamında da Turk ve Pentland’ın öne sürdüğü Özyüzler ile Yüz Tanıma yönteminden faydalanılmıştır. Yöntem üçüncü bölümde detaylı olarak anlatılmaktadır.

2.3. BAŞLICA YÜZ TANIMA YÖNTEMLERİ

Günümüze kadar yüz tanıma problemine çözüm olarak çeşitli yöntemler öne sürülmüştür. Bu yöntemlerin en sık kullanılanları [20] aşağıdaki gibidir.

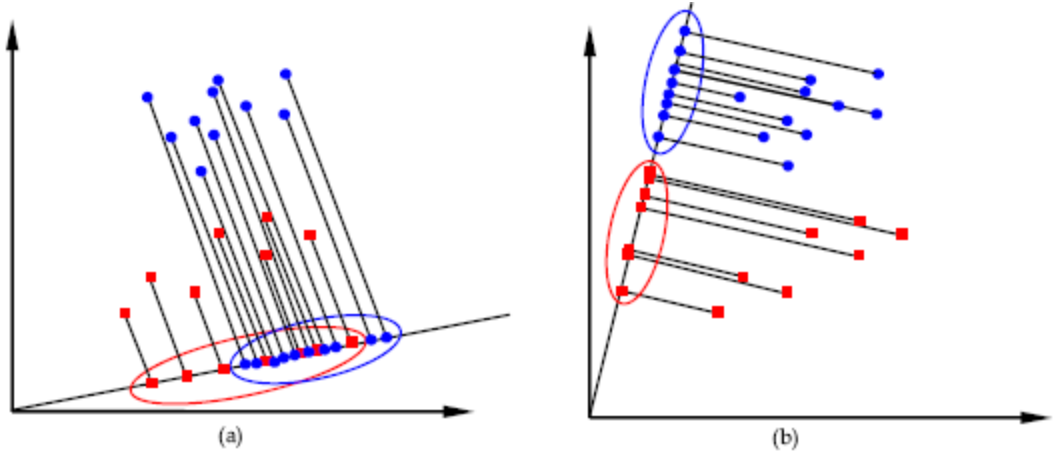
- 1) LDA (Doğrusal Diskriminant Analizi)
- 2) EBGM (Elastik Salkım Grafiği Eşleştirme)
- 3) 3 Boyutlu Yüz Tanıma
- 4) Genetik Algoritmalar İle Yüz Tanıma
- 5) Yapay Sinir Ağları İle Yüz Tanıma
- 6) PCA (Temel Bileşen Analizi – Özyüzler Yöntemi)
- 7) Destek Vektör Makineleri (DVM) Yardımıyla Yüz Tanıma

Bu yöntemlerden son üçü olan Yapay Sinir Ağları, PCA (Özyüzler) ve DVM ile yüz tanıma yöntemleri bu tez çalışmasına konu olarak seçildiklerinden çalışmanın 3. kısmı olan malzeme ve yöntem bölümünde detaylı olarak anlatılacaktır.

2.3.1. LDA (Doğrusal Diskriminant Analizi)

Doğrusal diskriminant analizi (Linear Discriminant Analysis – LDA) yöntemi 1936 yılında R.A. Fisher tarafından bir kümedeki örnekleri sınıflandırma amacıyla kullanılan bir yöntem olarak ileri sürülmüştür. Yöntemin yüz tanıma problemine uyarlamasında yöntemin mucidi Fisher'in adı kullanılmış ve “Fisherfaces” ismi kullanılmıştır.

Doğrusal diskriminant analizi yöntemi de temel bileşen analizi yöntemi (PCA) gibi veri sınıflandırma ve boyut küçültme işlemleri için kullanılır. Bu yöntemde test kümesini oluşturan veriler birbirlerine benzerlikleri açısından ortak sınıflarda gruplanır. Sonrasında önemli olan sınıflar arası dağılım matrisinin determinantının sınıf içi dağılım matrisinin determinantına oranını maksimize etmektir. Bu şekilde test kümesini oluşturan veriler, kendi içinde varyansı düşük ama birbirlerine göre varyansı büyük sınıflar halinde ifade edilir. Şekil 2.6(a)’da test kümesindeki verilerin herhangi bir izdüşümü ile sağlanan “kötü sınıf dağılımı” ve Şekil 2.6(b)’de noktaların LDA izdüşümü ile sağlanan “iyi sınıf dağılımı” görülmektedir.



Şekil 2.6 (a,b): Bir grup noktanın projeksiyonu

Konunun yüz tanıma problemine uygulanan hali olan Fisherfaces yönteminde amaç benzer yüz resimlerini veya aynı yüzün farklı ışıklandırma, arka plan vb. durumlarda çekilen resimlerini ortak bir sınıfta ve farklı yüz resimlerini de başka sınıflarda gruplayarak en iyi ayırt etmeyi sağlamaktır.

Sınıf içi dağılım matrisi S_w , ve sınıflar arası dağılım matrisi S_b , denklem 2.4 ve 2.5’deki gibi elde edilir.

$$S_w = \sum_{j=1}^Y \sum_{i=1}^{N_j} (\Gamma_i^j - \mu_j)(\Gamma_i^j - \mu_j)^T \quad (2.4)$$

$$S_b = \sum_{j=1}^Y (\mu_j - \mu)(\mu_j - \mu)^T \quad (2.5)$$

Burada Γ_i^j , j. sınıfın i. örneğini, μ_j , j. sınıfın ortalamasını, Y sınıf sayısını, N_j , j. sınıftaki örneklerin sayısını ve son olarak da μ , tüm sınıfların ortalamasını ifade etmektedir. Sınıf içi ve sınıflar arası dağılım matrisleri, LDA uzayını geren $H = [H_1, H_2, \dots, H_d]$ vektör kümesinin hesabında kullanıldığında amaç $\left| \frac{H^T S_b H}{H^T S_w H} \right|$ ’yi olabildiğince büyük değerli kılmaktır. H vektörleri, $S_w^{-1} S_b$ işleminden çıkan matrisin özvektörleridir.

Fisherfaces yöntemi “Malzeme ve Yöntem” başlığında anlatılacak olan “Özyüzler” yöntemiyle büyük benzerlik göstermektedir. Her iki yöntem de veri kümesinden kovaryans matrisler elde edip bunların özvektörlerini kullanarak bazı ayırt edici özellik vektörleri oluşturmak üzerine kuruludur. Ancak özyüzler yönteminde tüm yüzler tek bir sınıfta işlenirken Fisherfaces yönteminde doğrusal diskriminant analizi ile oluşturulan farklı sınıflarda işlenir.

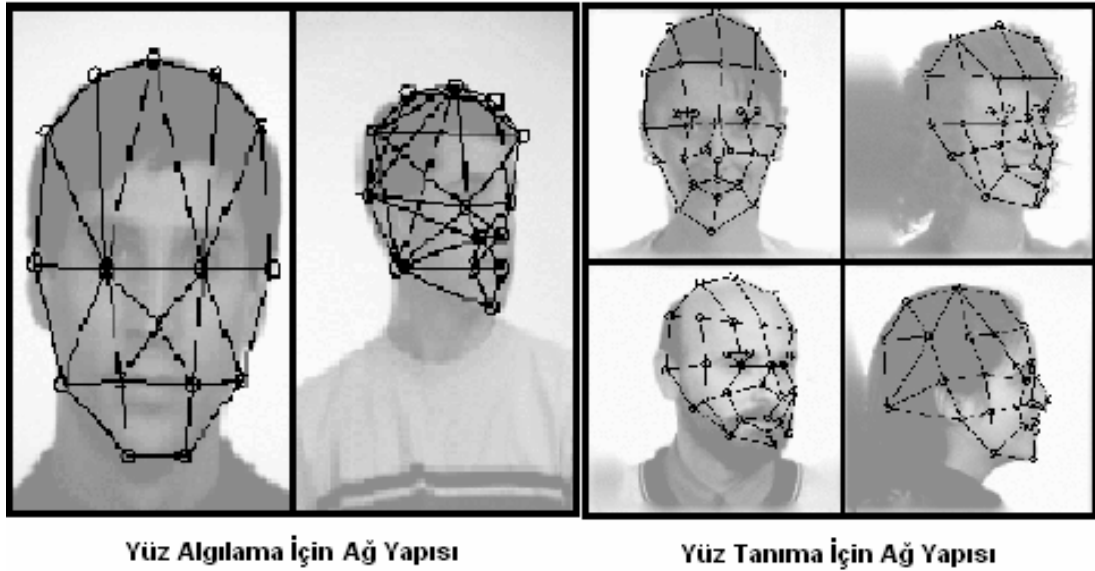
2.3.2. EBGM (Elastik Salkım Grafiği Eşleştirme)

Elastik Salkım Grafiği Eşleştirme (Elastic Bunch Graph Matching – EBGM) yöntemi özellik tabanlı eşleştirme yöntemlerinin en başarılılarından biridir. Yöntem yüzdeki göz bebeği, ağız kenarları, burun ucu, kulakların altı ve üstü vb. gibi bir kısım temel bölgedeki düğümlerden oluşturulan ağın yüz resmine konumlandırılması esasına göre çalışır.

EBGM yönteminde kullanılan yüz ağında N adet düğüm ve bu düğümleri birleştiren E adet kenar olduğunu varsayalım. Bu düğümlerin her biri J_n , $n = 1, \dots, N$ jetleri ile kenarlar da $\vec{\Delta x_e} = \vec{x_n} - \vec{x_{n'}}$, $e = 1, \dots, E$ ile gösterilen mesafeler ile işaretlidir (Burada e kenarı n ve n' düğümlerini birleştirmektedir.). Bir jet bir düğümün etrafındaki gri bölgeleri tanımlayan Gabor dalgacıkları kümesidir.

Yüzdeki temel bölgelerle ilişkili noktaların jet'lerinin kümesi "salkım" olarak adlandırılır. Örnek olarak bir ağız salkımı mevcut resimlerden elde edilen açık, kapalı, düz, yuvarlak vb. ağızların jet'lerini içerebilir. Bu salkımdan elde edilen jet değerlerinin kullanılmasıyla bir yüz resmindeki ağzın konumu ve durumu bulunabilir (Yüz ifadesi algılama). Düğümler arasındaki kenarların uzunluklarından da yüz tanıma işlemi yapılabilir.

Oluşturulan bir yüz ağında yüzün ayırt edici temel noktalarını yüzdeki her bölgeden eşit sayıda seçmek esastır. Ama amaca göre istenen bölgelere daha fazla sayıda düğüm yerleştirilebilir. Yüz algılama işlemi için kafanın dış hattına, yüz tanıma işlemi için de iç hattına daha fazla sayıda düğüm konabilir [21]. Bu durum Şekil 2.7'de görülmektedir.



Şekil 2.7 : EBGM yüz ağı modelleri

EBGM yöntemi, yüzdeki temel bölgeler düğümlerle işaretlendiğinden dolayı farklı açılardan verilen pozlarda dahi başarılı olabilmektedir. Ayrıca sadece yüz tanıma değil aynı zamanda yüz ve yüz ifadesi algılama gibi konularda da kullanılabilir. Ancak bu avantajlarına karşılık test kümesindeki yüzler için düğüm dağılımının elle yapılması ve yöntemin performansının ışıktandırma ile etkilenmesi dezavantajdır.

2.3.3. Üç Boyutlu Yüz Tanıma

Üç boyutlu yüz tanıma, tanıma işlemi için insan yüzünün 3 boyutlu geometrisinin kullanıldığı bir yöntemdir. Üç boyutlu yüz tanıma işleminin diğer yöntemlere göre en önemli avantajı yüzdeki katı noktaların sadece genişlik ve yükseklik değil aynı zamanda derinlik bilgisine de sahip olmasıdır. Böylece bu yöntem diğerleri gibi ışıktandırma, makyaj, yüz protezleri ve kafanın konumu gibi etkenlerden etkilenmeden tanımayı sağlayabilmektedir.

Bu avantajlarına karşılık 3 boyutlu yüz tanıma sistemlerinin en büyük dezavantajı kazanç modülünün karmaşıklığıdır. Yüzün üç boyutlu görüntüsünü elde etmek için “range imaging” adı verilen ve yüz üzerindeki noktaların yükseltilerini belirleyen bir yöntem kullanılmalıdır. Yüzün tam üç boyutlu haritasını çıkarmak yerine ön cepheden ve profilden alınan pozları da tanımayı iyileştirebilir [22]. Pratik uygulamalarda 2 boyutlu görüntü elde eden kameralar kullanıldığından ve maliyetinden dolayı bu yöntem üzerinde daha az çalışma yapılmaktadır.

Bu konuda yapılan çalışmalardan birinde [23] sabit yüz ifadesi ve ışıktandırma altında yüzün farklı açılardan resimleri çekilmiştir. Sonrasında çekilen resimlerden, yüzlerin yüz sınırları, göz bebekleri, burun sınırları gibi yüzü karakterize eden 83 kilit noktası yardımıyla üç boyutlu modeller oluşturulmuştur. Bu modellerin bir örneği Şekil 2.8’de görülmektedir. Son olarak da modelin üzerine cepheden çekilmiş yüz resmi dokusu yerleştirilmiştir. Yüz tanıma işleminde de önceden oluşturulan modeldeki test noktaları mevcut modellerinkiyle kıyaslanarak performans ölçümü yapılmıştır. Sonuç olarak da cepheden çekilmeyen pozlara dayalı bu sistemin sadece cepheden çekilen pozlara dayalı klasik uygulamalara göre daha başarılı olduğu belirlenmiştir [24].



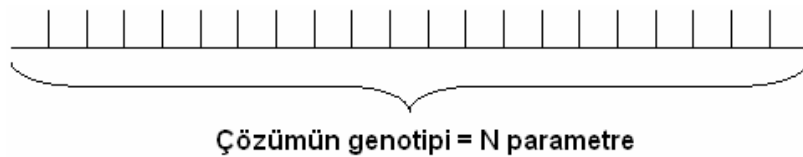
Şekil 2.8 : Farklı açılardan alınan pozlarla oluşturulan üç boyutlu yüz modeli
(1. ve 3. sıralarda çekilen gerçek pozlar, 2. ve 4. sıralarda bunlara bağlı olarak hazırlanan üç boyutlu model görülmektedir.)

2.3.4. Genetik Algoritmalar İle Yüz Tanıma

2.3.4.1. Genetik Algoritmalar Hakkında

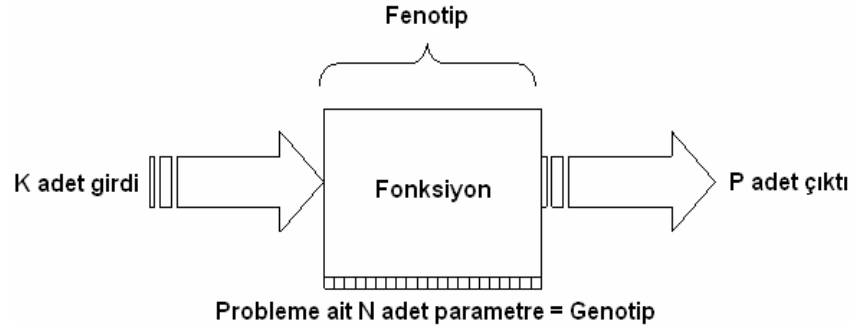
Evrım teorisine göre doğadaki tüm türler sürekli bir gelişim içindedir. Bu gelişime ayak uyduranlar hayatta kalmayı başarırken ayak uyduramayanlar yok olmaktadır. 1960 yılında araştırmacı John H. Holland bu gerçeği bilgisayar programları üzerinde denedi ve kendilerini belirtilen bir fonksiyona adapte edebilen “Genetik Algoritmaları” geliştirdi.

Genetik algoritmalar yöntemi çözümün optimizasyonu işlemi için tanımlanan iteratif döngülerden oluşur. Bu yöntemde temel olarak, seçilen uygunluk fonksiyonu bir çözümün probleme uygunluğunu ölçer. Problem için öne sürülen her çözüm “parametre” ismi verilen sayı kümeleriyle gösterilir. Bu parametrelere “gen” adı verilir. Genlerin birleşimiyle genotipler oluşur. Şekil 2.9’da bir genotip örneği görünmektedir.



Şekil 2.9 : N parametrelili bir genotip

Bu parametre kümeleri (genotipler) Şekil 2.10'daki gibi “fenotip” adı verilen seçim fonksiyonlarından geçirilerek en iyi genotipler elde edilmeye çalışılır. Burada önemli olan her problem için probleme uygun fenotiplerin seçilmesidir.



Şekil 2.10 : K adet parametrenin bir fenotipten geçirilişi

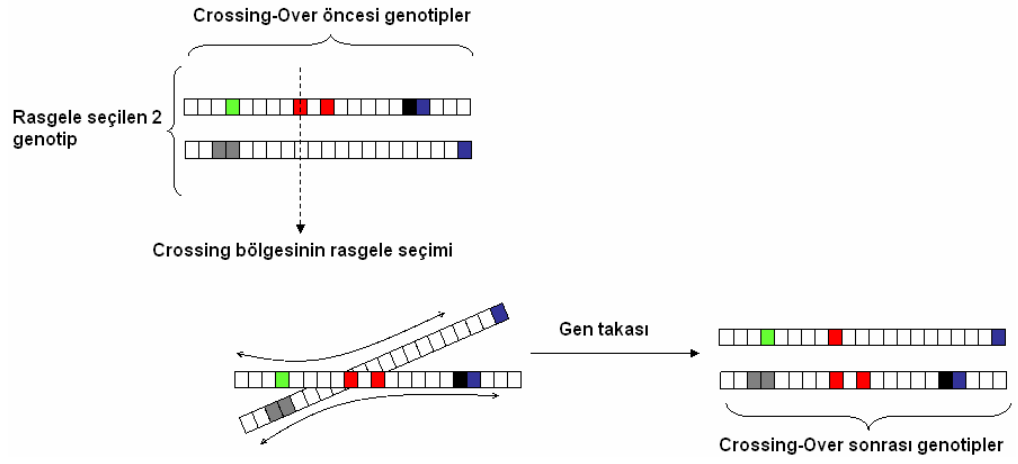
Bu aşamadan sonra fenotip fonksiyonu ile seçilen genotipler uygunluk fonksiyonu ile elenerek en uyumlu genotipler elde edilir. Genotiplerin çözüme uygunluğu arttıkça genetik algoritmadaki evrim döngüsünün tekrarlanma sayısı da artar. Bu döngü sırasında genotiplerin genlerinde değişiklikler meydana gelir. Bu değişiklikler iki şekilde olabilir :

- 1) Mutasyon : Bu işlem bir genotipteki rasgele seçilen bir genin Şekil 2.11'deki gibi değiştirilmesiyle elde edilir.



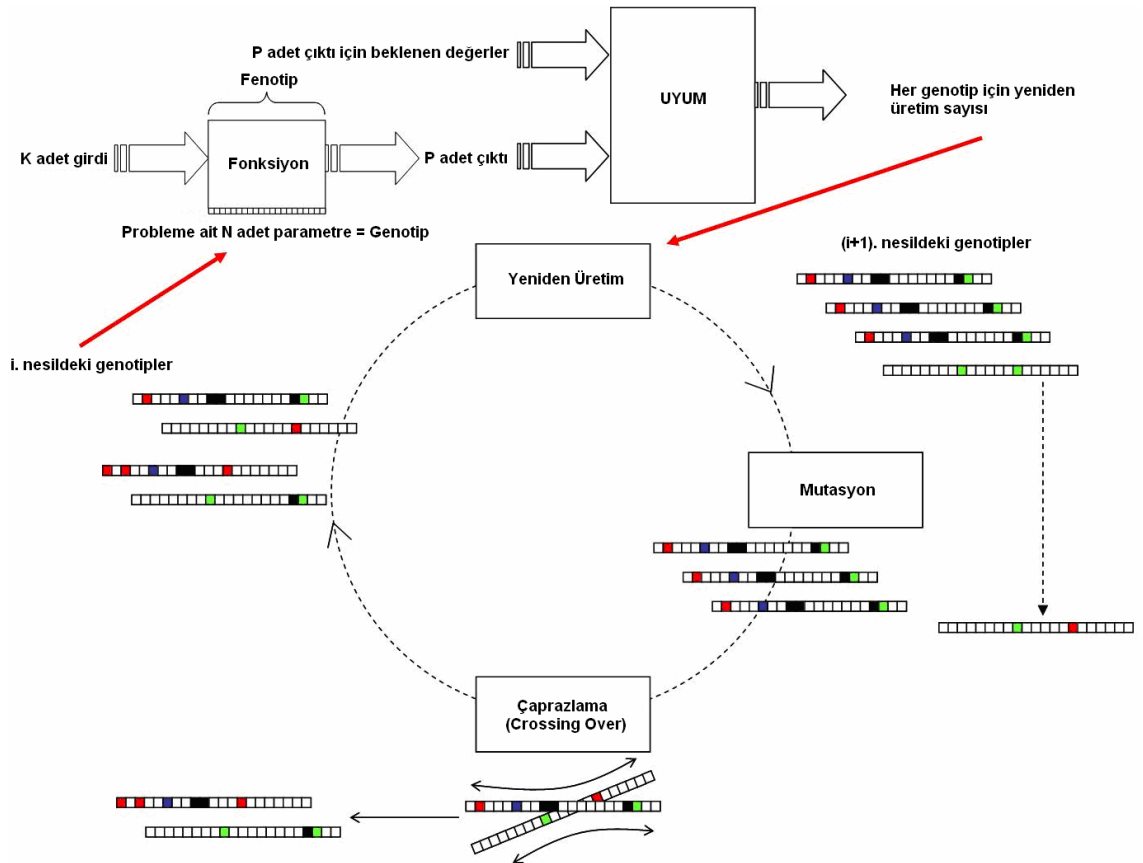
Şekil 2.11 : Mutasyon işlemi

- 2) Çaprazlama (Crossing – Over) : Bu işlemde rasgele seçilen iki genotip yine rasgele seçilen bir bölgeden sonraki genlerini Şekil 2.12'deki gibi değiştirirler.



