



**İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

YÜKSEK LİSANS TEZİ

**YÜZ TANIMA SİSTEMLERİNDE KULLANILAN
ÖZNİTELİK VEKTÖRLERİNİN OPTİMİZASYONU**

**Elektronik Müh. Serkan TÜZÜN
Elektrik-Elektronik Mühendisliği Anabilim Dalı
Elektrik-Elektronik Mühendisliği Programı**

**Danışman
Prof.Dr. Aydın AKAN**

Ocak, 2010

İSTANBUL



**İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

YÜKSEK LİSANS TEZİ

**YÜZ TANIMA SİSTEMLERİNDE KULLANILAN
ÖZNİTELİK VEKTÖRLERİNİN OPTİMİZASYONU**

**Elektronik Müh. Serkan TÜZÜN
Elektrik-Elektronik Mühendisliği Anabilim Dalı
Elektrik-Elektronik Mühendisliği Programı**

**Danışman
Prof.Dr. Aydın AKAN**

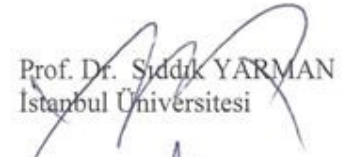
Ocak, 2010

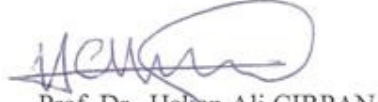
İSTANBUL

Bu çalışma 15/02/2010 tarihinde aşağıdaki jüri tarafından Elektrik Elektronik Mühendisliği Anabilim Dalı Elektrik Elektronik Mühendisliği programında Yüksek Lisans Tezi olarak kabul edilmiştir.

Tez Jürisi


Prof. Dr. Aydın AKAN (Danışman)
İstanbul Üniversitesi


Prof. Dr. Sıddık YARMAN
İstanbul Üniversitesi


Prof. Dr. Hakan Ali ÇIRPAN
İstanbul Üniversitesi


Prof. Dr. Osman Nuri UÇAN
İstanbul Üniversitesi


Prof. Dr. Ahmet SERTBAŞ
İstanbul Üniversitesi

ÖNSÖZ

Bu tezin hazırlanmasında değerli bilgi ve tecrübelerini esirgemeyen, lisans ve yüksek lisans öğrenimim sırasında ve tez çalışmam boyunca gösterdiği her türlü destek ve yardımlarından dolayı çok değerli hocam **Prof. Dr. Aydın AKAN**'a en içten dileklerle teşekkür ederim.

Ayrıca her zaman desteklerini yanımda hissettiğim arkadaşlarım Azime CAN, Esin KARAHAN ve Murat AKGÖL'e, değerli büyüklerim Salih SERDAR ve Vehbi BODUR'a ve özellikle kıymetli aileme teşekkürü borç bilirim.

Ocak, 2010

Serkan TÜZÜN

İÇİNDEKİLER

ÖNSÖZ	i
İÇİNDEKİLER	ii
ŞEKİL LİSTESİ.....	iv
TABLO LİSTESİ	v
SEMBOL LİSTESİ	vi
ÖZET	vii
SUMMARY	viii
1. GİRİŞ	1
1.1. BİYOMETRİK SİSTEMLERE GENEL BAKIŞ	2
1.2. BİR BİYOMETRİK SİSTEMİ OLARAK YÜZ TANIMA	2
2. GENEL KISIMLAR	7
2.1. İNSANLARDA YÜZ TANIMA	7
2.1.1. Yüz Tanıma Problemi	8
2.1.2. Yüz Tanıma, Nesne Tanımadan Farklı mı?	8
2.1.3. Nesne Tanıma Modelleri	11
2.2. GELİŞİM PSİKOLOJİSİ AÇISINDAN YÜZ TANIMA	13
2.3. BÜTÜNCÜL MODELLER VE ÖZNİTELİK TABANLI MODELLER	17
2.4. YÜZ, EN ETKİLEYİCİ BİYOMETRİK	19
2.4.1. Yüz İmgesi Veritabanları	20
2.4.2. Bilgisayarda Yüz Tanıma	23
2.4.3. Yüz Tanıma Modelleri	25
2.4.3.1. Bağımsız Bileşen Analizi.....	27

2.4.3.2. Temel Bileşen Analizi.....	28
2.4.3.3. Yapay Sinir Ağları.....	30
2.4.3.4. Dik Yüzler Yöntemi	34
3. MALZEME VE YÖNTEM.....	35
3.1. GABOR DALGACIKLAR	38
3.1.1. Dalgacık Dönüşümüne Kısa Bir Bakış.....	38
3.1.2. Dalgacık ve Fourier Dönüşümü	40
3.1.3. Gabor Dalgacıklar	44
3.2. PARÇACIK SÜRÜ OPTİMİZASYONU	46
3.2.1. Sosyal Davranışın Benzetimi.....	47
3.2.2. Parçacık Sürü Optimizasyonunun Uygulama Alanları	49
3.2.3. Parçacık Sürü Optimizasyonu	50
3.2.3.1. PSO Parametre Kontrolü.....	52
3.2.3.2. Geliştirilmiş PSO	53
3.3. KARAR VERME	53
3.3.1. Temel Bileşen Analizi	53
3.3.2. Öklid Mesafesi	55
3.4. ÖNERİLEN YÖNTEM.....	55
4. BULGULAR.....	57
5. TARTIŞMA VE SONUÇ	67
KAYNAKLAR.....	69
EKLER	77
ÖZGEÇMİŞ.....	93

ŞEKİL LİSTESİ

Şekil 1.1	: Değişik dönüşümler neticesinde imgeden farklı özneliklerin elde edilmesi... 3
Şekil 1.2	: Yüz tanıma için 22 geometrik noktanın belirlenmesi 4
Şekil 1.3	: Şablon eşlemeye dayalı Yüz Tanıma 5
Şekil 1.4	: Görünüş temelli metotlardan Yapay Sinir Ağı modeli 5
Şekil 2.1	: Mooney yüzleri 10
Şekil 2.2	: Üç haftalık bir bebeğin yetişkinin yüz hareketlerini taklit edebilmesi 16
Şekil 2.3	: Thatcher illüzyonu 17
Şekil 2.4	: Bilinen biyometriklerin yaygınlığı 20
Şekil 2.5	: Yüz Tanıma Sistemi Modeli..... 24
Şekil 2.6	: Özyüzler ile resmin yeniden oluşturulması..... 26
Şekil 2.7	: Elastik çizge modeli 26
Şekil 2.8	: Sinir hücresinden Yapay sinir ağına 30
Şekil 2.9	: Yapay Sinir Ağı modeli..... 32
Şekil 2.10	: RGB modelinin kartezyen koordinat sisteminde gösterimi 32
Şekil 3.1	: Birinci dereceden Daubechies 5 ölçek ve dalgacık fonksiyonu 39
Şekil 3.2	: KZFD'ün taban fonksiyonu ve zaman frekans düzlemi..... 41
Şekil 3.3	: Sabit Q'lu filtre bankasının zaman frekans çözünürlüğü 42
Şekil 3.4	: Dalgacık yönteminin temeli 44
Şekil 3.5	: Önerilen Yöntem 55
Şekil 4.1	: Eğitim kümesi yüz imgeleri - 1..... 57
Şekil 4.2	: Eğitim kümesi yüz imgeleri - 2..... 58
Şekil 4.3	: Gabor Dalgacıklar – 1 ($s = 1:5, \theta = \pi/8:\pi/8:\pi/2$)..... 59
Şekil 4.4	: Gabor Dalgacıklar – 2 ($s = 1:5, \theta = 5\pi/8:\pi/8:\pi$)..... 59
Şekil 4.5	: Bir kişiye ait resmin gabor filtrelere cevabı – 1 ($s = 1:5, \theta = \pi/8:\pi/8:\pi/2$) .. 60
Şekil 4.6	: Bir kişiye ait resmin gabor filtrelere cevabı – 2 ($s = 1:5, \theta = 5\pi/8:\pi/8:\pi$) ... 60
Şekil 4.7	: Bir kişiye ait 3 farklı pozun gabor filtreye cevapları ($s = 1, \theta = \pi$) 61
Şekil 4.8	: 16 kişiye ait aynı pozun gabor filtreye cevapları ($s = 1, \theta = \pi$) 61
Şekil 4.9	: Test kümesi yüz imgeleri – 1..... 62
Şekil 4.10	: Test kümesi yüz imgeleri – 2..... 62
Şekil 4.11	: Test kümesi kişi eşleme doğruluk değerleri..... 63
Şekil 4.12	: PSO ile oluşturulan Optimum Gabor Dalgacık 64
Şekil 4.13	: Eğitim kümesi imgelerinin Optimum Gabor Filtre'ye cevapları..... 65
Şekil 4.14	: Test kümesi imgelerinin Optimum Gabor Filtre'ye cevapları 65
Şekil 4.15	: Test kümesi kişi eşleme doğruluk değerleri..... 66
Şekil 4.16	: Eğitim kümesi - test kümesi imgeleri benzerlik resmi (TBA) 66

TABLO LİSTESİ

Tablo 2.1	: En önemli 2B yüz imgesi veritabanları.....	22
Tablo 2.2	: En önemli 3B yüz modeli veritabanları	23

SEMBOL LİSTESİ

A	: karıştırma matrisi
W	: ayırıştırma matrisi
\hat{X}	: fark matrisi
C	: kovaryans matrisi
ϕ_x	: özvektör
Λ_x	: özdeğer
ϵ_i	: öklit mesafesi
$F(\tau, \omega)$: KZFD fonksiyonu
$\psi(t)$: ana dalgacık fonksiyonu
$Wf(m,n)$: ayrık dalgacık dönüşümü
$\phi(x)$: ölçek fonksiyonu
$\varphi(x)$: dalgacık fonksiyonu
g^o	: gabor fonksiyonu tek kısmı
g^e	: gabor fonksiyonu çift kısmı
w_k	: dalgacık katsıysı
$\psi_n(x,y)$: gabor dalgacık
c_x, c_y	: konumdaki hareket parametresi
s_x, s_y	: ölçek parametresi
θ	: yönlendirme parametresi
x_i	: parçacık
v_i	: parçacığın hızı
$pbest$: parçacığın en iyi değeri
$gbest$: parçacıkların en iyi değeri
c_1, c_2	: öğrenme faktörleri

ÖZET

YÜZ TANIMA SİSTEMLERİNDE KULLANILAN ÖZİNİTELİK VEKTÖRLERİNİN OPTİMİZASYONU

Bugüne kadar gerçekleşen pek çok gelişmeye karşın yüz tanıma zorlu bir problem olmaya devam etmektedir. Yüz ifadeleri, poz ve aydınlık gibi parametrelerde meydana gelen küçük değişimler, yüz imgesinde bozulmalara neden olmaktadır. Buna rağmen, yüzdeki yerel niteliklerin bu tür bozulmalara karşı dayanıklı olduğu ve bir uzay-frekans analizi ile niteliklerin eldesinin mümkün olduğu düşünülmektedir. Bu amaç doğrultusunda, iyi biçimde belirlenmiş uzay-frekans yerelleştirmesi ile Dalgacık Analizi doğru bir seçim olarak karşımıza çıkmaktadır. Çeşitli dalgacık tabanları arasında Gabor Fonksiyonları, hem zamanda hem de frekansda en uygun çözünürlüğü sağlamaktadır. Örüntü tanıma için yerel niteliklerin elde edilmesinde, Gabor Dalgacıklar en uygun yaklaşım olarak görünmektedir.

Doğrusal olmayan sürekli fonksiyonların optimizasyonu için geliştirilmiş Parçacık Sürü Optimizasyonu (PSO), basitleştirilmiş sosyal bir modelin benzetimi esnasında bulunmuştur. Kökleri iki ana metodolojiye dayanmaktadır; genelde Yapay yaşam (Artificial life (A-life)), özelde Kuş sürüsü (Bird Flocking), Balık sürüsü (Fish Schooling) ve Sürü Teorisi (Swarm Theory). PSO, Evrimsel Hesaplama ile ilintilidir, bununla birlikte kökleri hem Genetik Algoritma hem de Evrimsel Programlamaya ulaşmaktadır.

PSO, basit bir fikir ve bir kaç satır bilgisayar kodu ile gerçekleştirmesi mümkün bir metoddan oluşmaktadır. Sadece temel matematik operatörlerine ihtiyaç duymakta, ayrıca bellek gereksinimi ve hız bakımından oldukça masrafsızdır.

Bu çalışmada, yüz tanımadaki kullanılan öz niteliklerin Gabor Dalgacık ile elde edilmesi üzerinde durulmuştur. Ayrıca parçacık sürü optimizasyonuna dayalı bir yüz tanıma öz nitelik optimizasyonu sunulmuş, farklı yüz kümeleri üzerinde testler yapılarak benzetim sonuçları yorumlanmıştır.

SUMMARY

OPTIMIZATION OF FEATURE VECTORS USED IN FACE RECOGNITION SYSTEMS

Despite remarkable progresses so far, the general task of face recognition remains a challenging problem, this is mainly due to the complex distortions that can be caused by variations in illumination, facial expressions and poses. It is widely believed that local features in face images are more robust against such distortions and a spatial–frequency analysis is often desirable to extract such features. With good characteristics of space–frequency localization, wavelet analysis is the right choice for this purpose. In particular, among various wavelet bases Gabor functions provide the optimized resolution in both the spatial and frequency domains. Gabor wavelets seem to be the optimal basis to extract local features for pattern recognition.

Particle swarm optimization has roots in two main component methodologies. Perhaps more obvious are its ties to artificial life (A-life) in general, and to bird flocking, fish schooling, and swarming theory in particular. It is also related, however, to evolutionary computation, and his ties to both genetic algorithms and evolutionary programming.

Particle swarm optimization comprises a very simple concept, and paradigm can be implemented in a few lines of computer code. It requires only primitive mathematical operators, and is computationally inexpensive in terms of both memory requirements and speed.

In this study, the extraction of features, used in face recognition, is examined. Gabor Wavelets are used to extract the features. Also a face recognition feature optimization technique, related to particle swarm optimization, is represented, and the results of the simulation are examined by testing different face sets.

1. GİRİŞ

Kişilerin fiziksel veya davranışsal özellikler kullanılarak tanınması/onaylanması olarak açıklanabilen biyometri bilimi, parmakizi, yüz, kulak, iris, retina, el geometrisi, ses gibi kişiyi fizyolojik veya davranışsal bir özelliğiyle temsil edebilecek kadar kişiye özgü, kişiden kişiye kolayca aktarılamayacak kadar güvenilir, hayatın başlangıcından sonuna kadar değişmeyen kalıcı özellikleri kapsar [1]. İlk zamanlarda yüksek güvenlik gereken ve bireyin kimliğinin doğru tespit edilmesinin çok önemli olduğu alanlarda kullanım alanı bulan biyometri, günümüzde sıkça karşımıza çıkan bir teknoloji haline gelmiştir [2]. Biyometri günümüzde turizm sektöründe, giriş-çıkış kapılarında kimlik tespiti ve benzer güvenlik fonksiyonlarında, bilgisayar ve bilgisayar ağlarında erişim düzeninin sağlanmasında ve kaynakların paylaşımında, ağ güvenliğinin sağlanmasında kullanılmaktadır. Ayrıca fiziksel giriş çıkış kontrol noktalarında, personel devam kontrol sistemlerinde, ticari işlemler ve benzer birçok alanda, banka güvenliğinde elektronik fon transferi veya ATM güvenliği gibi işlemlerde, çek ve kredi kartı işlemlerinde, gümrük ve göç işlemlerinde daha güvenli ve hızlı işlem için, ulusal kimliklendirme sistemlerinde daha güvenilir ve düzenli kimliklendirme için, seçmen ve sürücü kayıtlarında, internet işlemlerinde verimli bir şekilde uygulandığı ve insan hayatında geniş yelpazede çözümler sunan bir teknoloji olduğu görülmektedir [2,3]. Son yirmi yıl içerisinde biyometrik tanıma sistemleri konusu bir çok açıdan incelenmiş, konunun değişik yönlerine ve detaylarına yönelik çeşitli teknikler, verimli algoritmalar ve yöntemler kullanılarak konuyla ilgili birçok problem için çeşitli çözümler ortaya konmuştur [2-7]. İlk çalışmalar biyometrik sistemlerde doğruluğun ve güvenilirliğin artırılmasına, sistemin işleyişine ve iyileştirilmesine yönelik olmakla birlikte, son yıllarda araştırmacıların ilgisi biyometrik tekniklerin birleştirilerek kullanıldığı, doğruluk ve güvenliği bir üst seviyeye çıkaran çoklu biyometrik sistemler üzerine yoğunlaşmıştır [3,6-12].

1.1. BİYOMETRİK SİSTEMLERE GENEL BAKIŞ

Bir biyometrik sistem en genel anlamda bireyin biyometrik özelliğini alan, bu özellikten kişinin kimliklendirilmesinde kullanılacak olan özellik setini çıkararak ve kişiyi temsil eden bu anlamlı veri seti ile daha önceden aynı prensiplerle elde edilip veritabanına kaydedilmiş veri seti/setleri arasında karşılaştırma yapan bir tanıma/onaylama veya sınıflandırma sistemi olarak tarif edilebilmektedir [13]. Biyometrik sistemler çalışma şekli ve uygulama durumuna göre 4 gruba ayrılırlar. Bunlar kayıt modu (enrollment), onaylama modu (verification), tanıma modu (identification) ve izleme modu (screening) şeklinde sıralanmaktadır [1]. Temelde hepsinin çalışma prensibi aynı olmakla birlikte modlar uygulama şekli ve kullanım alanı konusunda bazı farklılıklara sahiptirler. Örneğin kayıt modu biyometrik verinin sisteme alınması, sınıflandırılması ve kaydedilmesi işlemlerini kapsamaktadır. Onaylama modu “kullanıcı adı + biyometrik veri” kombinasyonunu sisteme giriş olarak alıp birebir karşılaştırma (1-1) sonucu sistem cevabını oluşturmak şeklinde tanımlanmakta ve genelde personel takibi gibi giriş çıkış kontrol noktalarında tercih edilmektedir. Tanıma modu yalnızca biyometrik veriyi sisteme giriş olarak alıp veritabanındaki tüm kayıtlarla tek tek karşılaştırma (1-n) yaparak sistem cevabını oluşturmakta ve genelde suç ve suçlu takibi gibi işlerde kullanılmaktadır. İzleme modu ise herhangi bir kişinin arananlar listesinde olup olmadığının araştırılmasında görev yapmaktadır. Havaalanı güvenliği, herkesin kullandığı kamuya ait alanların güvenliği ve diğer takip ve gözlem gerektiren uygulamalarda tercih edilmektedir.

1.2. BİR BİYOMETRİK SİSTEMİ OLARAK YÜZ TANIMA

Hayvanlar ve insanlar evrim süreçleriyle iyileştirilmiş karmaşık görsel sistemleri sayesinde zorlu algılama problemlerini başarıyla çözerler. İnsan gözü öylesine karmaşık ve girift bir yapıya sahiptir ki, bu yapının tedrici olarak oluşmasının mümkün olmadığı hususu uzun zamandır Darwin’in evrim teorisine karşı yerini korumaktadır. Yine de mühendisler yarım yüzyıldır insanların algısal marifetlerini fizyolojik ve psikolojik bulguların da yardımıyla bilgisayarlarda modellemek için uğraşıyorlar.

İlk olarak insanlarda yüz tanımanın nasıl gerçekleştiği, ikinci olarak da bilgisayarlarda bu problemin nasıl çözülebileceği hususunu irdelemek faydalı olacaktır. Birinci soru

beynin nasıl çalıştığını araştıran bilişsel bilim açısından önemlidir. Ama bu soruya verilecek yanıt ikinci soru açısından da önem taşıyor; beyinde bulduğumuz süreçler bizi bilgisayar ortamında da etkili bir çözüme götürebilir. Bilgisayarda bu problemi çözmek ise hem daha güçlü bir insan-bilgisayar etkileşimi sağlamak için, hem de son zamanlarda çok önem verilen biyometrik güvenlik uygulamaları için gereklidir.

Bir bilgisayarda başarılı bir yüz tanıma sistemi yaratabilmek için, bir yapay öğrenme modeline ihtiyaç vardır. Yapay öğrenmeden anladığımız matematiksel ifadesi olan bir programın parametrelerinin belli bazı girdiler için (mesela yüz imgeleri) belli bazı çıktılar verecek şekilde (mesela yüzün bir erkeğe mi yoksa kadına mı ait olduğu) ayarlanmasıdır. Bu şekilde "öğrenen" bir sistem daha önceden görmediği yüzleri de doğru olarak sınıflandırabilir, yani genelleme yapabilir. Sistemin iyi genelleme yapabilmesi, girdiye ne tür dönüşümler (transformation) uyguladığına ve girdinin hangi özniteliklerini (feature) öğrendiğine bağlıdır (bkz. Şekil 1.1). Başarılı bilgisayar modelleri değişik özniteliklerin ve dönüşümlerin yüz tanıma problemine ne kadar katkı sağladığını sayısal olarak ortaya koyabilir ve bu şekilde bilişsel hipotezlerin desteklenmesinde veya çürütülmesinde rol oynayabilir.

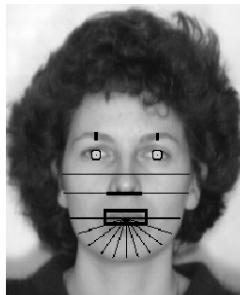


Şekil 1.1 : Değişik dönüşümler neticesinde imgeden farklı özniteliklerin elde edilmesi [14]

Bir otomatik yüz tanıma ve onaylama sistemi kişilerin durağan resimleri veya video görüntüleri üzerinden ilgili kişilerin veritabanına kayıtlı resimlerinin kullanılarak kimliklendirilmesi şeklinde tanımlanabilir. Yüz tanıma işlemi kişinin yüzüne ait özelliklerin değişmesine neden olan yüz ifadelerindeki değişiklikler, 3 boyutlu poz farklılıkları, makyaj, yüzün bir kısmını veya tamamını kapatan aksesuarlar, saç şekli ve benzeri birçok faktör olması nedeniyle oldukça karmaşık ve zor bir iştir [15]. Bir yüz tanıma sistemi karmaşık bir arka plandan yüzlerin ayırt edilmesi, yüze ait özellikler ve bu özellikler yardımıyla yüz yeri ve yönünün belirlenmesi ve tanıma/onaylama işleminin gerçekleştirilmesi olmak üzere temelde üç adımdan oluşmaktadır [15]. Yüz tanıma ile ilgili literatür incelendiğinde yüz tanıma ile ilgili yöntemlerin temelde 4 ana grupta toplandığı görülmektedir [6]. Bu yöntemler ana hatlarıyla aşağıda tanıtılmaktadır.

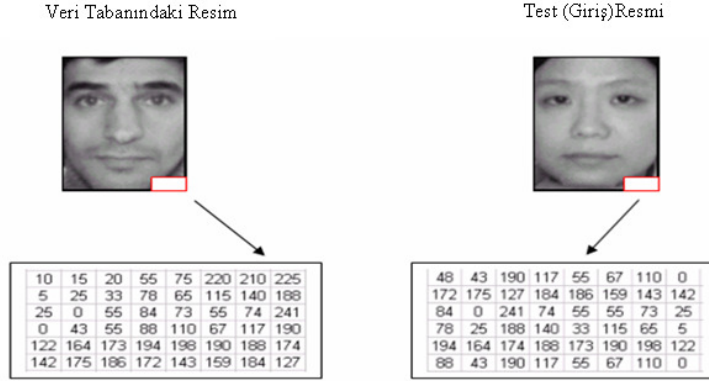
Bilgi temelli metotlar: Kural tabanlı metotlar olarak da bilinirler. Genellikle yüz yerinin tespiti ve yüze ait özellik setinin elde edilmesinde kullanılmak üzere tasarlanmış olan bu metotlar tipik bir yüzün insan beyninde nasıl şifrelendiği düşünülerek geliştirilmiştir.

Değişmeyen özelliklere dayalı yaklaşımlar: Bu algoritmalar yüzde var olan ve poz, bakış açısı, çeşitli aydınlatma koşullarına rağmen değişmeyen sabit kalan özellikleri bulma ve bunları kullanarak yüz yerinin tespiti ve yüz tanıma işleminin gerçekleştirilmesi için geliştirilen yöntemlerdir. Yüze ait özellikler, doku analizi, ten rengi veya birden fazla özellik kullanılarak işlem yapılan çalışmalar bu gruba girmektedir.



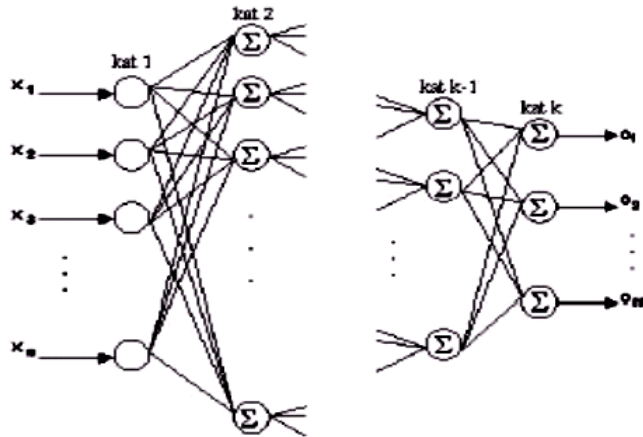
Şekil 1.2 : Yüz tanıma için 22 geometrik noktanın belirlenmesi [16]

Şablon eşlemeye dayalı metotlar: Bir yüze veya yüzün parçalarına ait çeşitli standart şablonlardan çok sayıda depolayıp karşılaştırma mantığına dayanan bir yöntemdir. Giriş resmiyle önceden tanımlanmış veya şekil değiştirebilen kayıtlı şablonlar arasındaki benzerlik oranına bakılarak yüz yeri tespit edilmeye ve yüz tanıma yapılmaya çalışılır.



Şekil 1.3 : Şablon eşlemeye dayalı Yüz Tanıma

Görünüş temelli metotlar: Bu yöntemlerde şablon karşılaştırmanın tersine, modeller veya şablonlar bir veri setinden öğrenilir. İlgili veri seti yüzün çeşitli şekillerinde, çeşitli şartlarda ve çeşitli pozisyonlarda alınan resimlerinden oluşan eğitim setidir. Bu öğrenilen modeller daha sonra yüz tanıma işlemi için kullanılmaktadır. Öz yüz yöntemi ve öz yüzün iyileştirilmiş versiyonları (doğrusal diskriminant analizi, ayırt edici ortak vektör yöntemi vs), dağılım tabanlı yöntemler, yapay sinir ağları, karar destek mekanizmaları, gizli markov modelleri gibi çeşitli yöntemler kullanılarak işlemler gerçekleştirilmektedir.



Şekil 1.4 : Görünüş temelli metotlardan Yapay Sinir Ağı modeli

Tezin 2. bölümünde 2.1’de insanlarda yüz tanıma üzerine geliştirilmiş teoriler ele alınmıştır. Bu teorilerin ortak noktası, beynin bir bilgisayar gibi veriyi aşama aşama işleyen bir organ olduğunu varsaymalarıdır. 2.2 bölümünde gelişim psikolojisi ve nöropsikolojinin bulguları, bu teorilerin ışığında değerlendirilmiştir. Özellikle cevap aranan sorular yüz tanıma yetisinin ne kadarının doğuştan geldiği ve ne kadarının öğrenildiği, beyinde yüz tanımaya özgü bir bölge olup olmadığıdır. 2.3 bölümünde yüz tanımanın bütüncül (holistic) olup olmadığı tartışılmıştır; bu ayrım özellikle bilgisayar modelleri açısından önem taşımaktadır. 2.4 bölümünde günümüzde yüz tanıma deyince akla gelen belli başlı bilgisayar modelleri sunulmuş ve bilişsel bilimden esinlenmiş modellerin hangi süreçleri ne kadar başarıyla kullandığına bakılmıştır.

Tezin 3. bölümünde çalışmamızda kullanılan temel yöntemler; Gabor dalgacıkları ve parçacık sürü optimizasyonu sunulmuştur.

Tezin 4. bölümünde parçacık sürü optimizasyonuna dayanarak ortaya konan bir yüz tanıma öznitelik optimizasyonu sunulmuştur. Farklı yüz kümeleri üzerinde testler yapılarak benzetim sonuçları verilmiştir. 5. bölümde elde edilen sonuçlar tartışılmış ve vargılar sunulmuştur.

2. GENEL KISIMLAR

Son zamanlarda gerçekleşen terör olayları neticesinde bugünün güvenlik mekanizmalarının tehlikeli kusurları ve zayıf yönleri idrak edilmiş, bu nedenle devlet kurumları güvenlik sistemleri geliştirmek için kaynakların önemli miktarını bu alanda yatırım yapmaya ayırmıştır. Rozet veya parola kimlik doğrulama tabanlı işlemleri kırmak çok kolay. Biyometrik geçerli bir alternatif olarak karşımıza çıkmakta lakin yetersizlikleri de mevcuttur. Örneğin, iris tarama çok güvenilir ancak çok müdahaleci; parmak izi sosyal açıdan kabul edilir, ancak rıza göstermeyen insanlar için geçerli bir uygulama olduğu söylenemez. Diğer taraftan, yüz tanıma kontrollü koşullarda altında çalıştırılrsa da, sosyal açıdan kabul edilebilir ve güvenilir olan arasında iyi bir uzlaşma temsil eder. Son on yılda, doğrusal/doğrusal olmayan yöntemler, yapay sinir ağları, dalgacıklar gibi pek çok algoritma önerilmiştir.

Bu bölümde bir dizi alt başlıklar halinde yüz tanımayı irdelemeye gayret edeceğiz. Evvelen, insanlarda yüz tanıma üzerine geliştirilmiş teoriler ele alınacaktır. Bu teorilerin ortak noktası, beynin bir bilgisayar gibi veriyi aşama aşama işleyen bir organ olduğunu varsaymalarıdır. Saniyen, gelişim psikolojisi ve nöropsikolojinin bulguları bu teorilerin ışığında değerlendirilecektir. Özellikle cevap aranan sorular yüz tanıma yetisinin ne kadarının doğuştan geldiği ve ne kadarının öğrenildiği, beyinde yüz tanımaya özgü bir bölge olup olmadığıdır. Akabinde, yüz tanımanın bütüncül (holistic) olup olmadığı tartışılacaktır; bu ayrım özellikle bilgisayar modelleri açısından önem taşımaktadır. Son olarak, günümüzde yüz tanıma deyince akla gelen belli başlı bilgisayar modelleri görülecek ve bilişsel bilimden esinlenmiş modellerin hangi süreçleri ne kadar başarıyla kullandığına bakılacaktır.