Şekil 2.12 : Crossing - Over işlemi

Genetik algoritma döngüsündeki tüm basamaklar Şekil 2.13'deki gibidir.

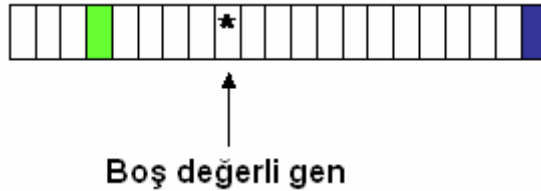


Şekil 2.13 : Genetik Algoritma Döngüsü

2.3.4.2. Genetik Algoritmaların Karakteristik Özellikleri

Genetik Algoritmaların karakteristik özellikleri şu şekildedir :

- Bir problem için pek çok sayıda “çözüm” (çok sayıda genotip ve fenotip) bulunur.
- Çok sayıda çözüm çaprazlama (crossing-over) işlemi ile bir araya getirilir. Böylelikle sadece sağ kalan çözümlerin parçalarını içeren yeni genotipler ortaya çıkar. Bu yeni genotiplerin probleme uyumluluğu rasgele seçilen genotiplerdeki rasgele değişimlerden elde edilen sonuçlardan daha yüksek olacaktır. Ancak probleme daha iyi uyum sağlayan genlerin çaprazlama ile oluşturulabilmesi için genler arasında “topoloji” özelliği bulunmalıdır. Bunun anlamı birbirine yakın genotiplerin çaprazlama sonuçlarının yine birbirine yakın olması gerektiğidir. Aksi takdirde yapılan işlem, uyum fonksiyonu tarafından elenecek olan rasgele yeni bir genotipin oluşmasına sebep olacaktır.
- Fenotipleri probleme çok iyi uyum sağladığından yeniden üretim işlemi sonucu birebir aynı pek çok genotipin oluşması mümkündür. Bu genotipler çaprazlama işlemi ile değişmeyeceğinden daha yavaş bir yöntem olan mutasyon ile değişirler.
- Genotiplerde boş değerli genler (wild cards) kullanmak mümkündür. Böyle genler mevcut tüm değerleri alabilir. Boş değerli gen içeren bir genotip, uyumluluk hesaplamasında söz konusu genin alabileceği tüm değerler göz önüne alınarak yeni genotiplere dönüşür. Şekil 2.14’de boş değerli bir gene sahip genotip örneği verilmiştir.



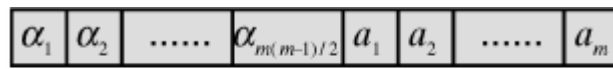
Şekil 2.14 : Boş değerli gen içeren bir genotip

2.3.4.3. Genetik Algoritmaların Yüz Tanıma Problemine Uygulanması

Genetik algoritmalar yöntemi ile yüz tanıma konusu üzerine yapılan en güncel çalışmalardan biri yüzleri ifade etmek için en iyi özelliklerin seçimine dayanan “Evrimsel Arayış” (Evolutionary Pursuit) [25] yöntemidir.

Evrimsel arayış (EA) yöntemi tanımlanan PCA alt uzaylarındaki rotasyon eksenleri boyunca yapılan bir arama işlemine dayanır. Yöntemin amacı Özyüzler yönteminde yüzlerin izdüşümünün alınacağı uzayın baz vektörlerini en iyi örüntü tanıma ve veri sıkıştırma sağlayacak şekilde genetik algoritma döngüsü ile seçmektir. Çalışmadaki evrim özelliği performans, doğruluk ve sınıflandırma indeksi terimlerine dayanan bir uyumluluk fonksiyonu ile sağlanmıştır. Doğruluk, öğrenme işleminin o ana kadarki başarısını gösterirken, sınıflandırma indeksi de gelecekteki denemelerde elde edilmesi beklenen uyumluluğu göstermektedir. Büyük değere sahip bir sınıflandırma indeksi gelecek denemelerden elde edilecek başarının yüksek olması için ek öğrenme adımlarına yol açmaktadır. Bu indeks ile öğrenme işleminin miktarı ve ne zaman sonlandırılacağı belirlenebilmektedir.

Yöntemde m boyutlu uzayda baz vektör çiftlerinin kendi aralarında $(0 - \Pi/2)$ arasında değişen açılarla döndürülerek en uygun bazların bulunması hedeflenmiştir. Her baz çifti arasındaki dönüş açısı 10 bitlik bir sayı ile ifade edilmiştir. Bu açılar α_k ile gösterilmektedir. Toplamda $m(m-1)/2$ adet baz çifti söz konusudur. Bunun yanında hangi eksenlerin döndürme işlemine tabi tutulacağını belirtmek için de m adet 1 bitlik yapı kullanılmıştır. Eksenler a_k ile gösterilmektedir. Şekil 2.15’de yöntemde kullanılan bir genotip yapısı görülmektedir.



Her α_k için 10 bit

Her eksen için 1 bit

Şekil 2.15 : Evrimsel Arayış yönteminde kullanılan bir genotip

Bu genotip evrim döngüsünde mutasyon ve çaprazlama işlemlerine tabi tutulup uyum fonksiyonundan geçirilerek uygun rotasyon açısı-baz çiftinin bulunmasında kullanılır.

3. MALZEME VE YÖNTEM

Bu çalışma kapsamında hazırlanan uygulama programı Visual Studio 6.0 IDE'sinde Visual Basic ile yazılmıştır. Program Pentium D 2.8 GHZ işlemcili ve 1 GB RAM'li bir sistem üzerinde geliştirilip test edilmiştir.

3.1. YAPAY SİNİR AĞLARI

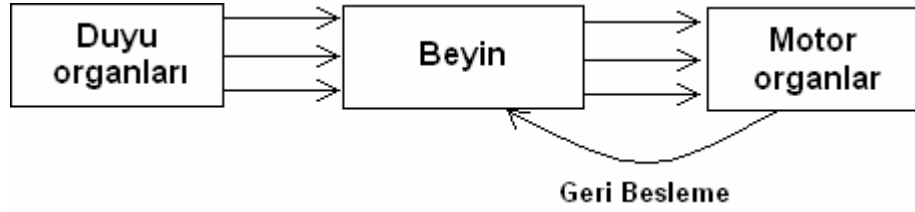
Yapay sinir ağları karmaşık problemlerin çözümünde kullanılabilen ve “nöron” adı verilen basit işlem elemanlarının kendi aralarında değişen etki katsayıları ile oluşturduğu sistemlerdir. Bu yöntem ilham kaynağını merkezi sinir sistemi ve nöronlar üzerine yapılan çalışmalardan almıştır. Bu sayede yapay sinir ağları ile problem çözümüne yönelik olarak kimseden yardım almadan öğrenen, yeni veriler oluşturan ve bilinen veriyi güncelleyebilen bilgisayar sistemlerinin tasarlanması hedeflenmiştir.

Günümüzde yapay sinir ağları çözümü için çok karmaşık algoritmaların tasarlanması gereken, maliyetli problemlerin çözümü için kullanılmaktadır. Paralel bilgi işleme yetenekleri, kolay tasarlanabilir olmaları, fazla değiştirilmeden farklı türlerde problemlere uygulanabilmeleri nedeniyle yapay sinir ağları çok çeşitli alanlarda hayata geçirilebilmektedir. Örneğin bir sistemin bilinen bir girdiye nasıl tepki vermesi gerektiğini öğrenen bir sinir ağı arıza analizinde, EEG gibi tıbbi sinyallerin analizi için eğitilen sinir ağları tıbbi cihazlarda, belirli nesneleri tanıma üzerine eğitilen sinir ağları nesne takibi ve örüntü tanıma sistemlerinde, bir üretim bandından sağlam çıkan ürünleri tanımak üzere eğitilen sinir ağları da fabrikalarda otomasyon sistemlerinde kullanılabilir.

Yapay sinir ağlarının en önemli yapı taşı olan nöronu anlayabilmek için bölüm 3.1.1'de konunun kökeni olan biyolojik sistemler verilmektedir.

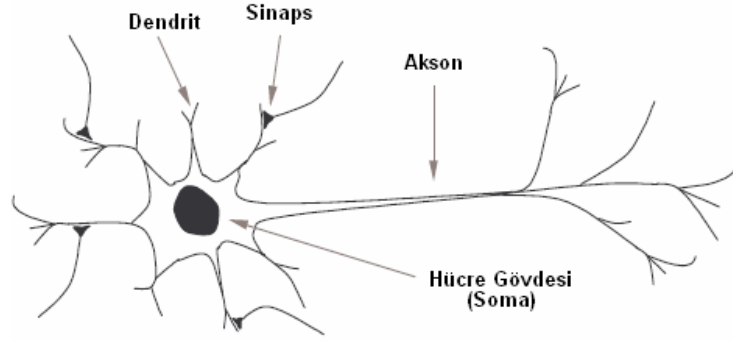
3.1.1. Yapay Sinir Ağlarının Kökeni

Bilgisayarda uygulanan yapay sinir ağları yönteminin kaynağı biyolojik sinir sistemidir. Biyolojik sinir sistemi çevreden gelen etkileri algılayan duyu organları, beynin ürettiği elektriksel sinyalleri yorumlayarak cevap veren motor organlar ve duyu organları ile motor organlar arasında ileri ve geri beslemelerle tepkiler üreten beyinden oluşur. Duyu organlarından alınan veriler merkezi sinir sistemindeki beyinde mevcut verilerle karşılaştırılır ve geliştirilir. Beynin ürettiği cevaplar da elektriksel sinyaller ile motor organlardaki alıcı sinirlere iletilerek istenen mekanik hareket gerçekleştirilir. Bu işlemler sırasında motor organlar geri besleme yaparak beyine o anki durumlarını bildirirler. Şekil 3.1’de bir sinir sistemindeki bilgi akışı görülmektedir.



Şekil 3.1 : Sinir sistemindeki bilgi akışı

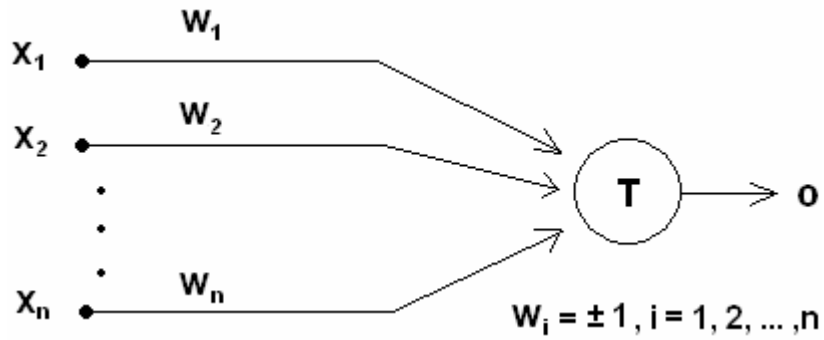
Bu sistemde en yüksek önceliğe sahip kısım etkilere karşı tepkiler üreten beyindir. Beyin her biri kendi başına bir işlem birimi olan “nöron” adı verilen hücrelerin oluşturduğu bir ağ sistemidir. Bir nöron hücre gövdesi (soma), akson ve dendrit’lerden oluşur. Dendrit’ler bir ağaç gibi dallanarak çok sayıda nörona bağlanırlar ve görevleri nörona komşu nöronların aksonlarından gelen mesajları almaktır. Aksonların görevi ise bir nörondan komşu nöronlara mesaj iletmektir. Bir nöronun aksonu ile diğer bir nöronun dendriti arasında bir temas organı olan “sinaps” bulunur. Sinaps’larda bir nörondan diğerine mesaj aktarımı gerçekleşir. Bir nöronun akson kolu boyunca mesaj elektriksel sinyal olarak iletilir. Ancak bu mesaj diğer nöronun dendrit’ine sinaps bölgesinde elektro-kimyasal bir süreçle iletilir. Şekil 3.2’de biyolojik bir nöron yapısal elemanlarıyla birlikte verilmiştir.



Şekil 3.2 : Biyolojik Nöron

3.1.2. Yapay Sinir Ağlarının Gelişimi

Yapay sinir ağlarının gelişiminin miladı 1943 yılı olarak kabul edilir. Bu yılda McCulloch ve Pitts hesaplama işlemleri için temel bir nöron tanımlamışlardır. Model aritmetik-lojik işlemleri yerine getirebilecek şekilde tasarlanmıştı ve bir çıktı verebilmesi için hesapladığı etkiyi bir eşik değeri ile kıyaslaması gerekmektedir. Ancak modelin hayata geçirilmesi o günlerdeki teknolojik yetersizlikten dolayı gerçekleşmemiştir. Bu model Şekil 3.3’de görülmektedir. Nöronun tetikleme kuralı da denklem 3.1’de verilmiştir.



Şekil 3.3 : McCulloch ve Pitts nöron modeli

$$o^{k+1} = \begin{cases} 1, & \text{eger } \sum_{i=1}^n w_i \cdot x_i^k \geq T \\ 0, & \text{eger } \sum_{i=1}^n w_i \cdot x_i^k < T \end{cases} \quad (3.1)$$

1949 yılında Donald Hebb nöronların bağlantılarındaki ağırlık katsayılarını güncelleştirmek için ilk yöntemi öne sürmüştür. Bugün bu yöntem Hebbian öğrenme metodu olarak bilinmektedir. Bu metot yapay sinir ağları alanında yapılacak yeni araştırmaların tetikleyicisi olmuştur.

1954 yılında ilk nörobilgisayar Minsky tarafından üretilmiştir. 1958 yılında da Frank Rosenblatt tarafından nöron benzeri bir makine olan “perceptron” geliştirilmiştir. Bu makine istenen örüntüleri sınıflandırmak üzere eğitilebilen bir makine olarak tasarlanmıştı. Bu fikir günümüzdeki uygulamalara da örnek olmuştur.

1960’lı yılların başlarında Bernard Widrow ve Marcian Hoff tarafından yeni bir öğrenme yöntemi olan Widrow-Hoff öğrenme kuralı öne sürüldü ve ADALINE (ADaptive LINEar combiner) isimli yapay sinir ağı temelli bir makine tanıtıldı. ADALINE’ın sonraki sürümü olan MADALINE (Many ADALINES) ile örüntü tanıma, hava durumu tahmini ve uyarlamalı kontrol işlemleri gerçekleştirildi.

1965 yılında Nils Nilsson o dönemde kullanılan öğrenme makineleri üzerine detaylı bir çalışma hazırladı. Nilsson çalışmasında mevcut makinelerin karşılaştığı dar boğazlardan bahsetmiştir. Ancak 1960’lı yıllarda bu çalışmada bahsedilen sorunların üzerine gidilmediğinden, günün ihtiyaçlarını karşılamada yetersiz kalan öğrenme kurallarının yerine yeni kurallar getirilmediğinden ve o zamanlarda bilinen çok katmanlı öğrenme modelleri uygulanmadığından yapay sinir ağları üzerine yapılan araştırmalar sekteye uğramıştır.

1960’lı yılların sonlarında daha az kaynakla çalışan hesaplama platformları ortaya çıkmaya başladığından yapay sinir ağları konusu 1986 yılına kadar uzun bir duraklama döneminde kalmıştır. Bu süreç içerisinde bu alandaki çalışmaların bir kısmı “yapay zekâ” konusu başta olmak üzere başka alanlara kaydırılmıştır. Ancak bu dönemde de konu üzerinde öncülük edecek bazı çalışmalar yapılmıştır. Örneğin 1980 yılında Fukushima ve Miyake “neocognitrons” adı verilen ve retina görüntülerinin örüntü tanıması için kullanılan yeni bir sinir ağı modeli önermişlerdir. Bu model çok katmanlı nöronlardan oluşmaktadır.

1982 ve 1986 yılları arasında yapay sinir ağı üzerine yöntemin potansiyelini gösteren pek çok makale hazırlanmıştır. 1984 yılında John Hopfield ilişkili bellek konusunda yinelemeli bir sinir ağı modeli öne sürmüştür. Bu çalışmasında Hopfield nöronları birbirleriyle tümüyle bağlı sinir ağlarının hesaplama özelliklerini formülize etmiştir.

Yapay sinir ağı konusu üzerine yapılan bir diğer çalışma da 1986 yılında James McClelland ve David Rumelhart tarafından gerçekleştirilmiştir. Bu çalışmalarında McClelland ve Rumelhart yeni öğrenme kuralları öne sürerek çok katmanlı sinir ağlarının hesaplama potansiyellerini göstermişlerdir. Yeni öğrenme kuralları, EEG ve ECG gibi tıbbi sinyallerin analizi ve otomasyon sistemlerinin geliştirilmesine katkıda bulunmuştur.

1986 yılından günümüze kadar yapay sinir ağı üzerine pek çok araştırma programı başlatılmıştır. Çok sayıda yayın yapılmakla beraber başlangıçta çok kısıtlı bir uygulama alanı olan sinir ağlarının artık daha çok konuya uyarlandığı görülmektedir. Uygulamaya yönelik olarak yapay sinir ağı içeren VLSI yongaları geliştirilmiştir. Bu hızlı gelişime ve donanımsal üretime rağmen nörobilgisayarların klasik bilgisayarların yerini alması beklenmemektedir. Çünkü klasik bilgisayarlar ile matematiksel hesaplamalar, üç boyutlu tasarım, yazılım geliştirme işlemleri hem çok ucuza hem de çok hızlı bir şekilde yapılabilir. Buna karşılık yapay sinir ağı sistemlerinin büyük boyutlu paralel işlemeye dayalı problemlerin çözümünde kullanılması daha olasıdır. Klasik bilgisayarların yeterli olamadığı sınıflandırma, ilişki kurma ve sonuç çıkarma gibi konuları içeren ses ve görüntü tanıma problemlerinde insan doğasına benzerliği nedeniyle yapay sinir ağlarına dayalı sistemler kullanılabilir.

3.1.3. Yapay Sinir Ağlarının Özellikleri

Yapay sinir ağı yöntemi, insan beynindeki nöronların çalışma şeklini örnek alarak tasarlandığından doğal olanın sahip olduğu bazı özelliklere sahiptir. Bu özellikler şu şekilde sıralanabilir :

- 1) Gerçekleme Kolaylığı : Yapay sinir ağı, her biri kendi üzerine düşen görevi yapmakla yükümlü çok sayıda nörondan oluştuğundan tasarımları başka sistemlere göre daha basittir. Her nöronun problemin çözümüne katkısı az olsa

da toplamda elde edilen paralel işleme gücüyle karmaşık problemler çözülebilir. Aynı zamanda bir sinir ağının fazla değiştirilmeden farklı problemlere de uygulanabilmesi yapay sinir ağlarının tasarım kolaylığını göstermektedir.

- 2) Paralel Çalışma : Klasik bilgisayarda yapılan işlemlerin çoğu seridir ve bir algoritma dahilinde basamaklar halinde işlenir. Yapay sinir ağları ise her biri kendi başına, tüm sistem açısından düşünüldüğünde paralel çalışan nöronlardan oluşur. Dolayısıyla bir nöronun yavaş çalışması yapılan işlemi sekteye uğratmaz.
- 3) Hatayı Tolere Etme : Seri işleme esasına dayanan sistemlerde işlem sırasında bir fonksiyon görevini yerine getiremezse ondan cevap bekleyen ardışık fonksiyonlar da çalışamaz ve işlem tamamlanamaz. Ancak paralel işleme esasına dayanan yapay sinir ağlarında, nöronların çok sayıda olması ve bir nöronun birden fazla nörona bağlanabilmesi nedeniyle üretilecek sonuç böyle bir durumdan etkilenmez.
- 4) Doğrusal Olmama : Bir f , fonksiyonunun doğrusal olması için hem $f(x+y)=f(x)+f(y)$ koşulunu (toplanabilirlik) hem de $f(a.x)=a.f(x)$ koşulunu (homojenlik) sağlaması gerekmektedir. Yapay sinir ağlarını oluşturan nöronlar bu özelliklere sahip değildir. Dolayısıyla tüm sinir ağı modeli doğrusal değildir.
- 5) Genelleme : Yapay sinir ağları eğitildikleri test kümelerine uygun çıkışları verdikten sonra daha önce karşılaşmadıkları yeni test verileri için de çıkışlar üretebilmektedirler. Örüntü tanıma bu duruma örnek olarak verilebilir. Genelleme yeteneği ile yapay sinir ağı tanınması için eğitildiği bir parmak izini başka parmak izleri ile karşılaştırabilir.
- 6) Çalışma Hızı : Yapay sinir ağları, VLSI donanımsal tasarımları gerçekleştirildiğinde paralel işlemenin gücünü de kullanarak çok hızlı bilgi işleme yeteneğine kavuşabilmektedir. Üretim bantlarında defolu ürün tespitinde kullanılabilen yapay sinir ağı sistemleri bu konuya bir örnektir.
- 7) Dinamik Öğrenme : Klasik hesaplama yöntemlerinde çözümü için bir algoritma hazırlanmayan bir problemin çözümü yapılamaz. Yapay sinir ağlarında ise sistemin, probleme istenen çözümü sağlayabilmesi için rasgele ağırlıklandırma kullanılabilir. Sinir ağı eğitim döngüsü boyunca aldığı girdiyi vericeği çıktıyla eşleştirecek katsayıları kendi hesaplayabilmektedir.

3.1.4. Yapay Sinir Ağlarının Uygulanışı

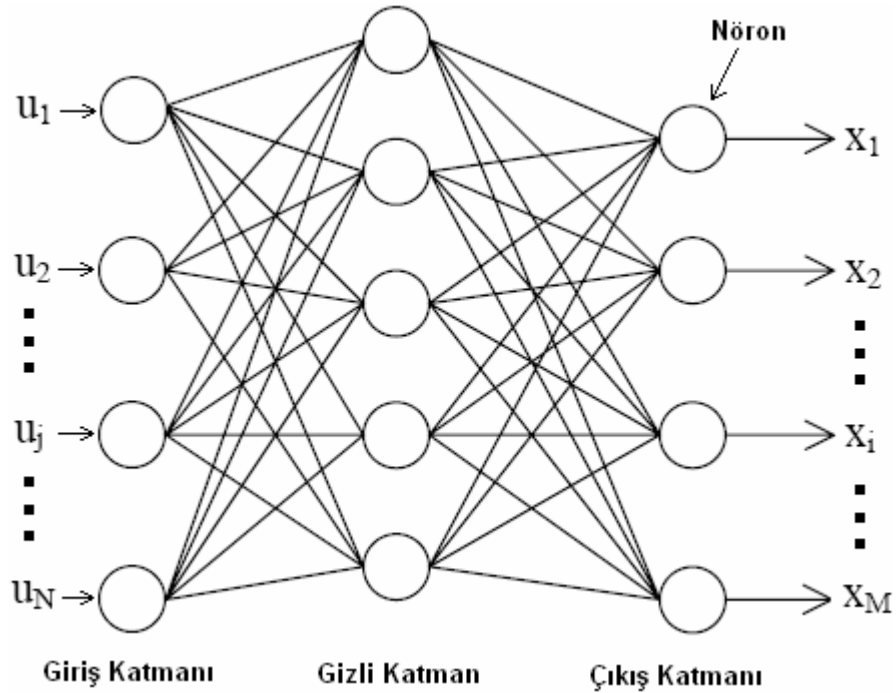
Yapay sinir ağlarına bir girdi kümesine karşı vermesi istenen çıktılar sunulur. Sinir ağının görevi deneme yanılma yöntemi ile “eğitim” adı verilen bir döngü kullanarak bünyesindeki nöron bağlantılarında bulunan ağırlık katsayılarını ayarlamaktır. Eğitim işleminde sinir ağının benzer girdi setleri için aynı çıktıyı üretmesi başka bir deyişle “genelleme” yapması beklenir. Bu nedenle buna uygun öğrenme algoritmalarının kullanılması gerekir.

Bir yapay sinir ağı eğitim döngüsü süresince sırasıyla;

- 1) Çıktılar üretir.
- 2) Ürettiği çıktıları hedef çıktılar ile kıyaslayarak hata hesaplar.
- 3) Ağırlık katsayılarını değiştirerek eğitim döngüsüne devam eder.

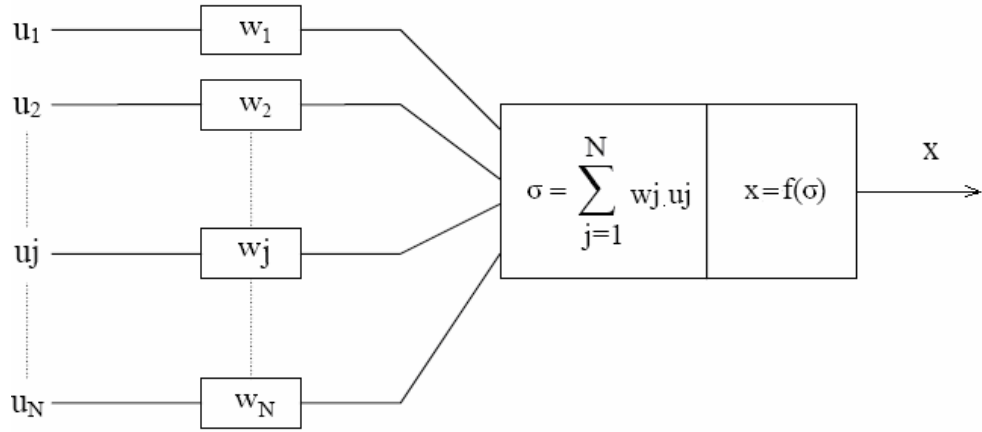
Eğitim, ağı ürettiği çıktı ile üretilmesi istenen çıktı arasındaki fark (hata) önceden belirlenen eşik değerinin (hata toleransı) altına düştüğünde sonlanır.

Şekil 3.4’de katmanlı bir yapay sinir ağı örneği verilmiştir.



Şekil 3.4 : 3 katmanlı yapay sinir ağı örneği

Şekil 3.4’de u_j ile gösterilen değerler sinir ağı sisteminin girdileri, x_i ile gösterilen değerler ise bu girdilere karşılık sinir ağının ürettiği çıktıları ifade etmektedir. Bu modelde sadece tek bir gizli katman gösterilmiştir ama probleme göre gizli katman birden fazla alt katmandan oluşabilir. Bu modelde katman çiftlerindeki nöronlar birbirlerine tamamiyle bağlı (fully-interconnected) durumdadırlar. Nöronlar arasındaki her bağlantıda giriş bilgisini ağırlıklandıran bir de ağırlık katsayısı bulunur. Bu katsayılar sistem eğitilmeden önce genelde 0-1 aralığından rasgele sayılar olarak seçilir. Şekil 3.5’de yapay bir nöron modeli ve denklem 3.2’de bu nöronun çalışma yapısı verilmektedir.



Şekil 3.5 : Yapay Nöron Modeli

$$\sigma = \sum_{j=1}^N w_j \cdot u_j, \quad x = f(\sigma) \quad (3.2)$$

Burada u_j değerleri nöronun girdi kümesini, w_j değerleri bu girdilere uygulanacak ağırlıklandırma katsayılarını, σ nöronun girişinde üretilen toplam net etkiyi, $f(\cdot)$ nöronun tetiklenip tetiklenmeyeceğini belirleyen aktivasyon fonksiyonunu ve $x = f(\sigma)$ de nöronun ürettiği çıktıyı ifade etmektedir. Bir nöronda kullanılabilecek aktivasyon fonksiyonlarına örnek olarak doğrusal, sigmoid ve tanjant hiperbolik gibi fonksiyonlar verilebilir.

3.1.4.1. Aktivasyon Fonksiyonları

Aktivasyon fonksiyonu yapay sinir ağı ile çözülmek istenen problemin yapısına göre değişiklik gösterir. Örneğin bir ağıın modelin ortalama davranışını öğrenmesi için sigmoid, ortalamadan sapma davranışını öğrenmek için de hiperbolik tanjant fonksiyonu kullanılabilir. Aktivasyon fonksiyonları genellikle $[0,1]$ veya $[-1,1]$ aralığından değerler üretecek şekilde seçilirler. Yapay sinir ağlarında kullanılabilen aktivasyon fonksiyonları şu şekildedir :

- Doğrusal Aktivasyon Fonksiyonu : $f(x) = a.x$ şeklinde tanımlanabilen ve doğrusal karakterli problemlerin çözümü için kullanılabilen fonksiyonlardır.
- Sigmoid Aktivasyon Fonksiyonu : İlk dönemlerden beri kullanılan aktivasyon fonksiyonlarıdır. Bu fonksiyonlar sınırlı ve türevi alınabilir fonksiyonlardır. Genelde unipolar (tek kutuplu – $[0,1]$ arasında sınırlı) ve bipolar (çift kutuplu - $[-1,1]$ arasında sınırlı) şekilleri kullanılır. Denklem 3.3 ve 3.4’de sırasıyla tek kutuplu ve çift kutuplu sigmoid fonksiyon örnekleri görülmektedir.

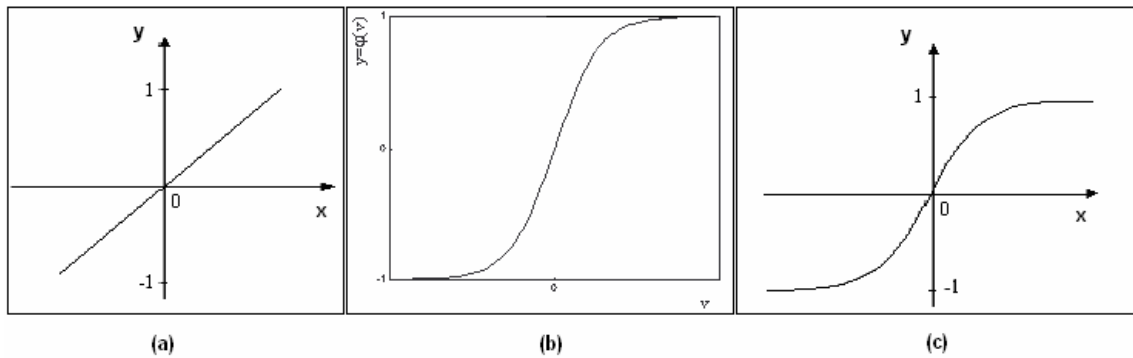
$$\text{Unipolar fonksiyon : } f(x) = \frac{1}{1 + e^{-\beta \cdot x}} \quad (3.3)$$

$$\text{Bipolar fonksiyon : } f(x) = \frac{2}{1 + e^{-\beta \cdot x}} - 1 \quad (3.4)$$

- Tanjant Hiperbolik Fonksiyon : Bipolar sigmoid fonksiyona benzeyen bir

fonksiyondur. $f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} = \frac{e^{2x} - 1}{e^{2x} + 1}$ şeklindedir.

Şekil 3.6’da sırasıyla doğrusal, unipolar ve tanjant hiperbolik fonksiyonların grafikleri verilmiştir.



Şekil 3.6 : Aktivasyon Fonksiyonları
(a) Doğrusal , (b) Sigmoid , (c) Tanjant Hiperbolik

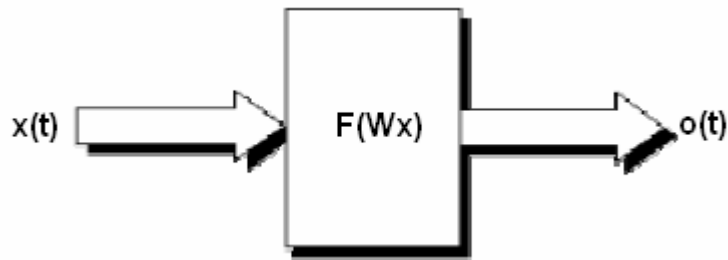
3.1.5. Yapay Sinir Ağlarının Sınıflandırılması

Yapay sinir ağları ağı oluşturan nöronlar arasındaki bağlantıların yapısı ve ağı ürettiği sonuçların işlenmesi gibi iki başlığa göre sınıflandırılabilir. Bunlardan ilki yapay sinir ağlarının yapısına göre sınıflandırılması, ikincisi de yapay sinir ağlarının kullanılan öğrenme tekniğine göre sınıflandırılmasıdır.

Yapay sinir ağları yapılarına göre “İleri Beslemeli Sinir Ağları” (Feed-Forward Neural Networks) ve “Geri Beslemeli Sinir Ağları” (Feed-Back Neural Networks) olarak ikiye ayrılır.

3.1.5.1. İleri Beslemeli Sinir Ağları

Bu tip sinir ağlarının ilk örneklerinde tek katman kullanılmıştır. Ancak genellikle çok katmanlı yapıda kullanılırlar. Nöronlar birbirlerine tek yönlü bağlantılar ile bağlıdır ve aynı katmandaki nöronlar kendi aralarında bağlantı kuramazlar. Bu tip ağlarda eğitim döngüsü sırasında üretilen ara sonucun sinir ağına yeniden girdi olarak verilmesi söz konusu değildir. Çünkü bu tip ağların dinamik hafızaları yoktur. Bu nedenle günümüzdeki uygulamaları da donanımsal tasarımdan çok bilgisayar yazılımı olarak gerçekleştirilmektedir. Şekil 3.7’de ileri beslemeli bir sinir ağı için blok diyagramı görülmektedir.



Şekil 3.7 : İleri beslemeli sinir ağı için blok diyagram

İleri beslemeli sinir ağlarının ilk örneği 1943 yılında tanımlanmış olan “Tek katmanlı Algılayıcı” (Single Layer Perceptron) tipi sinir ağlarıdır. Bu ağlar tek katmanlı çıkış nöronlarından oluşan ve girişlerin belirlenen ağırlık katsayılarının etkisiyle çıkışa yansıtıldığı ağlardır. Her nöron için giriş değerleri bu değerler için belirlenen ağırlık

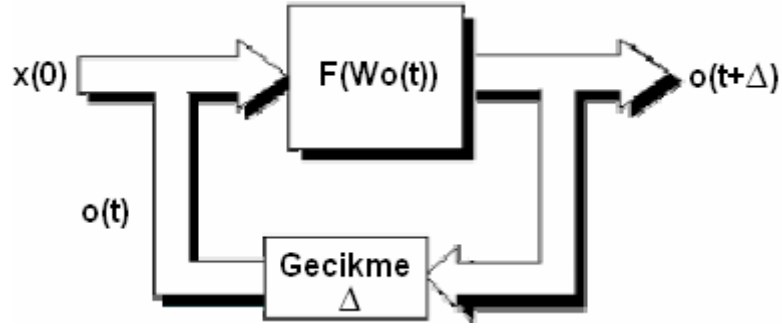
katsayıları ile çarpıldıktan ve oluşan çarpım değerleri toplandıktan sonra elde edilen sonuç bir eşik değeri (genellikle 0) ile kıyaslanır. Eğer sonuç eşik değerinden büyükse nöron tetiklenir ve yüksek değerli çıktı olan 1'i verir. Sonuç eşik değerinden büyük değilse nöron tetiklenmez ve çıktısı düşük değer olan -1 olur. Bu değerler uygulamadan uygulamaya fark gösterebilir. Bu şekilde bir aktivasyon fonksiyonuna sahip nöronlar Şekil 3.3'de gösterilen McCulloch-Pitts tipinde nöronlar olarak da bilinir.

Genel kullanım alanı bulan ileri beslemeli sinir ağı modeli ise “Çok Katmanlı Algılayıcı” (Multi Layer Perceptron) tipi sinir ağıdır. Bu yapıda nöronlar birden fazla katmana dağıtılmış ve aralarında yönlü bağlantılar belirlenmiştir. Aynı katmandaki nöronların arasında bağlantı bulunmamaktadır. Bu tipteki sinir ağları için genellikle Sigmoid tipte aktivasyon fonksiyonları kullanılmaktadır.

Çok katmanlı algılayıcı ağlar çeşitli öğrenme teknikleriyle kullanılabilir ancak yaygınlıkla kullanılan eğitim modeli “Hatayı Geriye Yansıtma” (Error Back Propagation)'dır. Bu modelde üretilen sonuçlar, ağın vermesi gereken sonuçlarla kıyaslanarak ağıdaki ağırlık katsayılarının doğru değerlerinden ne kadar uzakta olduğu çıkarılır. Sonrasında bu hata önceki katmanlara yansıtılarak daha iyi eğitim sonuçları alınması hedeflenir. Böylece sinir ağının doğru çıktılara yakınsaması sağlanır.

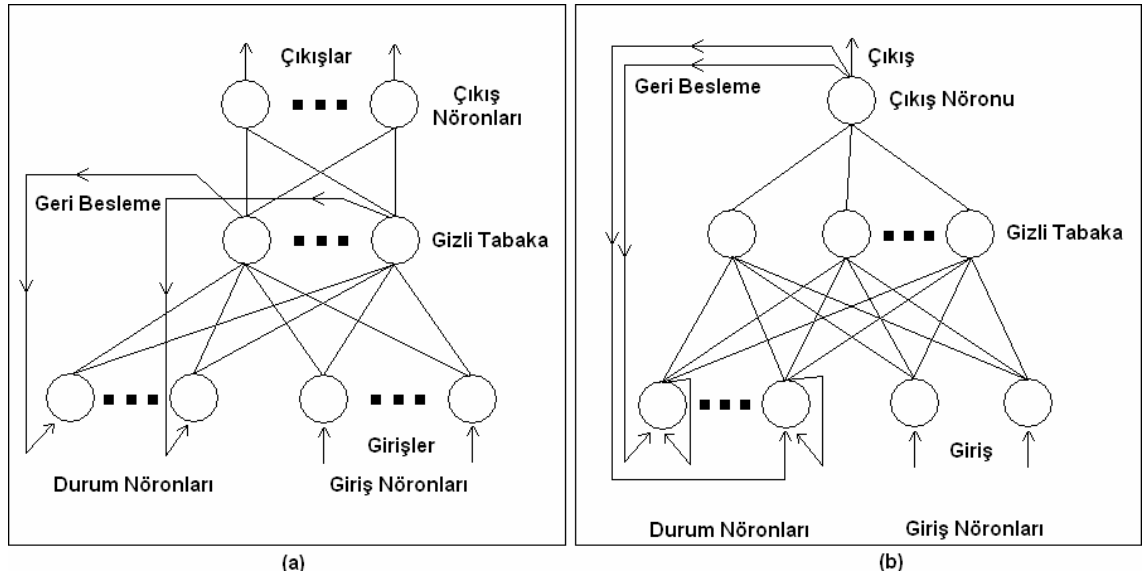
3.1.5.2. Geri Beslemeli Sinir Ağları

Geri beslemeli sinir ağları, nöron çıkışının sisteme yeniden girdi olarak verilebildiği ağlardır. Bu tip bir ağda veri akışı hem ileri hem de geri yönde gerçekleşir. Bu özelliğe sahip olabilmeleri için bu ağlar dinamik bellek yapısına ihtiyaç duymaktadır. Dinamik bellek ünitesinin bu ağdaki görevi bir nöronun t anında ürettiği çıktıyı Δ gibi bir süre boyunca saklayıp sisteme $t + \Delta$ anında yeniden girdi olarak vermektir. Bu tip ağlar sahip oldukları bellekli yapı nedeniyle donanımsal olarak gerçekleştirilebilmektedirler. Şekil 3.8'de geri beslemeli bir sinir ağının blok diyagramı görülmektedir.