2.1. İNSANLARDA YÜZ TANIMA

Yüz tanıma, bilgisayarlıları bugüne kadar en çok uğraştırmış örüntü tanıma problemidir, ama bu problemin ne kadar zor olduğunu ilk psikologlar fark etmiş ve

insan beyninde bu problemin nasıl çözüldüğü pek çok araştırmaya konu olmuştur. Bir iki kere gördüğümüz bir insan yüzünü aradan uzun bir zaman geçtikten sonra, farklı bir ışık altında, değişik bir pozda görsek bile tekrar tanıyabiliyoruz. Henüz bilgisayar sistemleri bu başarıyı gösteremiyorlar.

2.1.1. Yüz Tanıma Problemi

Aynı insan yüzünün farklı pozlardan ve değişik ışıklandırma koşullarında çekilmiş, farklı yüz ifadelerine sahip fotoğraflarını yan yana koyalım. İnsanlar bu yüzlerin aynı kişiye ait olduğunu zorluk çekmeden söyleyebiliyorlar. Oysa bir bilgisayar için bu çok büyük bir problem, çünkü imgeleri üst üste koyduğunuzda hiçbir piksel diğerini tutmuyor. Hatta iki kişinin aynı ışık altında, aynı açıdan çekilmiş fotoğrafları birbirlerine tek bir kişinin farklı ışıklandırmayla başka açılardan çekilmiş fotoğraflarından daha çok benziyorlar. Bu yüzden öznitelikleri istatistiksel olarak modelleyen bir öğrenme modeli doğrudan yüz imgeleri üzerinde çalışırsa başarısız oluyor.

Yapay öğrenme modelleri belli şekillerde genelleme yapmaya eğilimlidir. Bu eğilim, yahut yanlılık (bias) modelin nasıl kurulduğuna göre farklılık gösterir. Beynin doğuştan gelen yüz tanıma eğilimi öğrenmeyi kolaylaştıran bir yanlılıktır. Eğer beyni öğrenen bir makina olarak düşünürsek ve amacımız yüz tanıma problemini beyin kadar iyi çözmek ise, girdinin doğasını ve beynin bu probleme ne kadar ve ne şekilde yanlı yaklaştığını öğrenmemiz gerekiyor [17]. Acaba genel bir öğrenme sistemi bu iş için yeterli midir? Yoksa bu problemi ancak doğuştan sahip olduğumuz ve yüz tanıma için özelleşmiş bir altsistem sayesinde mi çözebiliyoruz? Yüz imgesi beyinde hangi aşamalardan geçerek işleniyor? Bu probleme ilişkin ne kadar ön bilgiyi öğrenme modelimize koymamız gerekiyor? Bu soruların her biri cevap bekleyen araştırma konularıdır.

2.1.2. Yüz Tanıma, Nesne Tanımadan Farklı mı?

Psikologlar insanlarda yüz tanımanın diğer nesnelere tanınmasından daha farklı bir biçimde gerçekleştiğini düşünmek için pek çok sebep olduğunu söylüyorlar. Örneğin beyindeki bazı nöronlar sadece görüş alanında bir yüz belirdiğinde harekete geçiyorlar. Birçok deney beyinde sadece yüz tanıma sırasında aktive olan bölgeler olduğunu ortaya koyuyor. Bazı beyin hasarlarından sonra hasta bildiği yüzleri tanıyamaz hale geliyor,

oyşa diđer nesneleri tanımakta güçlük çekmiyor (prosopagnosia). Bunu tam tersi bir durum da var, nesneleri tanıyamaz hale gelen hastalar bildikleri yüzleri rahatça tanıyorlar (agnosia). Fonksiyonlardaki bu ikili ayrılma (double dissociation) yüz tanımanın özel bir sistem tarafından gerçekleştirildiđinin en büyük göstergesi olarak görülüyor.

Benzer bir bulgu beyin hasarı sonucunda sadece ve sadece yeni yüzleri öğrenme becerisinin kaybedilebileceđini gösteriyor (prosopamnesia). Capgras sendromunda ise hastalar önceden bildikleri yüzleri tanıyorlar, ama tanıdık birini gördüklerinde duymaları gereken aşinalık hissi kayboluyor. Bunun sonucunda hasta, yakınının aslında gerçekten tanıdđı insan olmadığını, birilerinin onu taklit ettiđini iddia ediyor. Bu hastalarda yapılan arařtırmalar sonucunda yüz tanımanın beyinde iki farklı sürecin paralel çalışmasıyla gerçekleştiđi hipotezi ortaya atılmıştır. Bu süreçlerden birinin duyumsal (affective), diđerinin bilişsel (cognitive) olduđu öngörülüyor.

Beyin üzerinde yapılan arařtırmalar özellikle fusiform girus (fusiform face area) denilen bir bölgede yüz tanıma sırasında belirgin bir etkinlik olduđunu gösteriyor. Özellikle Mooney imgeleri ile yapılan deneylerin ilginç sonuçları var. Mooney imgeleri dengelenmiş siyah ve beyaz bölgelerden oluşan yüz resimleridir (bkz. Şekil 2.1). Bu imgelere bakanlar öncelikle bir yüz algılamayabilirler. Ancak yüz bir kere algılandıktan sonra, bütün yüz hatları yerine oturur ve tekrar bakıldığında yüz bu sefer kolayca bulunur. Mooney yüzlerine bakan kişilerde eđer yüz algılanmışsa fusiform girus bölgesinde aktivite gözleniyor. Bu önemli bir bulgu, çünkü imge deđişmiyor, sadece algı deđişiyor. Yani beyindeki aktivite girdinin doğasından deđil, algının doğasından kaynaklanıyor.

Beyin arařtırmalarını yüz tanımanın ayrıcalıklı bir konumu olduđu iddiasında destekleyen davranışbilimsel bulgular da var. Örneđin bir nesneyi ters çevirdiđinizde nesnenin tanınmasını zorlařtırmış olursunuz. Ters çevrilmiş yüzlerde bu etki çok daha belirgin bir biçimde ortaya çıkar. Tanaka ve Farah bu etkiyi şöyle açıklıyorlar: Nesneleri tanımak için onların görsel özniteliklerinden yararlanırız. Ama yüz tanımada bu özniteliklerin ne şekilde dađıldığı, hangisinin altta, hangisinin üstte olduđu, yani

özniteliklerin konfigürasyonu, özniteliklerin kendisinden daha önemli bir bir bilgidir [19]. Bu yüzden de çok basit iki nokta ve bir çizgiyi bile yüz olarak algılayabiliriz.



Şekil 2.1 : Mooney yüzleri [18]

Biyolojik öğrenmenin önemli süreçlerinden biri uzmanlaşmadır. Beyinde öğrenme aşama aşama gerçekleşir. İleri aşamalarda beyin, problemin doğasına ait bazı kısayollar öğrenir, yeni ve daha etkin gösterimler kullanmaya başlar ve daha zengin kavramsal ayrımlar yapar. Bu şekilde problemi daha hızlı ve daha doğru şekilde çözmeye başlar, ama bazı ayrımları yapma gücünden de feragat eder. Görsel ilüzyonların çoğu beynin bu özelliğinden yola çıkar. Konfigürasyonun önem kazanması da uzmanlaşmanın bilinen etkilerinden biridir; satranç ve go ustalarının oyun tahtasına baktıklarında tek tek taşları değil taşların durumlarına ait, daha yüksek seviye konfigürasyonlar gördükleri -örneğin hücum cepheleri, zayıf bölgeler, vs.- biliniyor. Ayrıca satranç ustaları gerçek satranç oyunlarına ait taş dizilimlerini kolayca ezberleyebilirken, rasgele dizilimlerde acemilerden bir farklılık göstermiyorlar [17].

Yüz tanımanın da aslında özel bir sistem gerektirmediğini, bütün saydığımız bulguların aslında insanın yüz tanımada uzmanlaşmasının sonucu olduğunu söyleyen çalışmalar

var. Gauthier ve Tarr [20] bir deney yaparak bu hipotezi deniyorlar. Denekleri "Greeble" adını verdikleri yapay yaratıkların türlü cinslerini ayırtetme konusunda eğitiyorlar. Denekler Greeble uzmanı olduktan sonra bir Greeble gördüklerinde, önceden beynin yüz tanıma bölgesi olarak düşünülen fusiform girus bölgesinde (fusiform face area) yoğun faaliyet gözleniyor. Dahası Greeble'larda konfigürasyon değişiklikleri yapıldığında uzmanlar bundan daha çok etkileniyorlar. Bu da yüz tanıma ve diğer uzmanlıklarda görülen ortak bir özellik.

Köpekleri hiç tanımayan biri, bir labrador gördüğünde onu bir "köpek" olarak algılar. Köpeklere aşina biri ise "Labrador" olarak algılayacaktır. İnsanlar da bir yüz gördüklerinde, onu herhangi bir yüz olarak algılamazlar, belli bir yüz olarak, "A'nın yüzü" veya "B'nin yüzü" olarak algırlarlar. Bu da bir uzmanlık göstergesidir.

2.1.3. Nesne Tanıma Modelleri

İnsan beyninin nesnelere nasıl tanıdığı uzun zamandır merak edilen bir konudur. Bu teorileri kısaca özetleyerek yüz tanıma açısından ne kadar geçerli olduklarına bakalım.

Seksenlerde önem kazanan yapısal tanımlama (structural description) teorisine göre beyin cisimleri parça ve bütün ilişkisi içinde algılar. Bu yaklaşımda bütün nesnelere bazı temel algı ünitelerinin birleşmesiyle ifade edilirler. Irving Biederman'ın "geon" adını verdiği bir dizi üç boyutlu şekil, David Marr'ın "codon" dediği basit öznitelikler bu temel algı ünitelerine örnek olarak gösterilebilir. Zamanla bu teori işlevini kaybetmiştir [17].

Bu analitik yaklaşıma karşı holistik, veya bütünsel diyebileceğimiz bir yaklaşım kaynağını daha eskilerden, Gestalt psikolojisinden alır. Bu yaklaşıma göre nesnelere parçalarına ayrılmaz, bir bütün olarak algılanırlar. Böyle bir algının mümkün olabilmesi için beyin gibi paralel çalışabilen bir sistem gereklidir.

Bir diğer teoriye göre beyin, nesnelere değişik açılardan, değişik büyüklüklerde ve farklı ışıklandırma şartları altında görür, bu görüntüleri fotoğraflarını çeker gibi ezberler ve sonradan kullanmak üzere saklar. Bu yaklaşıma göre nesne tanıma bilgisayarla örüntü

tanımada çok kullanılan “en yakın komşu” metoduna benzer bir şekilde yapılır. Bu yaklaşıma şablon modeli diyebiliriz.

Şablon modeliyle tutarlı bir diğer önemli hipotez de nesne tanımanın duruş-temelli (view-based) olduğu hipotezidir. Bir üç boyutlu nesneye farklı açılardan baktığımızda farklı görüntülerle karşılaşırız. Bu görüntülerin her birine “duruş” diyoruz. Nesnelerin günlük hayatta karşımıza hangi durumlarda çıktığına bağlı olarak bazı duruşlar o nesne için daha belirleyicidir. Mesela çoğu insan “bir at düşünün” denildiğinde atı 3/4-duruşta, yani sağ üstten sol alta doğru düşünür. Bu tip doğal duruşlar (canonic view) az görülen duruşlardan daha çabuk tanınır ve daha çok ayırdedici bilgi içerir. Yapılan deneylerde nesnelerin özgün duruşlarını tanımaya çalışan deneklerin de başarılı olduğu, ama tanımanın daha yavaş gerçekleştiği gözlenmiştir. Yani beyin bu tür bir genellemeyi yapmak için bir dizi ara işleme gereksinim duyar.

Yüz tanıma açısından bakacak olursak, şablon teorisi her tanıdığımız yüz için beynimizde çeşit çeşit şablonlar olduğunu söyler. Yeni bir yüz gördüğümüzde, varolan şablonlardan hangisine daha yakınsa o kişiye ait olduğu sonucuna varırız. Fakat olası şablonların sayısı çok yüksek olduğu için beynin bu gösterimi tercih ettiğini söylemek çok güç.

Pek çok deney hafif yan dönmüş yüzlerin tam karşıdan bakılan yüzlerden daha rahat tanındığını gösteriyor. Bu da yüz tanımanın duruş-temelli olduğu savını destekliyor. İki bilinen duruş arasındaki duruşların tanınabilmesi için beynin ne yaptığı tam olarak bilinmiyor, ama bu iş için bazı süreçlerin olduğu da şüphesiz.

Yapısal tanımlama teorisi her ne kadar popülerliğini yitirmiş olsa da, yapısal ilişkilerin beyinde ifade edildiği söylenebilir. Çok basit şematik yüzleri bile yüz olarak tanıyabildiğimize göre beyinde basit şablonlardan daha fazlası olmalı. Yüz tanımada duruşların önemli olması yüz tanımanın tamamen şablonlara dayandığı anlamına da gelmiyor. Şu anki veriler yetişkinlerde yüz tanımanın ağırlıklı olarak bütünsel işlemeye dayandığını gösteriyor.

2.2. GELİŞİM PSİKOLOJİSİ AÇISINDAN YÜZ TANIMA

Dilbilim camiasında dilin öğrenilmesine dair Chomsky ve karşıtları arasında yıllardır süren bir tartışma var. Tartışmanın özünde dil yetisinin ne kadarının doğuştan var olduğu yatıyor. Yüz tanıma konusunda da böyle bir tartışma var; argümanlar neredeyse aynı, ama olay henüz topyekün bir savaşa dönüşmedi. Bir tarafta “Bu kadarcık yüz görerek yüzleri böylesine iyi tanımak mümkün değildir, yüz tanıma doğuştan var olan, kendine has bir sistem tarafından gerçekleştirilir” diyen bir grup var. Bu gruba göre yüz tanıma Jerry Fodor'un [21] zihnin modülleri olarak adlandırdığı izole fonksiyonlardan biri. Diğer tarafta da “Hayır, beyin düşündüğünüzden daha güçlü bir öğrenme organıdır, azıcık evrimsel yardımla genel görme sistemi yüz tanımayı da başarır” diyen bir grup var. Bu seçenekleri değerlendirebilmek için soruna gelişim psikolojisi açısından bakmak gerekli.

Bilişsel yetiler (ya kısmen veya tamamen) doğuştan gelir, ya da bir öğrenme süreci ile geliştirilirler. Öğrenme çeşitli aşamalar ve gösterimler gerektirebilir. Bazı yetilerin öğrenilmesi diğerlerine bağlıdır. Gelişim psikolojisi açısından sorduğumuz sorular yüz tanımanın ne kadarının doğuştan geldiği, ne kadarının tecrübeyle öğrenildiği, bu öğrenmenin nasıl bir yol izlediği, dikkat, duygu ve iletişim gibi diğer fonksiyonlara ne kadar bağlı olduğu, ve hangi gösterimlerin kullanıldığıdır. Gelişim perspektifinden bakılınca yüz tanıma aşama aşama öğrenilen, her aşamada bir önceki aşamada kullanılan gösterimlere yeni fonksiyonların ve gösterimlerin eklendiği karmaşık bir sistem gerektiren bir problemdir [22].

Burada bir parantez açıp, beynin işleyişine dair bir noktaya dikkat çekelim. Basit bir öğrenme sistemi karmaşık bir problemi çözemez. Fakat karmaşık (çok parametresi olan) bir öğrenme sistemi için de öğrenme kolay bir iş değildir. Karmaşık problemleri çözmek için beynin stratejisi öncelikle daha basit bir problemi çözmek, ondan sonra da bu çözümü daha karışık olan problemde kullanmaktır. Problemi basite indirgemek için iki temel mekanizma vardır. Birinci mekanizma kısa süreli hafızadır. Tıpkı bir bidona su doldururken kullanılan huni gibi girdinin az ve düzenli olmasını sağlar. İkinci mekanizma da seçici dikkattir. Girdinin işlenebilecek kısmını öğrenme sistemine geçirir, karmaşık kısımları atar. Hafızanın gelişmesi ve seçici dikkatin seçtiği

özniteliklerin yavaş yavaş değişmesiyle karmaşık algılama sistemlerinin aşamalı eğitimi gerçekleşir.

Seçilen özniteliklerdeki değişime bir örnek verelim. Yeni doğmuş bir çocuk annesinin yüzünü çok erken, henüz birkaç günlükken tanımaya başlar. Ama annesi saçını farklı bir renge boyarsa tanıyamaz. Yani bebek saç rengi ve şeklini yüz tanımada kullanır. Oysa saç sürekli değişen ve -yetişkinler için- güvenilir bir özniteliktir, o yüzden büyüdükçe saçın tanımadaki önemi azalır. Yapılan araştırmalar iki-üç yaşındaki çocukların yüzlerde tek tek özniteliklere baktığını, bunlara sırayla dikkat ettiklerini saptamıştır. Çocuk beş yaşına geldiğinde bazı özniteliklerin öne çıktığı ve tutarlı bir biçimde tanımada kullanıldığı görülür. Bu yaştan sonra bütünsel algı gittikçe gelişir, analitik algıda gerileme olur. Yetişkinlerde bütünsel algı tamamen ön plandadır, sadece bu bütünsellik bozulduğunda yetişkin özniteliklere dikkat ederek yargıya varmaya çalışır. Bu şekilde yüz tanıma sistemi önce basit, yavaş ve özniteliklere dayalıyken, gittikçe daha hızlı, karmaşık ve bütünsel bir sistem haline gelir. Bütünselliğin bozulduğu durumlarda (mesela ağız, burun, ve gözlerin yerleri değiştirilince) bile yüz tanımanın mümkün olması, ilk gelişen öznitelik tabanlı altsistemlerin kaybolmadığını, karmaşık sistemin birer parçası olarak hayatlarına devam ettiklerini gösteriyor.

Beynin aşamalı öğrenme sistemi yapay sinir ağlarında da denenmiş, yüz tanıma ve dilin modellenmesinde başarıyla kullanılmıştır [23]. Bu yaklaşımda sistemin önce küçük bir hafıza birimi kullanması sağlanır. Sistem bu şekilde sadece basit bağlantıları öğrenebilir. Daha sonra hafıza kapasitesi yavaş yavaş artırılır. Sistem önceden öğrendiği bağlantıları unutmaz, bunlara ilaveten daha karışık bağlantılar da öğrenir. Aynı sistemi en baştan, bütün hafıza kapasitesini serbest bırakarak eğittiğimizde sistemin başarılı olmadığını görürüz.

Bilgisayar modellerine genel olarak baktığımızda, gösterimlerin bu şekilde değiştiği aşamalı eğitim süreçlerinin nadiren kullanıldığını görürüz. Bunun yerine girdinin çeşitli önışleme aşamalarından geçirildiği ve problemi zorlaştıran bazı özelliklerin ortadan kaldırılmasının amaçlandığı sistemler tercih edilir. Örneğin yüz tanımada ışıktandırmanın etkisini ortadan kaldırmak için histogram eşitlemesi kullanılır. Bu yöntem aslında genel ışıktandırma probleminin sadece bazı varsayımlar altında, bir

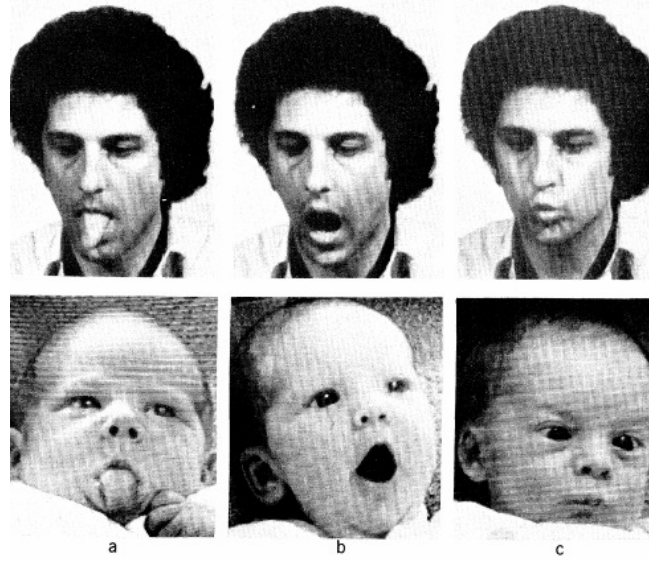
dereceye kadar çözümlenmesini sağlar; zira çoğu bilgisayar modeli, sınırları dikkatle çizilmiş bir alanda yüksek başarıyla çalışmak üzere tasarlanır.

Gösterimdeki aşamalı değişimin uzmanlaşmanın sonucu olduğunu belirtmiştik. Bu değişim rasgele değildir, yeni gösterimler uzmanın işini kolaylaştıracak niteliktedir. Yani gösterimin ne olacağını, problemin doğası belirler. Her insan bir yüz tanıma uzmanı olduğuna göre, yüzlerin insan beyninde nasıl gösterildiği yüz tanımanın ne için yapıldığına bağlıdır diyebiliriz. İnsanlar yüzlere göz kontağı kurma, iletişim sağlama, duygu tanıma, konuşma takibi için dudak okuma gibi amaçlarla bakarlar. O halde beyin uzmanlaşırken bu işleri kolaylaştıracak gösterimleri tercih edecektir. Seçici dikkat çalışmaları insanların yeni yüzlerde gerçekten de göz ve dudaklara özellikle baktığını gösteriyor.

Yeni doğmuş bebekler yüzlere ilgi duyarlar. Bu tercih genetik olarak kodlanmıştır yani yaratılıştandır. Bu durumun doğuştan geldiğini belirlemek için yapılan bir deneyde bebeklere üç değişik resim gösteriliyor. Birinci resimde normal bir insan yüzü var. İkinci resimdeki yüzün ağız, burun ve gözlerinin yerleri değiştirilmiş. Üçüncü resimdeyse sadece dış hatları belirgin olan içi boş bir yüz imgesi var. Bebekler doğduktan on dakika sonra birinci tip resimleri diğerlerine tercih ediyorlar.

Aslında birçok hayvan kendi cinsine karşı benzer tercihlerle doğar. Yavruyu annesine bağlayacak doğuştan gelen bir mekanizma yavrunun hayatta kalma şansını artıracığı için evrimsel değeri çok yüksektir, o yüzden doğal seçim böyle bir mekanizmayı destekleyecektir. Yüzlere duyulan erken ilgi, bebeğin diğer bireylerle iletişiminin ve dilin gelişiminin de başlangıç noktası olarak görülebilir (bkz. Şekil 2.2). Yüz tanımanın engellenmesi halinde, örneğin otistik bireylerde yüzlere dikkat edememe ve göz teması kuramama sonucu iletişim son derece düşük bir seviyede kalır, dil yetisi gelişemez.

Beynin yüzlere erken ilgi duyma özelliğine bakarak yüz tanıma doğuştan gelmektedir (innate) diyebilir miyiz?



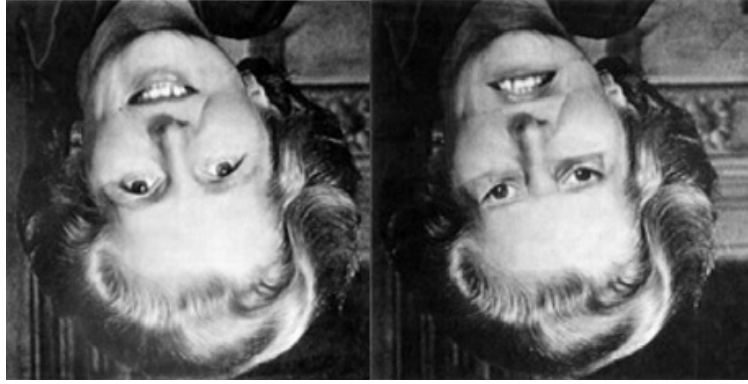
Şekil 2.2 : Üç haftalık bir bebeğin yetişkinin yüz hareketlerini taklit edebilmesi.

Bu beceriyle gösterdiği algı ve kontrol, bebeğin iletişim kurmasına ve öğrenmesine temel oluşturacak, doğuştan gelen bir sisteme işaret ediyor.

Bu soruya verilen en güçlü yanıt Jeff Elman, Anette Karmiloff-Smith ve UCSD'deki araştırmacılardan geliyor: Hayır, yüz tanıma konusunda beynin genetik temelli basit bir eğilimi olsa da, asıl öğrenme doğumdan sonra, daha genel bir sistem tarafından gerçekleştirilir [22, 23]. Çocuk uzun süre çeşitli açılardan insan yüzlerine bakar ve genel amaçlı sinir ağlarını bu girdinin istatistiksel özelliklerini öğrenmek için kullanır. Öncelikle öznitelikler öğrenilir, daha sonra çocuk yüz tanımada uzmanlaşmaya başlar. Çocuğun görme sistemi doğumu takiben bir süre daha yetişkinlerdeki hassasiyete ulaşamadığı için, öncelikle (saçlar, ten rengi gibi) çok belirgin öznitelikler öğrenilir. Uzmanlaşmayla birlikte bütünsel algı gelişir. Uzmanlaşma genel yeteneklerin kısmi kaybını da beraberinde getirir. Altı aylık çocuklar maymun yüzlerini birbirinden ayırdedebilirken, dokuz aylık çocuklar bunu başaramıyorlar, çünkü insana özgü özniteliklerdeki ayrımlara dikkat etmeyi, diğer ayrımları dikkate almamayı öğrenmişler. Yetişkinlere baktığımızda, başka bir ırkın yüzlerini ayırmakta zorlandıklarını görüyoruz (other race effect). Bu tip bir uzmanlaşma başka alanlarda da görülür, mesela Japonlar "l" ve "r" seslerini ayırdetme yetisini zamanla kaybederler, çünkü Japonca'da fonetik olarak bu sesler bir ayrıma karşılık gelmez. Halbuki bütün sağlıklı bebekler bu ayrımı yapabilir halde doğar.

2.3. BÜTÜNCÜL MODELLER VE ÖZNETELİK TABANLI MODELLER

Yüz imgelerini ters çevirmenin yüz tanımayı diğer nesnelere çok daha fazla etkilediğini söylemiştik. Bu etkinin sebebinin özniteliklerin ters çevirmeden fazla etkilenmemesi, buna karşın bütünsel bilginin bozulması olduğunu düşünüyoruz. Bu düşünceyi destekleyen başka bulgular da var. Örneğin yüze ait öznitelikler başka özniteliklerin yanında olduğu zaman tek başına olduklarından daha kolay tanınıyor. Mesela A'nın burnunu tek başına görünce tanıyamayan denekler, B'nin ağzı ve gözlerinin yanında tanıyabiliyor. Yani oluşan yüz konfigürasyonu tek tek özniteliklerin tanınmasını da kolaylaştırıyor. Buna yüzün üstünlüğü etkisi (face superiority) deniyor. Başka bir çalışma ise tek tek özniteliklerin ters çevrilmesinin yüz tanımayı fazla etkilemediğini gösteriyor.



Şekil 2.3 : Thatcher illüzyonu [24]

Bu çalışmalara dayanarak bütüncül (holistic) ve öznitelik-tabanlı (feature-based) yüz tanıma modelleri arasında bir ayrım yapıyoruz. Bu ayrımın bir ucunda yüz tanımanın tamamen bütüncül olduğunu, yüzlerin tek bir parça olarak algılandığını söyleyen araştırmacılar var [19]. Yüz tanımanın beynin daha çok sağ tarafında gerçekleşmesi, yüz tanımanın bütünsel olduğu varsayımıyla uyumlu görülüyor, zira beynin sağ tarafı daha çok bağlantıları kodluyor. Buna karşılık bir grup araştırmacı bu bulguların başka şekilde açıklanabileceğini savunuyor.

1980'lerde Thompson Margaret Thatcher'ın yüzündeki bazı öznitelikleri ters çevirerek ilginç bir deney yapmıştır [24]. Bu şekilde işlenen yüzler (Thatcher yüzleri) normalde garip görünüyorlar, ama ters çevrildiklerinde insanlar bunu fark etmiyor (bkz. Şekil

2.3). Thatcher illüzyonu bize özniteliklerin ayrı ayrı işlenip sonradan birleştirildiğini düşündürüyor. Ters çevirme sırasında bütünsel bilgi bozulduğu için tek tek özniteliklerden gelen bilgi kullanılıyor. Bu bilgi de ters çevirmeden nispeten az etkilendiği için yüzdeki gariplik fark edilmiyor.

Bu noktada durup insan gözünün çalışma şeklini hatırlamakta fayda var. İnsanda görme bir anda gerçekleşmez. Göz bütün algı alanını bir anda bütün detayıyla yakalayamaz, çünkü gözün detaylı bilgi alabilen bölümü fovea dediğimiz, göz sinirlerinin çok büyük bir bölümünün toplandığı küçük bir pencereden ibarettir. Bir yere baktığımızda foveadan detaylı bilgi gelir, onun dışındaki bölgelerden ise bulanık diyebileceğimiz çözünürlükte bilgi alırız. Bu yüzden gözün fovea bölgesi seçici dikkat mekanizması yardımıyla algı alanında gezinir, çeşitli yerlerden aldığı detaylı bilgi birleştirilir. Beyin gördüğü kısımları hemen işlemeye başlar ve bu işlem foveanın bir sonraki adımda nereye yöneleceğini de belirler.

Araştırmacılar ters çevrilmiş Thatcher yüzlerinin ters çevrilmiş yüzlerden daha hızlı tanıdığını bulmuşlar. Bunun sebebi insanlarda göz ve ağız bölgelerinin iletişimdeki önemleri sebebiyle seçici dikkat mekanizmasının ilk hedefleri arasında yer alması ve ters Thatcher yüzlerinde bu bölgeler düz oldukları için buradan daha sağlıklı bilgi gelmesidir. Bu bulgulara bakarak öznitelik-tabanlı tanımanın bütünsel tanımadan daha hızlı gerçekleştiği sonucuna varabiliriz. Eğer foveanın yönlendiği ilk birkaç bölgede tanımaya yetecek bilgi varsa, o zaman tanıma çok hızlı bir biçimde gerçekleşir. Yüz üstünlüğü etkisi de bu çerçeveden kolayca açıklanır; bütünsel bilgi foveanın sağlıklı bir şekilde göz ve ağız bölgelerine yönlendirilmesini sağlar.

Karikatürler üzerine yapılan çalışmalar da yüz tanıma konusunda ilginç açılımlar getiriyorlar. Karikatür sanatçıları bir yüzü karikatürize etmek için yüzün ortalamadan farklı özelliklerini abartır, önemsiz detaylarıysa ortadan kaldırır. Örneğin burun normalden biraz büyükse, karikatürde dev gibi çizilir. Bu şekilde yaratılan karikatürler çok az öznitelik içerseler bile tanıma için yeterlidirler. Bilinen yüzlerin çizimlerden tanınmasında abartılmış yüzlerin gerçeğine sadık yüzlerden daha iyi sonuç verdiği bulunmuştur. Karikatür etkisi sadece çizimlerde değil, fotoğraflarda da elde edilebilir. Ama çizimlerde gölgelerin olmaması bu etkinin daha belirgin olmasını sağlar.