Şekil 3.8 : Geri beslemeli sinir ağı için blok diyagram

Geri beslemeli sinir ağlarında geri besleme aynı katmandaki nöronlar arasında olabildiği gibi katmanlar arasındaki nöronlar arasında da olabilir. Bu şekilde bağlantılara sahip ağlara “Yinelenen Sinir Ağları” (Recurrent Neural Networks) adı verilir. Hopfield, Elman ve Jordan ağları bu tip ağlara örnek olarak verilebilir. Şekil 3.9 (a)’da Elman ağı ve Şekil 3.9 (b)’de Jordan ağı verilmiştir.



Şekil 3.9 : Yinelenen Sinir Ağları
(a) ELMAN AĞI , (b) JORDAN AĞI

Yapay sinir ağları kullandıkları öğrenme tekniğine göre “Danışmalı Öğrenme” (Supervised Learning) kullanan ve “Danışmasız Öğrenme” (Unsupervised Learning) kullanan sinir ağları olarak ikiye ayrılır.

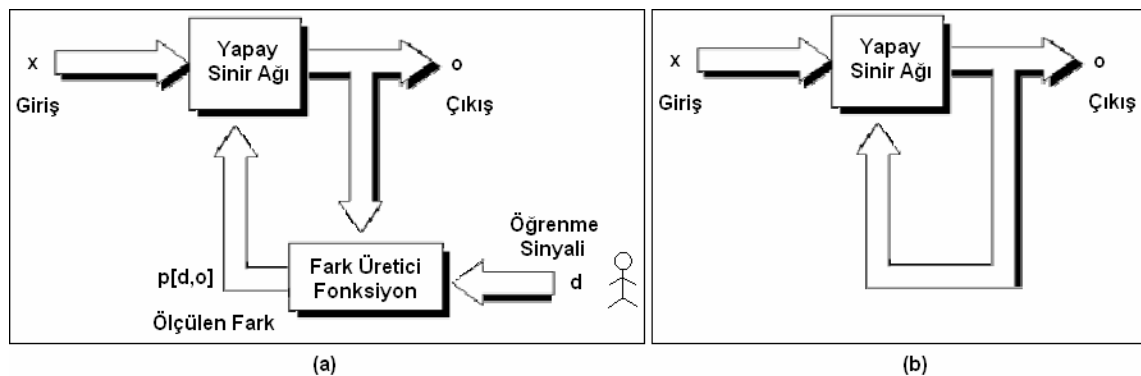
3.1.5.3. Danışmalı Öğrenme Kullanan Sinir Ağları

Danışmalı öğrenme yöntemi verilen eğitim kümesinden bir fonksiyon oluşturmayı amaçlar. Eğitim kümesi vektör bazında giriş verisini ve bu girişlere karşılık gelen istenen çıkış kümesini (Öğrenme Sinyali) içerir. Sisteme arzu edilen çıkışları bildiren kişiye “danışman” denir. Bu yöntemi kullanan bir sinir ağının görevi verilen giriş-çıkış çiftlerine göre ağırlık katsayılarını ayarlamaktır. Eğitim döngüsünün i. basamağında ağın ürettiği çıkış ile istenen çıkış arasındaki fark hata olarak isimlendirilir ve amaç bu hatayı “hatayı geriye yansıtma” (error-back propagation) gibi yöntemlerle azaltmaktır. Böylece sistemin verdiği çıktılar arzulanan çıktılara yakınsayabilir.

Bir danışmalı öğrenme yapısının kurulabilmesi için :

- 1) Eğitim kümesini oluşturacak örneklerin tipi belirlenmelidir. Örneğin bir yüz algılama problemi için giriş verisi yüz resmi iken, yüz tanıma problemi için giriş verisi yüz resminden elde edilen ayırt edici özellik vektörleri olabilir.
- 2) Giriş verisinin ve bu girişlere karşılık gelen arzu edilen çıktılarının belirlenmesi gerekir.
- 3) Giriş verisinin eğitimi gereksiz uzatmayacak kadar küçük boyutta ancak çıkışın doğru tahmin edilebilmesini sağlayacak kadara da büyük boyutta ifade edilmesi gerekmektedir.

Şekil 3.10 (a)’da danışmalı öğrenme yönteminin blok diyagramı verilmiştir.



Şekil 3.10 : Yapay Sinir Ağlarında öğrenme yöntemleri
(a) Danışmalı Öğrenme , (b) Danışmasız Öğrenme

Bir yapay sinir ağındaki i . nöronun k anında ürettiği çıkıştan, $k + 1$ anında kullanacağı ağırlık katsayılarının hesaplanması denklem 3.5’de görüldüğü gibidir.

$$W_i(k+1) = W_i(k) + c \cdot f(g_i(\sigma_i(k)), d_i(k)) U(k) \quad (3.5)$$

Burada $W_i(k) = [W_{i1}(k), W_{i2}(k), \dots, W_{im}(k)]$ vektörü, i . nöronun m adet ağırlık katsayısının k anındaki değerlerini, c , $[0-1]$ aralığından değer alan öğrenme katsayısını, $U(k) = [U_1(k), U_2(k), \dots, U_m(k)]$ vektörü, k anında sinir ağına uygulanan m adet girişin değerlerini, $d_i(k)$, i . nöronun k anında üretmesi istenen çıktıyı, $\sigma_i(k)$, i . nöronun k anında ürettiği net etkiyi, $g_i(\sigma_i(k))$, i . nöronun k anında ürettiği çıktıyı ve $f(.)$ fonksiyonu da fark üretici fonksiyonunu göstermektedir.

Şimdiye kadar öne sürülen öğrenme kuralları yukarıda adı geçen $f(.)$ fark üretici fonksiyonunun davranışına göre farklılık göstermişlerdir. Buna göre $r = f(g_i(\sigma_i), d_i)$ eşitliği göz önünde bulundurulmak üzere i . nöron için bazı danışmalı öğrenme kuralları şu şekilde verilebilir :

a. Perceptron Öğrenme Kuralı :

$$r = d_i - g_i(\sigma_i) \quad (3.6)$$

b. Widrow - Hoff Öğrenme Kuralı :

$$r = d_i - \sigma_i \quad (3.7)$$

c. Correlation Öğrenme Kuralı :

$$r = d_i \quad (3.8)$$

d. Outstar Öğrenme Kuralı :

$$c.r.U = \beta \cdot (D_i - W_i) , \quad D_i = \underbrace{[d_i, d_i, \dots, d_i]}_{m \dots a \text{ det}} , \quad \beta > 0 \quad (3.9)$$

e. Delta Öğrenme Kuralı :

$$r = (d_i - g_i(\sigma_i)) \cdot g_i'(\sigma_i) \quad (3.10)$$

$g_i'(\cdot)$ aktivasyon fonksiyonunun türevini simgelemektedir. Türev işlemi olduğundan sadece sürekli aktivasyon fonksiyonları ile kullanılabilir. Denklem 3.3'deki Unipolar fonksiyon ile kullanıldığında $g_i'(\sigma_i) = g_i(\sigma_i) \cdot (1 - g_i(\sigma_i))$ şeklinde, denklem 3.4'deki Bipolar fonksiyon ile kullanıldığında da $g_i'(\sigma_i) = \frac{(1 - g_i^2(\sigma_i))}{2}$ şeklinde ifade edilebilir.

3.1.5.4. Danışmasız Öğrenme Kullanan Sinir Ağları

Danışmasız öğrenme yönteminde arzu edilen çıktıların ne olduğu yapay sinir ağına daha önceden belirtilmez. Sadece giriş kümesi verilir. Bu öğrenme modeli bazen olası kullanım alanı olmasa da kümeleme işleminde kullanılabilir. Böyle bir kullanımda ağıın çıkışlarından elde edilen değerler “tetiklik parametresi” (vigilance parameter) adı verilen bir sabit ile kıyaslanarak eğitim kümesindeki girdilerin birbirlerine benzerliklerine göre kümelenmesi sağlanabilir. Gail Carpenter ve Stephen Grossberg tarafından geliştirilen ART (Adaptive Resonance Theory) ağları danışmasız öğrenme kullanan ağlara örnek olarak verilebilir. Şekil 3.10 (b)'de danışmasız öğrenme yönteminin blok diyagramı verilmiştir.

Denklem 3.5'den çıkartılan $r = f(g_i(\sigma_i), d_i)$ eşitliği göz önünde bulundurulmak üzere i. nöron için bazı danışmasız öğrenme kuralları şu şekilde verilebilir :

a. Hebbian Öğrenme Kuralı :

$$r = g_i(\sigma_i) \quad (3.11)$$

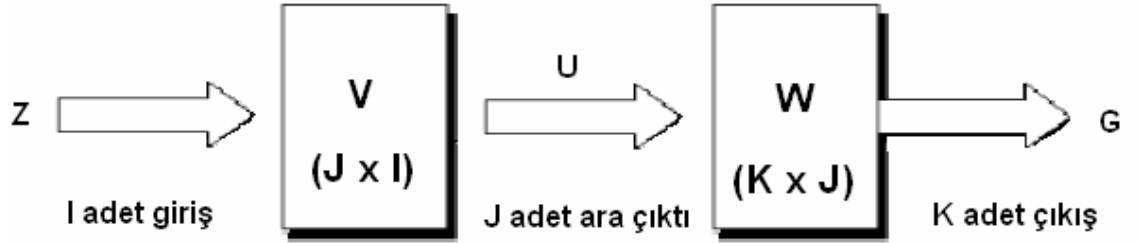
b. Winner – Take - All Öğrenme Kuralı : Bu öğrenme kuralında n adet nöron içinden ürettiği net etki (σ_i) en büyük olanın ağırlık katsayıları güncellenir. Diğer nöronların ağırlık sayıları güncellenmez.

$$(c.r.U)_{KAZANAN} = \alpha.(U - W_{KAZANAN}) , \alpha > 0 \quad (3.12)$$

3.1.6. Hatayı Geriye Yansıtma Algoritması

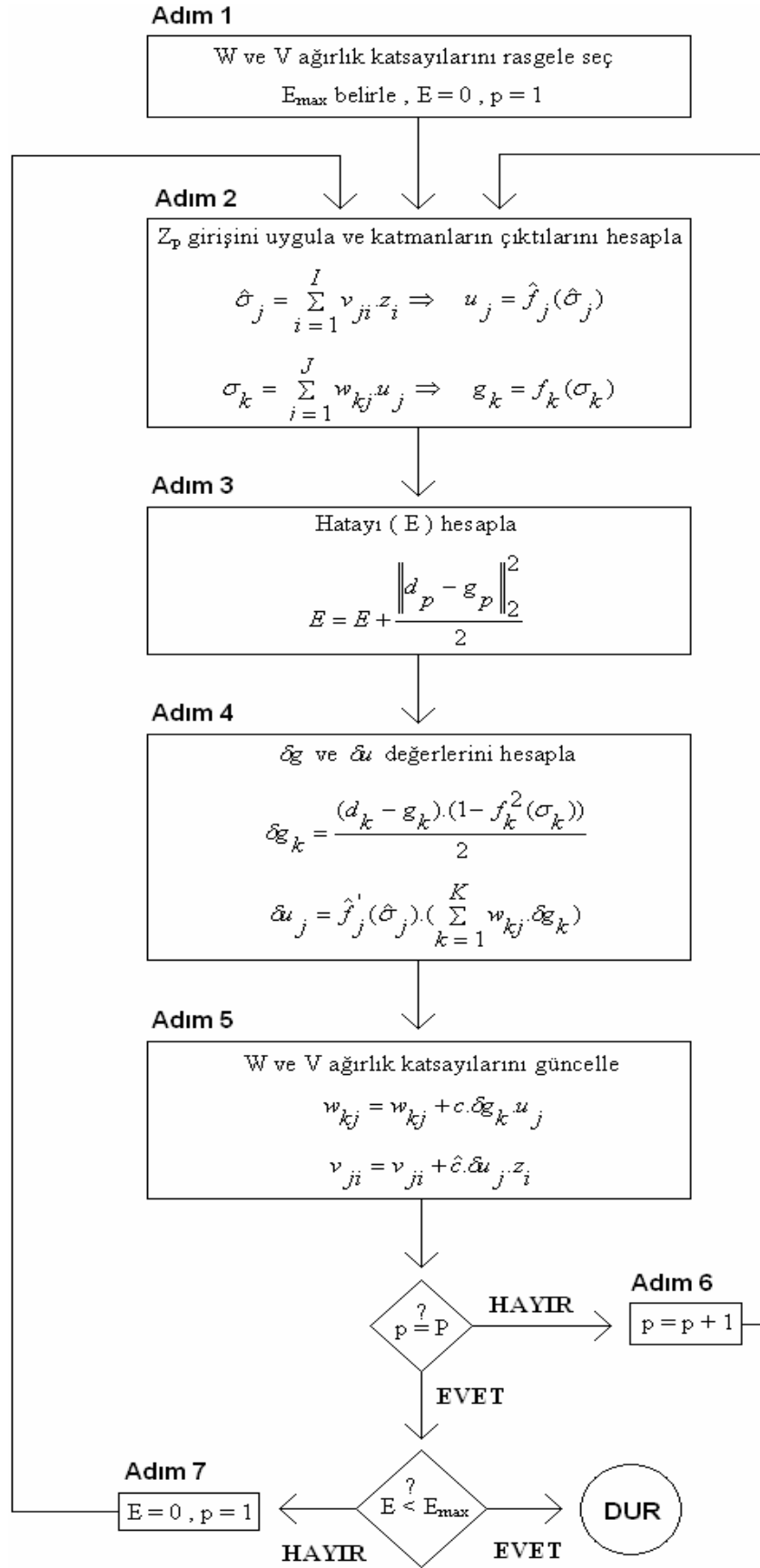
Hatayı geriye yansıtma algoritması çok katmanlı ileri beslemeli ağlarda yaygınlıkla kullanılan bir eğitim yöntemidir. Yöntem 1974 yılında Paul Werbos tarafından öne sürülmüştür [26]. Algoritma delta öğrenme kuralını kullanan bir danışmalı öğrenme yöntemidir. Bu nedenle nöronlarda türevi alınabilen aktivasyon fonksiyonlarının kullanılması gerekmektedir. Amaç çıkış katmanlarında hesaplanan eğitim hatasından faydalanarak her katmandaki ağırlık katsayılarını güncellemektir.

Algoritmanın işlediği örnek çok katmanlı ağı blok diyagramı Şekil 3.11’de verilmiştir.



Şekil 3.11 : Çok katmanlı sinir ağı için blok diyagram

Şekil 3.11 de gösterilen ağı $Z = \{z_1, z_2, \dots, z_p\}$ ile gösterilen P adet $(I \times 1)$ boyutlu giriş veri vektörü, $U = \{u_1, u_2, \dots, u_p\}$ ile gösterilen P adet $(J \times 1)$ boyutlu ara çıktı veri vektörü, $G = \{g_1, g_2, \dots, g_p\}$ ile gösterilen P adet $(K \times 1)$ boyutlu çıkış veri vektörü ve $D = \{d_1, d_2, \dots, d_p\}$ ile gösterilen P adet $(K \times 1)$ boyutlu istenen çıkışları tutan veri vektörü bulunmaktadır. p giriş vektörü indisini, E bir döngüdeki eğitim hatasını, $E_{\max}$ ulaşılması hedeflenen eğitim hatasını, $\hat{f}$ gizli katmanın aktivasyon fonksiyonunu, f çıkış katmanının aktivasyon fonksiyonunu, $\hat{\sigma}$ gizli katmanda üretilen net etkiyi, σ çıkış katmanında üretilen net etkiyi, $\hat{c}$ gizli katmandaki öğrenme katsayısını, c çıkış katmanındaki öğrenme katsayısını, $V_{(J \times I)}$ matrisi gizli katmanın ağırlık katsayılarını, $W_{(K \times J)}$ matrisi çıkış katmanının ağırlık katsayılarını, δg çıkış katmanındaki hatayı ve δu gizli katmandaki hatayı göstermek üzere hatayı geriye yansıtma algoritmasının akış diyagramı Şekil 3.12’de görüldüğü gibidir. 4. basamakta δu_j değerinin hesaplamasında türev işlemi olduğu gözden kaçırılmamalıdır.

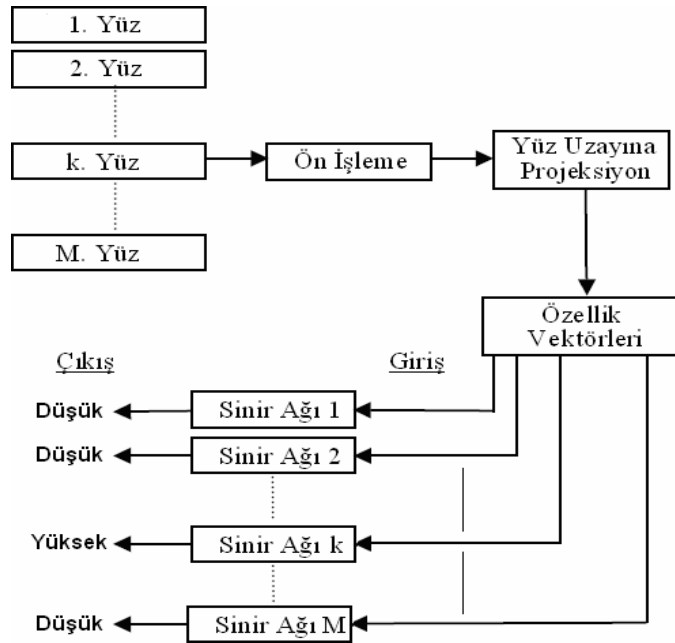


Şekil 3.12 : Hatayı geriye yansıtma algoritması için akış diyagramı

3.1.7. Yapay Sinir Ağlarının Yüz Tanıma Problemi İçin Kullanımı

Yapay Sinir Ağları, bu çalışmanın uygulamasında da görüleceği üzere yüz tanıma problemi için “ÖzYüzler” (Eigenfaces) yöntemiyle beraber kullanılabilir. Buna göre tanıma sisteminde her yüz resmi için özyüzler yöntemiyle önceden hesaplanarak kaydedilen özellik vektörleri yapay sinir ağına giriş olarak verilebilir. Bu şekilde sinir ağı her yüzün özellikleri için eğitilebilir. Bu yöntemin literatürdeki adı PCA-NN (Principal Component Analysis-Neural Networks)’dir.

Bu gibi bir sistemin eğitiminde her yüz resmi için bir sinir ağı oluşturulabileceği gibi tüm yüzleri kapsayan tek bir sinir ağı da oluşturulabilir. Örnek olarak her kişi için bir sinir ağı kullandığımızı ve k. kişiyi diğerlerinden ayırt etmek üzere bir ağ hazırladığımızı düşünelim. Bu durumda yapay sinir ağının k. kişinin resminden çıkarılan özellik vektörleri için yüksek değerli çıktı olan 1’i, ve bunun haricindeki yüzlerden edinilen özellik vektörleri için de düşük değerli çıktı olan 0’ı verecek şekilde eğitilmesi gerekir. Bu işlemi mevcut tüm kayıtlar için tekrarladığımızda her ağda o ağa ait olan yüzün karakteristiği kabul edilip yüksek değerli çıktılar oluşturulurken, farklı karakteristiğe sahip yüzler reddedilerek daha düşük çıkış değerleri üretilir. Böyle bir yapı Şekil 3.13’de verilmiştir.



Şekil 3.13 : Aranan k. yüz için özellik vektörü temelli sinir ağlarının çalışması

3.2. ÖZYÜZLER YÖNTEMİ

İnsan yüzü görüntüleri tüm görüntü uzayının alt uzayında bulunur. Bu alt uzayı ifade etmek için kullanılan yöntemlerden biri de “Temel Bileşen Analizi” (Principal Component Analysis – PCA)’dir.

Temel Bileşen Analizi yöntemi, 80’lerin sonunda Sirovich ve Kirby [18] tarafından insan yüzlerini ifade etmek için öne sürülmüştür. Yöntem verilen örnek kümesi içinde önce yüzlerin temel bileşenlerini yüz dağılımından elde edilen kovaryans matrisin özvektörleriyle ifade edilecek şekilde çıkarır. Yüz kümesindeki her yüz resmi hesaplanan en büyük özvektörlerin doğrusal birleşimiyle ifade edilir. Bu birleşim vektörüne “Özyüz” adı verilir. Sonrasında Turk ve Pentland [17] bu yöntemi yüz tanıma problemi için kullanmışlardır.

3.2.1. Özyüzler Yöntemi İçin Çalışma Basamakları

Özyüzler ile yüz tanıma yönteminin sistem eğitimi basamakları şu şekildedir :

- 1) Sisteme verilen resimler işleme alınmadan önce normalize işlemine tabi tutulur. Bu basamakta girilen resimdeki yüzün algılanması, arka plan etkisinin azaltılması (maskeleme vb.), hatalı ışıklandırma etkisinin azaltılması (örneğin renkli resmin gri tonlu yapıya dönüştürülmesi), çıkarılan yüz resminin sistemin kabul edeceği şekilde yeniden boyutlandırılması gibi ön işlemler gerçekleştirilir. Şekil 3.14’de yüz tanıma sistemine verilebilecek bir grup örnek yüz resmi görülmektedir.



Şekil 3.14 : Bir grup örnek yüz resmi

- 2) Sisteme N adet ve her biri m satır ve m sütundan oluşan yüz resmi verildiğini kabul edelim. Bu yüz resimlerini de $\Gamma_1, \Gamma_2, \dots, \Gamma_N$ şeklindeki ($m^2 \times 1$) boyutlu sütun vektörleri ile gösterelim. Bu resimlerin piksel değerlerinden denklem 3.13’de hesaplandığı gibi bir ortalama yüz resminin (Ψ) çıkarılması gerekir. Hesaplanan ortalama yüz resmi Şekil 3.15’de görülmektedir.

$$\Psi = \frac{1}{N} \cdot \sum_{i=1}^N \Gamma_i \quad (3.13)$$



Şekil 3.15 : Sisteme verilen N adet yüz resminden çıkarılan ortalama yüz resmi

- 3) Sisteme verilen her yüz resminin 2. adımda hesaplanan ortalama yüz resminden farkını gösteren Φ_i sütun vektörü denklem 3.14’deki gibi hesaplanır.

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

- 4) $D = [\Phi_1, \Phi_2, \dots, \Phi_N]$ şeklindeki ($m^2 \times N$) boyutlu matrizen denklem 3.15’de görüldüğü gibi bir kovaryans matris (C) elde edilir. Bu matrizen de λ_k özdeğerleri ve v_k özvektörleri çıkartılır. Ancak burada dikkat edilmesi gereken bir nokta C kovaryans matrisinin ($m^2 \times m^2$) boyutunda olmasıdır. Böyle bir matrisin özdeğer ve özvektörlerini hesaplamak kolay olmayacağından denklem 3.15’de gösterilen işlem $C = D^T \cdot D$ şeklinde yeniden düzenlenerek C matrisinin boyutu ($N \times N$)’e indirilebilir. Bu matrisin λ_n özdeğerleri ve v_n özvektörleri 5. basamakta özyüzlerin oluşturulmasında kullanılacaktır.

$$C = D \cdot D^T \quad (3.15)$$

- 5) $V = [v_1, v_2, \dots, v_N]$, C kovaryans matrisinin özvektörlerinden oluşan yeni bir matrisi ve $D = [\Phi_1, \Phi_2, \dots, \Phi_N]$ her yüz resminin ortalama yüzden farkını gösteren vektörlerden oluşan matrisi simgelemek üzere $U = [u_1, u_2, \dots, u_N]^T$ ile gösterilen ve N adet özyüzü barındıran özyüz matrisi denklem 3.16’daki gibi

hesaplanabilir. Bu işlemden sonra hesaplanan özyüzler kendilerine karşılık gelen özdeğerlerin büyüklüklerine göre sıralanır. Bir özyüze karşılık gelen özdeğer ne kadar büyükse o özyüzün ayırt edici özellikleri de o kadar fazladır. Uygulamaya bağlı olarak bu sıralama sonucu elde edilen ilk M adet özyüz sistem eğitiminde kullanılabileceği gibi hesaplanan tüm özyüzler de sistem eğitiminde kullanılabilir. Şekil 3.16'da sisteme verilen N adet yüz resminin ifadesinde kullanılacak olan 7 adet özyüz görülmektedir.

$$U = V.D^T \quad (3.16)$$



Şekil 3.16 : Eğitim kümesinden elde edilen 7 özyüz

- 6) $W = [w_1, w_2, \dots, w_N]$ ile gösterilen $(M \times N)$ boyutlu matris N adet yüz için hesaplanan $(M \times 1)$ boyutlu sütun vektörlerinden oluşur. W matrisinin çıkarımı denklem 3.17'deki gibidir.

$$W = U.D \quad (3.17)$$

Özyüzler ile yüz tanıma yönteminin tanıma işlemi basamakları ise şu şekildedir :

- 1) Sisteme tanınması için verilen yüz resmi tıpkı sistem eğitiminin 1. basamağında olduğu gibi tanımayı kolaylaştırmak amacıyla ön işlemlerden geçirilir.
- 2) Tanınması istenen yüz resminin Γ_T ile gösterilen ($m^2 \times 1$) boyutlu bir sütun vektörü ile ifade edildiğini düşünelim. Bu aşamada Γ_T vektörünün sistemde kayıtlı yüzlerden elde edilen ortalama yüz bilgisinden farkının denklem 3.18'deki gibi hesaplanması gerekir.

$$\Phi_T = (\Gamma_T - \Psi) \quad (3.18)$$

- 3) 2. adımda çıkarılan Φ_T fark vektörünün $U = [u_1, u_2, \dots, u_N]^T$ özyüz matrisi ile çarpılmasıyla tanınması istenen yüzün w_T özellik vektörü denklem 3.19'da görüldüğü gibi hesaplanır.

$$w_T = U \cdot \Phi_T \quad (3.19)$$

- 4) Hesaplanan w_T özellik vektörünün $W = [w_1, w_2, \dots, w_N]$ matrisini oluşturan w_i özellik vektörlerine yakınlığı veya uzaklığı tanınması istenen yüzün, o an özellik vektörü ile kıyaslaması yapılan kayıtlı yüze benzerliğini gösterir. Sistemin tanınan yüz olarak üreteceği yanıt, denklem 3.20'de hesaplanan Fark(i) değerlerinden en küçüğünü üreten i. resimdir.

$$\text{Fark}(i) = \sum_{j=1}^M |w_T(j) - w_i(j)|, \quad i = 1, 2, \dots, N \quad (3.20)$$

3.2.2. Özyüzler Yönteminin Diğer Yöntemlerle Karşılaştırılması

Özyüzlerle yüz tanıma yöntemi özellik temelli yüz tanıma ve yapay sinir ağlarıyla yüz tanıma gibi yöntemlere göre eğitim hızı açısından daha üstün bir yöntemdir. Bunda özellik temelli yüz tanıma yöntemlerinin yüz hatlarının sınırlarını bulmak için arama algoritmaları kullanmalarının ve yapay sinir ağları ile yüz tanıma yönteminin de yakınsama işlemi için bir eğitim döngüsü kullanmasının rolü büyüktür.

Özellik temelli yüz tanıma yöntemleri tıpkı yapay sinir ağlarında olduğu gibi operatör desteğine ihtiyaç duyan yöntemlerdir. Bu yöntemlerde arama yapılacak yüz bölgesi veya sinir ağının üretmesi istenen çıktı bir operatör tarafından belirlenir. Ancak özyüzler

yöntemi bu konuda operatöre ihtiyaç duymaz. Özyüzler yönteminde operatöre sisteme yeni yüzlerin eklenerek yeniden eğitilmesi gibi durumlarda ihtiyaç duyulabilir.

Özyüzler yöntemi sisteme verilen yüz resminin her pikselini işleyerek o yüze ait bir özellik vektörü çıkardığından yüz resmindeki istenmeyen bölgelerin en aza indirgenmesi tanıma büyük farklara yol açmaktadır. Bu nedenle özyüz yöntemine dayalı tanıma sistemlerinde değişen arka plana sahip resimler için bir yüz algılama algoritmasının ön işlem olarak kullanılması gereklidir.

Her yöntemde olduğu gibi resimdeki yüzün açısı da özyüzlerle yüz tanıma sistemi için tanıma performansını etkilemektedir. Bu nedenler yine bir ön işlem basamağı olarak dik olmayan yüz resimleri için döndürme işlemlerinin önceden uygulanması gerekmektedir.

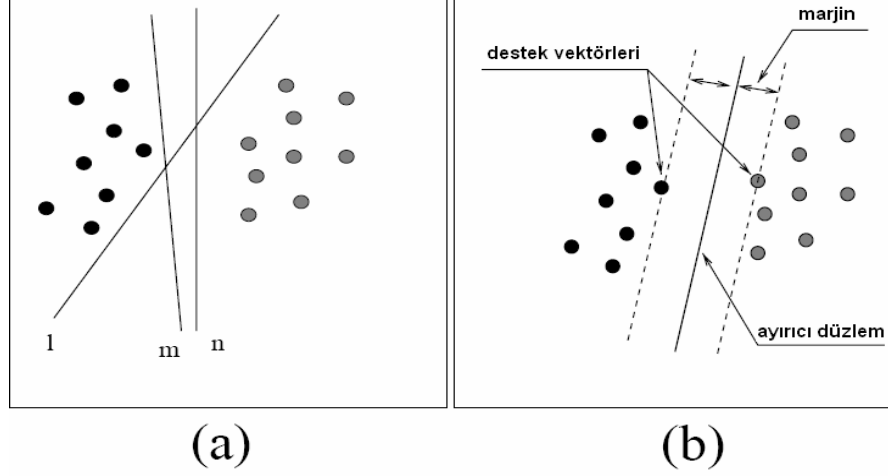
Özellik temelli yüz tanıma yöntemlerinin temel amacı yüzdeki temel organların yerlerini tespit etmektir. Ancak gözlük, sakal vb. gibi tanımaya etki eden detayların olduğu resimlerde bu yöntemlerin başarısı düşmektedir. Özyüzler yöntemi tanıma yüz resmindeki tüm piksellerden faydalandığı için bu konuda özellik temelli yöntemlere nazaran daha iyi sonuçlar verebilmektedir.

3.3. DESTEK VEKTÖR MAKİNELERİ YÖNTEMİ

Destek vektör makineleri (Support Vector Machines – SVM) iki veri kümesi grubunu aralarında oluşturulabilecek en büyük mesafe ile ayırmayı hedefleyen bir sınıflandırma yöntemidir. Yöntem Vapnik ve çalışma arkadaşları [29] tarafından genel amaçlı örüntü tanıma yöntemlerinde kullanılmak üzere ortaya çıkarılmıştır. Bu sınıflandırma yöntemiyle gerçekleştirilen örüntü tanıma işleminde amaç eğitim kümesindeki en yakın örnekleri birbirinden mümkün olan en büyük mesafe ile ayıran ve destek vektörü olarak anılan vektörlerle ayırmaktır.

Yöntemde eğitim kümesindeki her nokta $x_i \in \mathbb{R}^n$, $i = 1, 2, \dots, N$ formundadır ve her x_i noktası $y_i \in \{-1, 1\}$ şeklinde tanımlanan bir sınıfa aittir. Yöntemin amacı iki veri kümesi arasında ayırma işlemini, hesaplanacak destek vektörlerini birbirlerine en uzak mesafede konumlandırarak şekilde yapmaktır. Bu şekilde bulunan en iyi ayırıcı

düzleme “en uygun ayırıcı düzlem” denir. Şekil 3.17’de iki boyutlu bir uzayda test verisinin keyfi düzlemler ve en uygun ayırıcı düzlem ile doğrusal olarak ayrılışı görülmektedir. Sınır (marjin) değeri ayırıcı düzlem ile eğitim verileri arasındaki en kısa mesafe olarak tanımlanmaktadır.



Şekil 3.17 : Veri gruplarının düzlemlerle ayrılması
(a) Keyfi seçilen 3 farklı düzlemle ayırma , (b) En uygun ayırıcı düzlem ile ayırma

Destek vektör makineleri ile doğrusal sınıflandırma işleminde iki sınıf verisi denklem 3.21 ve 3.22’de görülen doğrusal sınırlar ile ayrılabilir. Burada w uygun bir ağırlık vektörü ve b ise bir eşik değeridir.

$$w \cdot x_i + b \geq 1, \quad y_i = 1 \quad (3.21)$$

$$w \cdot x_i + b \leq -1, \quad y_i = -1 \quad (3.22)$$

Denklem 3.21 ve 3.22’deki eşitsizlikler $y_i \cdot [(w \cdot x_i) + b] \geq 1, \quad i = 1, \dots, l$ şartıyla genelleştirilebilir. Sınır mesafesi (2 x marjin) ise $d = \frac{2}{\|w\|}$ olarak ön tanımlıdır. d

mesafesi ne kadar büyük olursa örnekler birbirlerinden o kadar iyi ayrılabilir. Bu nedenle d yi en büyük kılmak için w ’nun normunu en küçük kılmak gerekmektedir. Bu problem denklem 3.23’de gösterilen Lagrange çarpanları yöntemi ile çözülmektedir.

$$L(w, b, \alpha) = \frac{\|w\|^2}{2} - \sum_{i=1}^l \alpha_i \cdot \{y_i \cdot [(w \cdot x_i) + b] - 1\} \quad (3.23)$$

Burada α_i Lagrange çarpanlarını göstermektedir. Denklem 3.23'ün, w ve b değerlerine göre en küçüklenmesi ve $\alpha_i \geq 0$ değerlerine göre de en büyüklenerek çözülmesi sonucu en uygun ayırıcı düzlem parametresi olan w , $\sum_{i=1}^l \alpha_i \cdot y_i = 0$, $\alpha_i \geq 0$, $i = 1, \dots, l$ kısıtına bağlı olarak denklem 3.24 ile gösterilen şekilde elde edilebilir.

$$w = \sum_{i=1}^l \alpha_i \cdot y_i \cdot x_i , \quad \alpha_i \geq 0 , \quad i = 1, \dots, l \quad (3.24)$$

Herhangi bir x noktasının en uygun ayırıcı düzleme uzaklığı denklem 3.25'deki gibidir.

$$d(w, b; x) = \frac{|w \cdot x + b|}{\|w\|} \quad (3.25)$$

Denklem 3.24 ile elde edilen w değeri denklem 3.25'de yerine koyulduğunda x noktasının ayırıcı düzleme olan mesafesinin genelleştirilmiş hali şu şekilde gösterilir :

$$d(x) = \frac{(\sum_{i=1}^l \alpha_i \cdot y_i \cdot x_i) \cdot x + b}{\|\sum_{i=1}^l \alpha_i \cdot y_i \cdot x_i\|} \quad (3.26)$$

Denklem 3.26 ile hesaplanan d mesafesinin işareti x için sınıflandırma sonucunu, $|d|$ ise x 'in en uygun ayırıcı düzleme olan uzaklığını gösterir. $|d|$ mesafesi ne kadar büyük olursa x noktasının da o kadar iyi bir sınıflandırmaya tabi tutulması söz konusudur.

Ancak eğitim verilerini birbirinden doğrusal olarak ayırmak her zaman mümkün değildir. Bu gibi veri kümelerine doğrusal olmayan ayırma işlemi uygulanmak zorundadır. Bu, temel olarak basit bir işlemdir. Giriş uzayındaki her x noktası daha yüksek boyutlu bir uzaya (özellik uzayı) genişletilir ve test verileri yeni uzayda doğrusal olarak ayrılmaya çalışılır. Genişletme işlemi $z = \Phi(x)$ şeklinde gerçekleştirilir. x noktasının daha yüksek boyutlu bir uzay için sınıflandırma sonucu denklem 3.27'deki gibidir.

$$f(x) = w \cdot \phi(x) + b \quad (3.27)$$

Denklem 3.24'ün yerine konulmasıyla x noktasının sınıflandırmasının bir iç çarpım problemine dönüştüğü görülebilir.

$$f(x) = \sum_{i=1}^l \alpha_i \cdot y_i \cdot (\phi(x_i) \cdot \phi(x)) + b \quad (3.28)$$

Yüksek boyutlu bir uzayda $(\phi(x_i) \cdot \phi(x))$ işleminin gerçekleştirilmesi zor olduğundan bunun yerine $K(x_i, x) = (\phi(x_i) \cdot \phi(x))$ şeklinde tanımlanan bir kernel fonksiyonu kullanılabilir. Bu gibi işlemlerde en çok kullanılan kernel fonksiyonları şunlardır :

1) Polinom tipi fonksiyonlar :

$$K(x_i, x) = (x_i \cdot x + 1)^p \quad (3.29)$$

2) RBF tipi fonksiyonlar : σ , destek vektörünün etkilediği alana bağlı bir katsayı olmak üzere;

$$K(x_i, x) = \exp\left(-\frac{\|x_i - x\|^2}{2\sigma^2}\right) \quad (3.30)$$

3.3.1. Destek Vektör Makinelerinin Çok Sınıflı Yüz Tanıma Problemine Uygulanışı

Bilindiği üzere yüz tanıma probleminde sadece iki veri kümesi söz konusu değildir. Sürekli yeni yüz resimleri ile güncellenen bir sistemdeki eğitim verisi yüz sayısına bağlı olarak artmaktadır. Böyle çok sınıflı (q sınıflı) problemler için destek vektör makineleri ile sınıflandırma işleminde iki farklı yaklaşım öne sürülmüştür :

- a. “Hepsine karşı biri” yaklaşımında q sınıf için q destek vektör makinesi oluşturulur. Her DVM bir sınıfı kalan sınıflardan ayırır.
- b. “İkili ağaç” yaklaşımında karşılaştırılacak sınıflar çiftler halinde ikili ağaca yerleştirilir. İkili ağacın her düğümünde sınıflandırma işleminden geçen veri ağacın kök düğümüne kadar kalan diğer verilerle karşılaştırılır.

Bu iki yöntem kullanılan DVM sayısına göre kıyaslandığında $q \cdot (q-1)/2$ makine yerine sadece q makine kullanan hepsine karşı biri yaklaşımı daha çok tercih edilebilir. İlk yöntemde q karşılaştırma yapılırken ikinci yöntemde $q-1$ adet karşılaştırma yapılmaktadır. İki yöntemin de çalışma zamanı karmaşıklığı aynıdır [30].

3.4. UYGULAMA PROGRAMI

Tez çalışması kapsamında yüz tanıma konusuna Özyüzler ve PCA-NN yöntemlerini kullanarak yaklaşan bir uygulama programı hazırlanmıştır. Program Visual Studio 6.0 ortamında Visul Basic ile yazılmıştır. Programın testleri Pentium D 2.8 GHZ işlemcili ve 1 GB Ram'li bir bilgisayarda yapılmıştır.

Uygulama programı temel olarak 3 bölümden oluşmaktadır. Bunlar :

- 1) Sistemde kayıt ekleme, silme, güncelleme, listeleme gibi işlemlerin yapıldığı kısım,
- 2) Özyüzler yöntemiyle tanıma işleminin gerçekleştirildiği kısım,
- 3) Özyüzler yönteminden elde edilen özellik vektörlerinin yapay sinir ağı yönteminde kullanılmasıyla (PCA-NN) tanıma işleminin yapıldığı kısımdır.

Bu işlemler için program arayüzleri Şekil 3.18 (a), 3.18 (b) ve 3.18 (c)'de verilmiştir.