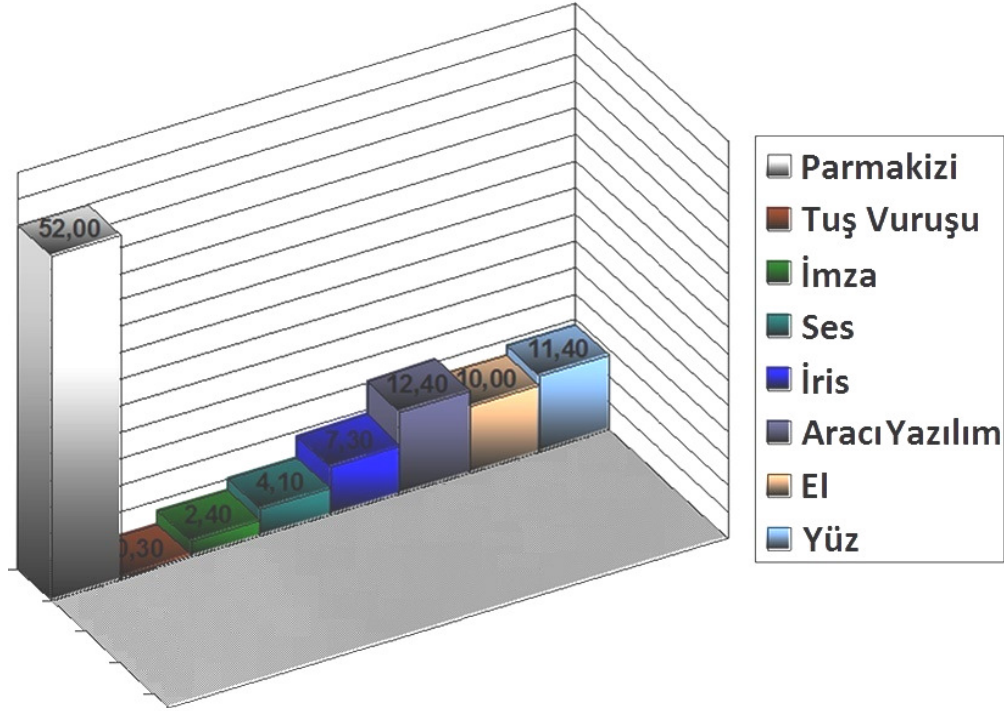
Bu bulgular yüz tanıma sisteminde hem öznitelik-tabanlı, hem de bütüncül veri işleme olduğunu ortaya koyuyor. Karikatürleştirme özniteliklerin daha iyi tanınmasını sağlıyor. Bu iyileştirme bütünsel bilginin zayıf olduğu durumlarda çok daha belirgin hale geliyor.

2.4. YÜZ, EN ETKİLEYİCİ BİYOMETRİK

Son zamanlarda gerçekleşen terörist saldırılar gibi olaylar en gelişmiş güvenlik sistemlerinde ciddi zayıflıklar olduğunu ortaya çıkarmıştır. Çeşitli devlet kurumları, biyometrik [25] diye ifade edilen vücut veya davranış özelliklerine dayalı güvenlik veri sistemlerinin geliştirilmesi için harekete geçmiştir. Genel olarak biyometrik sistemlerde, işlenmesi ve depolanması daha kolay bir şablon oluşturmak için ham verileri işlenir, ancak ihtiyaç duyulan bilginin çoğu ham verinin içindedir. Çok çekici bir teknolojidir, çünkü kişinin kim olduğundan ziyade, kişinin sahip olduğu ve ne kadar bildiği ölçütüne dayanan az gelişmiş teknolojilerin risklerini etkili bir biçimde ortadan kaldırarak, güvenlik veya erişim kontrolü gerektiren herhangi bir uygulamaya entegre edilebilir.

En yaygın biyometrikler parmak izi ve iris olsa da, son yıllarda insana özgü (parmak/avuç içi geometrisi, ses, imza, yüz gibi) diğer niteliklerde çalışma konusu olmuştur. Biyometriklerin dezavantajları olmasına karşın son yıllardaki ticari yaygınlığı Şekil 2.4'de gösterilmiştir. Iris tanıma son derece hassastır, hem pahalıdır hem de uygulama açısından pek kabul görmemiştir. Parmakizi güvenilirdir ve müdahaleci değildir, ama uygun olmayan kişiler düşünüldüğünde kullanılabilir olmadığı görünmekte. Aksine, yüz tanıma güvenilirlik ve sosyal kabul arasında iyi bir uzlaştırıcı, ayrıca güvenlik ve gizlilik bakımından dengeleyici görünmektedir. Yüz tanıma teknolojisi tabanlı herhangi bir tanıma sisteminin sivil haklar için çeşitli tehditler yarattığı gerçektir [27]. Evvelen, araştırma neticesinde elde edilen yanlış olumlu sonuçlar, masum insanların gizliliğine müdahale eder. Saniyen, kaybedilen bir belge kaybeden veya bir başkası tarafından kolayca değiştirebilirken, yüz şablon verisi çalınma tehlikesine sahip ve değiştirilememe riski mevcuttur. Buna rağmen, yüz tanıma teknolojilerini gerektiren ticari, güvenlik ve adli uygulamalar mevcuttur. Yüz tanıma, serbest satın alma koşullarında düşük bir güvenlik düzeyini sağlamaktadır. Fakat

habersiz ziyaretçilerden oluşan büyük kalabalıkların olduğu yerlerde çalışabilir durumda olması büyük bir avantajdır.



Şekil 2.4 : Bilinen biyometriklerin yaygınlığı [10]

Yüz tanıma, doğrulama ve kimlik tespiti olmak üzere iki gruba ayrılmaktadır. Yüz doğrulamada yüz imgesi ile kimliği iddia edilen kişiye ait şablon yüz görüntüleri kıyaslamak sureti ile 1:1 eşleme gerçekleştirilir. Buna karşın kimlik tespiti 1:N bir problemdir, ki burada giriş yüz imgesi yüz veri tabanını oluşturan her bir şablon yüz imgesi ile kıyaslanır.

2.4.1. Yüz İmgesi Veritabanları

Son on yıl içinde, tanıma oranı % 90'nın üzerine çıkan pek çok sistem ile yüz tanıma alanında büyük gelişmeler kaydedilmiştir. Yüz belirleme işleminin pek çok değişime tabi olması, gerçek dünya senaryolarının bir meydan okuması olarak devamına sebebiyet vermektedir. Yüz tanıma sistemi başarımını etkileyen beş ana faktör bulunmakta:

- Farklı cilt renkleri ve kamera içi kontrolünün farklılığı nedeni ile **aydınlanma** değişimleri. Yalnızca ortalama aydınlatma değişimleri altında yüz tanıma işini

iyi yapan 2B yöntemlerin sayısı azdır, poz ve aydınlanma değiştiğinde performans farkedilir biçimde düşmektedir.

- **Poz** değişiklikleri, iz düşüm bozulmalarına ve kendiliğinden görüntünün örtülmesine neden olarak kimlik tespiti sürecini etkilemektedir. Başın 32 dereceye kadar dönmesi durumunda kullanılabilen yöntemler olsa da, güvenlik amacı ile yerleştirilen kameraların konumuna bağlı olarak elde edilen görüntüler söz konusu çalışma aralığının dışına çıkabileceği için sorun tam anlamıyla çözülmüş değildir. Aksine, yüz ifadelerinde meydana gelen çığlık gibi belirgin yüz değişiklikleri hariç, kullanılan yöntemler **yüz ifadelerine** karşı görece gürbüzdür.
- Bir diğer faktör ise **zamandaki gecikme**dir. Özellikle uzun zaman dilimleri göz önüne alındığında, yüzün zamanla değişimi doğrusal değildir. Bu problem diğerleri ile karşılaştırıldığında çözümü daha güçtür ve özellikle yaş değişimi ile ilgili çok fazla çalışmaya mevcut değildir.
- Son olarak **örtünme** yüz tanıma performansını önemli bir biçimde etkilemekte, bilhassa yüzün üst tarafında bulunması durumunda.

Bu değişimlerin biri veya bileşimi durumunda önerilen yöntemlerin ne kadar iyi çalıştığını değerlendirmek için, birkaç farklı yüz imgesi veritabanı oluşturulmuştur. Meydana gelen sorunların (Test veritabanı sayısı, galeri boyutu ve soruşturma kümeleri gibi diğer parametreler ile birlikte) tipi ve sayısı, yüz tanıma sistemlerinin ne kadar gürbüz olduğunun göstergesidir. Bu durum pek çok araştırmacıyı, imgelerinde mümkün olduğu kadar değişiklik mevcut olan yüz imgesi veritabanları oluşturma yönünde teşvik etmiştir. FERET [28], CMU-PIE [29] ve AR Faces [30] en popüler 2 boyutlu yüz imge veritabanlarıdır. Her veritabanı bir grup senaryo içeren belirli zorlukları belirlemek üzere tasarlanmıştır. Galeri boyutunun büyüklüğü ve soruşturma kümesi söz konusu olduğunda FERET, poz ve aydınlanma değişimleri söz konusu olduğunda ise CMU daha uygun bir veritabanıdır. Son olarak doğal örtülü yüz imgeleri kullanılmak istendiğinde, AR Yüzleri doğru bir tercih olacaktır.

3B yüz modellerini içeren veritabanları ise oldukça azdır ve içerdikleri bilgi ise sınırlıdır. En popüler 2B yüz imgesi ve 3B yüz modeli veritabanları Tablo 2.1 ve Tablo

2.2’de ad, gri tonlu/renkli, kiři sayısı, kiři başına imge sayısı, mevcut bozulmalar (aydınlanma (i), poz (p), yüz ifadesi (e), örtü (o), zamanda gecikme (t), içeride/dıřarıda (i/o)), mevcudiyet (ücretsiz/ücretli) ve web adresi gibi ana özellikleri ile sunulmuřtur.

Tablo 2.1 : En önemli 2B yüz imgesi veritabanları [26]

İsim	RGB/ Gri ton	İmge boyutu	Kiři sayısı	Resim/ kiři	Kořullar	Mevcudiyet	Web adresi
AR Yüz Veritabanı*	RGB	576x768	126 70 Erkek 56 Kadın	26	i,e,o,t	Evet	http://rvl1.ecn.purdue.edu/~aleix/aleix_face_DB.html
Richard’s MIT Veritabanı	RGB	480x640	156 82 Erkek 74 Kadın	6	p,o	Evet	
CVL Veritabanı	RGB	640x480	114 108 Erkek 6 Kadın	7	p,e	Evet	http://www.lrv.fri.uni-lj.si/facedb.html
The Yale Veritabanı B*	Gri Tonlu	640x480	10	576	p,i	Evet	http://cvc.yale.edu/projects/yalefacesB/yalefacesB.html
The Yale Veritabanı*	Gri Tonlu	320x243	15 14 Erkek 1 Kadın	11	i,e	Evet	http://cvc.yale.edu/prprojec/yalefaces/yalefaces.html
PIE Veritabanı*	RGB	640x486	68	~608	p,i,e	Evet	http://www.ri.cmu.edu/projects/project_418.html
UMIST Veritabanı	Gri Tonlu	220x220	20	19-36	p	Evet	http://images.ee.umi.st.ac.uk/danny/database.html
Olivetti Att – ORL*	Gri Tonlu	92x112	40	10		Evet	http://www.uk.research.att.com/facedatabase.html
(JAFFE) Veritabanı	Gri Tonlu	256x256	10	7	e	Evet	http://www.mis.atr.co.jp/~mlyons/jaffe.html
The Human Scan Veritabanı	Gri Tonlu	384x286	23	~66		Evet	http://www.humanscan.de/support/downloads/facedb.php
The University of Oulu Physics-Based Yüz Veritabanı	Gri Tonlu	428x569	125	16	i	50\$	http://www.ee.oulu.fi/research/imag/color/pbfd.html
XM2VTSDB	RGB	576x720	295		p	Ön 153\$ Yan 229.5\$	http://www.ee.surrey.ac.uk/Research/VSP/xm2vtsdb/

FERET*	Gri RGB	256x384	30000	p,i,e,i/o,t	Evet	http://www.itl.nist.gov/iad/humanid/feret/
--------	------------	---------	-------	-------------	------	---

(*) İşareti ile en çok kullanılan 2B yüz veritabanları gösterilmektedir. Koşullar altında görülen kısaltmalar: (i) aydınlanma, (p) poz, (e) yüz ifadesi, (o) örtü, (i/o) içeride/dışarıda ve (t) zamanda gecikme anlamlarını taşımaktadır.

Tablo 2.2 : En önemli 3B yüz modeli veritabanları [26]

İsim	Tip	Veri Boyutu	Kişi sayısı	3B model/ kişi	Koşullar	Doku imgesi	Mevcudiyet	Web adresi
3D RMA	Noktalar kümesi	4000 nokta	120 106 Erkek 14 Kadın	3	p	Hayır	Evet	http://www.sic.rma.ac.be/~beumier/DB/3d_rma.html
SAMPL	Ara imge	200x200	10	33 (2) 1 (8)	p,e	Hayır	Evet	http://sampl.eng.ohio-tate.edu/~sampl/
Univ. Of York 1	Ara imge	-	97	10	p,e,o	Hayır	Evet	http://www-users.cs.york.ac.uk/~tomh/3DfaceDatabase.html
Univ. Of York 2	Ara imge	-	350	15	p,e	Hayır	Evet	http://www-users.cs.york.ac.uk/~tomh/3DfaceDatabase.html
GavabDB	Üç yüzey		61 45 Erkek 16 Kadın	9	p,e	Hayır	Evet	http://gavab.escet.urjc.es/recursos_en.html

Koşullar altında görülen kısaltmalar: (p) poz, (e) yüz ifadesi ve (o) örtü anlamlarını taşımaktadır.

2.4.2. Bilgisayarda Yüz Tanıma

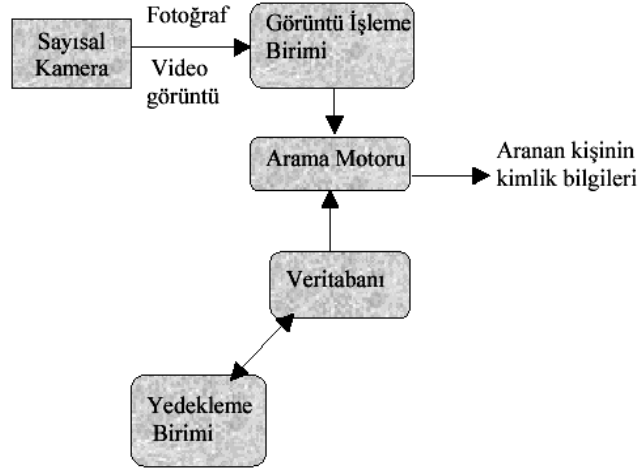
Bilgisayarda yüz tanıma dendiğinde anladığımız şey bir bilgisayar programının bir yüz imgesini girdi olarak alması ve bu yüzün kime ait olduğunu belirlemesidir. Biyometrik uygulamalarda bazen yüz ile birlikte kimlik bilgisi de verilir, o zaman bilgisayarın verilen yüzün gerçekten de o kişiye ait olup olmadığını belirlemesi istenir. Bu biraz daha basit bir problemdir.

Genel bir yüz tanıma sistemi üç kısımdan oluşur (bkz. Şekil 2.5):

- **Veritabanı:** Sistemin tanıyacağı yüzlerle ilgili bilgileri saklar. Bu bilgi yüzlerin imgeleri de olabilir, bu imgelerden çıkartılmış bazı öznitelikler de olabilir. Veritabanının büyüklüğü için bir üst sınır yoktur, milyonlarca yüz içerebilir,

fakat literatüre bakıldığında bu sayının 50 ile 5.000 arasında değiştiği, genelde 1.000 civarında olduğu görülür.

- **Öğrenme sistemi:** Yüzlerin tanınması için bir yapay öğrenme sistemi eğitilir. Veritabanındaki yüzler bu sistemin *eğitim kümesini* oluşturur. Parametrik bir sistem kullanılıyorsa, eğitim bu parametrelerin değerlerinin bulunması ile olur.
- **Önişleme sistemi:** Genellikle zor problemlerde bir önişleme aşaması ile girdinin daha kolay tanınması sağlanır. Yüz tanımda önişleme sistemi ışıklandırmanın ayarı, poz normalizasyonu, imgenin standart bir büyüklüğe getirilmesi, yüzün ortalanması, arka fonun temizlenmesi, tanımayı zorlaştıracak saç, sakal ve gözlüklerin ortadan kaldırılması gibi aşamalar içerebilir. Önişleme de aslında bir tür öğrenme gerektirir ve önişleme ne kadar iyiyse, öğrenme sisteminin işi o kadar kolaylaşır.



Şekil 2.5 : Yüz Tanıma Sistemi Modeli

Sistem eğitildikten sonra yeni bir yüz imgesi verildiğinde önce önişleme sistemi devreye girer, imgeden bir dizi öznitelik çıkartır. Bir başka deyişle, imgeyi öznitelik uzayına taşır. Bu dönüşümden çıkan öznitelikler öğrenme sistemine geçer. Öğrenme sistemi de veritabanındaki yüzlerden hangisinin (ya da hangilerinin) bu yüze en yakın olduğunu bulur.

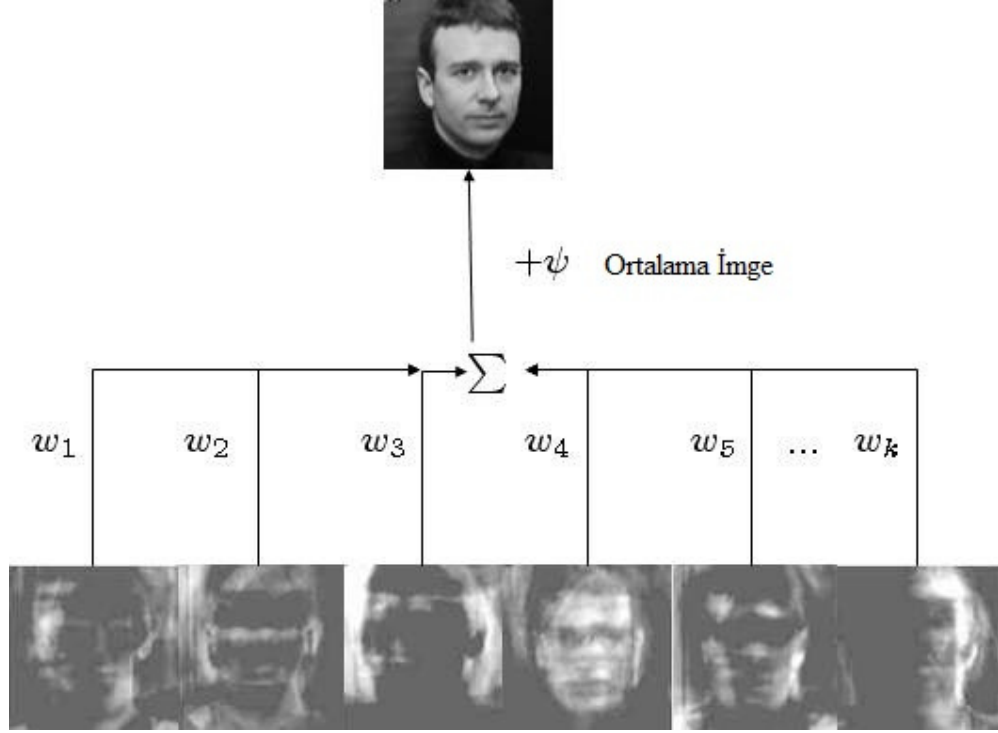
Chellappa, Wilson ve Sirohey [31] yüz tanıma ile ilgili çalışmalarını özetledikleri makalede araştırmacıları uyarıyorlar:

“Yüz tanıma algoritmaları ve sistemleri tasarlayanlar psikofiziksel ve nörofizyolojik bulgulardan haberdar olmalı, ama sadece pratik anlamda işe yarayacak olanları modellerinde kullanmalıdırlar.”

Bu uyarının anlamlı olması için, bilgisayarda yüz tanıma araştırmalarını üç gruba ayırmamız gerekir. Birinci grup yüz tanımayı bir mühendislik problemi olarak ele alır ve amacı bu problemi mümkün olan en az kaynakla, gerektiği kadar iyi çözmektir. Yukarıdaki uyarı daha çok bu grubu ilgilendirir. İkinci grup ağırlıklı olarak psikoloji ve bilişsel bilim tarafında, bilgisayar modellerini beyinde yüz tanıma ile ilgili geliştirilen hipotezleri sınamakta kullanan gruptur. Bu grubun amacı problemi daha iyi bir şekilde çözmek değil, beyindeki süreçlerin modellerini yapmaktır. Üçüncü bir grup da bu ikisinin arasında, henüz bilişsel bilimin pratik anlamda işe yarayıp yaramadığı sınanmamış bulgularını modelleyerek mühendislere yeni araçlar sunmayı hedefler. Aslında amacı daha iyi yüz tanımak olan, ama modellerinde biyolojik sistemlere göndermeler yapan ve beynin işleyişiyle tutarlı olmaya çalışan araştırmacıları da bu gruba koyabiliriz.

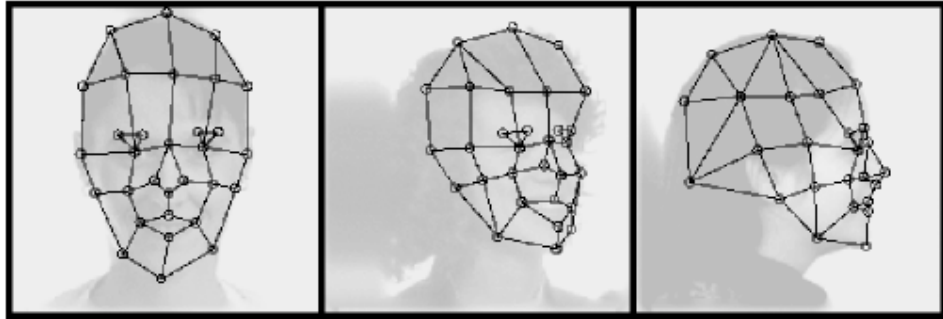
2.4.3. Yüz Tanıma Modelleri

Biyolojik yüz tanıma sistemleriyle tutarlı olma iddiası taşıyan ilk yüz tanıma modeli Turk ve Pentland'ın özyüz (eigenface) modelidir [32]. (bkz. Şekil 2.6) Yayınlandığı tarihten bu yana iki binin üzerinde atıf alan bu çalışmada yüz tanıma için Temel Bileşenler Analizi (Principal Components Analysis, PCA) önerilir. PCA yönteminde eğitim kümesindeki imgelerin varyansının en yüksek olduğu boyutlar seçilir ve imgelerin bu boyutlara izdüşümü alınır. Bu boyutların her birine özyüz adı verilir. İzdüşüm alma, yüzün bu özyüzlerin ağırlıklı toplamı olarak ifade edilmesi anlamına gelir. Tanıma bu dönüşümün ardından veritabanındaki en yakın şablonu bulmak olarak yapılır. Özyüz metodu bütüncül yaklaşıma örnektir. İmgelerin büyüklüklerine, pozlarına ve konumlarına karşı hassas olduğundan, her yüz için değişik pozlarda şablonlar saklayan bir çeşitlemesi de geliştirilmiştir.



Şekil 2.6 : Özyüzler ile resmin yeniden oluşturulması

Bir diğer biyolojik yaklaşım da elastik çizge (elastic graph) modelidir [33]. (bkz. Şekil 2.7) Bu modelde Gabor dalgacıkları ile yüzlerden öznitelikler çıkartılır. Gabor dalgacıkları beyindeki görme sisteminin ilk aşaması olan V1'deki basit hücrelere benzer bir işlem yaparlar. İmgelerden gelen öznitelikler daha önceden eğitim kümesinde bulunup saklanmış olan özniteliklerle karşılaştırılır ve bir yüz konfigürasyonuna uyan en iyi öznitelik seti bulunur. Bu metod öznitelik-tabanlı yaklaşıma örnektir.



Şekil 2.7 : Elastik çizge modeli

İki modelin başarısının da insanlardaki yüz tanıma sonuçlarıyla benzeştiği gözlenmiştir. Genel olarak elastik çizge modeli öznitelikler arasındaki yapısal ilişkiyi ayrıca modellediği için daha başarılıdır. Bu iki modele temelde benzeyen, fakat farklı öznitelikler kullanan birçok çalışma vardır. Bunların dışında insanlarda yüz tanıma ile ilgili değişik savlara denemek amacıyla da modeller geliştirilmiştir. Bu modellere birkaç örnek verelim.

Ramasubramanian ve Venkatesh [34] yüz imgelerine ayrık kosinüs değişimi (Discrete Cosine Transform, DCT) uyguluyorlar. Kullandıkları bütünsel yöntem yüksek bir başarı yüzdesi yakalıyor. Bu modelde de Gabor dalgacıkları gibi insandaki görme sisteminde var olduğu düşünülen dönüşümlere benzer bir dönüşüm var. (İnsanların görme sistemi düşük frekanslardaki bilgiye karşı daha hassastır. Yüz imgelerinde de enerji daha çok düşük frekanslarda bulunur.)

Almanya'da Max Planck Enstitüsü'nün Biyolojik Siberetik konusunda çalışan Tübingen araştırma merkezinde insanlarda ve bilgisayarlarda yüz tanımayı karşılaştıran çalışmalar yapılıyor. Bunlardan birinde yüzlerden parçalar kesilip karıştırılıyor ve bütüncül bilgi ortadan kaldırılıyor [35]. Deneklerin bu şekilde yüzleri ne kadar başarıyla tanıdığına bakılıyor. Daha sonra parçaları karıştırılmış yüzler tanıma tamamen başarısız olana kadar bulanıklaştırılıyor. Yani hem öznitelik, hem de bütünsel bilgi ortadan kalkmış oluyor. Daha sonra aynı bulanıklaştırma normal yüz imgelerine uygulanıyor. Böylece öznitelik bilgisi siliniyor, sadece bütünsel bilgi kalıyor. Bu şekilde öznitelik bilgisiyle bütünsel bilginin yüz tanımaya ne kadar katkı sağladığını karşılaştırmak mümkün oluyor. Benzer bir tanıma testi öznitelikler ve bunların arasındaki ilişkileri ayrı ayrı modelleyen bir bilgisayar programına veriliyor ve programın tanıma performansının deneklerinkine son derece benzediği görülüyor.

2.4.3.1. Bağımsız Bileşen Analizi

Bağımsız bileşen analizi (Independent Component Analysis, ICA) yöntemi, çok değişkenli veri kümesinde saklı olan yapıları belirleyen istatistiksel bir yöntemdir. ICA yönteminde, bağımsız bileşen olarak tanımlanan saklı faktörler, bilinmeyen bir karıştırma matrisi ile karıştırılarak gözlem verilerini oluştururlar. Gözlem vektörü x , kaynak vektörü s ve karıştırma matrisi A ile gösterildiğinde temel katışım modeli eşitlik

2.1'deki gibi olur. ICA yöntemindeki amaç gözlem vektörü kullanılarak, bilinmeyen A ve s değerlerinin tahmin edilmesidir [36, 37].

$$x = A.s \quad (2.1)$$

Doğrusal ICA'da karıştırma matrisinin kare, kaynakların Normal (Gaussian) olmayan bir dağılıma sahip olduğu kabul edilir. ICA'da bağımsız bileşenlerin hesaplanması için yüksek dereceden bilgilerin bilinmesi gerekmektedir. Verilerin Normal dağılıma uygun olduğu varsayıldığında bu bilgiler sıfırdır. Bu yüzden ICA, Normal dağılıma sahip verilerde sonuç vermemektedir. Karıştırma matrisinin kare kabul edilmesi, kaynak sayısı ile bağımsız bileşen sayısının eşit olması demektir. Bu varsayım ayrıştırma matrisinin tahminini kolaylaştırmaktadır. Çünkü karıştırma matrisinin tersi alınarak eşitlik 2.2'deki gibi ayrıştırma matrisi bulunur [36, 37].

$$s = W.x, \quad W = A^{-1} \quad (2.2)$$

Burada **W**, ayrıştırma matrisidir. ICA'da amaç kaynaklar arasındaki istatistiksel bağımlılığı minimize edecek şekilde **W** matrisinin bulunmasıdır.

ICA'da yaygın olarak kullanılan 2 farklı mimari yapı vardır [38]. Birinci yapıda görüntüler değişkenleri, piksel değerleri de deneme değerlerini temsil etmektedir. Bu yaklaşım, görüntülerin bağımsızlığına veya görüntü fonksiyonlarının bağımsızlığına duyarlıdır. Bu yaklaşıma göre bir görüntüdeki bir pikselin değeri bir diğer görüntüdeki aynı pikselin değerinden tahmin edilemiyorsa, bu iki görüntü birbirinden bağımsız demektir. İkinci yapıda birinci durumun transpozunu alır. Bu durumda piksel değerleri değişken, görüntüler ise deneme olur. Bu yaklaşım da, piksellerin bağımsızlığına veya piksel fonksiyonlarının bağımsızlığına duyarlıdır. Bu yaklaşıma göre aynı görüntü içindeki birbirine benzer piksellerden birinin değeri diğeri kullanılarak tahmin edilemiyorsa pikseller birbirinden bağımsız demektir.

2.4.3.2. Temel Bileşen Analizi

Temel Bileşen Analizi (Principal Component Analysis, PCA) veya diğer adıyla Karhunen-Loève transformu işaret işleme sahasında yaygın olarak kullanılan istatistiksel hesaplama tekniğidir [39]. PCA'daki esas fikir, doğrusal olarak dönüştürülmüş bileşenlerle maksimum değişim miktarını ifade eden bileşenleri

bulmaktadır. Danışmansız bir metod olan PCA yüksek boyutlu verileri daha düşük boyutlu verilere indirgemeye yarayan güçlü bir veri analiz tekniğidir. PCA kısaca şu şekilde özetlenebilir.