(a)

Şekil 3.18 (a) : Kayıt düzenleme işlemleri için program arayüzü

Özyüzler ile Yüz Tanıma İşlemi

Sistemi Eğit Resim Seç Kişiyi Bul

Minimum Doğru Eşleşme Oranı : % 60

Bulunan Kişinin Bilgileri

ID : 84638554

Ad : üç

Soyad : üç

Cinsiyet : ☒ Bay ☐ Bayan

Doğum Tarihi : 01.01.1900

Meslek :

Telefon No : 0

E-Mail Adresi :

İkametgah Adresi :

Seçilen Resim

Bulunan Resim

Doğru Eşleşme Oranı : + % 82 -

(b)

Şekil 3.18 (b) : Özyüzler yöntemi ile yüz tanıma işlemi için program arayüzü

PCA-NN ile Yüz Tanıma İşlemi

Sistemi Eğit Resim Seç Kişiyi Bul

Bulunan Kişinin Bilgileri

ID : 84638554

Ad : üç

Soyad : üç

Cinsiyet : ☒ Bay ☐ Bayan

Doğum Tarihi : 01.01.1900

Meslek :

Telefon No : 0

E-Mail Adresi :

İkametgah Adresi :

Seçilen Resim

Bulunan Resim

Sinir Ağının Ürettiği Değer : + 0,1749 -

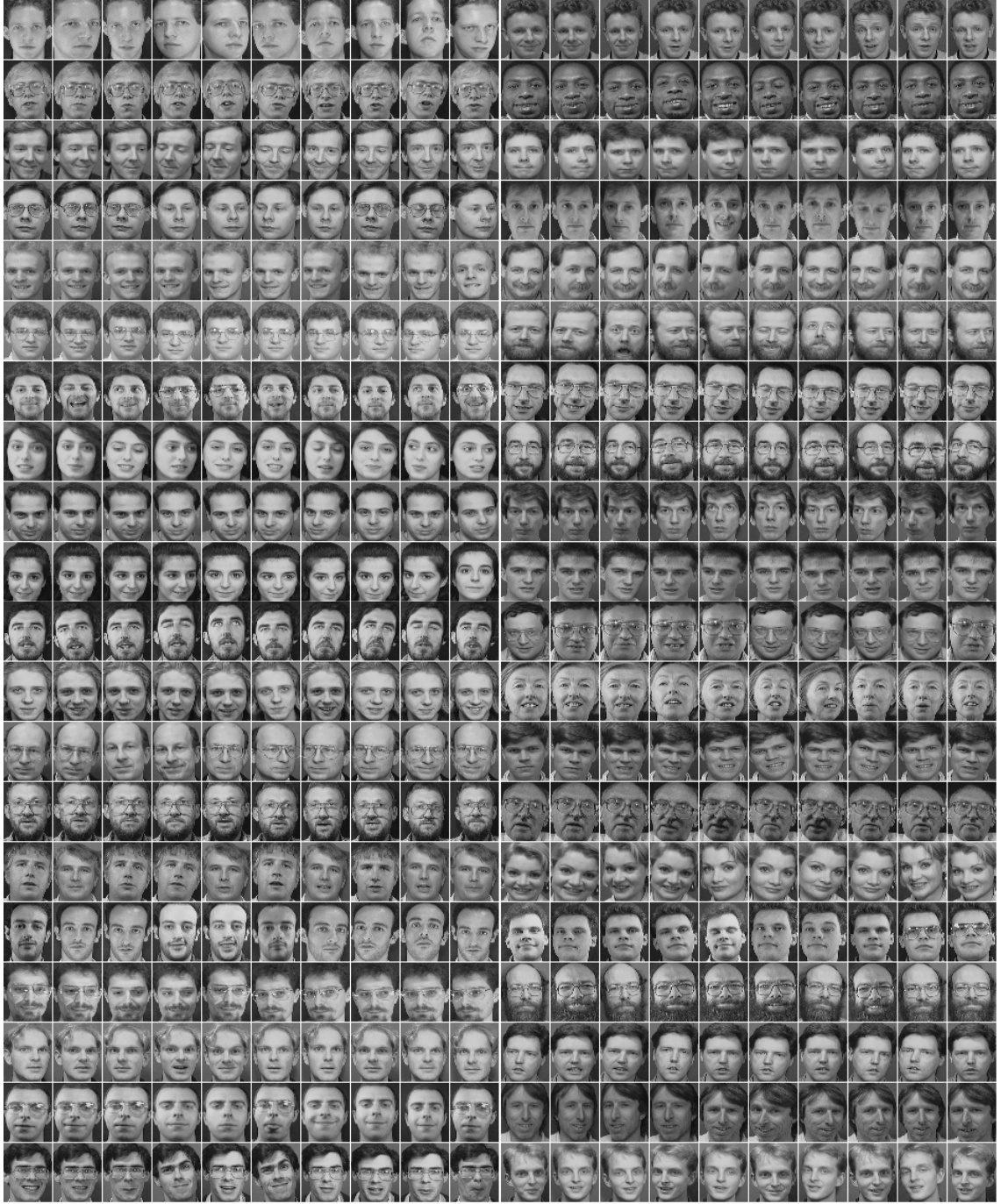
(c)

Şekil 3.18 (c) : PCA-NN yöntemi ile yüz tanıma işlemi için program arayüzü

Uygulama programı sisteme kaydedilecek resimlerdeki yüzleri algılamak için 3. parti bir yazılım kullanmaktadır. Bu yazılım fotoğraftaki yüzleri ten renklerine göre algılayıp bulundukları koordinatları vermektedir. Kayıt işleminde algılanan yüz resmi 128 x 128 piksel boyutunda ölçeklenerek kullanıcı bilgisiyle beraber saklanmaktadır. Ancak bu çalışmada programının testinde eğitim kümesi olarak AT&T Laboratories'den [27] ORL yüz veritabanı kullanılmıştır. Bu veritabanından girilen 92 x 112 ebadındaki resimler bu ebada en yakın ayarlanan 93 x 117 boyutuna genişletilerek kaydedilmiştir.

Eğitim kümesi tümü gri tonlu resimlerden oluştuğundan yüz algılama modülü kullanılamamıştır. Ancak veritabanındaki resimlerin sabit bir arka plan görüntüsüne sahip olması ve sadece yüz resmini içeriyor olması bu dezavantajı bir ölçüde ortadan kaldırmıştır.

ORL veritabanı 40 kişinin 10 farklı pozdan çekilen ve tamamı gri tabanlı olan yüz resimlerinden oluşmaktadır. Bu resimler veri tabanı hakkında fikir yürütülebilmesi için Şekil 3.19'da toplu halde verilmiştir. Programa kişi kaydı sırasında kişinin 1. pozunu ele alınmıştır.



Şekil 3.19 : Kullanılan eğitim seti (ORL yüz veritabanı)

3.4.1. Özyüzler İle Yüz Tanıma Bölümü

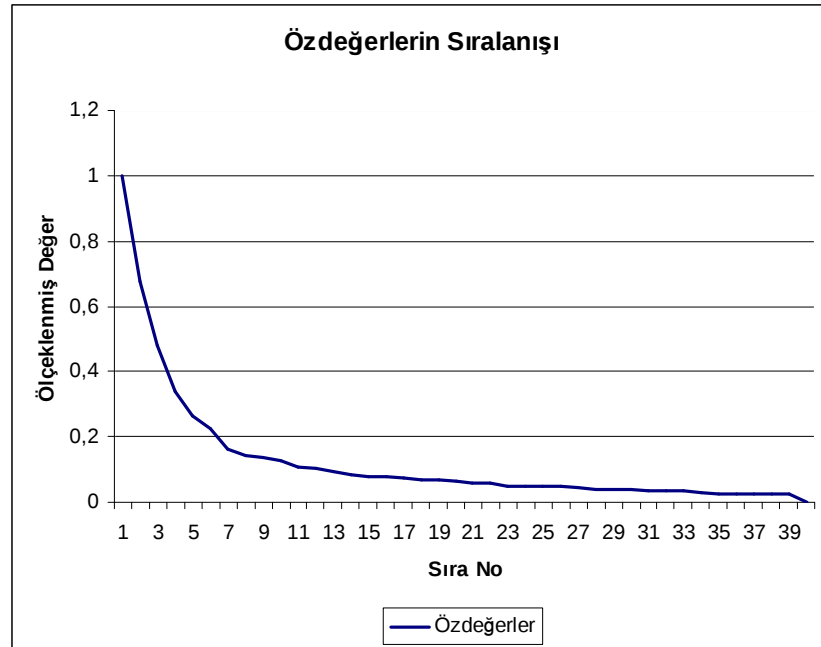
Kayıt işlemlerinin tamamlanmasıyla beraber uygulama programının yüz veritabanı için eğitilmesi gerekmektedir. Bu maksatla her yüz resmi $(m \times n)$ boyutlu bir matristen, satırları ucuca eklenmiş $(1 \times m.n)$ boyutlu bir vektöre dönüştürülür. Oluşturulan bu

vektör üzerine ışıklandırmanın etkisini azaltmak için gri tona çevirme işlemi uygulanmaktadır. Ancak kullanılan ORL veritabanındaki tüm yüz resimleri gri tonda olduğundan bu ön işlem basamağı bu eğitim kümesi için uygulanmamıştır.

Yüz resimlerinin vektör formatına getirilmesiyle sistemin eğitime geçilir ve sırasıyla;

- Tüm yüz resimlerinden ortalama bir yüz resminin çıkarılması,
- Her yüze ait özyüzün çıkarılması,
- Özyüzleri kullanarak her yüzü ifade eden bir özellik vektörünün çıkarılması,
- Ortalama yüz, özyüzler ve özellik vektörlerinin dosyalarak saklanması işlemleri gerçekleştirilir.

ORL veritabanı için yapılan sistem eğitiminde 40 kişinin 1 numaralı pozları kullanılmıştır. Bunun neticesinde 40 x 40 boyutlu kovaryans matrinden 40 adet özyüz hesaplanmıştır. Her özyüze karşılık gelen özdeğerlerin büyükten küçüğe sıralanması Şekil 3.20'de görülmektedir. Bir özyüze karşılık gelen özdeğer ne kadar büyükse o özyüzün tanıma işlemine katkısı da o kadar büyük olmaktadır.



Şekil 3.20 : Özdeğerlerin Sıralanışı
(Hesaplanan gerçek özdeğerler şekilde görülenlerin 128.843.294 katıdır.)

Yüzleri tanımlayan özellik vektörlerinin hesaplanmasında çıkarılan özyüzlerin hepsi kullanılmış ve sonuçta her yüz için (40×1) boyutlu bir özellik vektörü elde edilerek sistem eğitimi tamamlanmıştır.

Yüz tanıma işleminde de kullanıcının programa verdiği $(m \times n)$ boyutlu yüz resmi $(1 \times m.n)$ boyutuna getirilmekte ve tanıma işlemi için sırasıyla;

- 1) Aranılan yüz ile eğitim setinin ortalama yüzü arasındaki fark vektörü hesaplanır,
- 2) Bu fark vektörü özyüzler matrisiyle çarpılarak aranılan yüze ait özellik vektörü çıkarılır,
- 3) Bu özellik vektörünün mevcut 40 yüze ait özellik vektörlerine yakınlığı hesaplanıp elde edilen mesafeler küçükten büyüğe sıralanır,
- 4) Sıralanan mesafeler yüzdelik sisteme göre kullanıcının seçtiği kabul edilebilir tanıma oranı ile karşılaştırılıp ya tanınan kişiye ait resim getirilir ya da tanınmanın gerçekleşmediği şeklinde bir uyarı verilir.

3.4.2. PCA-NN İle Yüz Tanıma Bölümü

Bu çalışmaya konu olan karma yöntem özyüzler yönteminden elde edilen özellik vektörlerinin bir yapay sinir ağına uygulandığı PCA-NN yöntemidir. Bu yöntem kapsamında uygulamanın ikinci aşamasında gizli katmanındaki nöron sayısı, eğitim katsayısı ve kabul edilebilir hata eşiği değerleri kullanıcı seçimine göre belirlenen bir ileri beslemeli sinir ağı kullanılmıştır.

ORL veritabanının kullanıldığı sistem eğitiminde bu sinir ağının giriş katmanında (40×1) boyutundaki özellik vektörü için 40 giriş nöronu ve çıkış katmanında da 40 yüz resmi için 40 adet çıkış nöronu bulunmaktadır. Eğitim esnasında sinir ağına hangi yüze ait özellik vektörü verildiyse o yüze ait çıkış nöronunun çıktı değeri 1'e (yüksek) diğer yüzlere ait çıkış nöronlarının çıktı değerleri de 0'a (düşük) eşitlenmektedir. Sinir ağının eğitimi için $[0 - 1]$ aralığında çıkış veren ve denklem 3.3'de gösterilen tek kutuplu aktivasyon fonksiyonu kullanılmıştır. Öğrenme metodu olarak da denklem 3.10'da görülen delta öğrenme kuralı kullanılmıştır.

PCA-NN modelinin tanımda kullanılabilmesi için öncelikle sistemde kayıtlı tüm yüzler için özyüz ve özellik vektörlerinin çıkarılması gerekmektedir. Tanıma aşamasında kullanıcının seçtiği yüz resmi için de özyüzler yöntemi kullanılarak bir özellik vektörü hesaplanır. Bu özellik vektörü, eğitilmiş sinir ağına giriş olarak verilerek çıkış katmanındaki her nöronun üreteceği [0-1] aralığındaki çıkışlar incelenir. Bir nörona ait çıkış ne kadar büyükse tanınması istenen yüz ile bu nöronla bağlantılı yüz o kadar çok benzemektedir. Nöronun ürettiği çıkış düşükse tersi bir durum söz konusudur. Sonuçta nöronların ürettikleri çıkışlar büyükten küçüğe doğru sıralanırken sistemde kayıtlı resimler de tanınması istenen resme en çok benzeyenden en az benzeyene doğru sıralanmış olur.

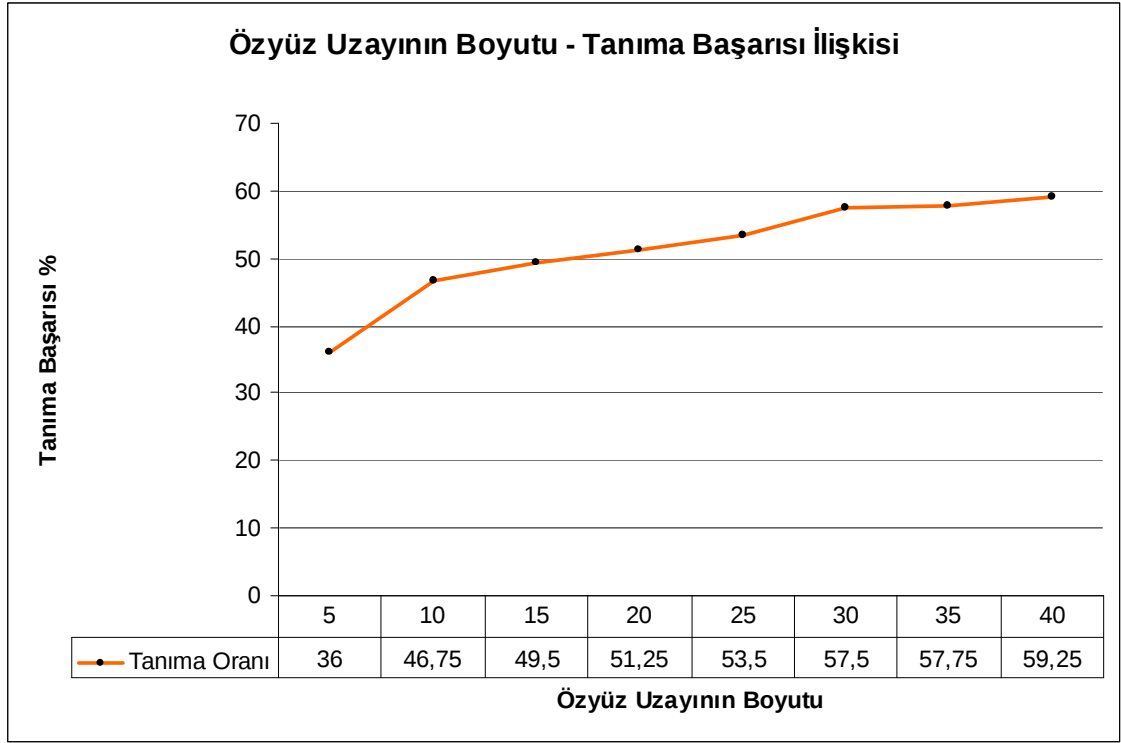
3.4.3. PCA-SVM İle Yüz Tanıma Bölümü

Uygulama programının yanı sıra Özyüzler yöntemiyle elde edilen özellik vektörleri Destek Vektör Makineleri (SVM) yöntemiyle de sınıflandırılmıştır. Bu amaçla özyüz eğitim işlemi sırasında ORL test kümesindeki 400 resmin tamamı için özellik vektörleri hesaplatılmış ve bu vektörler gerçekte hangi yüz resmine ait oldukları bilgisiyle beraber bir csv dosyasına kaydedilmiştir. Bu dosya, bünyesinde çeşitli sınıflandırma fonksiyonları barındıran 3. parti bir yazılım olan WEKA [28] yazılımına girdi olarak verilerek özellik vektörlerinin yüzlere göre sınıflandırılması yapılmıştır. Yazılımda sınıflandırıcı fonksiyon olarak bir destek vektör makinesi sınıflandırıcısı olan SMO seçilmiştir.

4. BULGULAR

Tez çalışması kapsamında önceki bölümde detaylı olarak anlatılan Özyüzler Yöntemi (PCA), PCA-NN ve PCA-SVM karma yöntemleri kullanılarak yüz tanıma işlemi incelenmiştir. Hazırlanan uygulamaların hepsi Özyüz temelli yüz tanıma işleminden elde edilen özellik vektörlerini giriş olarak kullanmaktadır. Çıkışı ise 3 farklı sınıflandırma yöntemine göre belirlemektedirler. Bu sınıflandırıcılar sırasıyla: özellik vektörlerinin farkının hesaplanmasına dayalı olan “Vektör Farkına göre Sınıflandırıcı” (VD) (denklem 3.20), “Yapay Sinir Ağları” (NN) ve “Destek Vektör Makineleri” (SVM) sınıflandırıcılarıdır.

Özyüzler yönteminde, sistemin eğitimi tanımaya katkısı en fazla olan (kovaryans matrsten elde edilen özvektörlerden özdeğeri en büyük olanlar) yüz resimlerinden elde edilen özyüz uzayıyla gerçekleştirilir. Böylece sisteme yeni bir kişinin resmi eklendiğinde tekrar sistem eğitimi yapmaya gerek kalmaz. Yeni eklenen resim, mevcut resimlerden elde edilen özyüz uzayıyla işlenerek kendine özgü özellik vektörleri hesaplanır. Ancak özyüz uzayında kullanılan yüz resimlerinin sayısının toplam tanıma başarısına doğrudan bir etkisi bulunur. Bu nedenle çalışma kapsamında incelenen ilk konu özyüz uzayının boyutundaki değişimin sonuçtaki toplam başarıya etkisi olmuştur. Bu amaçla ORL veritabanındaki 40 kişinin 1. pozlarının kayıtlı olduğu bir sistemde, özyüz uzayında işlenecek resimlerin sayısı sırasıyla 5’ten 40’a kadar beşer beşer artırılarak her seferinde toplam tanıma başarısı ölçülmüştür. Sınıflandırıcı olarak VD (Vector Discrimination) sınıflandırıcı kullanılmıştır. Elde edilen sonuçlar Şekil 4.1’de görülmektedir.

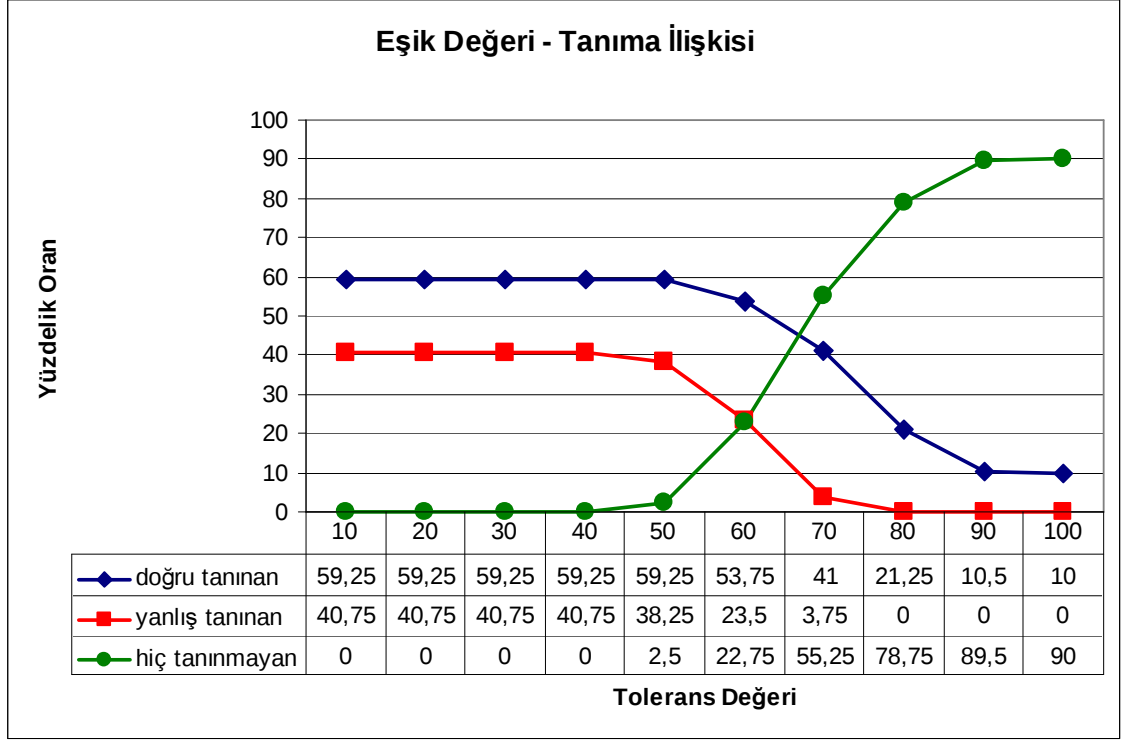


Şekil 4.1 : Özyüz Uzakı Boyutu – Tanıma Başarısı İlişkisi

Şekilde 4.1’de görüldüğü üzere özyüz uzayının boyutundaki artışın toplam tanıma başarısı üzerinde doğrudan bir etkisi söz konusudur. Bunun nedeni özyüz uzayına düşürülen her yeni yüz resminin aynı zamanda tüm özyüzlere ve onlardan elde edilen özellik vektörlerine katkıda bulunması ve olası yanlış tanımları engellemesidir. Bununla beraber özyüz uzayının boyutundaki artış, her yüzü tanımlayan özellik vektörlerinin de boyutunu artırdığından yüzleri birbirlerinden ayırt etmede kullanılabilecek daha fazla özellik ortaya çıkmaktadır. Böylece daha iyi bir sınıflandırma yapılarak toplam tanıma başarısı artırılabilir.

Yüz tanıma işleminde kullanılan özyüz uzayının özelliklerinin yanında sınıflandırma işlemi için kullanılan tanıma “eşik değeri”nin de önemi büyüktür. Eşik değeri, sınıflandırma işleminde aranan resim ile o an karşılaştırma işlemi için eğitim setinden seçilen bir resmin özellik vektörlerinin kabul edilebilir benzeşme oranını ifade eden bir değerdir. Bu değer, tanıma işlemi sonunda operatörün danışmanlık etkisi bulunmayan sistemlerde çok önemli bir role sahiptir ve olası yanlış tanımları en aza indirebilmek amacıyla mümkün olduğunca uygun seçilmelidir. Şekil 4.2’de VD sınıflandırma

kullanılan bir sistemde eşik değerinin tanıma oranına etkisi görülmektedir. Bu testte 40 kişinin 1. pozundan oluşan bir eğitim kümesi kullanılmıştır.

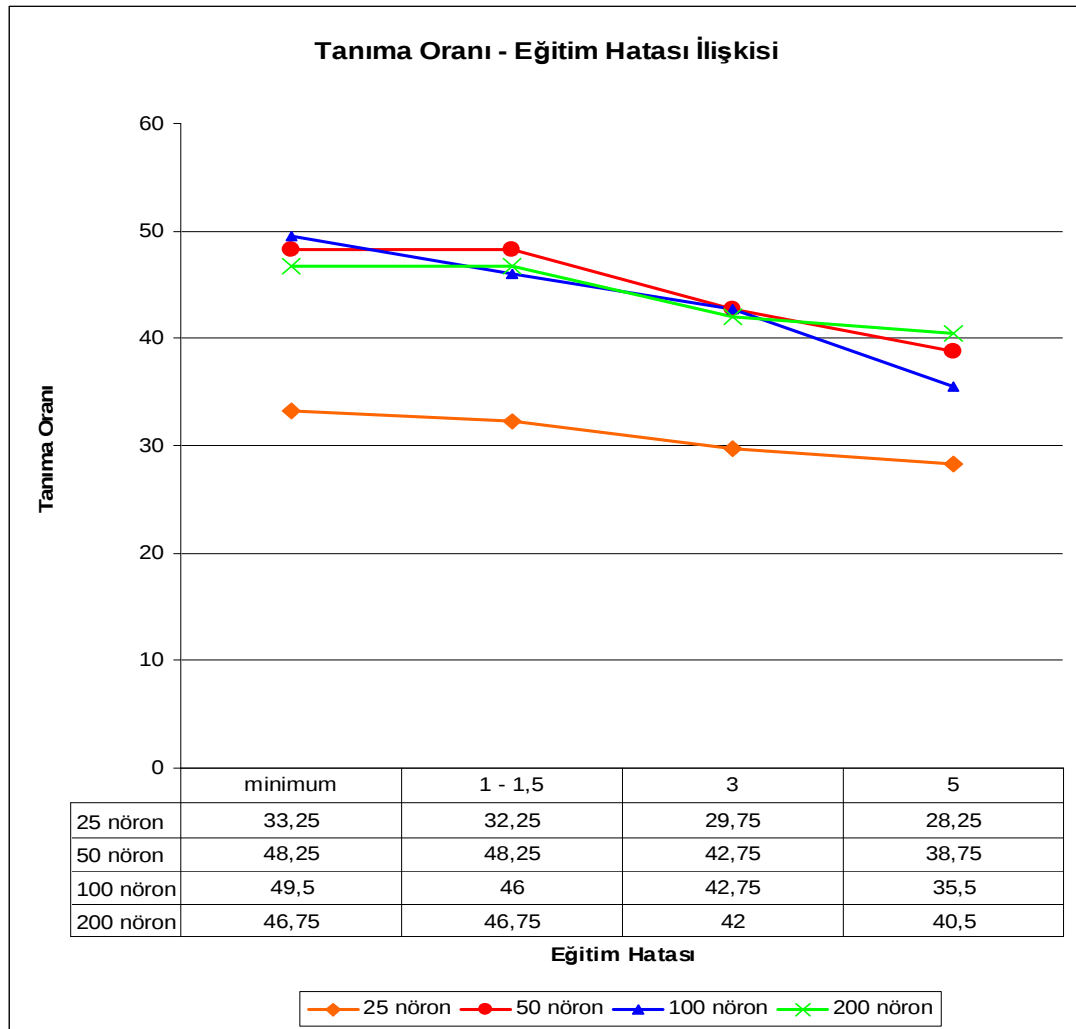


Şekil 4.2 : Eşik Değeri – Tanıma İlişkisi

Şekil 4.2’de görüldüğü üzere karşılaştırılan iki resmin özellik vektörlerinin benzerliği %10 seviyesinden %40 seviyesine kadar çekildiğinde doğru tanıma, yanlış tanıma ya da hiç tanıma oranlarında hiçbir değişiklik olmamaktadır. Ancak %50’lik eşik değeri seviyesinde doğru tanıma oranı %59,25 değerinde sabit kalırken yanlış tanınan resimlerin oranında %2,5’lik (toplamda 400 pozdan 10’u) bir azalmanın olduğu görülmektedir. Bu noktada yanlış tanınacak olan 10 poz sistem tarafından hiç tanınmayan resimler olarak elenmiştir. Eşik değeri %60 seviyesine çekildiğinde bir önceki duruma (%50 eşik değerine) göre doğru tanıma oranında %5,5’lik (22 poz) bir azalma, yanlış tanıma oranında da %14,75’lik (59 poz) bir azalma olduğu görülmektedir. Eşik değeri %70 seviyesine getirildiğinde de toplamda tanıma oranı %41 seviyesine düşerken yanlış tanıma oranı da %3,75 seviyesine kadar düşmüştür. Tanıma işleminde son kararı bir insan operatörün vermediği sistemler için eşik değerinin, doğru tanıma oranını mümkün olduğunca az etkileyecek ancak yanlış tanıma oranını da

mümkün olduğunca fazla azaltarak sistemde kayıtlı olmayan pozları “hiç tanınmayan” olarak sınıflandıracak şekilde seçilmesi gerekir. ORL veritabanındaki 40 kişinin 1. pozlarından oluşan eğitim seti için bu özelliği sağlayacak eşik değerinin %50 ile %70 arasında seçilebileceği görülmektedir.

Özyüzler yöntemiyle eğitim kümesine ait özellik vektörleri çıkarıldıktan sonra 1. karma yönteminde (PCA-NN) yapay sinir ağları sınıflandırıcısıyla da tanıma işlemi gerçekleştirilmiş ve 40 resimden oluşan eğitim kümesi için 40 girişli ve 40 çıkışlı çok katmanlı bir yapay sinir ağı yapısı kullanılmıştır. Şekil 4.3’de gizli katmandaki değişik sayıdaki nöron için hata oranındaki değişimin, PCA-NN yöntemiyle tanıma başarısına etkisi görülmektedir. Bu uygulamada eğitim katsayısı ve momentum değerleri 0,5 olarak seçilmiştir.



Şekil 4.3 : Tanıma Oranı – Eğitim Hatası İlişkisi

Şekil 4.3’de gösterilen PCA-NN uygulamasında gizli katmandaki nöron sayısı sırasıyla 25, 50, 100 ve 200 olarak seçilmiştir. Farklı boyuttaki gizli katmanlar arasından 25 nörona sahip örnek diğer örneklerle göre belirgin bir şekilde en düşük tanıma oranını göstermiştir. Bu aşamada gizli katmanda kullanılan nöron sayısındaki artışın tanıma oranına olumlu yönde etki ettiği söylenebilir ancak 200 nöronluk yapıda erişilebilen minimum hatanın gizli katmanda kullanılan nöron sayısına bağlı olarak daha yüksek olması nedeniyle en iyi tanıma başarısı gizli katmanında 100 nöron bulunduran yapıdan elde edilebilmiştir. Bu uygulamaların gizli katmanlarında kullanılan nöron sayısı – erişilen eğitim hatası eşleşmeleri Tablo 4.1’de verilmiştir. Bununla beraber tanıma başarısının eğitim hatasıyla ters orantılı olduğu da görülebilir.

Tablo 4.1 : Gizli katmanda kullanılan nöron sayısı – erişilen eğitim hatası eşleşmesi

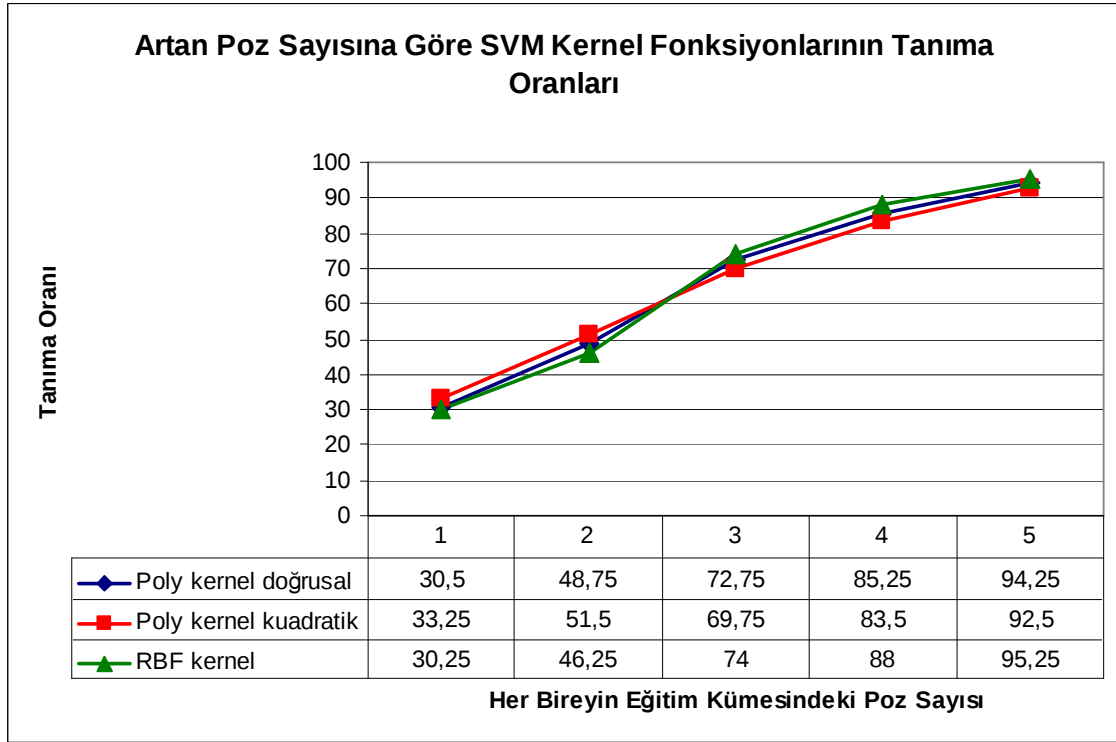
Nöron Sayısı	25	50	100	200
Erişilen minimum hata	0,5	0,5	0,5	1,5

Özyüzler yöntemiyle elde edilen özellik vektörleri YSA’nın yanı sıra Destek Vektör Makinesi (SVM) ile de sınıflandırılmıştır. Bölüm 3.3’de de belirtildiği gibi SVM sınıflandırıcıda, test kümesindeki verileri daha yüksek boyutlu bir uzayda ifade etmek amacıyla kernel fonksiyonu adı verilen fonksiyonlar kullanılmaktadır. Yüz tanıma uygulamasında bu fonksiyonlardan en yaygın kullanılan iki türü Poly-kernel ve RBF kernel kullanılmıştır. Poly-kernel fonksiyonunda (Denklem 3.29) p üstel değeri 1 olarak seçildiğinde doğrusal poly kernel fonksiyonu, 2 olarak seçildiğinde de kuadratik poly kernel fonksiyonu söz konusudur. WEKA uygulama yazılımı ile yapılan testlerde kullanılan çekirdek fonksiyonları için seçilen parametreler şu şekildedir :

- 1) Polykernel Doğrusal : $c = 100$, $\gamma = 1$, BuildLogisticModel = True
- 2) Polykernel Kuadratik : $c = 100$, $\gamma = 2$, BuildLogisticModel = True
- 3) RBF Kernel : $c = 100$, $\gamma = 0,1$, BuildLogisticModel = False

Her birey için artan poz sayısı (1’den 5’e kadar) oluşturulan eğitim kümeleri üzerinde SVM sınıflandırıcının farklı kernel fonksiyonları ile kullanılması sonucu elde edilen tanıma oranları Şekil 4.4’de görülmektedir. Artan poz sayısına göre ORL veritabanındaki bireylerin eğitim kümesine dahil edilen poz numaraları 1 poz için : 1.

poz, 2 poz için : 1. ve 5. pozlar, 3 poz için : 1. , 5. ve 9. pozlar, 4 poz için : 1. , 5. , 7. ve 9. pozlar, 5 poz için : 1. , 3. , 5. , 7. ve 9. pozlar şeklinde seçilmiştir. Poz numaralarının bu şekilde seçilmesinin nedeni her birey için mümkün olduğunca değişik tipteki resimlerin eğitim kümesine verilmek istenmesidir. Zira ORL veritabanında arka arkaya sıralanan pozlarda genelde çok az bir değişim görülmektedir.



Şekil 4.4 : Artan Poz Sayısı – SVM Kernel Fonksiyonlarının Tanıma Oranları İlişkisi

Şekil 4.4’de de görülebileceği gibi SVM yöntemiyle yapılan sınıflandırmada her birey için eğitim kümesinde 1 veya 2 poz kullanıldığında en yüksek tanıma oranını kuadratik tipteki poly kernel, en düşük tanıma oranını ise RBF kernel vermektedir. Kişi başına düşen poz sayısı 3-5 aralığından seçildiğinde ise en yüksek tanıma oranı RBF kernel ile, en düşük tanıma oranı ise kuadratik tipteki poly kernel ile elde edilmiştir. Burada seçilen eğitim kümesinin boyutundaki artışa bağlı olarak RBF kernel’in diğerlerine göre daha iyi bir sınıflandırma sonucu verdiği görülmektedir. Doğrusal tipteki polykernel fonksiyonu ise değişen poz sayılarının tamamında diğer iki yöntemin arasında bir tanıma performansı göstermiştir.

Tez uygulaması kapsamında son olarak, kullanılan üç tipteki sınıflandırıcının (VD, NN, SVM) artan poz sayısına göre gösterdikleri tanıma performansı incelenmiştir. Birey başına düşen poz sayısının 1’den 10’a kadar değiştiği eğitim kümeleri için kullanılan poz numaraları Tablo 4.2’de görülmektedir.

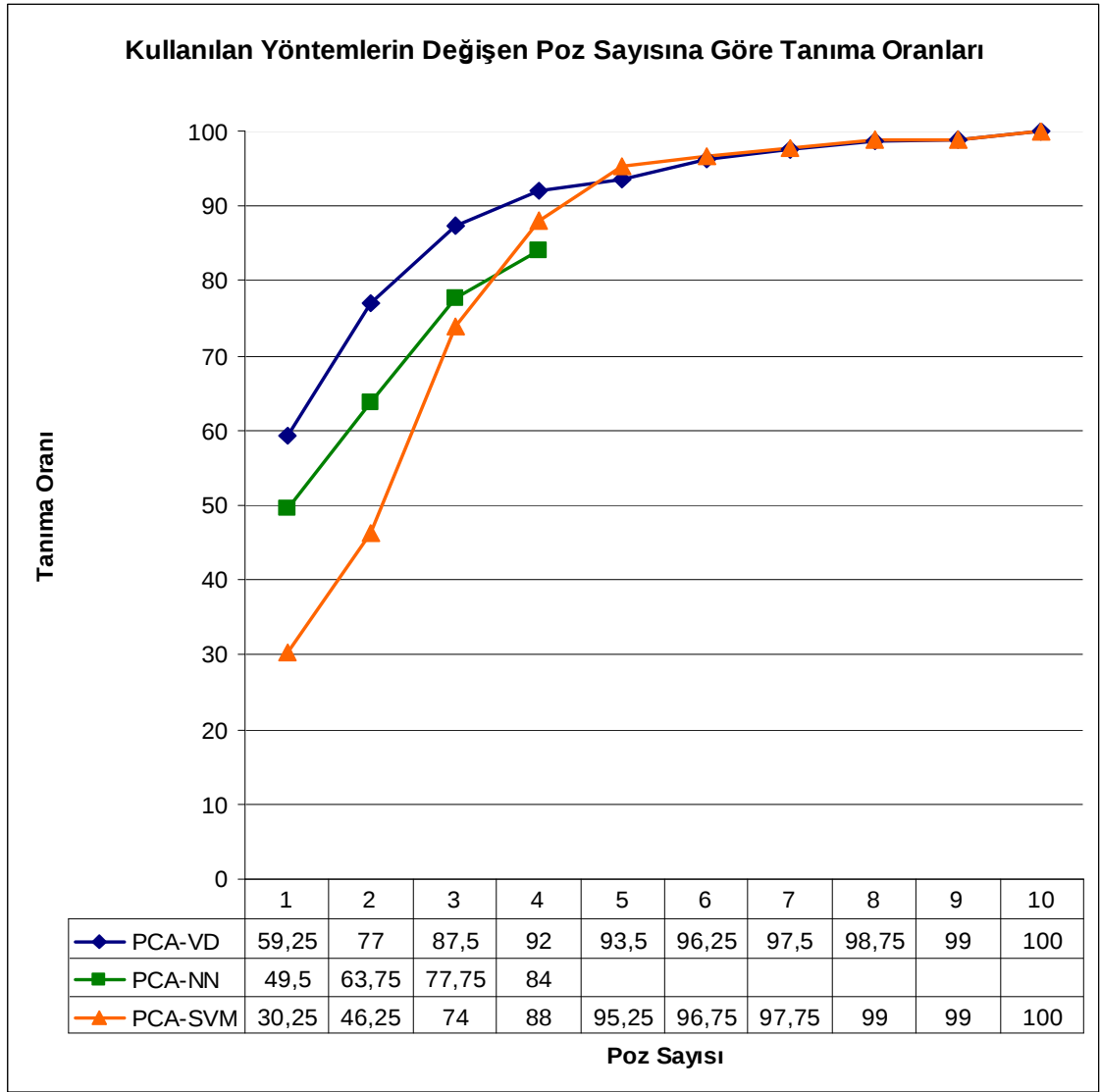
Tablo 4.2 : Artan poz sayısı için eğitimde kullanılan pozlar

Poz No	1	2	3	4	5	6	7	8	9	10
Poz Sayısı										
1 poz	X									
2 poz	X				X					
3 poz	X				X				X	
4 poz	X				X		X		X	
5 poz	X		X		X		X		X	
6 poz	X		X		X	X	X		X	
7 poz	X		X	X	X	X	X		X	
8 poz	X		X	X	X	X	X	X	X	
9 poz	X	X	X	X	X	X	X	X	X	
10 poz	X	X	X	X	X	X	X	X	X	X

PCA-NN yöntemindeki YSA sınıflandırıcısı için gizli katmanda en yüksek tanıma oranını sağlayan 100 nöron seçilmiştir. Eğitim katsayısı ve momentum da 0,5 olarak alınmıştır. PCA-NN yöntemi, yapay sinir ağı yapısının kabul edilebilir hata oranına erişmesi için gereken eğitim döngüsü çok uzun sürdüğünden 1–4 arası pozdan oluşan eğitim kümelerinde kullanılmıştır.