X , n adet d boyutlu x_i verisi içeren bir matris olsun,

$$X = [x_1, x_2, \dots, x_n] \quad (2.3)$$

İlk olarak n adet verinin ortalama değeri \bar{x} bulunur, ortalama değeriyle her bir x_i verisinin farkı alınarak \tilde{X} fark matrisi bulunur.

$$\tilde{X} = [x_1 - \bar{x}, x_2 - \bar{x}, \dots, x_n - \bar{x}] \quad (2.4)$$

Elde edilen fark matrisi aşağıdaki ifadede kullanılarak C kovaryans matrisi hesaplanır.

$$C = \frac{1}{n} \tilde{X} \tilde{X}^T \quad (2.5)$$

Aşağıdaki ifade doğrulanacak şekilde C kovaryans matrisinin özvektörleri, $\phi_{\tilde{X}}$ ve özdeğerleri, $\Lambda_{\tilde{X}}$ bulunur.

$$C \phi_{\tilde{X}} = \phi_{\tilde{X}} \Lambda_{\tilde{X}} \quad (2.6)$$

Bu aşamayla elde edilen d adet d boyutlu özvektör ve köşegen matris biçiminde d adet özdeğer aşağıdaki gibidir.

$$\Lambda_{\tilde{X}} = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \lambda_d \end{bmatrix} \quad \Phi_{\tilde{X}} = [\phi_1, \phi_2, \dots, \phi_d] \quad (2.7)$$

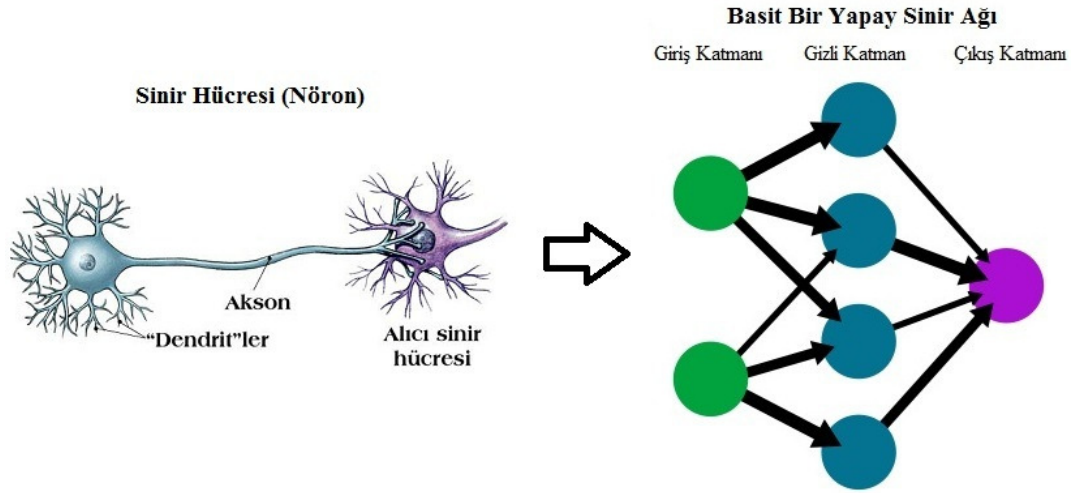
m adet, yüksek özdeğere karşılık gelen özvektörler seçilerek $d \times m$ boyutlu iz düşüm matrisi W elde edilir.

$$x'_i = W^T (x_i - \bar{x}) \quad (2.8)$$

ifadesiyle d boyutlu x_i verisiyle daha düşük boyutlu x'_i elde edilir.

2.4.3.3. Yapay Sinir Ağları

Temelde yapay sinir ağları biyolojik sinir hücresini model olarak alır. Kabul edilmesi gerekir ki yapay sinir ağlarının gerçekçi bir model yaklaşımı yapması zaman alacaktır. Yapay sinir ağlarının yüz tanıma üzerindeki çalışmalarına geçmeden önce yapay sinir ağlarının genel kazanımlarından söz etmek lazım. Bununla birlikte yapay sinir ağlarının bilgi işleme gücünü, paralel dağılmış yapısından, öğrenebilme ve genelleme yeteneğinden aldığı söylenebilir. Genelleme, eğitim ya da öğrenme sürecinde karşılaşılmayan girişler için YSA' nın uygun tepkileri üretmesi olarak tanımlanır. Bu üstün özellikleri, YSA' nın karmaşık problemleri çözebilme yeteneğini gösterir.



Şekil 2.8 : Sinir hücresinden Yapay sinir ağına

Günümüzde birçok bilim alanında YSA, aşağıdaki özellikleri nedeniyle etkin olmuş ve yüz tanıma gibi birçok alanda uygulanmıştır [40].

- **Doğrusal Olmama:** YSA' nın temel işlem elemanı olan hücre, doğrusal değildir. Dolayısıyla hücrelerin birleşmesinden meydana gelen YSA da doğrusal değildir ve bu özellik bütün ağa yayılmış durumdadır. Bu özelliği ile YSA, doğrusal olmayan karmaşık problemlerin çözümünde en önemli araç olmuştur.
- **Öğrenme:** YSA' nın arzu edilen davranışı gösterebilmesi için amaca uygun olarak ayarlanması gerekir. Bu, hücreler arasında doğru bağlantıların yapılması ve bağlantıların uygun ağırlıklara sahip olması gerektiğini ifade eder. YSA' nın

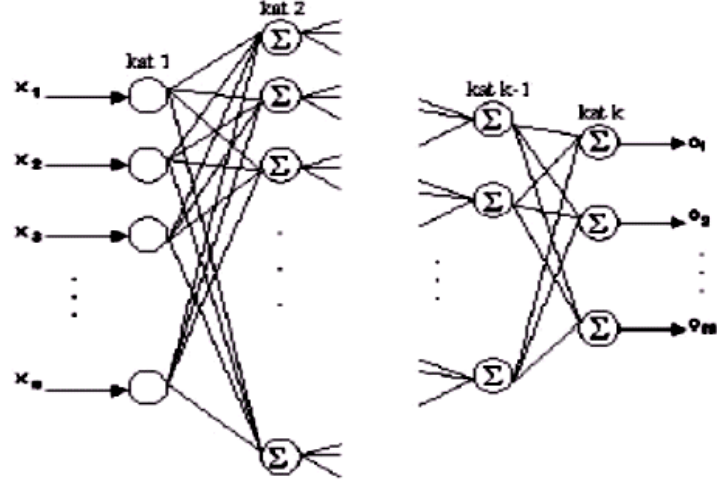
karmaşık yapısı nedeniyle bağlantılar ve ağırlıklar önceden ayarlı olarak verilemez ya da tasarlanamaz. Bu nedenle YSA, istenen davranışı gösterecek şekilde ilgilendiği problemden aldığı eğitim örneklerini kullanarak problemi öğrenmelidir.

- **Genelleme:** YSA, ilgilendiği problemi öğrendikten sonra eğitim sırasında karşılaşmadığı test örnekleri için de arzu edilen tepkiyi üretebilir. Örneğin, karakter tanıma amacıyla eğitilmiş bir YSA, bozuk karakter girişlerinde de doğru karakterleri verebilir ya da bir sistemin eğitilmiş YSA modeli, eğitim sürecinde verilmeyen giriş sinyalleri için de sistemle aynı davranışı gösterebilir.
- **Uyarlanabilirlik:** YSA, ilgilendiği problemdeki değişikliklere göre ağırlıklarını ayarlar. Yani, belirli bir problemi çözmek amacıyla eğitilen YSA, problemdeki değişimlere göre tekrar eğitilebilir ve değişimler devamlı ise gerçek zamanda da eğitime devam edilebilir. Bu özelliği ile YSA, uyarlamalı örnek tanıma, sinyal işleme, sistem tanılama ve denetim gibi alanlarda etkin olarak kullanılır.
- **Hata Toleransı:** YSA, çok sayıda hücrenin çeşitli şekillerde bağlanmasından oluştuğu için paralel dağılmış bir yapıya sahiptir ve ağına sahip olduğu bilgi, ağdaki bütün bağlantılar üzerine dağılmış durumdadır. Bu nedenle, eğitilmiş bir YSA'nın bazı bağlantılarının hatta bazı hücrelerinin etkisiz hale gelmesi, ağına doğru bilgi üretmesini önemli ölçüde etkilemez. Bu nedenle, geleneksel yöntemlere göre hatayı tolere etme yetenekleri son derece yüksektir.

Yapay sinir ağları yüz tanıma kullanıldığı gibi diğer tüm biyometrik tanımalarda kullanılmakta ve bu tanıma yöntemlerinde büyük başarılar sağlamaktadır. Yapay sinir ağlarının biyometrik tanıma ve özellikle yüz tanıma getirdiği yenilikler yukarıda saydıklarımızın dışında; yüz tanımanın sisteme getirdiği veri fazlalığını ve işlemci üzerine düşen yükü ve yoğunluğu azaltması da söylenebilir.

Şekil 2.9'da verilen, ağ yapısı geri hata yayımlı sinir ağı blok şemasıdır. Yüz tanıma uygulamalarında tercih edilen bir ağ yapısı olan bu modeldir. Tanıma için kullanılan ağ genel olarak giriş katmanı, çıkış katmanı ve bir saklı katmandan oluşmaktadır. Yüz tanıma kullanılan yapay sinir ağı yapılarının olabilecek en az sayıda katman kullanarak gerçekleştirilme sebebi fazla katmanlı ağlarda öğrenme işleminin yada

tanıma işleminin gerçekleşmemesi yada fazla katmanlı ağların sistemin çok zaman harcaması ve işlemciye düşen yük bakımından uygulanabilir olmamasıdır.

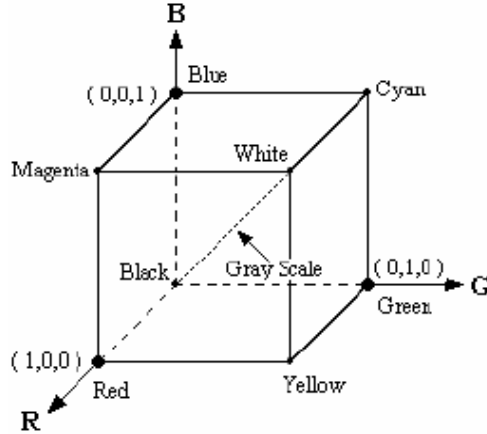


Şekil 2.9 : Yapay Sinir Ağı modeli

Yapay sinir ağlarının kullanılarak yapıldığı çalışmalar sonucunda tanıma yapan bilgisayar sistemine düşen yük azaltılmış ve bu nedenle de yapay sinir ağı gerçek zamanlı yüz tanıma yada gerçek zamanlı video kaydında yüz seçmek için kullanılmıştır.

2.4.3.4. Dik Yüzler Yöntemi [41]

RGB modeli ile ifade edilen bir imgenin her bir beneğine karşı gelen renk R, G, B'nin belli katsayılar ile birleşiminden oluşur. Bu durumda R, G, B için ayrı matrisler oluşturulabilir. Diğer taraftan gri seviyeleri olarak gösterilen bir imgedeki renk bileşenleri Şekil 2.10'da görülen siyah ve beyaz köşeleri birleştiren köşegen üzerindedir.



Şekil 2.10 : RGB modelinin kartezyen koordinat sisteminde gösterimi

Bu yöntemde kullanılan veritabanı gri seviyeli resimlerden oluşmaktadır. Dolayısıyla RGB modelinde ifade edildiği gibi **R**, **G** ve **B**'ye karşı gelen değerler aynıdır. Algoritmada her bir imge tek bir matris olarak ifade edilir. Veri tabanımızda bulunan **M** adet yüz imgesi kullanılarak, **M** elemanlı dik-düzgün (ortonormal) bir matris sistemi oluşturulur. Böylece veri tabanında bulunan matrisler dışında onlardan üretilen **M** adet dik-düzgün bir alt küme elde edilmiş olur. Verilen bir yüz imgesinin, bu dik küme elemanlarının gerekli katsayılar ile lineer kombinasyonu hesaplanarak yaklaşık gösterimi elde edilir. Bu gösterimin veri tabanındaki imgelere olan mesafesi en küçük yapılarak hedef yüz belirlenebilmektedir.

Dikleştirme için **Gramm Schmidt Algoritması** [42] kullanılmaktadır:

{ F_1, F_2, \dots, F_M } dik-düzgün olmayan veri tabanı matrisleri olsun. Dik düzgün { f_1, f_2, \dots, f_M } matrisleri aşağıdaki adımları izleyerek bulunabilmektedir:

$$\begin{aligned}
 E_1 &= F_1 \Rightarrow f_1 = E_1 / \|E_1\| \\
 E_2 &= F_2 - \langle F_2, f_1 \rangle f_1 \Rightarrow f_2 = E_2 / \|E_2\| \\
 &\quad \downarrow \\
 E_M &= F_M - \sum_{k=1}^{M-1} \langle F_M, f_k \rangle f_k \Rightarrow f_M = E_M / \|E_M\|
 \end{aligned} \tag{2.9}$$

ve burada $\| \cdot \|$ $N \times N$ boyutlu matrisler uzayında Öklit normunu göstermektedir. Veri tabanındaki herhangi bir A imgesinin bu dik küme elemanları üzerine dik iz düşümleri kullanılarak, A imgesi şu şekilde yaklaşık olarak ifade edilebilir:

$$A' = \sum_{k=1}^M \langle A, f_k \rangle f_k \tag{2.10}$$

Burada $\langle \cdot, \cdot \rangle$ iç çarpımı göstermektedir. Böyle tanıma problemlerinde genel olarak, bir A imgesinin veri tabanındaki karşılığı, A' kestiriminin veri tabanı elemanları ile arasındaki Öklit mesafesi $C_i = \|A' - \Gamma_i\|$ en küçük yapılarak elde edilir:

$$\min_i \{ \|A' - \Gamma_i\| \} \tag{2.11}$$

Bu yaklaşım ile benzetimlerde her zaman çok başarılı sonuçlar elde edilememiştir. Bu nedenle önerilen yöntemde aşağıdaki şekilde hesaplanan bir karar eşiği kullanılmıştır:

- Girişte verilen imge ile izdüşüm sonucu elde edilen kestirim arasındaki Öklit uzaklığı hesaplanır.
- Kestirilen imge ile veri tabanında tüm imgeler arasındaki uzaklıklar hesaplanır ve en küçük uzaklık seçilir.
- İlk iki adımda bulunan değerler arasındaki farkın deneysel olarak belirlenen bir eşik seviyesinin altında olup olmadığına bakılarak karar verilir.

3. MALZEME VE YÖNTEM

İnsan yüzünün otomatik olarak tanınması, son yıllarda aktif bir araştırma alanı olmuştur. Araştırmaların önemi yanısıra, izleme, güvenlik, haberleşme ve insan bilgisayar akıllı etkileşimi gibi kanunî mecburiyeti olan alanlar ile ticarî uygulamaların da sayısı artmaktadır. Günümüze kadar yüz tanıma için farklı yaklaşımlar önerilmiştir, bunları kabaca çözümleyici (analytic) ve bütünsel (holistic) yaklaşımlar olarak iki sınıfa ayırabiliriz.

Çözümleyici yaklaşımlar, yüzdeki yerel nitelikler (local features), yüz niteliklerinin biçimleri, yüze ait belli noktaların birbirleri arasındaki mesafe ve açılar gibi değerleri kullanmaktadır. Örnek olarak, yüz niteliklerinden elde edilen yeğinlik (intensity) değerini kullanabilir. Çözümleyici yaklaşımların asıl getirisi, yüze ait önemli nitelik noktalarında esnek bir deęiştirim ile poz deęişikliklerini telafi edebilmesidir. Brunelli ve Poggio [43] çalışmalarında, yüz bölgeleri göz, burun ve ağız şablonları ile karşılaştırılarak yüz tanıma işlemini geometrik kısıtlamalardan bağımsız olarak gerçekleştirilmiştir. Destek Vektör Makineleri (Support Vector Machines, SVM) sınıflandırıcı kümesi uygulanarak farklı yüz bileşenleri elde edilir ve bileşenlere ait gri değerleri ile bir nitelik vektörü oluşturulmuştur [44]. Geliştirilmiş 2 Boyutlu Saklı Markov Modelinin (Hidden Markov Model, HMM) kullanıldığı bir başka uygulamada [45] yüz imgesi üzerinde üstüste bindirilmiş bir pencere kaydırılarak elde edilen Ayrık Kosinüs Dönüşümü (Discrete Cosine Transform, DCT) katsayıları ile bir gözlemci yöneyi oluşturulmuştur.

Çözümleyici yaklaşımlar yüzden elde edilen belirgin yüz niteliklerini karşılaştırırken, bütünsel yaklaşımlar bütün yüzden elde ettiği bilgiyi kullanır. Temel Bileşen Analizi (Principal Component Analysis, PCA) bilinen bir bütünsel metoddur. PCA Karhunen-Loeve dönüşümünü kullanan istatistiksel bir tekniktir. Turk ve Pentland, yüz gösterimi ve tanıma için PCA tekniğini kullanan bir özyüz yöntemi geliştirmişlerdir [46]. PCA, ortalama karesel hata bakımından en uygun ifade biçimidir, ancak farklı kişilere ait yüz

imgeleri arasındaki daha önemlidir [47]. Bu yorum dikkate alınarak Doğrusal Ayırtaç Analizi (Linear Discriminant Analysis, LDA) [48] Fisher yüz yöntemleri [49] için uygulanmıştır. LDA'nın öngörüsü, sınıf içi saçılımı düşürürken sınıflar arası saçılımı büyütür, fakat iyi bir gerçekleştirme için örneklerden oluşan eğitim kümesinin geniş olması gereklidir. Bu durum yüz tanıma uygulamaları için pek uygun görünmemekte olup, LDA uygulamasından önce PCA ile boyut indirgeme yoluna gidilmektedir [50]. Yapay Sinir Ağları (Neural Networks, NN) kullanılarak yerel nitelikler sınıflandırılmaktadır.

Genel teknikler ile yüzün önden görünen resimleri için başarımlar iyidir, lakin farklı açılardan alınan görüntülerde sıkıntı söz konusudur. Bu nedenle bu yöntemlerde normalizasyon önemli ve kaçınılmaz bir işlemdir, yüzdeki belirgin noktalar (gözler, burun delikleri, ağız merkezi, vb.) kullanılarak giriş yüz imgesi gerekli döndürme ve yeniden boyutlandırma süreçlerine tabi tutulur. Normalizasyon sonrasında giriş imgeleri model yüz ile hizalanır ve yüz tanıma gerçekleştirilir.

Bugüne kadar gerçekleşen pek çok gelişmeye karşın yüz tanıma zorlu bir problem olmaya devam etmektedir. Yüz ifadeleri, poz ve aydınlık gibi parametrelerde meydana gelen küçük değişimler, yüz imgesinde bozulmalara neden olmaktadır. Buna rağmen, yüzdeki yerel niteliklerin bu tür bozulmalara karşı dayanıklı olduğu ve bir uzay-frekans analizi ile niteliklerin eldesinin mümkün olduğu düşünülmektedir [51-53]. Bu amaç doğrultusunda, iyi biçimde belirlenmiş uzay-frekans yerelleştirmesi ile Dalgacık Analizi doğru bir seçim olarak karşımıza çıkmakta [54,55]. Çeşitli dalgacık tabanları arasında Gabor Fonksiyonları, hem zamanda hem de frekansta en uygun çözünürlüğü sağlamaktadır [56,57]. Örüntü tanıma için yerel niteliklerin elde edilmesinde, aşağıda yer alan nedenler bakımından Gabor Dalgacıklar en uygun yaklaşım olarak görünmektedir:

- **Biyolojik Sebepler:** Gabor Dalgacıkların şekli ile beyin arka lobunda yer alan görme işlevini yerine getiren hücrelerin algılama alanları benzerlik göstermektedir [57].
- **Matematiksel Nedenler:** Yerel uzaysal frekansların ölçümü için uygun olan Gabor Dalgacıklarıdır [58, 59].

- **Gözleme Dayalı Nedenler:** Gabor Dalgacıkların, doku bölütleme [60,61], elle yazılan rakamları tanıma [62] ve parmakizi tanıma [63] gibi alanlarda nitelik uzayında meydana gelen bozulmaları tolere ettiği gözlemlenmiştir.

Yüz tanıma alanında Gabor Dalgacıkların kullanımına 1993 yılında Lades ve diğ. [64] tarafından önerilen Devingen Bağlantı Mimarisi (Dynamic Link Architecture, DLA) öncülük etmiştir. Burada yüzler biçimi değiştirilebilir düğümleri mevcut bir dikdörtgen çizge ile ifade edilir. Düğümlerde Gabor Dalgacıklar ile yerel nitelikler belirlenir. Söz konusu düğümler “Gabor Jets” olarak da ifade edilir. Wiskott ve diğ. [65] DLA’i geliştirerek, yüz içinde yer alan nirengi noktalarına çizgenin düğümlerini yerleştirmek sureti ile Esnek Bağ Çizge Eşleme (Elastic Bunch Graph Matching, EBGM) metodunu oluşturmuştur. O zamandan bu yana esnek çizge tabanlı pek çok yöntem önerilmiştir [66-69]. Bu yöntemlerin tamamı, nitelik vektörleri yüz üzerinden seçilen noktalardan elde edildiği için çözümleyici yaklaşımlardır.

Son zamanlarda Gabor Dalgacıklar [70-72] yüz tanımda genel formda kullanılmaya başlanmıştır. Bütüncül yaklaşımlar, nitelik gösterimi için Gabor Dalgacıklar uygulandıktan sonra görüntünün tümünü kullanırlar.

Performans bakımından, EBGM yöntemi rekabetçi bir performans sergileyerek FERET değerlendirmesinde [73] üst sırada yer almıştır. Yapılan son yüz doğrulama rekabetinde (FVC2004) [74], en iyi iki yöntemde de nitelik eldesinde Gabor Dalgacıklar kullanılmıştır. Nottingham Üniversitesi tarafından kullanılan yaklaşımda imgenin tamamından Gabor nitelikleri elde edilirken, Binghua Üniversitesi bütüncül ve bileşen tabanlı sınırlandırmaları birlikte kullanmıştır.

Yüz gösteriminde Gabor Dalgacıkların başarı ile kullanılmış olması, çalışmamızda bizim de Gabor Dalgacıkları uygulamamızın nedenidir. İmgeyi Gabor Dalgacıkları ile filtrelemeye dayalı çalışmamızın emsallerinden ayrılan tarafı, Parçacık Sürü Optimizasyonu (PSO) kullanarak Gabor Dalgacıklar ile elde edilen nitelik vektörünün optimizasyonudur.

3.1. GABOR DALGACIKLAR

3.1.1. Dalgacık Dönüşümüne Kısa Bir Bakış

Dalgacık serileri bir çok farklı alana uygulanabilen bir yöntem olup, bunlar arasında uygulamalı matematik, sinyal işleme teknikleri, ses ve görüntü sıkıştırma teknikleri başta gelmektedir. Bu üç alanda yapılan çalışmalarda wavelet serileri ve dönüşümleri oldukça hassas sonuçlar vermiştir. Dalgacık Dönüşümü, bir işaretin $\psi(t)$ örnek fonksiyonunun ölçekleme ve ötelemeleriyle oluşturulan taban fonksiyonları ile analizine dayanır. Bu taban fonksiyonları hızlı değişimli yani yüksek frekans ve yavaş değişimli yani alçak frekanslı fonksiyonlardır.

Dalgacıkların temel başlangıcı Joseph Fourier'e ve O'nun Fourier Dönüşümü'ne kadar gitmektedir. 1807'den sonra Fourier denklemlerinin ortaya çıkmasıyla matematikçiler sinyali tanıma için frekans alanında çalışmaya yöneldiler.

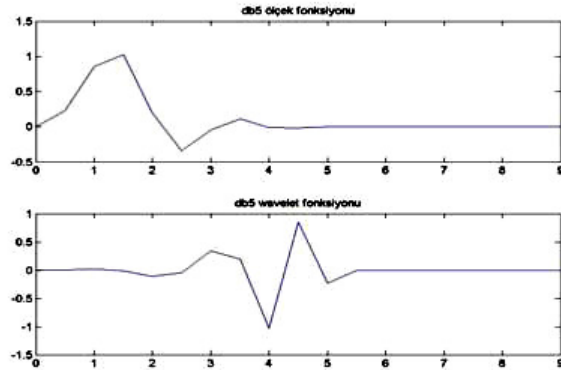
Dalgacıklar ilk olarak 1909'da Haar Wavelet olarak adlandırılan A. Haar'ın [75] tezinin ekler kısmında görülmüştür. Haar dalgacık fonksiyonunun en önemli özelliği sıkı desteğe sahip olmasıdır. Buna karşın Haar dalgacık fonksiyonu sürekli türeve sahip değildir. Haar Dalgacıkları bazı sınırlı uygulamalar için geçerli olup, bilinen en basit ve en eski wavelet fonksiyonudur.

1930'lu yıllarda değişken ölçekli taban fonksiyonları üzerine yapılan çalışmalarda [76], Littlewood, Paley ve Stein, ölçeği değiştirken enerjisini koruyan fonksiyonlar elde ettiler. 1950-1960 yıllarında Littlewood-Paley teorisi kısmi diferansiyel denklemler ve integral denklemlerine ve Calderon-Zygmund teorisi olarak harmonik analiz alanına uygulandı. 1960-1980 yılları arasında ise Guido Weiss ve Ronald Coifman, bir fonksiyon uzayının en basit elemanları olarak tanımladıkları ve atom adını verdikleri fonksiyonlar üzerine çalıştılar [77]. Amaçları bu atomlar sayesinde fonksiyonlar uzayını yeniden oluşturabilmektir. 1977'lerde Esteban ve Galand yeni bir süzgeç kavramını ortaya attı ancak bu yolla ana sinyalin yeniden elde edilmesinde hata çok yüksekti.

Dalgacık terimi ilk kez 1980'lerde John Morlet ve A. Grossman tarafından kuantum fiziği çalışmalarında kullanıldı. John Morlet [78], Kısa Zamanlı Fourier Dönüşümü'ne

alternatif çalışmalar yapmaktaydı. Kısa Zamanlı Fourier Dönüşümü'nde, zamanda pencerelemiş işaretin Fourier Dönüşümü alınmakta ve pencere ötelenip, tüm işaret taranarak aynı işlem tekrar edilmektedir. Bu dönüşümde pencere fonksiyonu dar alınırsa yüksek frekanslar, geniş alınırsa alçak frekanslar daha iyi analiz edilebilmekteydi. Morlet, bu iki işlemi tek bir dönüşümde yapmak için dönüşüm fonksiyonlarını farklı şekilde oluşturdu. Pencerelemiş bir kosinüs fonksiyonunu ve bu fonksiyonun zamanda daraltılmış biçimlerini kullanarak, pencere fonksiyonunun zamandaki yerine ve pencerenin ölçeğine bağlı yeni dönüşüm fonksiyonları geliştirdi. Morlet geliştirdiği fonksiyonları sabit biçimli dalgacıklar olarak adlandırdı. Daha sonra ise dalgacıklar olarak kısaltıldı.

Grossman, A [79], kuantum mekaniğindeki evre uyumlu durumları kullanarak, Morlet'in önerdiği dalgacık fonksiyonlarına dayanan integral için ters dönüşümü elde etti. 1985'de Mallat çoklu çözünürlük analizini geliştirdi, 1987'de dalgacık ve süzgeç grupları arasındaki ilişkiyi ortaya çıkardı. Meyer [80] kendi adıyla anılan ilk dalgacıkları ortaya attı. Bu Haar dalgacıkların aksine, sürekli uygulamalarda kullanılabilen bir fonksiyon idi. Sürekli türeve sahip ilk dalgacık fonksiyonları idi. Yıllar geçtikçe, Ingrid Daubechies bir takım dik tabanlı dalgacık serilerini ortaya atarak günümüzdeki bir çok uygulamaya temel teşkil etmiştir.



Şekil 3.1 : Birinci dereceden Daubechies 5 ölçek ve dalgacık fonksiyonu

Geçen son on yılda, dalgacıkların kullanımında artan bir eğilim gözlenmiştir. Bu alanlar arasında görüntü işleme teknikleri, tıbbi uygulamalar, radar, akustik, sonar sistemleri, veri sıkıştırma gibi alanları sayabiliriz.

Tanım olarak, bir dalgacık, ortalama değeri sıfır olan ve zamanla sınırlı bir dalga şeklindedir. Zaman ekseninde kaydırma ve ölçekleme parametreleri dalgacıkların temelini oluşturmaktadır. Fourier serilerinin temel fonksiyonları sinüs ve kosinüs ifadelerinden meydana gelmektedir. Buna karşın çok sayıda dalgacık fonksiyonları vardır. Dalgacık dönüşümü değişik uzunluktaki bölgeleri kapsayan pencereleri içeren yeni bir teknik olarak karşımıza çıkmaktadır [81, 82]. Dalgacık dönüşümü kullanarak bir sinyalin ayrıştırılması ve tekrar oluşturulması genel olarak üç aşamadan oluşur [83].

- Ayırık dalgacık dönüşümü kullanarak sinyalin bileşenlerine ayrılması,
- Ortaya çıkan katsayıların eşikleme metotlarıyla yumuşatılması,
- Eşikleme yapılmış ayırık dalgacık katsayılarından orijinal sinyalin tekrar elde edilmesi.

3.1.2. Dalgacık ve Fourier Dönüşümü

Dalgacık Dönüşümü, bir işaretin $\psi(t)$ örnek fonksiyonunun ölçekleme ve ötelemeleriyle oluşturulan taban fonksiyonları ile analizine dayanır. Bu taban fonksiyonları hızlı değişimli yani yüksek frekans ve yavaş değişimli yani alçak frekanslı fonksiyonlardır.

Dalgacık dönüşümün gelişimi Fourier dönüşümü ile yakından ilgilidir, bu nedenle teoriyi irdelemeye Fourier Dönüşümü'nden başlamak isabetli olacaktır. Aşağıdaki denklemde Sürekli Fourier Dönüşümü görülmektedir.