PCA-SVM yöntemindeki Destek Vektör Makineleri sınıflandırıcısında, artan poz sayısına göre en iyi sınıflandırma oranlarının elde edildiği RBF kernel kullanılmıştır.

Üç yöntemle elde edilen tanıma oranları Şekil 4.5’de görülmektedir.



Şekil 4.5 : Kullanılan Yöntemlerin Artan Poz Sayısına Göre Tanıma Oranları

Şekil 4.5’de de görüldüğü üzere birey başına düşen 1’den 3’e kadar değişen poz sayısı için en iyi tanıma oranını PCA-VD (Özyüzler - Vektör Farkı) yöntemi vermiş, onu PCA-NN yöntemi takip etmiştir. 4 pozluk eğitim setinde yine VD sınıflandırıcı en iyi tanıma oranını verirken onu SVM sınıflandırıcı takip etmiştir. 5-10 poz aralığında ise SVM sınıflandırıcının, VD sınıflandırıcıya tanıma oranı açısından az da olsa (en fazla % 1.75) üstünlük sağladığı görülmektedir. Buradan SVM sınıflandırıcının eğitim kümesindeki örnek sayısının az olduğu durumlarda diğer yöntemler kadar iyi sınıflandırma sonuçları veremediği ancak eğitim kümesindeki artan örnek sayısı ile beraber diğer yöntemlerden daha iyi sınıflandırma sonuçları verebileceği sonucuna varılabilir.

5. TARTIŞMA VE SONUÇ

Bu tez çalışması kapsamında literatürde adı geçen yüz tanıma yöntemlerinin bazılarına değinilmiş ve bu yöntemler içinden seçilen “Özyüzler ile Yüz Tanıma” yöntemi kullanılarak eğitim kümesindeki resimlerin özniteliklerini gösteren özellik vektörleri elde edilmiştir. Bu özellik vektörleri, “Vektör Farkına göre Sınıflandırma” (PCA-VD), “Yapay Sinir Ağları ile sınıflandırma” (PCA-NN) ve “Destek Vektör Makineleri ile sınıflandırma” (PCA-SVM) yöntemlerinin uygulanmasıyla tanıma işleminde kullanılmıştır. Böylece üç farklı sınıflandırıcının kullanıldığı bir karma yüz tanıma sistemi elde edilmiştir.

Bir yüz tanıma uygulaması için sisteme verilen giriş bilgisinin formu çok önemlidir. Böyle bir uygulamada kullanılacak test kümesinin, yüzdeki detayların net olarak seçilebildiği ve buna bağlı olarak her yüzün kendine özgü özelliklerinin kolaylıkla çıkarılabildiği bir yapıda olması arzulanmaktadır. Ancak gerçek dünya uygulamalarında bu her zaman mümkün değildir. Işıklandırma şiddetindeki farklılıklar, kazanç modülünden (kamera, fotoğraf makinesi vb.) kaynaklanan gürültü, resmin arkaplanında bulunan ve tanımayı güçleştiren unsurlar, resimdeki yüzün eğimi gibi etkenler tanımayı güçleştiren unsurlardır. Bu noktada yüz algılama ve bazı ön işleme modülleri devreye sokularak yüz resminin karakteristik özelliklerini belirginleştirmeye yönelik çabalar gösterilebilir. Hazırlanan uygulama programında bahsedilen bu etkenlerden mümkün olduğunca az etkilenmek amacıyla boyutları sabit olan, ışıklandırma etkisini azaltmak için gri formata çevrilmiş, sadece yüz bölgesinden çekilmiş pozları barındıran ve 40 kişinin toplamda 400 pozundan oluşan ORL yüz veritabanı kullanılmıştır. Bu test kümesi yüz tanıma uygulamalarında kullanılan standart bir yüz veritabanıdır ve içerdiği resimler sadece yüzdeki ifade değişimlerine göre değil aynı zamanda cepheden ve profilden verilen pozlara göre de değişmektedir.

Günümüzde yaygınlıkla kullanılan yüz tanıma yöntemlerinin sınıflandırma başarıları kullandıkları yöntemlere göre ürettikleri hata oranı (yanlış tanınan resimlerin yüzdeliği) ile

belirlenmektedir. Bu yöntemlerin bazılarının, ORL veritabanı ile beraber kullanımlarından elde edilen hata oranları Tablo 5.1’de [31] gösterilmiştir. Bu sonuçların alındığı uygulamaların sonuncusu hariç tamamında eğitim kümesinde kişi başına düşen poz sayısı 5 olarak seçilmiştir.

Tablo 5.1 : ORL veritabanı için bazı yöntemlerle çıkarılan hata oranları

YÖNTEM	HATA ORANI	EĞİTİMDE KULLANILAN YÜZ SAYISI
CNN	%3.8	5
SVM + PCA	%3	5
ICA	%15	5
Özyüzler	%9.5	5
Elastik Eşleme	%20	5
Gabor Süzgeçleri + Sıra Korelasyonu	%8.5	5
Continuous n-tuple classifier	%2.7	5
SVM + Gabor Dalgacık Modeli	%1.25	4

Tasarlanacak bir yüz tanıma sisteminde kullanılacak olan yöntemi sadece Tablo 5.1’deki yöntemlerin ürettikleri hata oranlarının düşüklüğüne ya da yüksekliğine bakarak seçmek yeterli değildir. Aynı zamanda kullanılacak olan yöntemin yapılmak istenen işin doğasına uygun olması (işleme süresi, gerçek zamanlı çalışma, kullanılacak resimlerin boyutu, güncellenme sıklığı vb.) da gerekmektedir. Örneğin bir özellik temelli yüz tanıma yöntemi olan Elastik Eşleme yöntemi genelde küçük boyutlu resimlerin kullanıldığı sistemler için uygun değildir. Çünkü bu yöntem yüzdeki belirli bölgelerin detaylarına yönelik olarak işlemektedir. Bunun aksine yüz resmindeki tüm pikselleri işleyen PCA ve LDA gibi yöntemler küçük boyutlu resimlerle kullanılabilirler [32]. Aynı zamanda PCA ve LDA yöntemleri Elastik Eşleme yöntemine göre çok daha düşük işleme zamanına gereksinim duymaktadırlar. Buna karşılık PCA tabanlı (Özyüzler gibi) yöntemler ışıklandırma etkisinden büyük ölçüde etkilenirken Elastik Eşleme yöntemi bundan daha az etkilenmektedir [33]. Ayrıca sisteme yeni eklenen her resimle beraber daha iyi tanıma başarısı için Özyüzler yönteminde baştan bir eğitim döngüsü yapılması gerekirken, Elastik Eşleme yöntemi için böyle bir durum söz konusu değildir.

Bu çalışmada ORL yüz veritabanı esas alınarak yapılan incelemelerde çeşitli konulara değinilmiştir. Öncelikle her resmin özniteliklerini gösteren özellik vektörünün hesaplanmasında kullanılan özyüz uzayının boyutundaki değişimin sonuçta elde edilen toplam tanıma performansına etkisi incelenmiştir. Bilindiği üzere bir yüz tanıma sistemine kaydedilen her yeni yüz, o yüzün kaydedildiği sınıfta yeni bir eğitim döngüsünü beraberinde getirebilmektedir. Ancak bu durum içeriği sürekli olarak güncellenen bir tanıma sistemi için uygun değildir. Özyüzler yöntemiyle çalışan bir yüz tanıma sisteminde kaydedilen her yeni yüz resmi için tüm kayıtlı resimleri de içerecek bir eğitim döngüsünü baştan yapma zorunluluğu yoktur. Yeni yüz resmi mevcut özyüz uzayına düşürülüp, karakteristik özelliklerini içeren özellik vektörü hesaplandıktan sonra tanıma işleminde kullanılmak üzere saklanır. Ancak yapılan incelemede görüldüğü üzere böyle bir zorunluluk olmasa da sisteme eklenen her yeni yüzle baştan bir eğitim sürecine girilmesinin gerekli olduğu görülmüştür. Zira özyüz uzayının boyutu ile toplam tanıma performansı arasında doğru orantı bulunmaktadır. Bunun böyle olmasındaki en büyük etken özyüz uzayının boyutunun artmasıyla her yüzü tanımlayan özellik vektörlerinin de boyutunun artmasıdır. Böylece sınıflandırıcılar daha net bir seçme işlemi uygulayabilmektedir (Şekil 4.1).

Yüz tanıma sistemleri temel olarak karşılaştırdığı iki örnek resmin birbirine benzerliğini ölçen ve tanımlanan eşik değerlerine göre bu örneklerin aynı ya da farklı olduğunu ortaya koyan sistemlerdir. Bu noktada, insan operatör desteği olmayan otomatik bir yüz tanıma sistemi, doğru tanıma oranını mümkün olduğunca yüksek, yanlış tanıma oranlarını da azami ölçüde düşük tutmalıdır. Böyle bir sistem için ORL veritabanını kullanan bir uygulamada bu eşik değerinin %50 - %70 aralığında seçilebileceği görülmüştür (Şekil 4.2).

Ayrıca bu çalışma ile çok katmanlı yapay sinir ağı kullanan karma yapı bir yüz tanıma sisteminde gizli katmanda kullanılan nöron sayısının çok düşük olması durumunda iyi sınıflandırma sonuçları elde edilemeyeceği sonucuna varılmıştır. Aynı sonuç, gizli katmanda kullanılan nöron sayısının çok yüksek olduğu bazı durumlarda eğitim hatası yeterince azaltılamadığından da ortaya çıkmaktadır. Çünkü eğitim hatası ile doğru sınıflandırma arasında bir ters orantı bulunmaktadır (Şekil 4.3).

Tez çalışması sonucunda, kullanılan sınıflandırıcıların kendi aralarında değişen parametrelere göre ürettikleri tanıma oranları da belirlenmiştir. Bu aşamada yüz tanıma sisteminde kayıtlı örnek sayısının artışına bağlı olarak Destek Vektör Makineleri ile sınıflandırmanın (PCA-SVM) daha başarılı sonuçlar verdiği ancak daha küçük boyutlu eğitim kümelerinde ise Özyüzler yöntemiyle her yüz için elde edilen özellik vektörlerinin farkına göre yapılan sınıflandırmanın (PCA-VD) daha başarılı olduğu görülmüştür. Guo, Li ve Chan [34] de yaptıkları çalışmada ORL veritabanından seçtikleri 200 pozlu Özyüzler yöntemiyle sistem eğitime tabi tutmuş ve elde ettikleri nitelik vektörlerini SVM ve NCC (Nearest Center Classification) sınıflandırıcıları ile sınıflandırmışlardır. Bu çalışmalarında minimum sınıflandırma hatalarını SVM yöntemi için %3, NCC yöntemi için de %5.25 olarak elde etmiş ve SVM yönteminin sınıflandırma başarısını göstermişlerdir. Benzer şekilde Dai ve Zhou [35] da PCA tabanlı özellik çıkarıcı uygulamalarında ORL veritabanı için SVM ve NMC (Nearest Mean Classification) sınıflandırıcılarını karşılaştırmış ve SVM ile daha yüksek tanıma oranları (SVM: %91.1 – NMC: %70.3) elde etmişlerdir. Bu tez çalışmasında ise seçilen 200 pozdan oluşan eğitim kümesinden elde edilen sınıflandırma hataları PCA-SVM yöntemi için %4.75, PCA-VD yöntemi için de %6.5'dir (Şekil 4.5).

Uygulama programında kullanılan PCA-NN yöntemi ile diğer yöntemlere göre üstün sınıflandırma sonuçları elde edilememiştir. Ancak bu konuda literatürde başarılı PCA-NN uygulamaları da bulunmaktadır. Eleyan ve Demirel [36] PCA ve LDA yöntemlerini ayrı ayrı uygulamış ve bu yöntemlerle elde ettikleri özellik vektörlerini yapay sinir ağları kullanarak sınıflandırmışlardır. Bu uygulamalarında eğitim kümesindeki artan poz sayısı için NN sınıflandırıcı Öklid mesafesiyle sınıflandırmaya göre daha yüksek tanıma oranları (4 pozlu uygulamada Öklid mesafesiyle sınıflandırma başarısı : %77 , NN ile sınıflandırma başarısı : %80) göstermiştir.

Bu çalışmanın neticesinde ileriye yönelik olarak yapılacak uygulamalarda “Doğrusal Diskriminant Analizi” (Linear Discriminant Analysis – LDA) yöntemini kullanarak eğitim kümesinin sınıflandırılmasına dayalı bir yüz tanıma sisteminin tasarlanması planlanmaktadır.

KAYNAKLAR

- 1) 2008, *Biometrics* [online], Wikipedia, <http://en.wikipedia.org/wiki/Biometric> [Ziyaret Tarihi: 1 Mart 2008]
- 2) 2008, *Keystroke dynamics* [online], Wikipedia, http://en.wikipedia.org/wiki/Keystroke_dynamics [Ziyaret Tarihi: 1 Mart 2008]
- 3) KILINÇ, Erdal, 2005, *Yüzler kameraya takılacak* [online], Milliyet, <http://www.milliyet.com.tr/2005/07/09/guncel/gun00.html> [Ziyaret Tarihi: 2 Şubat 2008]
- 4) JAIN, A. K. (28-30 Nisan 2004), "Biometric recognition: how do I know who you are?", Signal Processing and Communications Applications Conference, 2004. Proceedings of the IEEE 12th: 3 – 5
- 5) GALTON, F., "Personal identification and description-II", Nature 201-203, 28 Haziran 1988
- 6) HJELMAS, E. and LOW B. K., "Face Detection: A Survey," Computer Vision and Image. Understanding, vol. 83, pp. 236-274, 2001
- 7) JIANG X., BINKERT M., ANCHERMANN B., and BUNKE H., "Towards detection of glasses in facial images", Pattern Anal. Appl. 3, 2000, 9–18.
- 8) WONG C., KORTENKAMP D. and SPEICH M., "A mobile robot that recognises people", in IEEE Int. Conf. on Tools with Artificial Intelligence, 1995.
- 9) ROWLEY H. A., BALUJA S. and KANADE T., "Neural network-based face detection", IEEE Trans. Pattern Anal. Mach. Intell. 20, Ocak 1998, 23–38.
- 10) AKALIN Volkan, 2003, *Face recognition using eigenfaces and neural networks*, Yüksek Lisans tezi, Orta Doğu Teknik Üniversitesi.
- 11) BERGER, M. O. and MOHR, R., "Towards autonomy in active contour models", Proc. 10th Int. Conf. on Pattern Recognition, pp. 847-851, (1990).
- 12) KASS, M., WITKIN, A. and TERZOPOULOS, D., "Snakes: Active contour models", Int. J. of Computer Vision, pp. 321-331, (1987).

- 13) YUILLE, A. L., COHEN, D. S. and HALLINAN, P. W., "Feature extraction from faces using deformable templates", Proc. of CVPR, (1989).
- 14) SHACKLETON A. and WELSH W. J., "Classification of facial features for recognition", in IEEE Proc. of Int. Conf. on Computer Vision and Pattern Recognition, Hawaii, 1991, pp. 573–579.
- 15) CHOW G. and LI X., "Towards a system for automatic facial feature detection", Pattern Recog. 26, 1993, 1739–1755.
- 16) LANITIS A., TAYLOR C. and COOTES T. F., "Automatic interpretation and coding of face images using flexible models", IEEE Trans. Pattern Anal. Mach. Intell. 19, 1997.
- 17) TURK M. and PENTLAND A., "Eigenfaces for Recognition", Journal of Cognitive Science, s. 71-86, 1991.
- 18) SIROVICH L. and KIRBY M., "Low-dimensional procedure for the characterization of human faces", J. Opt. Soc. Amer. 4, 1987, 519–524.
- 19) PENTLAND A., MOGHADDAM B. and STRARNER T., "View-based and modular eigenspaces for face recognition", in IEEE Proc. of Int. Conf. on Computer Vision and Pattern Recognition, 1994.
- 20) 2007, *Algorithms* [online], face-rec.org, <http://www.face-rec.org/algorithms/> [Ziyaret Tarihi: 10 Şubat 2008]
- 21) WISKOTT L., FELLOUS J., KRUGER N. and MALSBURG C. V., "Face Recognition by Elastic Bunch Graph Matching", IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), Vol. 19, No.7, pp. 775-779, 1997.
- 22) GORDON, G., "Face recognition from frontal and profile views", Proc. Int'l Workshop on Face and Gesture Recognition, pp. 47-52 (1996).
- 23) HU, Y., JIANG, D., YAN, S., ZHANG, L., ZHANG, H., 2004, "Automatic 3D Reconstruction for Face Recognition". Proc. 6th IEEE International Conference on Automatic Face and Gesture Recognition, p.843
- 24) 2006, *Face Recognition and 3D Face Modeling* [online], microsoft.com, <http://download.microsoft.com/download/C/8/8/C88B493F-0A8B-4040-A935-CF8AB63859C7/FaceRecognition.Technical.doc> [Ziyaret Tarihi: 14 Mart 2008]
- 25) LIU C. and WECHSLER H., "Evolutionary Pursuit and Its Application to Face Recognition", IEEE Trans. Patt. Anal. Mach. Intell. 22, (2000) 570-582.

- 26) 2008, *Back Propagation* [online], Wikipedia, <http://en.wikipedia.org/wiki/Backpropagation> [Ziyaret Tarihi: 15 Mart 2008]
- 27) 2002, *The Database Of Faces* [online], Cambridge University Computer Laboratory, <http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html> [Ziyaret Tarihi: 20 Şubat 2008]
- 28) 2008, *Weka 3 - Data Mining with Open Source Machine Learning Software in Java* [online], The University Of Waikato, <http://www.cs.waikato.ac.nz/ml/weka/> [Ziyaret Tarihi: 15 Nisan 2008]
- 29) VAPNIK V. N., “Statistical Learning Theory”, John Wiley & Sons, New York, 1998.
- 30) HEISELE B., HO P. and POGGIO T., “Face Recognition with Support Vector Machines”, Global versus Component-based Approach. Proc. of ICCV’01, VOL-II, (2001) 688-694.
- 31) KEPENEKCI B. and AKAR G. B., “Face classification with support vector machine”, Signal Processing and Communications Applications Conference, 2004. Proceedings of the IEEE 12th Volume , Issue , 28-30 April 2004 Page(s): 583 – 586.
- 32) ZHAO W., CHELLAPPA R., PHILLIPS P. J., and ROSENFELD A., “Face recognition: A literature survey”, ACM Comput. Surv., 35(4):399–458, 2003.
- 33) ZHANG J., YAN Y. and LADES M., “Face recognition: Eigenface, elastic matching, and neural nets”, Proceedings of IEEE. v85 i9. 1423-1435.
- 34) GUO G, LI S. Z. and CHAN K., “Face Recognition by Support Vector Machines”, Proc. of 4th IEEE Intern. Conf. On Automatic Face and Gesture Recognition of AFG’ 00, (2000) 96-201.
- 35) GUANG D. and CHANGLE Z., “Face Recognition Using Support Vector Machines with the Robust Feature”, Proceedings of the 2003 IEEE International Workshop on Robot and Human interactive Communication Millbrae. Calimia, USA. Oct. 31 -Nov. 2,2003.
- 36) DELAC K. and GRGIC M. , 2007 , *Face Recoginiton* - (ELEYAN A. and DEMIREL H. , “PCA and LDA based Neural Networks for Human Face Recognition”), I-Tech , Vienna , 978-3-902613-03-5.

ÖZGEÇMİŞ

Ergün GÜMÜŞ 1984 yılında Herrenberg/Almanya’da doğdu. Ortaokul ve Lise öğrenimini İstanbul Vefa Anadolu Lisesi’nde gördü. 2006 yılında İstanbul Üniversitesi Bilgisayar Mühendisliği Bölümü’nden mezun oldu. Aynı yıl Eylül ayında İstanbul Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı’nda yüksek lisans öğrenimine başladı.