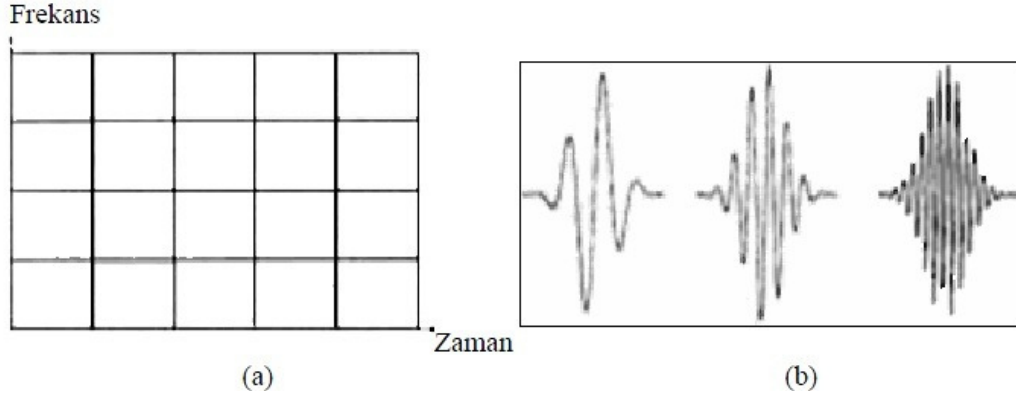
$$F(\omega) = \int_{-\infty}^{\infty} f(t) \cdot e^{-j\omega t} dt \quad (3.1)$$

Denklem (3.1)'de Fourier dönüşümü sonsuz zaman aralığında temel sinüs fonksiyonlarını ($e^{-j\omega t}$) kullanmaktadır. Bu yaklaşım, frekansı zamanla fazla değişmeyen kararlı sinyallerin analizinde oldukça hassas ve kullanışlı bir yöntemdir. Bir sinyalin standart Fourier Dönüşümü, zaman ve frekanstaki içeriği ile ilişkilidir. Fourier Dönüşümü ile sinyal bileşenlere ayrılır ve bileşenin gücü belirlenir. Ancak, bu işlev tüm frekans karakteristiğini içermez. Standart Fourier Dönüşümü'nde frekans ortamında lokal bir noktadan bahsederken, bu zaman ortamında tüm zaman aralığına karşı gelmektedir. Yani zaman bilgisi kaybolmakta, hangi frekans bileşeninin hangi zaman aralığında varolduğu bilinmemektedir. Bu nedenle, durağan olmayan sinyaller için standart Fourier Dönüşümü kullanışlı değildir. Çünkü durağan olmayan sinyaller, sabit

bir periyoda sahip değildir ve ani değişimler içerir. Kısa Zamanlı Fourier Dönüşümü ise zaman hakkında bilgi edinmek için geliştirilen bir yöntemdir. Kısa Zamanlı Fourier Dönüşümü aşağıdaki denklemde görülmektedir.

$$KZFD\{f(t)\} \equiv F(\tau, \omega) = \int_{-\infty}^{\infty} f(t) \cdot \omega(t - \tau) \cdot e^{-j\omega t} dt \quad (3.2)$$

Denklem (3.2)'de görüldüğü gibi pencereleme ana fonksiyonu $\omega(t - \tau)$ olarak seçilmiştir. Kısa Zamanlı Fourier Dönüşümü hem frekans tabanlıdır hem de zamana bağlıdır. Zaman hakkında bilgi sahibi edinilebilir ancak bu bilgi $\omega(t - \tau)$ pencere aralığı ile sınırlıdır.

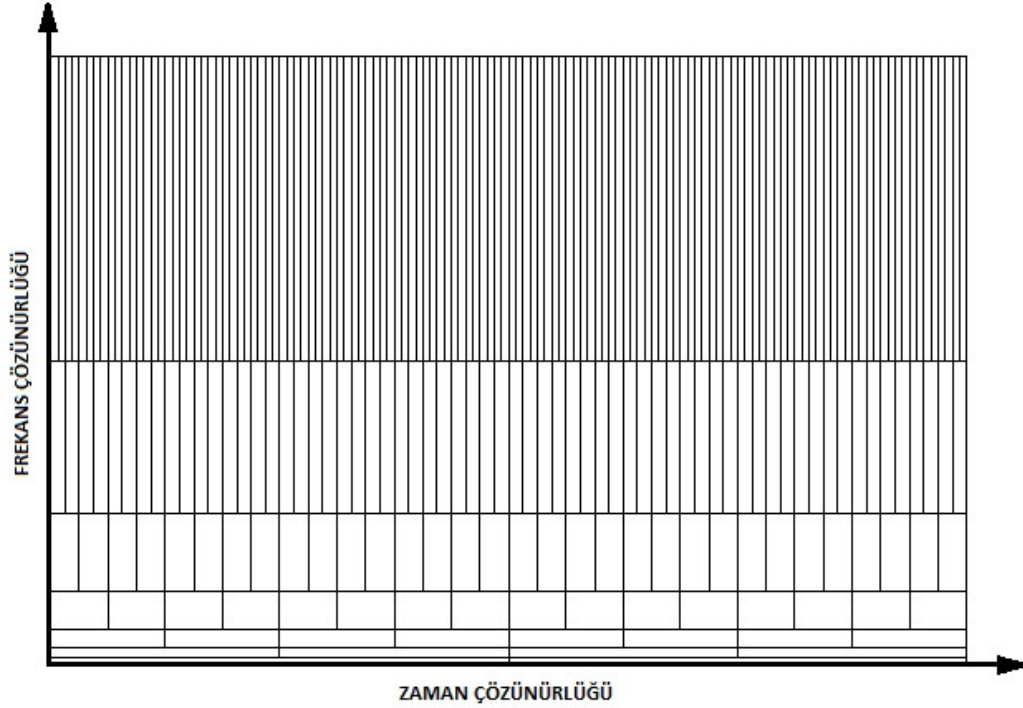


Şekil 3.2 : KZFD'nin taban fonksiyonu ve zaman frekans düzlemi;
(a) KZFD için zaman frekans çözünürlük kaplaması, (b) KZFD için taban fonksiyonları

Dalgacık Dönüşümü'nde zaman ve frekans çözünürlükleri Δt ve Δf 'i sabit tutmak yerine, her iki çözünürlüğü, çok çözünürlüklü bir analiz elde etmek üzere zaman-frekans düzleminde değişken olarak kullanabiliriz. Heisenberg eşitsizliği kuralına tabi olduğumuzu unutmamak gerekir. Frekans arttıkça zaman çözünürlüğü artmalı, frekans azaldıkça frekans çözünürlüğü artmalıdır [84].

Süzgeç bankası terminolojisiyle yaklaşılıcak olursa, analiz filtre bankası, sabit göreceli band genişlikli band geçiren filtrelerden oluşur. Bu yaklaşımla oluşturulan zaman-frekans düzleminin çözünürlüğü Şekil 3.3'de gösterilmiştir. Filtre bankasındaki analiz filtrelerinin frekans tepkileri logaritmik ölçeğe göre düzenli bir biçimde yerleştirilir. Bu filtreler doğal durumda oktavlara dağıtılır.

Bu yaklaşımla, yüksek frekanslarda seçime bağlı zaman çözünürlüğü iyi ve seçime bağlı frekans çözünürlüğü de düşük frekanslarda iyi olur. Sonuç olarak iki çok yakın sıçrama, zaman çözünürlüğü artacak şekilde yüksek frekanslara gidildiğinde ayrıştırılabilir. Dalgacık analizi en iyi, kısa süreli yüksek frekanslı bileşenler ve uzun süreli düşük frekanslı bileşenlerden oluşan sinyallerde başarılı olur. Farklı frekanslarda çözünürlüğü değiştirme yaklaşımı dalgacık paketleri olgusu anlatılarak açıklanabilir.



Şekil 3.3 : Sabit Q'lu filtre bankasının zaman frekans çözünürlüğü

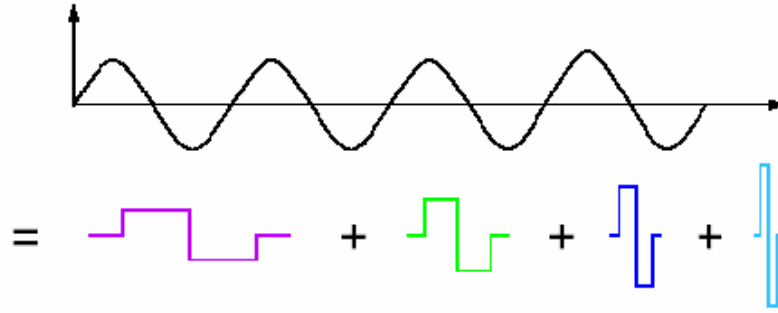
Heisenberg belirsizlik ilkesine bağlı kalarak, sinyale göre rastgele zaman frekans çözünürlüğü seçilebilir. Bu durumda, filtre bankasındaki analiz filtrelerinin tümünün dürtü tepkileri aynı $\psi(t)$ prototipinin sıkıştırılarak yada genişletilerek ölçeklenmiş biçimleridir. Örneğin ölçekleme faktörü a için, filtre dürtü tepkisi,

$$\psi^a(t) = \left[\frac{1}{\sqrt{|a|}} \right] \psi \left[\frac{t}{a} \right] \quad (3.3)$$

$\frac{1}{\sqrt{|a|}}$ sabiti normalizasyon amacıyla kullanılır. Bu formüle göre SDD,

$$W(a, b) = \left(\frac{1}{\sqrt{|a|}}\right) \int f(t) \bar{\psi}\left(\frac{t-b}{a}\right) dt \quad (3.4)$$

şeklinde tanımlanır [85]. Denklem (3.4)'de $\psi(t)$ ana dalgacık fonksiyonu, a: ölçek (frekans) parametresi, b: zaman (konum) parametresidir. Yüksek frekans analizlerinde kısa pencere seçimleri yeterli olup, alçak frekans analizlerinde ise uzun pencere aralıkları seçilmelidir. Aynı $\psi(t)$ prototipi, tüm dürtü tepkileri için kullanılır. Hiçbir ölçekte farklı olmayan bu prototipe, temel dalgacık denir [86].



Şekil 3.4 : Dalgacık yönteminin temeli

Şekil 3.4'de dalgacık yönteminin temeli görülmektedir. Fourier tekniğinin aksine, bir sinüs işareti çok sayıda frekansı ve genliği farklı kare dalgaların toplamından meydana gelmektedir. Sürekli Dalgacık Dönüşümü (Continuous Wavelet Transform, CWT) sonsuz sayıda girişe ihtiyaç duyduğundan bilgisayar analizi için uygun değildir. Bilgisayar analizleri için denklem (3.5)'de görülen Ayrık Dalgacık Dönüşümü (Discrete Wavelet Transform, DWT) kullanılır.

$$Wf(m, n) = 2^{-m/2} \int f(t) \varphi(2^{-m}t - n) dt \quad (3.5)$$

Denklem (3.5)'de; m parametresi frekansı belirler ve n parametresi ise konumu (zamanı) belirler. Herhangi bir sürekli fonksiyon, verilen bir ölçekleme fonksiyonu ve onun dalgacık türevleri cinsinden gösterimi aşağıdaki dalgacık serileri gibidir:

$$f(t) = \sum_{k=-\infty}^{+\infty} C_k \phi(t - k) + \sum_j \sum_k d_{j,k} \varphi(2^j t - k) \quad (3.6)$$

Bu açılımdaki ilk toplam $f(t)$ 'nin düşük çözünürlüklü veya kaba yakınsaması olan bir fonksiyon verir. İkinci toplamda her artan j değeri için, daha yüksek veya hassas çözünürlüklü bir fonksiyon eklenir.

$$\phi(x) = \sqrt{2} \sum_n h_0 \phi(2x - n), \text{ölçek fonksiyonu} \quad (3.7)$$

$$\varphi(x) = \sqrt{2} \sum_n h_1 \varphi(2x - n), \text{dalgacık fonksiyonudur} \quad (3.8)$$

(3.7) eşitliğinde h_0 katsayılar kümesi ölçekleme fonksiyonu katsayılar kümesi olarak adlandırılır (veya ölçekleme filtresi ya da ölçekleme vektörü). $\sqrt{2}$, ölçekleme fonksiyonunun normunu iki ölçeği ile sabitler. Eşitlik (3.7) ölçekleme fonksiyonlar teorisinin temelidir ve literatürde çeşitli isimlerle adlandırılır. Rafine denklem, çok çözünürlüklü analiz denklemi veya açılma (dilation) denklemi isimleri kullanılır. Bir dalgacık sisteminin tasarımı, h_0 katsayılar kümesinin seçilmesidir. (3.8) eşitliğinde h_1 katsayılar kümesi dalgacık fonksiyon katsayıları (veya dalgacık filtresi) olarak isimlendirilir.

Dalgacık katsayıları belirli koşullar altında, tıpkı Fourier Seri katsayılarının kullanılmasının benzeri bir şekilde, orijinal fonksiyonun analizi, tanımlanması, yakınsaması ve filtrelenmesi amacıyla kullanılabilir. Eğer ölçekleme fonksiyonu iyi ayarlanmışsa, bu durumda yüksek ölçeklerde, sinyalin örnekleri örnekleme katsayılarına çok yakındır.

Fourier dönüşümü yalnızca sin ve cos fonksiyonlarına ihtiyaç duymaktadır. Dalgacık dönüşümleri sonsuz sayıda ana fonksiyonlara sahiptir. Bu yüzden Fourier dönüşümleri durağan sinyal analizleri için oldukça iyi bir yaklaşımdır. Sonuç olarak dalgacık dönüşümleri geçici sinyal analizi için oldukça hassas olup, sinyallerdeki süreksizlik analizleri bu yolla çok daha güvenilir yapılmaktadır.

3.1.3. Gabor Dalgacıklar

Gabor fonksiyonları sahip oldukları iyi vasıf nedeni ile imgeyi hazırlama, anlama ve tanıma aşamalarında kullanılmaktadır. Fourier analizi ile belirlenemeyen, yerel uzay ve frekans tabanındaki bilgi, Gabor dalgacıklar ile ayrıştırılabilmektedir. Gabor fonksiyonu

[87], elips biçimli bir Gauss fonksiyonu ile çok yüzeyli bir dalganın çarpımından oluşur. Tek boyut için aşağıdaki gibi ifade edilir:

$$g_{\sigma, \omega_0}(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2} \frac{x^2}{\sigma^2}\right) \exp(j\omega_0 x) \quad (3.9)$$

İki boyut için de aşağıdaki gibi ifade edilir:

$$g_{\sigma, \omega_0}(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left[-\frac{1}{2} \left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right)\right] \exp[j\omega_0(x+y)] \quad (3.10)$$

σ : Gauss fonksiyonu için değerlendirme ölçütüdür, ω_0 : çok yüzeyli dalganın uzay frekansıdır. Gabor fonksiyonu tek (g^o) ve çift (g^e) kısımlarına ayırırsak aşağıda yer alan eşitlikleri elde ederiz:

$$g_{\sigma, \omega_0}^e(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left[-\frac{1}{2} \left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right)\right] \cos(\omega_0(x+y)) \quad (3.11)$$

$$g_{\sigma, \omega_0}^o(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left[-\frac{1}{2} \left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right)\right] \sin(\omega_0(x+y)) \quad (3.12)$$

$\sigma\omega_0 \approx 1$ sağlandığında, Gabor fonksiyonunun çift kısmı kenar belirlemede daha başarılı olmaktadır [88]. Gabor fonksiyonu bağımlı ve bağımsız ortogonal taban fonksiyonlarından oluşur, verilen fonksiyon bu fonksiyon grubu üzerine ayrıştırıldığında, belirli bir yerel frekans tanımlayıcı metod karşımıza çıkar. Bu işlem Gabor Dönüşümü olarak isimlendirilir. $\omega_0 = 1$, $\sigma = 1$ kabul edildiğinde, Gabor fonksiyonunun tek kısmı ölçek, hareket ve dönüş parametrelerinin değişimi süresince ana dalgacık gibi davranır. Gabor dalgacık, her zaman ortogonal değildir, bu nedenle Gabor'un katsayıları hesaplanırken, ikil ortogonal dalgacık yaklaşımı uygulanır.

$\{\psi_i\}$ $\{\tilde{\psi}_j\}$ iki grup dalgacık olsun, tüm i ve j değerlerinde $\langle \psi_i, \tilde{\psi}_j \rangle = \delta_{ij}$ çiftlikgenlik koşulu sağlanıyorsa söz konusu iki grup dalgacık ortogondur. $\tilde{\psi}$ ve ψ çift dalgacıktır. Çiftlikgenlik koşulu itibari ile, dalgacık katsayısı w_k , dual dalgacık $\tilde{\psi}_k$ ile $w_k = \langle f, \tilde{\psi}_k \rangle$ eşitliğinden elde edilir.

Dalgacık Dönüşümü ile imgeye ait kenar, şekil, açı gibi niteliklerin elde edilmesi mümkündür. Gabor fonksiyondan istifade edilerek elde edilmiş Gabor dalgacık,

$$\begin{aligned}
\psi_n(x, y) = \exp \left(-\frac{1}{2} \left(\left[\frac{1}{s_x} \left((x - c_x) \cos \theta - (y - c_y) \sin \theta \right) \right]^2 \right. \right. \\
\left. \left. + \left[\frac{1}{s_y} \left((x - c_x) \sin \theta + (y - c_y) \cos \theta \right) \right]^2 \right) \right) \\
\times \sin \left(\frac{1}{s_x} \left((x - c_x) \cos \theta - (y - c_y) \sin \theta \right) \right)
\end{aligned} \tag{3.13}$$

eşitlik (3.13)'de görülmektedir. $n = (c_x, c_y, \theta, s_x, s_y)^T$ dalgacığa ilişkin parametre vektörüdür. Burada c_x ve c_y konumdaki hareket parametresi, θ yönlendirme parametresi, s_x ve s_y ölçek parametresidir. Dalgacık katsayısı w_k , dual dalgacık $\vec{\psi}_k$ ile $w_k = \langle f, \vec{\psi}_k \rangle$ eşitliğinden elde edilir [89].

3.2. PARÇACIK SÜRÜ OPTİMİZASYONU

Biyolojik sistemlerden esinlenilerek geliştirilen birçok yöntem hesaba dayalı problemlerin çözümünde kullanılmaktadır. İnsan beyninin basitleştirilmiş bir modeli olan yapay sinir ağları ile evrimden esinlenerek geliştirilen genetik algoritma bu yöntemlere örnek olarak gösterilebilir. Sosyal sistemlerden esinlenerek geliştirilmiş yöntemler de mevcuttur. Bu yöntemlerde ise sistemi meydana getiren bireylerin kendi çevreleri ile etkileşimi, sistemdeki diğer bireyler ile iletişimi ve bireylerin ortak davranışları incelenip modellenmiştir. Bu davranışlar sürü zekası olarak isimlendirilmektedir.

Hesaplama alanında kullanılan ve sürülerden esinlenilerek geliştirilmiş iki popüler metot; Karınca Koloni Optimizasyonu (KKO) ve Parçacık Sürü Optimizasyonu (PSO)'dur. KKO karıncaların davranışları modellenerek geliştirilmiş bir yöntemdir ve ayrık optimizasyon problemlerinde başarı ile uygulanmıştır [91]. PSO ise sosyal hayatın basitleştirilmiş bir benzetimidir ve zamanla optimizasyon yöntemi olarak hesaba dayalı problemlerin çözümünde kullanılmaya başlanmıştır.

Parçacık Sürü Optimizasyonu (Particle Swarm Optimization) (PSO) son yıllarda kullanılan en yeni akıllı hesap yöntemlerinden biridir. PSO karmaşık denklem takımı

içeren doğrusal olmayan problemlerde başarı ile kullanılmaktadır. Klasik optimizasyon yöntemlerinden en önemli farkı türev kullanmamasıdır.

Parçacık Sürü Optimizasyonu (PSO) 1995'te Dr. Eberhart ve Dr. Kennedy tarafından geliştirilmiş popülasyon temelli sezgisel bir optimizasyon tekniğidir. Kuş veya balık sürülerinin iki boyutlu davranışlarından esinlenilerek geliştirilmiş popülasyon tabanlı bir optimizasyon tekniğidir [90].

Doğrusal olmayan sürekli fonksiyonların optimizasyonu için geliştirilmiş Parçacık Sürü Optimizasyonu (PSO), basitleştirilmiş sosyal bir modelin benzetimi esnasında bulunmuştur. Kökleri iki ana metodolojiye dayanmaktadır; genelde Yapay yaşam (Artificial life (A-life)), ayrıntıda Kuş sürüsü (Bird Flocking), Balık sürüsü (Fish Schooling) ve Sürü Teorisi (Swarm Theory). PSO, ayrıca Evrimsel Hesaplama ile ilintili ve hem Genetik Algoritma hem de Evrimsel Programlama ile bağlantılıdır.

Dr. Eberhart ve Dr. Kennedy tarafından geliştirilen PSO, basit bir fikir ve bir kaç satır bilgisayar kodu ile gerçeklemesi mümkün bir metoddan oluşmaktadır. Sadece temel matematik operatörlerine ihtiyaç dumakta ve bellek gereksinimi ve hız bakımından oldukça masrafsızdır.

3.2.1. Sosyal Davranışın Benzetimi

Bilimadamları kuş sürüsü veya balık sürüsünü oluşturan bireylerin sürü içerisindeki hareketlerini farklı yaklaşımlar ile bilgisayar ortamında modellemiş bulunmaktadır. Reynolds [93] ile Heppner ve Grenander'in [92] kuş sürüleri benzetim çalışmaları dikkat çekicidir. Kuş sürülerinin hareket düzenindeki estetik, Reynolds'un merakını uyandıran. Heppner'in bir hayvan bilimcisi olarak dikkatini çeken ise, büyük sayıda kuşların meydana getirdiği sürünün senkronize hareketleri, ani istikamet değişimine çabuk adapte olmaları, saçılmaları akabinde tekrar bir araya gelmeleri ve bu davranışın altında yatan kurallardır.

Her iki bilimadamı da hücrel otomasyon tarafından modellenen yerel süreçlerin, kuşun sosyal davranışının sezilemeyen grup dinamikleri altında yattığı anlayışına sahiptir. Her iki model de bireyler arası mesafenin değişimine dayalıdır. Sürünün

eşzamanlı davranışı, kuşların kendileri ve komşuları ile aralarındaki optimum mesafeyi sağlamak için harcadıkları eforun fonksiyonu olarak düşünülebilir.

“Teoride sürüyü oluşturan bireyler, sürüdeki diğer bireylerin yiyecek araştırırken edindiği tecrübelerden ve keşiflerden faydalanabilir. Bu menfaat, yiyecek kaynağının arazi üzerindeki dağılımı sezilemediğinde ve yiyecek maddeler için rekabetin dezavantajlarının ağır bastığı durumlarda belirleyici olabilir.” Sosyobiolog E. O. Wilson’ın [94] balık sürülerine ilişkin yazısında da dile getirildiği gibi, bireylerin evrimi boyunca bilginin sosyal paylaşımı evrimsel avantaj sağlamaktadır. Bu hipotez, parçacık sürü optimizasyonunun gelişiminde temel teşkil etmiştir.

Daha önce de belirtildiği gibi Parçacık Sürü Optimizasyonu (PSO), Kuş veya balık sürülerinin iki boyutlu davranışlarından esinlenilerek geliştirilmiş populasyon tabanlı bir optimizasyon tekniğidir. Bu iki boyutlu davranışlar; çevrelerine adapte olabilmek, zengin yiyecek kaynakları bulabilme ve avcılardan kaçabilme gibi ‘bilgi paylaşma’ yaklaşımı gerektirmektedir. Bilgi paylaşımı sayesinde sürüler, yiyecek ararken ya da avcıdan kaçarken, hedefe en yakın olan sürü elemanını takip ederler ve kendi hızlarını ve konumlarını en başarılı elemana göre güncellerler.

Balık sürüsü yada kuş sürüsünü modellemek ile birebir örtüşmese de, benzetimin geliştirilmesinde bir neden insan davranışını modellemektir. Kuşlar ve balıklar, eş ve yiyecek aramak, yırtıcılardan korunmak ve sıcaklık gibi ortam değişkenlerini optimize etmek için bedensel hareketlerini ayarlarlar. İnsanlar ise sadece bedensel hareketlerini ayarlamaz ayrıca bilişsel ve tecrübeye dayalı değişkenlerini günceller. Bizler genelde uygun adımda yürüyüp ve birlikte eşzamanlı dönüş yapmayız (bunu yapabilme yeteneğine sahip olsak da), daha ziyade inanış ve görüşlerimizi sosyal grubumuzdaki üyeler ile rahatımızı sağlayacak şekilde uyarlama eğilimindeyizdir.

Bu durum, bilgisayarda benzetimin başarımı açısından büyük bir fark oluşturur, en belirgin nedeni ise çarpışmadır. İki kişi birbiri ile çarpışmadan aynı görüş ve inanışa sahip olabilir ancak iki kuşun uzayda aynı noktada çarpışmadan bir arada olması mümkün değildir. İnsanın sosyal davranışının incelenmesinde, balık/kuş hareketi analogisi içerisine gerekli değişimin uyarlanması mantıklı görünüyor. Hareket çeşitleri

gibi nicel ve nitel deęişimler klasik Aristocu görüő ile tutarlıdır. Üç boyutlu fiziki uzayda hareket etmenin yanında çarpışmaları önlemek için çarpışmadan bağımsız çok boyutlu soyut uzayda insanlar deęişirler. Fiziki uzay elbette bilgi girişlerini etkilemektedir ancak psikolojik tecrübe açısından deęersiz bir bileően olduęu tartışılmazdır. İnsanlar küçük yaşlarda iken çarpışmamayı öğrenirler ancak n boyutlu psikososyal uzayda durum deęişmekte, burada onlarca deneyime ihtiyaç vardır. Zaten pek çoęumuz da arzu ettięimiz yeteneklere tam olarak ulaşamadıęımızın farkındayız.

3.2.2. Parçacık Sürü Optimizasyonunun Uygulama Alanları

Literatür incelendięinde görülür ki, Parçacık Sürü Optimizasyonu pek çok alanda başarı ile uygulanmıştır. Literatürde mevcut PSO uygulamalarına birkaç örnek:

- Görüntü işleme alanında PSO tabanlı bir imge bölütleme yaklaşımı önerilmiştir [95]. Mevcut görüntüye ait histogram, Gauss olasılık fonksiyonlarından oluşan bir karışım ile oluşturulmuş ve PSO algoritması ile olasılık fonksiyoları karışımının parametreleri hesaplanmıştır. Böylece yoğunluk fonksiyonu ile gerçek histogram arasındaki hata en aza indirilmiştir.
- Sınırlamalar içeren bazı mekanik tasarım problemlerinde de kullanılmıştır [96]. Bu problemlerde minimize edilecek fonksiyonun yanısıra fonksiyonların parametrelerini sınırlayan başka denklem takımları da mevcuttur. Hayli karmaşık bu problemlerin PSO ile çözümü dięer yöntemlerle karşılaştırılmış ve PSO'nun en avantajlı ve başarılı yöntem olduęu saptanmıştır.
- Güç sistemlerinde, lineer olmayan optimizasyon problemi olarak tanımlanmış reaktif güç ve gerilim kontrol (Voltage/VAR control) problemine uygulanmıştır [97]. Burada minimize edilecek durum, gerilim kararlılıęının sağlanması koşulu ile hedef güç sistemindeki toplam güç kaybıdır.
- FPGA yerleőtirme ve yönlendirmesinde kullanılmıştır [98]. Algoritma, yapılandırılabilir lojik bloklar (CLB) arasındaki bağlantı uzunluklarının minimize edilmesini sağlamıştır.
- Şebekeden bağımsız fotovoltaik sistem için tasarlanan bulanık lojik denetleyicisinin optimizasyonunda kullanılmıştır [99]. Denetleyicinin tasarımında kullanılan üyelik fonksiyonunu ve kural kümesini optimize etmiştir.

- Hareketli sensörler veya robotlar sürüsünün haberleşmesinde; yangın söndürme, mayın bulma ve radyoaktivitenin belirlenmesi gibi yerel ve küresel işlerin azami uygulanması için kullanılmıştır [100].

3.2.3. Parçacık Sürü Optimizasyonu

Daha evvel belirtildiği gibi PSO, kuş sürülerinin davranışlarının bir benzetimidir. Kuşların uzayda yeri bilinmeyen bir yiyeceği araması, bir probleme çözüm aramaya benzetilir. Kuşlar yiyecek ararken yiyeceğe en yakın kuşu takip ederler. Parçacık olarak adlandırılan her tekil çözüm, arama uzayındaki bir kuş gibi düşünülebilir. Parçacık hareket ettiğinde, kendi koordinatlarını bir fonksiyona gönderir ve böylece parçacığın uygunluk değeri ölçülmüş olur. (yani yiyeceğe ne kadar uzaklıkta olduğu ölçülmüş olur.) Bir parçacık, koordinatlarını, hızını (çözüm uzayındaki her boyutta ne kadar hızla ilerlediği), şimdiye kadar elde ettiği en iyi uygunluk değerini ve bu değeri elde ettiği koordinatları hatırlamalıdır. Çözüm uzayındaki her boyuttaki hızının ve yönünün her seferinde nasıl değişeceği, komşularının en iyi koordinatları ve kendi kişisel en iyi koordinatlarının bir birleşimi olacaktır.

Çözüm uzayı problemdeki değişken veya bilinmeyen sayısına bağlı olarak çok boyutta olabilir. Görsel olarak insanların resmetmekte zorlandığı 4 ve daha fazla boyutlu karmaşık problemlerde çalışmanın PSO için herhangi bir zorluğu bulunmamaktadır. Örneğin PSO, bir yapay sinir ağının eğitiminde kullanılacaksa ve ağda 50 tane bağlantı ağırlığı mevcutsa problem 50 boyutlu bir uzayda çözülecektir.

PSO, bir grup rasgele çözümle (parçacık sürüsü) başlatılır ve güncellemelerle optimum çözüm bulunmaya çalışılır. Her tekrarlama (iterasyonda), parçacık konumları, iki en iyi değere güncellenir. İlki; o ana kadar parçacığın elde ettiği en iyi çözümü sağlayan koordinatlarıdır. Bu değer "*pbest*" olarak adlandırılır ve hafızada saklanmalıdır. Diğer en iyi diğer ise, popülasyonda o ana kadar tüm parçacıklar tarafından elde edilen en iyi çözümü sağlayan koordinatlarıdır. Bu değer global en iyidir ve "*gbest*" ile gösterilir. Örneğin D adet parametreden oluşan n adet parçacık olduğunu varsayalım. Bu durumda popülasyon aşağıdaki gibi ifade edilir.

$$x = \begin{bmatrix} x_{11} & x_{12} & \dots & \dots & x_{1D} \\ x_{21} & x_{22} & \dots & \dots & x_{2D} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ x_{n1} & x_{n2} & \dots & \dots & x_{nD} \end{bmatrix}_{n \times D} \quad (3.14)$$

Yukarıdaki matriste, i 'ninci parçacık $x_i = [x_{i1}, x_{i2}, \dots, x_{iD}]$ olarak ifade edilir. Önceki en iyi uygunluk değerini veren i 'ninci parçacığın pozisyonu $pbest_i = [p_{i1}, p_{i2}, \dots, p_{iD}]$ olarak ifade edilir. $gbest$ ise her iterasyonda tüm parçacıklar için tektir ve $gbest = [p_1, p_2, \dots, p_D]$ şeklinde gösterilir. i 'ninci parçacığın hızı (her boyuttaki konumunun değişim miktarı) $v_i = [v_{i1}, v_{i2}, \dots, v_{iD}]$ olarak ifade edilir. İki en iyi değer bulunmasından sonra parçacık hızları ve konumları aşağıda verilen (3.15) ve (3.16) eşitliklerine göre güncellenir.

$$v_i^{k+1} = v_i^k + c_1 \cdot rand_1^k \cdot (pbest_i^k - x_i^k) + c_2 \cdot rand_2^k \cdot (gbest^k - x_i^k) \quad (3.15)$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \quad (3.16)$$

Denklem (3.15)'de, c_1 ve c_2 öğrenme faktörleridir. c_1 ve c_2 , her parçacığı $pbest$ ve $gbest$ pozisyonlarına doğru çeken, stokastik hızlanma terimlerini ifade eden sabitlerdir. c_1 , parçacığın kendi tecrübelerine göre hareket etmesini, c_2 ise sürüdeki diğer parçacıkların tecrübelerine göre hareket etmesini sağlar. Düşük değerler seçilmesi parçacıkların hedef bölgeye doğru çekilmeden önce, bu bölgeden uzak yerlerde dolaşmalarına imkan verir. Ancak hedefe ulaşma süresi uzayabilir. Diğer yandan, yüksek değerler seçilmesi, hedefe ulaşmayı hızlandırırken, beklenmedik hareketlerin oluşmasına ve hedef bölgenin es geçilmesine sebep olabilir. Bu algoritma üzerinde araştırmacıların yaptığı denemelerde $c_1 = c_2 = 2$ olarak almanın iyi sonuçlar verdiği belirtilmiştir. Denklemdeki $rand_1$ ve $rand_2$, $[0,1]$ arasında düzgün dağılımlı rasgele sayılardır. k ise iterasyon sayısını belirtmektedir. Aşağıda PSO algoritması için gerekli olan prosedür özetlenmiştir.

For *her parçacık için başlangıç koşulları*

End

Do For *her parçacık için uygunluk değerini hesapla, eğer uygunluk değeri*

pbest'ten daha iyi ise; şimdiki değeri yeni pbest olarak ayarla

End

Tüm parçacıkların bulduğu pbest değerlerinin en iyisini, tüm parçacıkların gbest'i olarak ayarla

For *her parçacık için*

(3.15) deklemine göre parçacık hızını hesapla

(3.16) deklemine göre parçacık pozisyonunu güncelle

End

While *maksimum iterasyon sayısına veya minimum hata koşulu sağlanana kadar devam et*

3.2.3.1. PSO Parametre Kontrolü

PSO'da ayarlanması gereken çok sayıda parametre yoktur. Aşağıda parametreler ve tipik değerleri verilmiştir:

- **Parçacık sayısı:** 20 ile 40 arasındadır. Birçok problem için 10 parçacık kullanmak yeterlidir. Bazı zor veya özel problemlerde ise 100 veya 200 parçacık kullanılması gerekebilir.
- **Parçacık boyutu:** Optimize edilecek probleme göre değişmektedir.
- **Parçacık aralığı:** Optimize edilecek probleme göre değişmekle birlikte farklı boyutlarda ve aralıklarda parçacıklar tanımlanabilir.
- **Vmax:** Bir iterasyonda, bir parçacıkta meydana gelecek maksimum değişikliği (hız) belirler. Genellikle parçacık aralığına göre belirlenir. Örneğin X_1 parçacığı (-10, 10) aralığında ise $V_{max}=20$ ile sınırlandırılabilir.
- **Öğrenme Faktörleri:** c_1 ve c_2 genellikle 2 olarak seçilir. Fakat farklı da seçilebilir. Genellikle c_1, c_2 'ye eşit ve [0, 4] aralığında seçilir.
- **Durma Koşulu:** Azami iterasyon sayısına ulaşıldığında veya değer fonksiyonu istenilen seviyeye ulaştığında algoritma durdurulabilir.

PSO'nun şu andaki güncel versiyonu elde edilmeden önce çeşitli denemeler yapılmış ve bir çok yöntem ortaya atılmıştır. Algoritmanın başarımını arttırmak için Yuhui Shi ve R.C. Eberhart tarafından yapılan çalışmalar sunucunda 1998 yılında *Geliştirilmiş PSO* algoritması ortaya çıkarılmıştır [101,102].

3.2.3.2. Geliştirilmiş PSO

Bu versiyonun en önemli farklılığı; parçacığın, komşularının ve kendisinin en iyi pozisyonları arasında arama yapmamasıdır. Komşularının en iyi pozisyonu (*gbest*) ile tüm parçacıkların en iyi orta pozisyonu arasında arama yapılır. Yani elde edilen *pbest* değerlerinin ortalaması alınır.

$p_i^* = [p_{i1}^*, p_{i2}^*, \dots, p_{iD}^*]$ olmak üzere; bireyin en iyi değeri aşağıdaki gibi hesaplanır:

$$p_{ij}^* = (p_{1j} + p_{2j} + \dots + p_{nj}) / n \quad (3.17)$$

Denklem (3.17)'de $j = 1, 2, \dots, D$ olmak üzere n adet parçacığın her boyuttaki en iyi (*pbest*) değerlerinin ortalaması alınmakta ve yeni p_i^* vektörü elde edilmektedir. Buna göre (3.15) denklemi aşağıdaki biçime dönüşür.

$$v_i^{k+1} = w \cdot v_i^k + c_1 \cdot rand_1^k \cdot (p_i^{*k} - x_i^k) + c_2 \cdot rand_2^k \cdot (gbest^k - x_i^k) \quad (3.18)$$

Denklemde w eylemsizlik ağırlığıdır, $w < 1$ olarak seçilmeli ve her iterasyonda doğrusal olarak azaltılmalıdır. PSO'da eylemsizlik ağırlığı global ve yerel arama yeteneğini dengelemek için kullanılır. Büyük eylemsizlik ağırlığı global arama, küçük eylemsizlik ağırlığı ise yerel arama yapılmasını kolaylaştırır. Eylemsizlik ağırlığı yerel ve global araştırma arasındaki dengeyi sağlar ve bunun sonucunda yeterli optimal sonuca daha az iterasyonda ulaşılır. Buradaki her parçacık; sürüdeki sadece en iyi parçacığın değil sürüdeki diğer tüm parçacıkların tecrübelerinden de yararlanmış olur.

3.3. KARAR VERME

3.3.1. Temel Bileşen Analizi

Temel Bileşen Analizi (Principal Component Analysis, PCA) veya diğer adıyla Karhunen-Loève transformu işaret işleme sahasında yaygın olarak kullanılan istatistiksel hesaplama tekniğidir [39]. PCA'daki esas fikir, doğrusal olarak dönüştürülmüş bileşenlerle maksimum değişim miktarını ifade eden bileşenleri bulmaktır. Danışmansız bir metod olan PCA yüksek boyutlu verileri daha düşük

boyutlu verilere indirgemeye yarayan güçlü bir veri analiz tekniğidir. PCA kısaca şu şekilde özetlenebilir.

X , n adet d boyutlu x_i verisi içeren bir matris olsun,

$$X = [x_1, x_2, \dots, x_n] \quad (3.19)$$

İlk olarak n adet verinin ortalama değeri \bar{x} bulunur, ortalama değerle her bir x_i verisinin farkı alınarak \tilde{X} fark matrisi bulunur.

$$\tilde{X} = [x_1 - \bar{x}, x_2 - \bar{x}, \dots, x_n - \bar{x}] \quad (3.20)$$

Elde edilen fark matrisi aşağıdaki ifadede kullanılarak C kovaryans matrisi hesaplanır.

$$C = \frac{1}{n} \tilde{X} \tilde{X}^T \quad (3.21)$$

Aşağıdaki ifade doğrulanacak şekilde C kovaryans matrisinin özvektörleri, $\Phi_{\tilde{X}}$ ve özdeğerleri, $\Lambda_{\tilde{X}}$ bulunur.

$$C \Phi_{\tilde{X}} = \Phi_{\tilde{X}} \Lambda_{\tilde{X}} \quad (3.22)$$

Bu aşamayla elde edilen d adet d boyutlu özvektör ve köşegen matris biçiminde d adet özdeğer aşağıdaki gibidir.

$$\Lambda_{\tilde{X}} = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \lambda_d \end{bmatrix} \quad \Phi_{\tilde{X}} = [\phi_1, \phi_2, \dots, \phi_d] \quad (3.23)$$

m adet, yüksek özdeğere karşılık gelen özvektörler seçilerek $d \times m$ boyutlu iz düşüm matrisi W elde edilir.

$$x'_i = W^T (x_i - \bar{x}) \quad (3.24)$$

ifadesiyle d boyutlu x_i verisiyle daha düşük boyutlu x'_i elde edilir.

3.3.2. Öklid Mesafesi

x ve y ; d boyutuna sahip sütun vektörleri olmak üzere;

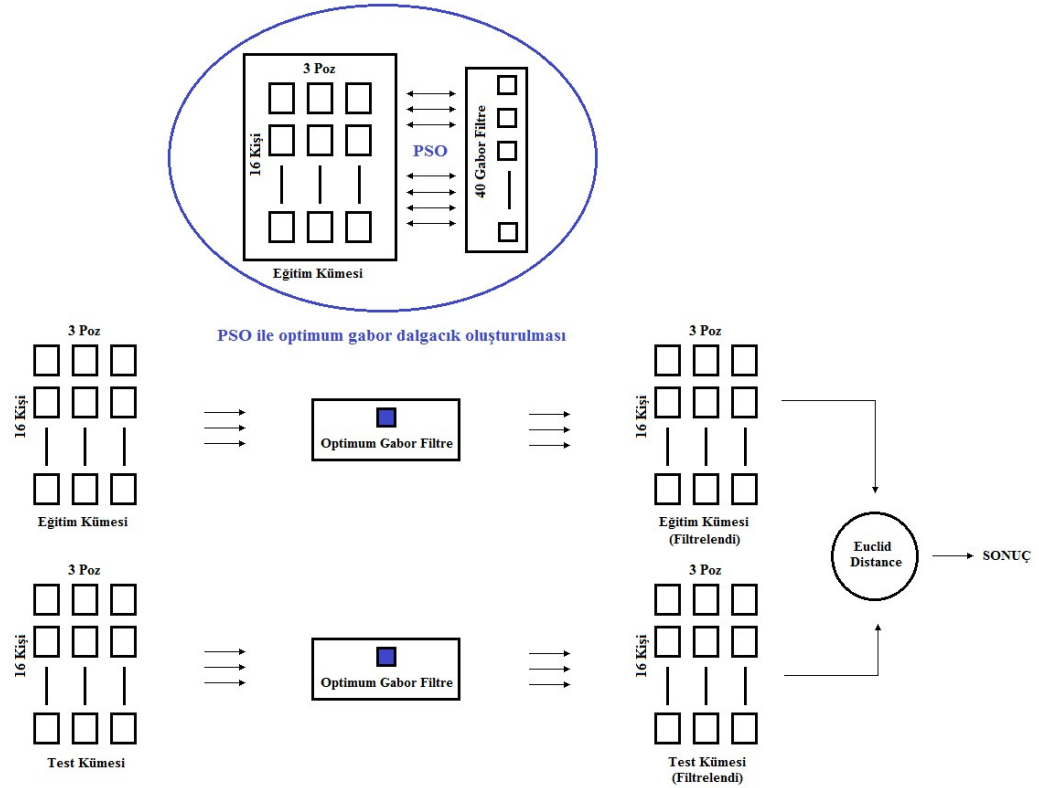
$$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_d \end{bmatrix} \quad \underline{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_d \end{bmatrix} \quad (3.25)$$

Öklid mesafesi $D(x,y)$;

$$D(x,y) = \sqrt{\sum_{i=1}^d |x_i - y_i|^2} \quad (3.26)$$

şeklinde ifade edilir [103, 104].

3.4. ÖNERİLEN YÖNTEM



Şekil 3.5 : Önerilen Yöntem

Çalışmamızda Olivetti Att – ORL veritabanına ait yüz imgeleri kullanılmıştır. Olivetti Att veritabanı 40 kişinin 10 farklı yüz imgesinden oluşmaktadır. Nitelik vektörlerinin eldesinde kullanılmak üzere 16 kişiye ait 3 farklı poz için seçilen resimler ile 48 yüz imgesinden oluşan bir eğitim kümesi elde edilmiştir. Test kümesini aynı 16 kişiye ait 3'er imge seçilerek oluşturulmuştur. Test kümesindeki imgeler ile eğitim kümesindeki imgeler birbirlerinden farklıdır.

Nitelik vektörü eldesinde kullanılan Gabor dalgacıkların PSO ile optimizasyonu aşağıda yer alan adımlardan oluşmaktadır:

1. 5 ölçek, 8 yönlendirme ile 40 adet parametre oluşturulur.
2. Parametreler ile (3.13) eşitliği uyarınca 40 adet Gabor Dalgacık üretilir.
3. Eğitim kümesini oluşturan imgeler 40 adet filtre ile filtrelenir.
4. Her poz için filtrelenmiş imgeler arası farklar Öklid Mesafesi ile hesaplanır.
5. 3 poz için de elde edilen fark değerlerini yükselten parametre ile pbest'i günceller. (Burada arzu edilen elde edilecek filtre ile farklı kişilere ait imgelerin filtreye verdikleri cevapların birbirlerinden farklı olması, yani kişi tepitinde belirleyici olmasıdır.)
6. Başlangıçtan mevcut iterasyona kadar en yüksek fark değerini sağlayan parametre ile gbest güncellenir.
7. (3.15) ve (3.16) denklemleri uyarınca hız ve konum güncellenir.
8. Maksimum iterasyon sayısına ulaşılmadıysa tekrar 2. adıma geçilir.

Maksimum iterasyon sayısına ulaşıldıktan sonra elde edilen $(c_x, c_y, \theta, s_x, s_y)$ parametresi ile Optimum Gabor Dalgacık oluşturulur.

Eğitim kümesinde ve Test kümesinde yer alan her bir imgeye ait nitelik vektörleri Optimum Gabor Dalgacık ile hesaplanır.

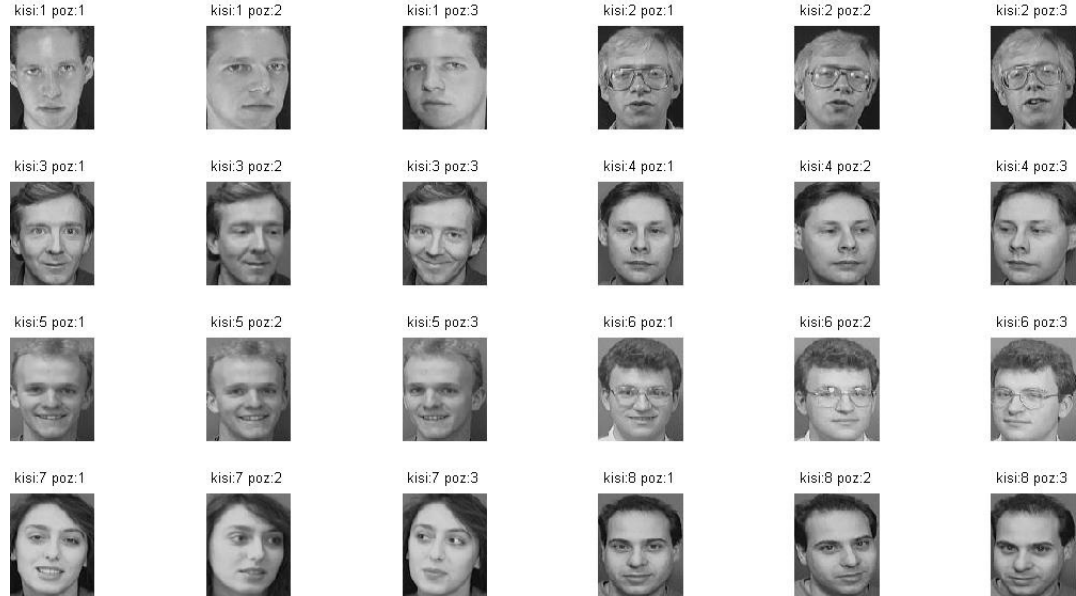
Elde edilen nitelik vektörleri ile kişi tespitinde iki farklı yöntemden faydalanılır.

- Öklid mesafesi ile vektörler arası uzaklıklar hesaplanarak test kümesindeki imgenin hangi bireye ait olduğu tespit edilir.
- Test ve eğitim kümesi bireyelerine ait nitelik vektörleri Temel Bileşen Analizi'ne tabi tutularak kişi tespiti yapılır.

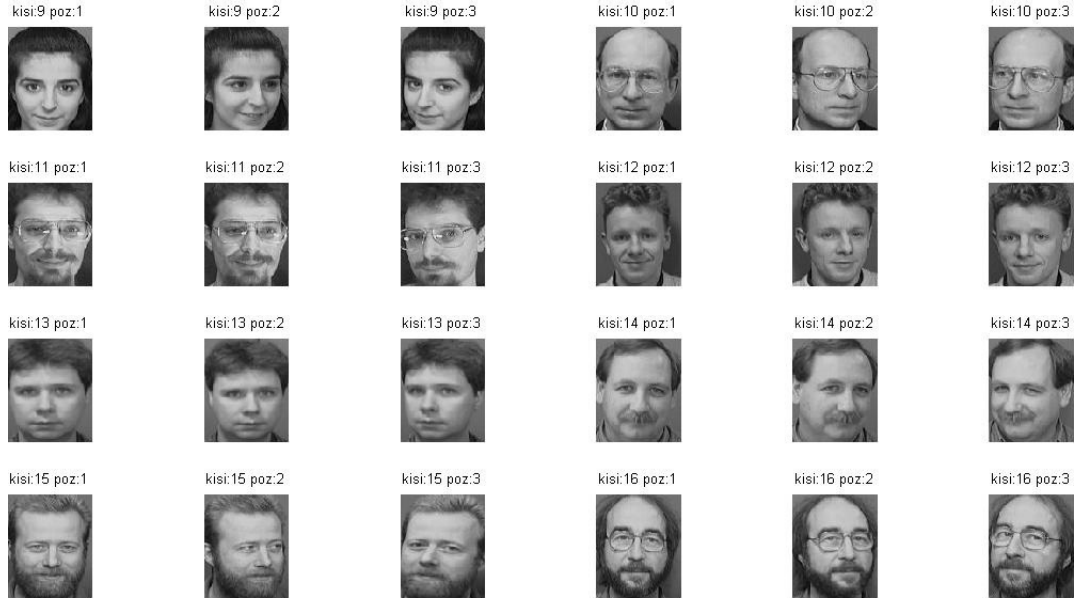
4. BULGULAR

Son on yıl içinde, tanıma oranı % 90'nın üzerine çıkan pek çok sistem ile yüz tanıma alanında büyük gelişmeler kaydedilmiştir. Yüz belirleme işleminin pek çok değişime tabi olması, gerçek dünya senaryolarının bir meydan okuması olarak devamına sebebiyet vermektedir. Aydınlanma (i), poz (p), yüz ifadesi (e), örtü (o) ve zamanda gecikme (t) gibi faktörler yüz tanıma sistemi başarımını etkilemektedir. Bu faktörlerin biri veya bileşimi durumunda önerilen yöntemlerin ne kadar iyi çalıştığını değerlendirmek için, birkaç farklı yüz imgesi veritabanı oluşturulmuştur.

Nitelik vektörlerinin eldesinde kullanılmak üzere 16 kişiye ait 3 farklı poz için Olivetti Att – ORL veritabanından seçilen resimler ile 48 yüz imgesinden oluşan bir eğitim kümesi elde edilmiştir. Şekil 4.1 ve Şekil 4.2'de eğitim kümemizi oluşturan yüz imgeleri görülmektedir.



Şekil 4.1 : Eğitim kümesi yüz imgeleri - 1



Şekil 4.2 : Eğitim kümesi yüz imgeleri - 2

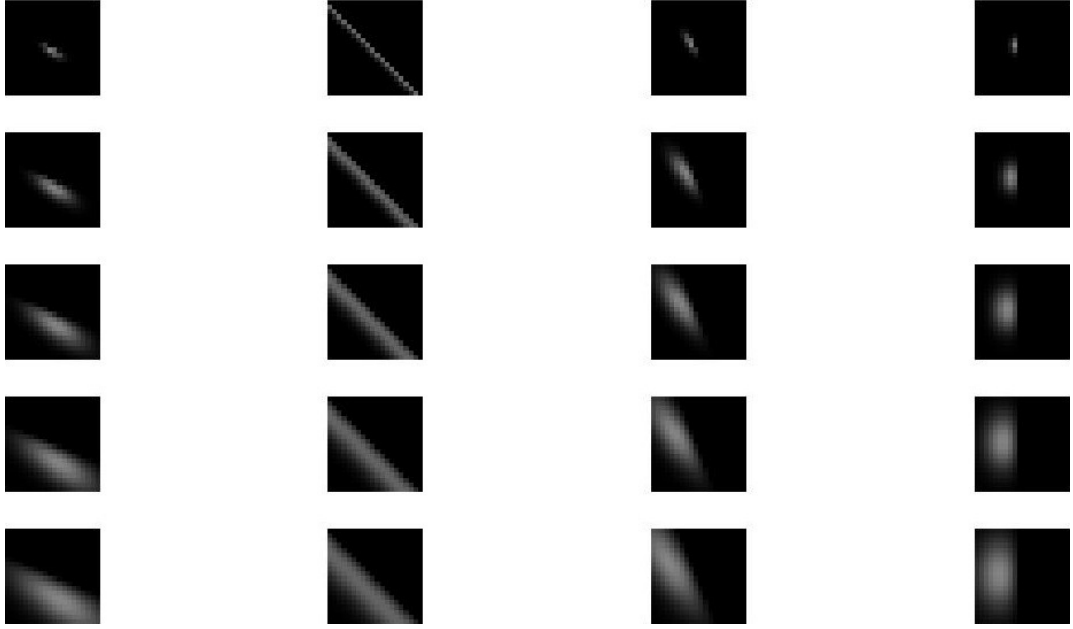
Malzeme ve Yöntem başlığı altında verdiğimiz bilgiler ışığında, yüz gösteriminde Gabor Dalgacıkların başarı ile kullanılmış olması, çalışmamızda bizim de Gabor Dalgacıkları uygulamamıza sebep teşkil etmiştir.

Dalgacık Dönüşümü ile imgeye ait kenar, şekil, açı gibi niteliklerin elde edilmesi mümkündür. Gabor fonksiyondan istifade edilerek elde edilen Gabor dalgacık,

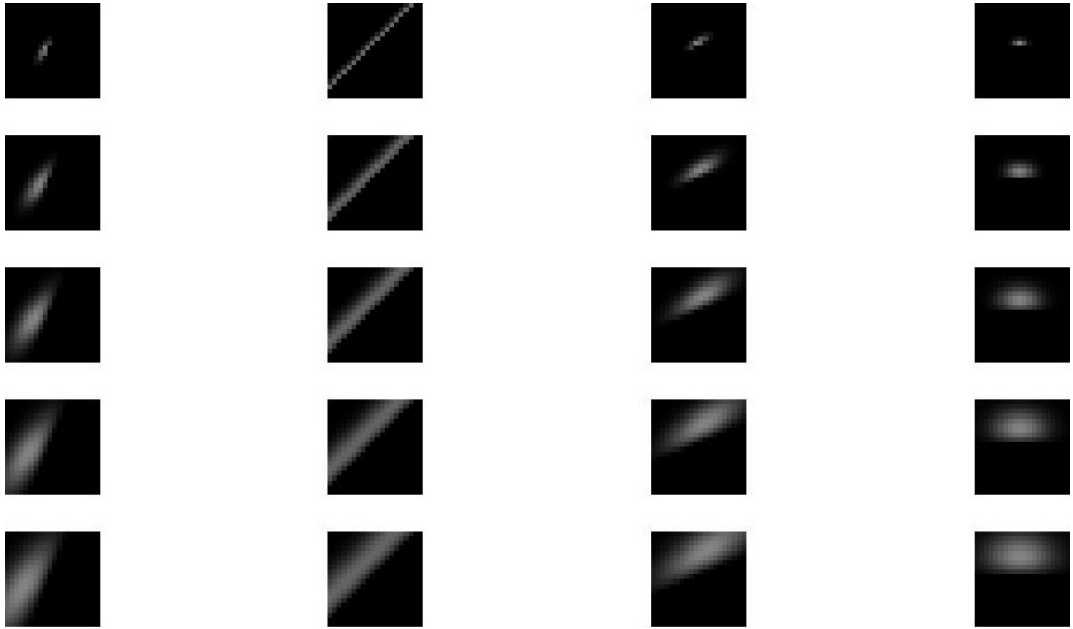
$$\begin{aligned} \psi_n(x, y) = \exp \left(-\frac{1}{2} \left(\left[\frac{1}{s_x} \left((x - c_x) \cos \theta - (y - c_y) \sin \theta \right) \right]^2 \right. \right. \\ \left. \left. + \left[\frac{1}{s_y} \left((x - c_x) \sin \theta + (y - c_y) \cos \theta \right) \right]^2 \right) \right) \\ \times \sin \left(\frac{1}{s_x} \left((x - c_x) \cos \theta - (y - c_y) \sin \theta \right) \right) \end{aligned} \quad (4.1)$$

eşitlik (4.1)'de görülmektedir. $n = (c_x, c_y, \theta, s_x, s_y)^T$ dalgacığa ilişkin parametre vektörüdür. Burada c_x ve c_y konumdaki hareket parametresi, θ yönlendirme parametresi, s_x ve s_y ölçek parametresidir.

Eşitlik 4.1 uyarınca oluşturulan 5 farklı ölçek ve 8 farklı açığa sahip 40 Gabor dalgacığı Şekil 4.3 ve Şekil 4.4'de görüldüğü gibidir.

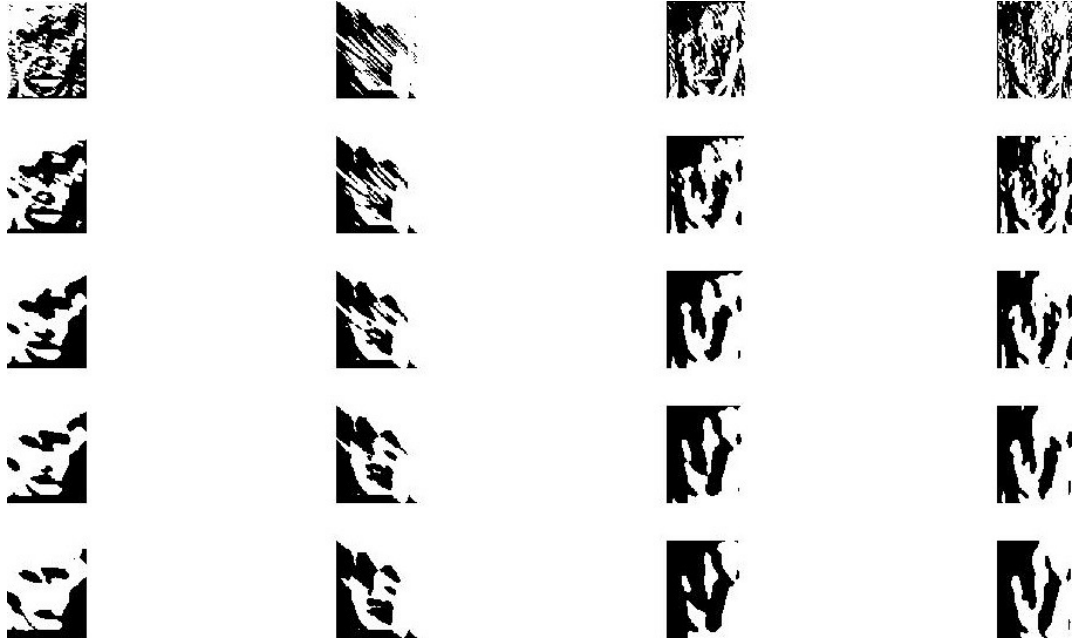


Şekil 4.3 : Gabor Dalgacıklar – 1 ($s = 1:5$, $\theta = \pi/8:\pi/8:\pi/2$)

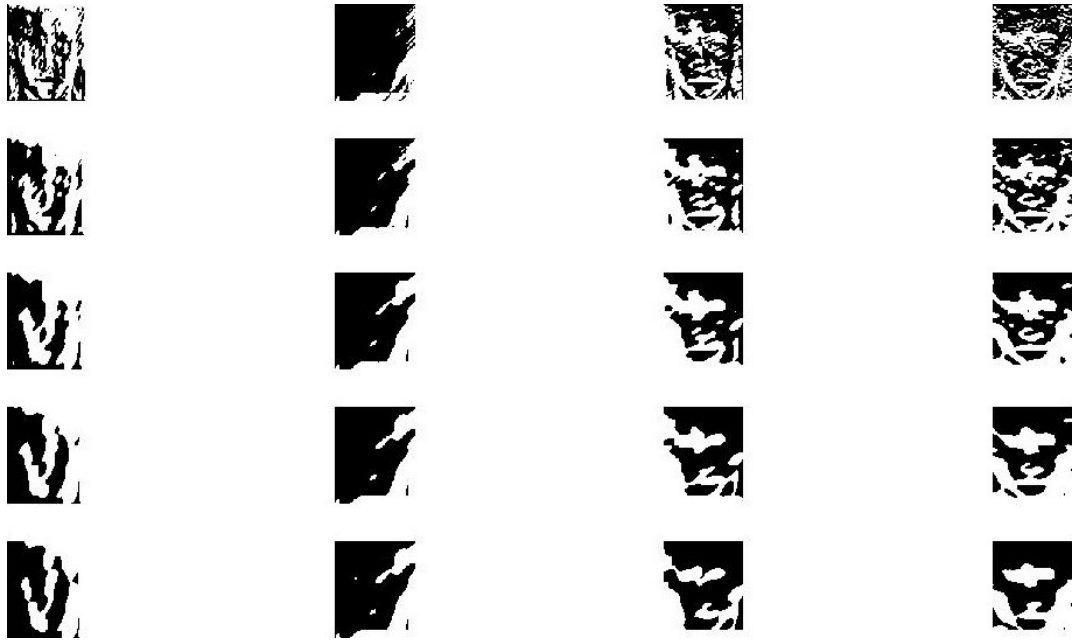


Şekil 4.4 : Gabor Dalgacıklar - 2 ($s = 1:5$, $\theta = 5\pi/8:\pi/8:\pi$)

Eğitim kümesinin Gabor dalgacık filtrelere cevabı birkaç açıdan incelenmiştir. Şekil 4.5 ve Şekil 4.6'da eğitim kümesinde yer alan 1. kişinin 1 numaralı resmine ilişkin Gabor filtre cevapları görülmektedir.



Şekil 4.5 : Bir kişiye ait resmin gabor filtrelere cevabı - 1 ($s = 1:5$, $\theta = \pi/8:\pi/8:\pi/2$)



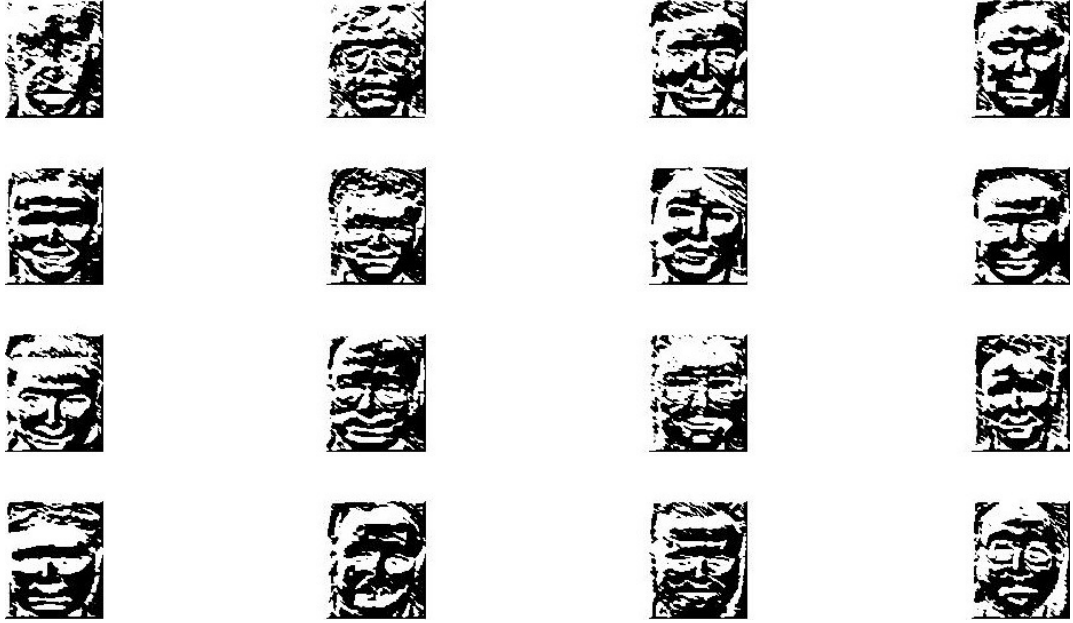
Şekil 4.6 : Bir kişiye ait resmin gabor filtrelere cevabı - 2 ($s = 1:5$, $\theta = 5\pi/8:\pi/8:\pi$)

Şekil 4.7'de ise aynı kişiye ait 3 farklı poz için, aynı filtreden elde edilen cevaplar görülmektedir.



Şekil 4.7 : Bir kişiye ait 3 farklı pozun gabor filtreye cevapları ($s = 1, \theta = \pi$)

Şekil 4.8'de, 16 kişinin aynı pozuna, aynı filtrenin cevapları görülmektedir.



Şekil 4.8 : 16 kişiye ait aynı pozun gabor filtreye cevapları ($s = 1, \theta = \pi$)

İlk olarak 5 ölçek 8 yönelim ile elde edilen Gabor filtreler ile her bir imgeye ait nitelik vektörleri oluşturulur. Eğitim ve test kümesini oluşturan imgeler 112x92 boyutlarındadır. Gabor filtre sayısının 40 olması, tek imge için elde edilen nitelik vektörünün uzunluğunu 412160 gibi bir değere ulaştırmaktadır. Dolayısı ile matematiksel işlemlerin süresi uzamaktadır. Bu nedenle, filtrelenen imgeler ölçeklenerek nitelik vektörü uzunluğu 25760'a düşürülmüştür.

Elde edilen nitelik vektörleri arasında Öklid uzaklığı hesaplanarak test kümesindeki imgelerin (bkz. Şekil 4.9 ve Şekil 4.10) eğitim kümesinde hangi kişilere denk geldiği

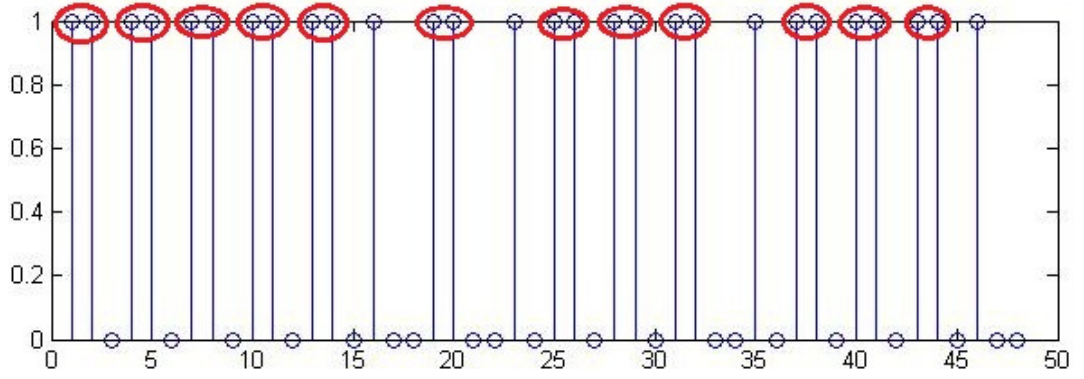
tespit edilmiştir. Şekil 4.11’de test imgesini oluşturan imgelerden hangilerinin doğru kişi ile eşlendiği görülebilir.



Şekil 4.9 : Test kümesi yüz imgeleri – 1



Şekil 4.10 : Test kümesi yüz imgeleri – 2



Şekil 4.11 : Test kümesi kişi eşleme doğruluk değerleri

Çalışmamızın ikinci aşamasında Parçacık Sürü Optimizasyonu ile Gabor Dalgacık parametrelerinin eniyilenmesi sureti ile nitelik vektörlerinin (filtre cevaplarının) optimizasyonu gerçekleştirilmiştir. Parçacık sürü optimizasyonuna ilişkin algoritma tezin 3. bölümünde detaylı biçimde açıklanmıştır. Ancak kısa bir hatırlatma salık verilir:

PSO, bir grup rasgele çözümle (parçacık sürüsü) başlatılır ve güncellemelerle optimum çözüm bulunmaya çalışılır. Her tekrarlama (iterasyonda), parçacık konumları, iki en iyi değere güncellenir. İlki; o ana kadar parçacığın elde ettiği en iyi çözümü sağlayan koordinatlardır. Bu değer "*pbest*" olarak adlandırılır ve hafızada saklanmalıdır. Diğer en iyi değer ise, popülasyonda o ana kadar tüm parçacıklar tarafından elde edilen en iyi çözümü sağlayan koordinatlardır. Bu değer global en iyidir ve "*gbest*" ile gösterilir. Beş adet parametreden oluşan 40 adet parçacığımız olduğunu varsayalım. Bu durumda popülasyon aşağıdaki gibi ifade edilir. $n = (c_x, c_y, \theta, s_x, s_y)$

$$x = \begin{bmatrix} c_{x1} & c_{y1} & \theta_1 & s_{x1} & s_{y1} \\ c_{x2} & c_{y2} & \theta_2 & s_{x2} & s_{y2} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ c_{x40} & c_{y40} & \theta_{40} & s_{x40} & s_{y40} \end{bmatrix}_{40 \times 5} \quad (4.2)$$

(4.2) eşitliğinde yer alan matriste, her satır 5 parametreden oluşan parçacıklara denk gelmektedir. Önceki en iyi uygunluk değerini veren i 'ninci parçacığın pozisyonu $pbest_i = [p_{i1}, p_{i2}, \dots, p_{i5}]$ olarak ifade edilir. $gbest$ ise her iterasyonda tüm parçacıklar için tektir ve $gbest = [p_1, p_2, \dots, p_5]$ şeklinde gösterilir. i 'ninci parçacığın hızı (her boyuttaki konumunun değişim miktarı) $v_i = [v_{i1}, v_{i2}, \dots, v_{i5}]$ olarak ifade edilir. İki en iyi değer

bulunmasından sonra parçacık hızları ve konumları aşağıda verilen (4.3) ve (4.4) eşitliklerine göre güncellenmektedir.

$$v_i^{k+1} = w \cdot v_i^k + c_1 \cdot rand_1^k \cdot (pbest_i^k - x_i^k) + c_2 \cdot rand_2^k \cdot (gbest^k - x_i^k) \quad (4.3)$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \quad (4.4)$$

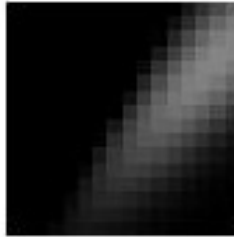
Çalışmamızda gerçekleştirilen optimizasyon süreci aşağıda yer alan adımlardan oluşmaktadır:

1. 5 ölçek, 8 yönlendirme ile 40 adet parametre oluşturulur.
2. Parametreler ile (4.1) eşitliği uyarınca 40 adet Gabor Dalgacık üretilir.
3. Eğitim kümesini oluşturan imgeler 40 adet filtre ile filtrenilir.
4. Her poz için filtrelenmiş imgeler arası farklar Öklid Uzaklığı ile hesaplanır.
5. 3 poz için de elde edilen fark değerlerini yükselten parametre ile pbest'i günceller. (Burada arzu edilen elde edilecek filtre ile farklı kişilere ait imgelerin filtreye verdikleri cevapların birbirlerinden farklı olması, yani kişi tepitinde belirleyici olmasıdır.)
6. Başlangıçtan mevcut iterasyona kadar en yüksek fark değerini sağlayan parametre ile gbest güncellenir.
7. (4.3) ve (4.4) denklemleri uyarınca hız ve konum güncellenir.
8. Maksimum iterasyon sayısına ulaşılmadıysa tekrar 2. adıma geçilir.

PSO ile elde edilen $(c_x, c_y, \theta, s_x, s_y)$ parametreleri sırası ile aşağıdaki gibidir:

$$c_x = 0.3921, \quad c_y = -0.1919, \quad \theta = 5.2234, \quad s_x = 5.0472, \quad s_y = 3.319.$$

Elde edilen optimum parametre ile oluşturulan Gabor filtre Şekil 4.12' de görülmektedir.

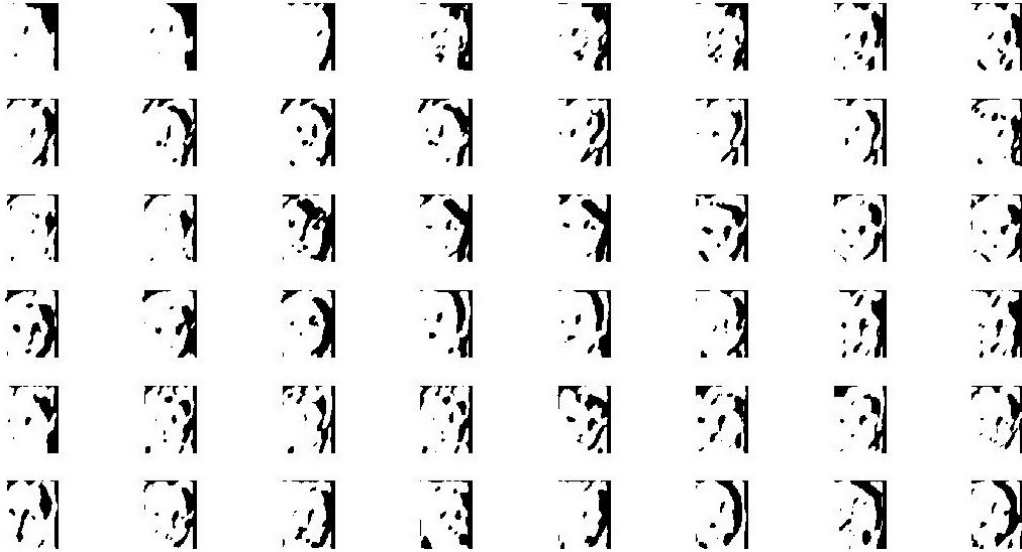


Şekil 4.12 : PSO ile oluşturulan Optimum Gabor Dalgacık

Eđitim ve test kümelerini oluřturan imgeler elde ettiđimiz tek Gabor Filtre ile filtrelenerek her bir imgeye ait nitelik vektörü elde edilir. (bkz. Őekil 4.13 ve Őekil 4.14)



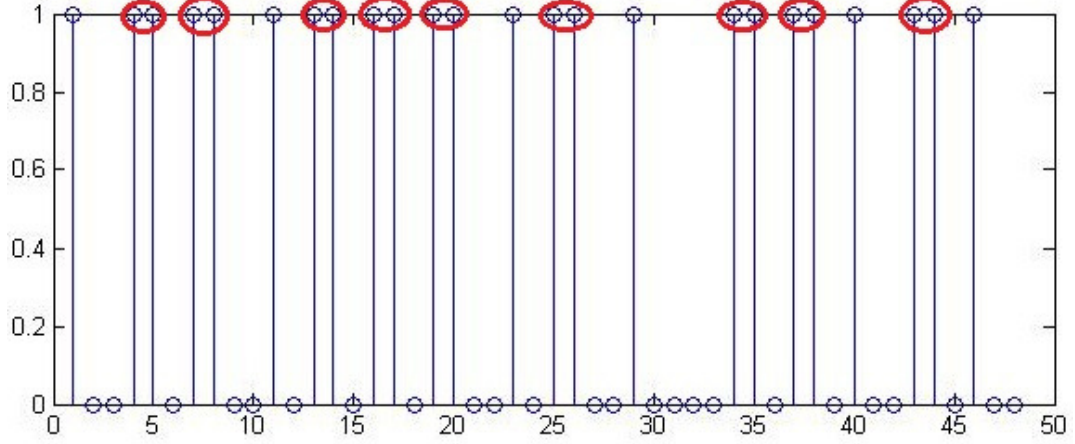
Őekil 4.13 : Eđitim kümesi imgelerinin Optimum Gabor Filtre'ye cevapları



Őekil 4.14 : Test kümesi imgelerinin Optimum Gabor Filtre'ye cevapları

Optimum Gabor Filtre ile elde edilen nitelik vektörleri arasında Öklid uzaklıđı hesaplanarak test kümesindeki imgelerin (bkz. Őekil 4.9 ve Őekil 4.10) eđitim

kümesinde hangi kişilere denk geldiği tespit edilmiştir. Şekil 4.15’de test imgesini oluşturan imgelerden hangilerinin doğru kişi ile eşlendiği görülebilir.



Şekil 4.15 : Test kümesi kişi eşleme doğruluk değerleri

Test kümesindeki imgelerin eğitim kümesinden hangi kişilere ait olduğunu belirlemek için Öklid mesafesi dışında Temel Bileşen Analizi de kullanılmıştır. TBA sonucunda elde edilen farkların minimum olduğu kısımlar koyu renkli bölgelerdir. Şekil 4.16’da görülen diyagonal karartı, test imgesi ile eğitim kümesi imgelerinin çakıştığı konumları göstermektedir. Yatay eksen soldan sağa 1 – 16 test imgelerini ifade ederken, dikey eksen 1 – 16 eğitim kümesi imgelerini ifade etmektedir.



Şekil 4.16 : Eğitim kümesi - test kümesi imgeleri benzerlik resmi (TBA)

5. TARTIŞMA VE SONUÇ

Bugüne kadar gerçekleşen pek çok gelişmeye karşın yüz tanıma zorlu bir problem olmaya devam etmektedir. Yüz ifadeleri, poz ve aydınlık gibi parametrelerde meydana gelen küçük değişimler, yüz imgesinde bozulmalara neden olmaktadır. Buna rağmen, yüzdeki yerel niteliklerin bu tür bozulmalara karşı dayanıklı olduğu ve bir uzay-frekans analizi ile niteliklerin eldesinin mümkün olduğu düşünülmektedir. Bu amaç doğrultusunda, iyi biçimde belirlenmiş uzay-frekans yerelleştirmesi ile Dalgacık Analizi doğru bir seçim olarak karşımıza çıkmaktadır. Çeşitli dalgacık tabanları arasında Gabor Fonksiyonları, hem zamanda hem de frekansda en uygun çözünürlüğü sağlamaktadır. Örüntü tanıma için yerel niteliklerin elde edilmesinde, Gabor Dalgacıkları en uygun yaklaşım olarak görünmektedir.

Doğrusal olmayan sürekli fonksiyonların optimizasyonu için geliştirilmiş Parçacık Sürü Optimizasyonu (PSO), basitleştirilmiş sosyal bir modelin benzetimi esnasında bulunmuştur. Kökleri iki ana metodolojiye dayanmaktadır; genelde Yapay yaşam (Artificial life (A-life)), özelde Kuş sürüsü (Bird Flocking), Balık sürüsü (Fish Schooling) ve Sürü Teorisi (Swarm Theory). PSO Evrimsel Hesaplama ile ilintilidir, bununla birlikte kökleri hem Genetik Algoritma hem de Evrimsel Programlamaya ulaşmaktadır.

PSO, basit bir fikir ve bir kaç satır bilgisayar kodu ile gerçekleştirmesi mümkün bir metoddan oluşmaktadır. Sadece temel matematik operatörlerine ihtiyaç duymakta ve bellek gereksinimi ve hız bakımından oldukça masrafsızdır.

Bu çalışmada, PSO ile Gabor Dalgacık parametrelerinin eniyilenmesi yardımıyla yüz tanımadaki kullanılan nitelik vektörlerinin optimizasyonu gerçekleştirilmiştir. PSO ile 40 farklı parçacık kullanılarak Optimum Gabor Dalgacık oluşturulmuştur.

Çalışmanın ilk aşamasında 40 adet Gabor dalgacık kullanılarak nitelik vektörleri elde edilmiş, yüz tanıma başarımının %60 olduğu görülmüştür. İkinci aşamada ise PSO yöntemi yardımıyla elde edilen Optimum Gabor Filtre kullanılarak nitelik vektörleri hesaplanmış, yüz tanıma başarımının %50 olduğu görülmüştür.

Benzetim sırasında tespit edilen bir diğer husus da veri boyutudur. 5 ölçek ve 8 yönlendirmeye sahip 40 adet gabor filtreden oluşan belirleyici filtre kümesi ile imgelerin tamamı filtrelendiğinde elimizdeki verinin hacmi 412160 gibi bir değere ulaştırmaktadır. Bu durum matematiksel işlem yükünün artmasına ve çalışma hızının yavaşlamasına neden olmaktadır. Filtrelenen imgeler ölçeklenerek veri boyutu indirgenmiştir ancak bir miktar veri kaybı yaşanmıştır.

Gabor Dalgacık sayısının bire indirilmesi ile veri hacmi %97.5 azalırken, yüz tanıma başarımındaki kayıp %10 gibi bir değerle sınırlanmıştır.

Yüz imgelerinin normalizasyonu ile başarımların analizleri üzerinde çalışmalarımız devam etmektedir.

KAYNAKLAR

1. JAIN, A. K., ROSS, A., and PANKANTI, S., 2006, Biometrics: a tool for information security, *IEEE Transactions on Information Forensics and Security*, 2006, vol.1, no.2, 125-143.
2. MAIO, D., MALTONI, D., JAIN, A. K., and PRABHAKAR, S., 2003, *Handbook of fingerprint recognition*, Springer- Verlag, New York, 0387954317.
3. BOUCHAFFRA D. & AMIRA, A., 2007, Structural Hidden Markov Models for Biometrics: Fusion of Face and Fingerprint, *In Special Issue of Pattern Recognition Journal, Feature Extraction and Machine Learning for Robust Multimodal Biometrics*, 2007, Article in press, available online.
4. JAIN, A. K., HONG, L., PANKANTI, S., and BOLLE, R., 1997, An identity authentication system using fingerprints, *Proceedings of the IEEE*, 1997, vol.85, no.9, 1365-1388.
5. JAIN, L. C., HALICI, U., HAYASHI, I., LEE, S. B., TSUTSUI, S., 1999, *Intelligent biometric techniques in fingerprint and face recognition*, CRC press, New York, 0849320550.
6. YANG, M. H., KRIEGMAN, D. J., and AHUJA, N., 2002, Detecting faces in images: a survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.24, no.1, 2002, 34-58.
7. ZHAO, W., CHELLAPPA, R., PHILLIPS, P. J., and ROSENFELD, A., 2003, Face recognition: a literature survey, *ACM Computing Surveys*, 2003, vol.35, 399-459.
8. FIERREZ-AGUILAR, J., GARCIA-ROMERO, D., ORTEGA- GARCIA J., and GONZALEZ-RODRIGUEZ, J., 2005, Adapted user dependent multimodal biometric authentication exploiting general information, *Pattern Recognition Letters*, 2005, vol.26, no.16, 2628-2639.
9. JAIN, A.K., PANKANTI, S., PRABHAKAR, S., HONG, L., ROSS, A., and WAYMAN, J. L., 2004, Biometrics: A Grand Challenge, *Proceedings of the International Conference on Pattern Recognition*, 2004, Cambridge, UK, August, vol. II, 935-942.
10. INDOVINA, M., ULUDAG, U., SNEICK, R., MINK A., and JAIN, A., 2003, Multimodal biometric authentication methods: a cots approach, *Proc. MMUA 2003, Workshop on Multimodal User Authentication*, 2003, 99-106.

11. FIERREZ-AGUILAR, J., ORTEGA-GARCIA, J., GONZALEZ-RODRIGUEZ, J., and BIGUN, J., 2005, Discriminative multimodal biometric authentication based on quality measures, *Pattern Recognition* 38 (5), 2005, 777-779.
12. JAIN, A. K., HONG, L., and KULKARNI, Y., 1998, F2ID: A personal identification system using faces and fingerprints, *Proc. 14th International Conference on Pattern Recognition*, 1998, Brisbane, 1373 – 1375.
13. JAİN, A. K., ROSS, A., PRABHAKAR, S., 2004, An introduction to biometric recognition, *IEEE Transaction on Circuits and Systems for Video Technology*, 2004, vol.14, no.1, 4-19.
14. BRUCE, V., & YOUNG, A., 1998, *In the eye of the beholder: The science of face perception*, Oxford University Press, New York, 0198524404.
15. CEVİKALP, H., NEAMTU, M., WILKES, M., BARKANA A., 2005, Discriminative common vectors for face recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2005, vol.27, no.1, 4-13.
16. BRUNELLI, R. & POGGIO, T., 1992, Face Recognition Through Geometrical Features, *Proceedings of the Second European Conference on Computer Vision*, 1992, 792-800.
17. SALAH, A. A., 2005, İnsan ve Bilgisayarda Yüz Tanıma, *Üçüncü Bilgi İşleyen Makine Olarak Beyin – Beyin 2005*, Albert Long Hall, Boğaziçi Üniversitesi.
18. ANDREWS, T. J. & SCHLUPPECK, D., 2004, Neural responses to Mooney images reveal a modular representation of faces in human visual cortex, *NeuroImage*, 2004, vol.21, 91-98.
19. TANAKA, J. W. & FARAHA, M. J., 1993, Parts and wholes in face recognition, *Quarterly Journal of Experimental Psychology: Human Experimental Psychology*, 1993, vol.46A, 225-245.
20. GAUTHIER, I. & TARR, M. J., 1997, Becoming a “Greeble expert”: Exploring the face recognition mechanism, *Vision Research*, 1997, vol.37, 1673-1682.
21. FODOR, J. A., 1983, *The Modularity of Mind*, MIT Press, Cambridge, MA, 0-262-06084-1.
22. KARMILOFF-SMITH, A., 1992, *Beyond modularity: A developmental perspective on cognitive science*, Cambridge MA: MIT Press, 0-262-11169-1.
23. ELMAN, J. L., BATES, E. A., JOHNSON, M. H., KARMILOFF-SMITH, A., PARISI, D., PLUNKETT, K., 1996, *Rethinking innateness: A connectionist perspective on development*, Cambridge, MA: MIT Press, 0-262-05052-8.
24. THOMPSON, P., 1980, Margaret Thatcher - A new illusion, *Perception*, 1980, vol.9, 483-484.

25. PERRONNIN, F. & DUGELAY, J. L., 2003, An introduction to biometrics and face recognition, *Proc. IMAGE'2003: Learning, Understanding, Information Retrieval, Medical*, 2003, Cagliari, Italy.
26. ABATE, A. F., NAPPI, M., RICCIO, D., SABATINO, G., 2007, 2D and 3D face recognition: A survey, *Pattern Recognition Letters*, 2007, vol.28, 1885-1906.
27. SHERRAH, J., 2004, False alarm rate: A critical performance measure for face recognition, *Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, May 2004, 189-194.
28. PHILLIPS, J. P., MOON, H., RIZVI, A. S., RAUSS, P. J., 2000, The FERET evaluation methodology for face-recognition algorithms, *IEEE Transactions on Pattern Analysis Machine Intelligence*, 22 (10), 1090-1104.
29. SIM, T., BAKER, S., BSAT, M., 2003, The CMU pose, illumination, and expression database, *IEEE Transactions on Pattern Analysis Machine Intelligence*, 25 (12), 1615-1618.
30. MARTINEZ, A. M., 2002, Recognizing imprecisely localized, partially occluded, and expression variant faces from a single sample per class, *IEEE Transactions on Pattern Analysis Machine Intelligence*, 24 (6), 748-763.
31. CHELLAPPA, R., WILSON, C. L., SIROHEY, S., 1995, Human and machine recognition of faces: a survey, *Proceedings of the IEEE*, vol.83, no.5.
32. TURK, M., & PENTLAND, A., 1991, Eigenfaces for recognition, *Journal of Cognitive Neuroscience*, vol.3, no.1, 71-86.
33. WISKOTT, L., FELLOUS, J. M., KRÜGER, N., VON DER MALSBERG, C., 1997, Face recognition by elastic bunch graph matching, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.19, no.7, 775-779.
34. RAMASUBRAMANIAN, D. & VENKATESH, Y. V., 2001, Encoding and recognition of faces based on the human visual model and DCT, *Pattern Recognition*, vol.34, 2447-2458.
35. WALLRAVEN, C., SCHWANINGER, A., BÜLTHOFF, H. H., 2004, Learning from humans: computational modeling of face recognition, *Proceedings of ECVW*.
36. HYVARINEN, A., 1999, Survey on independent component analysis, *Neural Comp. Surveys*, vol.2, 94-128.
37. HYVARINEN, A. & OJA, E., 2000, Independent component analysis: algorithms and applications, *Neural Networks*, vol.13, 411-430.
38. BARTLETT, M. S., MOVELLAN J. R. and SEJNOWSKI, T. J., 2002, Face recognition by independent component analysis, *IEEE Transactions on Neural Networks*, Vol.13, no.6, 1450-1464.

39. KIRBY, M. & SIROVICH, L., 1990, Application of the Karhunen-Loeve procedure for the characterization of human faces, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.12, 103-108.
40. ERGEZER, H., DİKMEN, M. ve ÖZDEMİR, E., 2003, Yapay sinir ağları ve tanıma sistemleri, *PiVOLKA*, 2(6), 14-17.
41. TÜZÜN, S. & AKAN, A., 2005, Yüz uzayının dikleştirilmesine dayanan yeni bir yüz tanıma yöntemi, *Elektrik-Elektronik-Bilgisayar Mühendisliği 11. Ulusal Kongresi ve Fuarı Bildirileri*, İstanbul, 2005.
42. GOLUB, G. H. & VAN LOAN, C. F., 1996, *Matrix Computations (3rd ed.)*, Johns Hopkins, 978-0-8018-5414-9.
43. BRUNELLI, R. & POGGIO, T., 1993, Face recognition: features versus templates, *IEEE Trans PAMI*, 15(10), 1042-1052.
44. HEISELE, B., HO, P., POGGIO, T., 2001, Face recognition with support vector machines: global versus component-based approach, *International conference on computer vision (ICCV '01)*, 2001, Vancouver, 688-694.
45. NEFIAN, A. & HAYES, M., 1999, An embedded hmm-based approach for face detection and recognition, *Proceedings of IEEE international conference on acoustics, speech and signal processing*, 1999, 3553-3556.
46. TURK, M. & PENTLAND, A., 1991, Eigenfaces for recognition, *Journal Cognitive Neuroscience*, 3(1), 71-86.
47. ADINI, Y., MOSES, Y., ULLMAN, S., 1997, Face recognition: the problem of compensating for changes in illumination direction, *IEEE Trans PAMI*, 19(7), 721-732.
48. FISHER, R. A., 1936, The use of multiple measures in taxonomic problems, *Annals Eugen*, vol.7, 179-188.
49. BELHUMEUR, P. N., HESPANHA, J. P., KRIEGMAN, D. J., 1997, Eigenfaces vs. Fisherfaces: recognition using class specific linear projection, *IEEE Trans PAMI*, 19(7), 711-720.
50. ZHAO, W., 1998, Discriminant analysis of principal components for face recognition., In: *Wechsler, H., Phillips, P. J., Bruce, V., Soulie, F. F., Huang, T. S., (eds) Face recognition: from theory to applications*, Springer, Berlin Heidelberg New York, 73-85.
51. ZHAO, W., 2000, Face recognition: a literature survey, *Technical Report*, University of Maryland.
52. CHELLAPA, R., WILSON, C., SIROHEY, S., 1995, Human and machine recognition of faces: a survey, *Proc. IEEE*, 83(5), 705-740.

53. SCHOLKOPF, B., 1997, Comparing support vector machines with Gaussian kernels to radial basis function classifiers, *IEEE Trans Signal Process*, 45(11), 2758-2765.
54. QIAN, S. & CHEN, D., 1996, *Joint time-frequency analysis: method and applications*, Prentice Hall, Englewood Cliffs, 0132543842.
55. DAUBECHIES, I., 1990, The wavelet transform, time-frequency localization and signal analysis, *IEEE Trans Inf Theory*, 6(5), 961-1005.
56. GABOR, D., 1946, Theory of communications, *J. Inst. Electr. Eng.*, 93, 429-457.
57. DAUGMAN, J. G., 1985, Uncertainty relation for resolution in space, spatial-frequency, and orientation optimized by two dimensional visual cortical filters, *J. Opt. Soc. Am. A Opt. Image Sci. Vis.*, 2(7), 1160-1169.
58. KRUGER, V. & SOMMER, G., 2002, Gabor wavelet networks for efficient head pose estimation, *Image Vis. Comput.*, 20(9-10), 665-672.
59. KRUGER, V. & SOMMER, G., 2002, Wavelet networks for face processing, *J. Opt. Soc. Am. A Opt. Image Sci. Vis.*, 19(6), 1112-1119.
60. JAIN A. K. & FARROKHANIA, F., 1991, Unsupervised texture segmentation using Gabor filters, *Pattern Recognition*, 24(12), 1167-1186.
61. WELDON, T. P., HIGGINS, W. E., DUNN, D. F., 1996, Efficient Gabor filter design for texture segmentation, *Pattern Recognition*, 29(12), 2005-2015.
62. HAMAMOTO, Y., 1998, A Gabor filter-based method for recognizing handwritten numerals, *Pattern Recognition*, 31(4), 395-400.
63. LEE, C. J. & WANG, S. D., 1999, Fingerprint feature extraction using Gabor filters, *Electr. Letters*, 35(4), 288-290.
64. LADES, M., 1993, Distortion invariant object recognition in the Dynamic Link Architecture, *IEEE Trans. Comput.*, 42(3), 300-311.
65. WISKOTT, L., 1997, Face recognition by elastic bunch graph matching, *IEEE Trans PAMI*, 19(7), 775-779.
66. MU, X. Y. & HASSOUN, M. H., 2003, Combining Gabor features: summing vs. voting in human face recognition, *2003 IEEE international conference on systems, man and cybernetics*.
67. DUC, B., FISCHER, S., BIGUN, J., 1999, Face authentication with Gabor information on deformable graphs, *IEEE Trans Image Process*, 8(4), 504-516.
68. JIAO, F., 2002, A face recognition method based on local feature analysis, *Proceedings of the 5th Asian conference on computer vision*, 188-192.

69. LIAO, R. & LI, S., 2000, Face recognition based on multiple facial features, *Proceedings of the 4th IEEE international conference on automatic face and gesture recognition*, 239-244.
70. LIU, C. J. & WECHSLER, H., 2002, Gabor feature based classification using the enhanced Fisher linear discriminant model for face recognition, *IEEE Trans Image Process*, 11(4), 467-476.
71. AYINDE, O. & YANG, Y. H., 2002, Face recognition approach based on rank correlation of Gabor-filtered images, *Pattern Recognition*, 35(6), 1275-1289.
72. SHEN, L., BAI, L., FAIRHURST, M., 2006, Gabor wavelets and Generalized Discriminant Analysis for face identification and verification, *Image Vis. Comput.* (in press).
73. PHILLIPS, P. J., 2000, The FERET evaluation methodology for face-recognition algorithms, *IEEE Trans PAMI*, 22(10), 1090-1104.
74. MESSER, K., 2004, Face authentication test on the BANCA database, *Proceedings of international conference on pattern recognition*, Cambridge, 523-532.
75. HAAR, A., 1910, Zur Theorie der orthogonalen Funktionensysteme [On the theory of orthogonal function systems], *Mathematische Annalen*, 69 (1910), 331-371, Translated by Georg Zimmermann.
76. YOSHIDA, N., 1994, The Littlewood-Paley-Stein inequality on an infinite dimensional manifold, *J. Funct. Anal.*, 122, 402-427.
77. COIFMAN, R. R. & WEISS, G., 1977, Extensions of Hardy spaces and their use in analysis, *Bull. Amer. Math. Soc.*, 83 (1977), 569-645.
78. GROSSMANN, A. & MORLET J., 1984, Decomposition of Hardy functions into square integrable wavelets of constant shape, *SIAM J. Math. Anal.* 15 (1984), 723-736.
79. GROSSMANN, A., MORLET, J. & PAUL, T., 1985, Transforms associated to square integrable group representations I, *J. Math. Physics*, 26 (1985), 2473-2479.
80. MEYER, Y., 1987, Ondelettes, fonctions splines et analyses graduees [Wavelets, spline functions, and multiresolution analysis], *Rend. Sem. Mat. Univ. Politec. Torino*, 45 (1987), 1-42, Translated by John Horvath.
81. KOCAMAN, Ç., 2003, *Dalgacık Tabanlı Transformatör Koruma Algoritması*, Yüksek Lisans Tezi, Ondokuz Mayıs Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik-Elektronik Mühendisliği Anabilim Dalı, Samsun, 2003.
82. POLIKAR, R., 2001, *The Wavelet Tutorial Lecture Notes*, Dept. of Electrical and Computer Engineering, Rowan University, 2001.

83. SMITH, M. J. & BARNWELL, T. P., 1984, A Procedure for Designing Exact Reconstruction Filter Banks for Tree Structured Sub-band Coders, *In Proc. IEEE Int. Conf. Acoust., Speech, and Signal Proc.*, 1984, San Diego.
84. UÇAN, O. N., OSMAN, O. & ALBORA, A. M., 2006, *Görüntü İşleme Teknikleri ve Mühendislik Uygulamaları*, Nobel Yayın Dağıtım, Ankara, 9944-77-051-5.
85. RUSKAI, M. B., BEYLKIN, G., COIFMAN, R., DAUBECHIES, I., MALLAT, S., MEYER, Y. and RAPHAEL, L., 1992, *Wavelets and Their Applications*, Jones and Bartlett Books in Mathematics, Boston: Jones and Bartlett, 1992, 153–178.
86. MALLAT, S., 1989, A Theory for Multiresolution Signal Decomposition: the Wavelet Representation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11, 674-693.
87. TAI SING LEE, 1996, Image representation using 2D Gabor wavelet, *IEEE Trans on Pattern Analysis and Machine Intelligence*, vol.18, no.10, 959-971.
88. MEHROTRA, R., 1992, Gabor filter-based edge detection, *Pattern Recognition*, vol.25, no.12, 1479-1494.
89. KRUGER, V. & SOMMER, G., 2000, Gabor wavelet networks for object representation, *Technical Report*, Nr. 2002, University of Kiel, February 2000.
90. KENNEDY, J., & EBERHART, R. C., 1995, Particle swarm optimization, *Proc. IEEE int'l conf on neural Networks, Vol. IV*, 1995, Piscataway, NJ: IEEE service center, 1942-1948.
91. KALINLI A., KARABOĞA N., KARABOĞA D., 2001, A modified touring ant colony optimization algorithm for continuous functions, *16th International Symposium on Computer and Information Science (ISCIS XVI)*, 2001, Antalya: Işık University, 437-444.
92. HEPPNER F. & GRENANDER U., 1990, *A stochastic nonlinear model for coordinated bird flocks*. In: E. Krasner (ed.), *The ubiquity of chaos*, AAAS Publications, Washington, DC, 233-238.
93. REYNOLDS, C. W., 1987, Flocks, Herds and Schools: a distributed behavioral model, *Computer Graphics*, 21 (4), 25-34.
94. WILSON, E. O., 1975, *Sociobiology: The new synthesis*, Belknap Press, Cambridge, MA, 0674000897.
95. CHIH-CHIN LAI, 2006, A Novel Image Segmentation Approach Based on Particle Swarm Optimization, *IEICE Transactions*, 89-A (1), 324-327.
96. HU, X., EBERHART, R. C., SHI, Y., 2003, Engineering optimization with particle swarm, *IEEE Swarm Intelligence Symposium*, 2003, Indianapolis, USA: IEEE service center.

97. YOSHIDA, H., KAWATA, K., FUKUYAMA, Y., NAKANISHI, Y., 1999, A particle swarm optimization for reactive power and voltage control considering voltage stability, *Proc. Intl. Conf. on Intelligent System Application to Power Systems*, 1999, Rio de Janeiro, Brazil, 117-121.
98. GUDISE, V. G. & VENAYAGAMOORTHY, G. K., 2004, FPGA Placement and Routing Using Particle Swarm Optimization, *IEEE Computer Society Annual Symposium on VLSI*, February 2004, USA, 307-308.
99. WELCH, R. & VENAYAGAMOORTHY, G. K., 2007, A Fuzzy-PSO Based Controller for a Grid Independent Photovoltaic System, *IEEE Symposium on Computational Intelligence and Data Mining*, April 2007, USA: IEEE service center.
100. VENAYAGAMOORTHY, G. K. & DOCTOR, S., 2004, Navigation of Mobile Sensors Using PSO and Embedded PSO in a Fuzzy Logic Controller, *39th IEEE IAS Annual Meeting on Industry Applications*, 2004, USA, 1200-1206.
101. SHI, Y. & EBERHART, R. C., 1998, A modified particle swarm optimizer, *Proceedings of the IEEE International Conference on Evolutionary Computation*, 1998, Piscataway, NJ: IEEE Press, 69-73.
102. SHI, Y. & EBERHART, R. C., 1998, Parameter selection in particle swarm optimization, *Evolutionary Programming VII: Proc. EP 98*, 1998, New York: Springer-Verlag, 591-600.
103. RABINER, L. R. & SHAFER, R. W., September 1978, *Digital Processing of Speech Signals*, Prentice Hall Inc., pp. 396-444.
104. RABINER, L. R. and JUANG, B. H., April 1993, *Fundamentals of Speech Recognition*, Prentice Hall Inc., pp. 97-123.

EKLER

EK-A

```
% Tez Adi      : Yuz Tanima Sistemlerinde Kullanilan Oznitelik Vektorlerinin
                  Optimizasyonu – (5 olcek 8 yonelimli 40 adet filtre ile yuz tanima)
% Yazar        : Serkan TUZUN
% e-posta      : eng.serkantuzun@yahoo.com
```

```
close all;
clear all;
clc;
```

```
% GABOR FILTRELERIN OLUSTURULMASI
% -----
```

```
R=16; C=16;
toplam_filtre=40;
GW1=zeros(toplam_filtre,R,C);
```

```
Cx=0;
Cy=0;
```

```
i=1;
for Sx=1:0.5:3
    Sy = Sx;
    for theta=pi/8:pi/8:pi
        GW = GaborWavelet1(R, C, Cx, Cy, theta, Sx, Sy);
        GW1(i, :, :) = GW(:, :, :);
        subplot(5,8,i)
        imshow(GW);
        i=i+1;
    end
end
```

```
load -mat yuzler
```

```
filtered_image = zeros(28,23,40,16,3);
```

```
for fsayi=1:toplam_filtre
    for kisi=1:16
        for poz=1:3
            imge = imfilter(Kisiler(:, :, kisi, poz), GW1(fsayi, :, :));
```



```

        imge2=imresize(imge,0.25);
        filtered_image(:,:,fsayi,kisi,poz)=imge2;
    end
end
end

```

```

k=1;
figure;
for kisi=1:16
    for poz=1:3
        subplot(6,8,k)
        imshow(filtered_image(:,:,10,kisi,poz));
        k=k+1;
    end
end

```

```

% filtrelenen görüntülerin normalize edilmesi
% -----

```

```

enkucuk=min(min(min(min(min(filtered_image))));
filtered_image=filtered_image+abs(enkucuk);

```

```

enbuyuk=max(max(max(max(max(filtered_image))));
filtered_image=filtered_image./abs(enbuyuk);

```

```

filtered_image=round(filtered_image);
filtered_image=filtered_image.*255;

```

```

k=1;
figure;
for kisi=1:16
    for poz=1:3
        subplot(6,8,k)
        imshow(filtered_image(:,:,10,kisi,poz));
        k=k+1;
    end
end

```

```

% TEST KUMESI IMGELERININ GABOR FILTRELER ILE FILTRELENMESI
% -----

```

```

load -mat test_kumesi

```

```

for fsayi=1:toplam_filtre
    for kisi=1:16
        for poz=1:3
            imge=imfilter(test_imge(:,:,kisi,poz),GW1(fsayi,,:));
            imge2=imresize(imge,0.25);

```

```

        filtered_test(:,:,fsayi,kisi,poz) = imge2;
    end
end
end

```

```

k=1;
figure
for ks=1:16
    for pz=1:3
        subplot(6,8,k)
        imshow(filtered_test(:,:,1,ks,pz));
        k=k+1;
    end
end

```

```

% filtrelenen goruntulerin normalize edilmesi
% -----

```

```

enkucuk=min(min(min(min(min(filtered_test))));
filtered_test=filtered_test+abs(enkucuk);

```

```

enbuyuk=max(max(max(max(max(filtered_test))));
filtered_test=filtered_test./abs(enbuyuk);

```

```

filtered_test=round(filtered_test);
filtered_test=filtered_test.*255;

```

```

k=1;
figure;
for kisi=1:16
    for poz=1:3
        subplot(6,8,k)
        imshow(filtered_test(:,:,10,kisi,poz));
        k=k+1;
    end
end

```

```

% Egitim ve test kumesi filtre cevaplarinin vektöre cevrilmesi
% -----

```

```

for kisi=1:16
    for poz=1:3
        for fsayi=1:40
            fltrd_imge(:,fsayi,kisi,poz)=reshape(filtered_image(:,:,fsayi,kisi,poz),(28*23),1);
        end
        fltrdd_imge(:,kisi,poz)=reshape(fltrd_imge(:,:,kisi,poz),(28*23*40),1);
    end
end

```

```

for kisi=1:16
    for poz=1:3
        for fsayi=1:40
            fltrd_test(:,fsayi,kisi,poz)=reshape(filtered_test(:, :, fsayi, kisi, poz), (28*23), 1);
        end
        fltrrdd_test(:,kisi,poz)=reshape(fltrd_test(:, :, kisi, poz), (28*23*40), 1);
    end
end

```

```

% Euclid Distance ile en yakin resmin belirlenmesi

```

```

% -----
k=1;
for ks=1:16
    for puz=1:3
        for kisi=1:16
            for poz=1:3
                fark(:,kisi,poz,ks,pz)=sqrt(abs(fltrrdd_test(:,ks,pz).^2-
fltrrdd_imge(:,kisi,poz).^2));
                fark_deger(kisi,poz,ks,pz)=sum(fark(:,kisi,poz,ks,pz))/25760;
            end
        end
        mn=min(min(fark_deger(:, :, ks, puz)));
        [dogru_kisi(k) dogru_puz(k)]=find(fark_deger(:, :, ks, puz)==mn);
        k=k+1;
    end
end

```

```

dogru=0;
for k=1:48
    if dogru_kisi(k)==(int8(k-1)/3)+1
        grafik(k)=1;
        dogru=dogru+1;
    else
        grafik(k)=0;
    end
end
end
figure
stem(grafik)

```

EK-B

```
% Tez Adi      : Yuz Tanimada Kullanilan Oznitelik Vektorlerinin Optimizasyonu
%              Kisim-1
% Yazar        : Serkan TUZUN
% e-posta      : eng.serkantuzun@yahoo.com
```

```
close all;
clear all;
clc;
```

```
% veritabanini olusturan yuzlerin Kisiler matrisine aktarimi
% veritabani, 16 kisinin 3 farkli acidan elde edilmiş görüntülerinden oluşur.
% poz ile kastedilen, 1: onden, 2: sagdan, 3: soldan acili görünumdur.
% -----
```

```
Kisiler = zeros(112,92,16,3);k=1;
for kisi=1:16
    for poz=1:3
        Kisiler(:,:,kisi,poz) =
double(imread(streat('C:\MATLAB\AT&T_Faces\veritabani\resim',num2str(kisi),',',nu
m2str(poz),'.jpeg')));
%     subplot(4,6,k)
%     imshow(uint8(Kisiler(:,:,kisi,poz)))
%     title(streat('kisi: ',num2str(kisi),' poz: ',num2str(poz)));
%     k=k+1;
    end
end

save yuzler.mat Kisiler;
clear;
```

```
% GABOR OLUSTURMA VE PARCACIKLARA ILK DEGER ATANMASI
% -----
```

```
R=16; C=16;
toplam_filtre=40;
GW1 = zeros(toplam_filtre,R,C);

for i=1:toplam_filtre
    X(i,:,1)=[round(rand*2); round(rand*2); rand*pi; round(rand*5); round(rand*5)];
    i=i+1;
end
```

```

save parcacik X;
clear;

% OPTIMIZASYON DONGUSU BASLAMAK UZERE
% -----
% -----
load parcacik

opt_dongu=1;
w=0.7;
sinir=10;
V=zeros(40,5);
pbest(:,:,1)=X(:,:,1);
gbest=X(25,:,1);
c1=2;
c2=c1;

while opt_dongu<sinir
% -----
% -----
load parcacik
R=16; C=16;
toplaml_filtre=40;
for i=1:toplaml_filtre
    GW = GaborWavelet1(R, C, X(i,1,opt_dongu), X(i,2,opt_dongu), X(i,3,opt_dongu),
X(i,4,opt_dongu), X(i,5,opt_dongu));
    GW1(i,:,:)= GW(:,:);
    subplot(5,8,i)
    imshow(GW);
    i=i+1;
end

save p_eniyi pbest
save g_eniyi gbest
save dalgacik GW1
save indis opt_dongu
save hiz V
clear;

% 40 adet Gabor Wavelet ile yüzlerin filtrelenmesi
% -----

load -mat yuzler
load -mat dalgacik
R=16; C=16;
filtered_image = zeros(112,92,40,16,3);
toplaml_filtre=40;

```

```

for fsayi=1:toplaml_filtre
    for kisi=1:16
        for poz=1:3
            filtered_image(:,:,fsayi,kisi,poz) = imfilter(Kisiler(:,:,kisi,poz),GW1(fsayi,:,:));
        end
    end
end

```

```

k=1;
for kisi=1:16
    for poz=1:3
        subplot(6,8,k)
        imshow(filtered_image(:,:,10,kisi,poz));
        k=k+1;
    end
end

```

```

save filtrelenmis filtered_image;
clear;

```

```

% filtrelenen goruntulerin normalize edilmesi
% -----

```

```

load -mat filtrelenmis

```

```

enkucuk=min(min(min(min(min(filtered_image))));
filtered_image=filtered_image+abs(enkucuk);

```

```

enbuyuk=max(max(max(max(max(filtered_image))));
filtered_image=filtered_image./abs(enbuyuk);

```

```

filtered_image=round(filtered_image);
filtered_image=filtered_image.*255;

```

```

save filtrelenmis filtered_image;
clear;

```

```

% farkli kisilere ait, filtrelenmis yüz imgeleri arasındaki farkların hesaplanması
% -----

```

```

load -mat filtrelenmis

```

```

load -mat parcacik

```

```

load -mat indis

```

```

load -mat p_eniyi

```

```

load -mat g_eniyi

```

```

load -mat hiz

```

```

w=0.7;

```

```

toplaml_filtre=40;
c1=2;
c2=c1;

for fsayi=1:toplaml_filtre
    for poz=1:3
        k=1;
        for kisi=2:16
            for i=1:(kisi-1)
farkkisi(:, :, fsayi, poz, k)=sqrt(abs(filtered_image(:, :, fsayi, kisi, poz).^2-
filtered_image(:, :, fsayi, i, poz).^2));
farkkisi_deger(k, fsayi, poz, opt_dongu)=sqrt(sum(sum(farkkisi(:, :, fsayi, poz, k).^2))/(112*
92));
                k=k+1;
            end
        end
    end
end

if opt_dongu == 1

    rast1=rand(1,5);
    rast2=rand(1,5);
    sinir=10;
    if opt_dongu<sinir
        for i=1:40
            V(i,:)=w.*V(i,:)+c1.*rast1.*(pbest(i,:,opt_dongu)-X(i,:,opt_dongu))+
c2.*rast2.*(gbest-X(i,:,opt_dongu));
            X(i,:,opt_dongu+1)=X(i,:,opt_dongu)+V(i,:);
        end
    end

elseif opt_dongu == 2
    pbest(:, :, 2)=pbest(:, :, 1);
    save p_eniyi pbest
    for fsayi=1:toplaml_filtre
        for poz=1:3
            cost(fsayi, poz, opt_dongu-
1)=sum(sqrt(abs(farkkisi_deger(:, fsayi, poz, opt_dongu).^2-
farkkisi_deger(:, fsayi, poz, opt_dongu-1).^2)))/120;
        end
    end

    rast1=rand(1,5);
    rast2=rand(1,5);
    sinir=10;

    if opt_dongu<sinir

```

```

    for i=1:40
        V(i,:)=w.*V(i,:)+c1.*rast1.*(pbest(i,:,opt_dongu)-X(i,:,opt_dongu))+
c2.*rast2.*(gbest-X(i,:,opt_dongu));
        X(i,:,opt_dongu+1)=X(i,:,opt_dongu)+V(i,:);
    end
end

else
    pbest(:, :, opt_dongu)=pbest(:, :, opt_dongu-1);
    save p_eniyi pbest
    for fsayi=1:toplam_filtre
        for poz=1:3
            cost(fsayi,poz,opt_dongu-
1)=sum(sqrt(abs(farkkisi_deger(:,fsayi,poz,opt_dongu).^2-
farkkisi_deger(:,fsayi,poz,opt_dongu-1).^2)))/120;
            end
            if (cost(fsayi,1,opt_dongu-1)>cost(fsayi,1,opt_dongu-
2))&&(cost(fsayi,2,opt_dongu-1)>cost(fsayi,2,opt_dongu-
2))&&(cost(fsayi,3,opt_dongu-1)>cost(fsayi,3,opt_dongu-2))
                pbest(fsayi,:,opt_dongu)=X(fsayi,:,opt_dongu);
                save p_eniyi pbest
            end
        end
    end

    for k=1:opt_dongu-1
        for i=1:toplam_filtre
            dgsm(i,k)=0;
            toplam(i,k)=cost(i,1,k)+cost(i,2,k)+cost(i,3,k);
            for poz=2:3
                for ckr=1:(poz-1)
                    dgsm(i,k)=dgsm(i,k)+abs(cost(i,poz,k)-cost(i,ckr,k));
                end
            end
        end
    end

    [dgsm2(:,k) knm_dgsm(:,k)]=sort(dgsm(:,k));
    [toplam2(:,k) knm_toplam(:,k)]=sort(toplam(:,k),'descend');
end

for k=1:opt_dongu-1
    i=1;
    y=0;
    f_sayi(k)=0;
    while y==0 & i<toplam_filtre+1
        if knm_dgsm(i,k)==knm_toplam(i,k)
            f_sayi(k)=knm_dgsm(i,k)
            y=1;
        else
            i=i+1;
        end
    end
end

```



```

        y=0;
    end
    if f_sayi(k)==0
        f_sayi(k)=knm_dgsm(1,k)
    end
end
end

for k=1:opt_dongu-2
    cost_diz(k)=cost(f_sayi(k),1,k+1);
end
r=find(cost_diz==max(cost_diz));
if size(r)<2
    ff_sayi_indis=r;
else
    ff_sayi_indis=r(1);
end

gbest=pbest(f_sayi(ff_sayi_indis),:(ff_sayi_indis+1))
save g_eniyi gbest

rast1=rand(1,5);
rast2=rand(1,5);
sinir=10;
if opt_dongu<sinir
    for i=1:40
        V(i,:)=w.*V(i,:)+c1.*rast1.*(pbest(i,:,opt_dongu)-X(i,:,opt_dongu))+
c2.*rast2.*(gbest-X(i,:,opt_dongu));
        X(i,:,opt_dongu+1)=X(i,:,opt_dongu)+V(i,:);
    end
end

end
save parcacik X;
opt_dongu=opt_dongu+1;
sinir=10;
end

```

EK-C

```

% Tez Adi      : Yuz Tanimada Kullanilan Oznitelik Vektorlerinin Optimizasyonu
%              : Kisim-2 (Okliid Mesafesi ile Kisinin Tespit Edilmesi)
% Yazar       : Serkan TUZUN
% e-posta     : eng.serkantuzun@yahoo.com

```

```

close all
clear all
clc

```

```

% Parcacigin en iyi degeri ile optimum filtrenin olusturulmasi
% -----
load -mat g_eniyi

R=16; C=16;
GW = GaborWavelet1(R, C, gbest(1,1), gbest(1,2), gbest(1,3), gbest(1,4), gbest(1,5));

figure
imshow(GW);

% Egitim kumesi yuz imgelerinin optimum filtre ile filtrelenmesi
% -----
load -mat yuzler
for kisi=1:16
    for poz=1:3
        filtered_imge(:, :, kisi, poz) = imfilter(Kisiler(:, :, kisi, poz), GW);
    end
end

k=1;
figure
for kisi=1:16
    for poz=1:3
        subplot(6,8,k)
        imshow(filtered_imge(:, :, kisi, poz));
        title(strcat('kisi: ', num2str(kisi), ' poz: ', num2str(poz)));
        k=k+1;
    end
end
save filtrelenmis_opt filtered_imge;

% Test kumesi yuz imgelerinin optimum filtre ile filtrelenmesi
% -----
for ks=1:16;
    for pz=1:3
        test_imge(:, :, ks, pz) =
double(imread(strcat('C:\MATLAB\AT&T_Faces\test\test', num2str(ks), '\', num2str(pz), '.
jpeg')));
        filtered_test(:, :, ks, pz) = imfilter(test_imge(:, :, ks, pz), GW);
    end
end

save test_kumesi test_imge

k=1;
figure

```

```

for ks=1:16
    for pz=1:3
        subplot(6,8,k)
        imshow(filtered_test(:,:,ks,pz));
        k=k+1;
    end
end

% Euclid Distance ile en yakin resmin belirlenmesi
% -----
k=1;
for ks=1:16
    for pz=1:3
        for kisi=1:16
            for poz=1:3
                fark(:,:,kisi,poz,ks,pz)=sqrt(abs(filtered_test(:,:,ks,pz).^2-
filtered_imge(:,:,kisi,poz).^2));
                fark_deger(kisi,poz,ks,pz)=sum(sum(fark(:,:,kisi,poz,ks,pz)))/(112*92);
            end
        end
        mn=min(min(fark_deger(:,:,ks,pz)));
        [dogru_kisi(k) dogru_poz(k)]=find(fark_deger(:,:,ks,pz)==mn);
        k=k+1;
    end
end

dogru=0;
for k=1:48
    if dogru_kisi(k)==(int8(k-1)/3)+1
        grafik(k)=1;
        dogru=dogru+1;
    else
        grafik(k)=0;
    end
end
figure
stem(grafik)

```

EK-Ç

```

% Tez Adi      : Yuz Tanimada Kullanilan Oznitelik Vektorlerinin Optimizasyonu
%              : Kisim-3 (Temel Bileşen Analizi ile Kisinin Tespit Edilmesi – Asama 1)
% Yazar       : Serkan TUZUN
% e-posta     : eng.serkantuzun@yahoo.com
close all
clear all
clc

```

```
% Parcacik en iyi degeri ile optimum filtrenin olusturulmasi
```

```
% -----
```

```
load -mat g_eniyi
```

```
R=16; C=16;
```

```
GW = GaborWavelet1(R, C, gbest(1,1), gbest(1,2), gbest(1,3), gbest(1,4), gbest(1,5));
```

```
figure
```

```
imshow(GW);
```

```
load -mat yuzler
```

```
for kisi=1:16
```

```
    for poz=1:3
```

```
        filtered_imge(:,:,kisi,poz) = imfilter(Kisiler(:,:,kisi,poz),GW);
```

```
    end
```

```
end
```

```
k=1;
```

```
figure
```

```
for kisi=1:16
```

```
    for poz=1:3
```

```
        subplot(6,8,k)
```

```
        imshow(filtered_imge(:,:,kisi,poz));
```

```
        title(strcat('f_kisi: ',num2str(kisi),' f_poz: ',num2str(poz)));
```

```
        k=k+1;
```

```
    end
```

```
end
```

```
save filtrelenmis_opt filtered_imge;
```

```
for ks=1:16;
```

```
    for pz=1:3
```

```
        test_imge(:,:,ks,pz)
```

```
double(imread(strcat('C:\MATLAB\AT&T_Faces\test\test',num2str(ks),',',num2str(pz),'.jpeg')));
```

```
        filtered_test(:,:,ks,pz) = imfilter(test_imge(:,:,ks,pz),GW);
```

```
    end
```

```
end
```

```
save test_kumesi test_imge
```

```
k=1;
```

```
figure
```

```
for ks=1:16
```

```
    for pz=1:3
```

```
        subplot(6,8,k)
```

```
        imshow(filtered_test(:,:,ks,pz));
```

```
        k=k+1;
```

```
    end
```

```
end
```

```

% PCA ile en yakin resmin belirlenmesi
% -----
k=1;
for kisi=1:16
    for poz=1:3
        filt_im(:,:,k)=filtered_imge(:,:,kisi,poz);
        filt_test(:,:,k)=filtered_test(:,:,kisi,poz);
        k=k+1;
    end
end

% ortalamanin hesaplanmasi
% -----
ort_im=zeros(112,92);
for k=1:48
    ort_im=ort_im + filt_im(:,:,k);
end
ort_im=ort_im./48;

figure
imagesc(ort_im(:,:,)),colormap('gray');

for k=1:48
    filt_im_ort(:,:,k)=filt_im(:,:,k)-ort_im;
    filt_test_ort(:,:,k)=filt_test(:,:,k)-ort_im;
end

for sayi=1:48
    A(:,sayi)=reshape(filt_im_ort(:,:,sayi),(112*92),1);
end

L=A'*A;
[U,Lmd]=eig(L);

for l=1:48
    eigen(:,l)=zeros(112,92);
    for n=1:48
        eigen(:,l)=eigen(:,l)+U(l,n)*filt_im_ort(:,:,n);
    end
end

for k=1:48
    for eg=1:48
        w(:,:,eg,k)=eigen(:,:,eg).'* filt_test_ort(:,:,k);
        W(:,:,eg,k)=eigen(:,:,eg).'* filt_im_ort(:,:,k);
    end
end

```

```

for sayi=1:48
    for eg=1:48
        for k=1:92
            for i=1:92
                t=92*eg-92;
                omega(i+t,k,sayi)=w(i,k,eg,sayi);
                OMG(i+t,k,sayi)=W(i,k,eg,sayi);
            end
        end
    end
end

for sayi=1:48
    for eg=1:48
        eps(eg,sayi)=norm(omega(:, :, sayi)-OMG(:, :, eg))/norm(OMG(:, :, eg));
    end
end
save hata eps;

```

EK-D

```

% Tez Adi      : Yuz Tanimada Kullanilan Oznitelik Vektorlerinin Optimizasyonu
%              : Kisim-4 (Temel Bileşen Analizi ile Kisinin Tespit Edilmesi – Asama 2)
% Yazar       : Serkan TUZUN
% e-posta     : eng.serkantuzun@yahoo.com

```

```

close all
clear all
clc

```

```

load hata;

```

```

k=1;
for sayi=1:3
    subplot(3,1,k)
    stem(eps(:,sayi))
    k=k+1;
end

```

```

eps1=eps;%sqrt(eps);

```

```

for sayi=1:48
    for eg=1:16
        eps2(eg,sayi)=eps1(((eg-1)*3+1),sayi)+eps1(((eg-1)*3+2),sayi)+eps1(((eg-1)*3+3),sayi);
    end
end

```

```

for sayi=1:16
    for eg=1:16
        eps3(eg,sayi)=eps2(eg,((sayi-1)*3+1))+eps2(eg,((sayi-1)*3+2))+eps2(eg,((sayi-1)*3+3));
    end
end

```

```

k=1;
figure
for sayi=1:4
    subplot(4,1,k)
    stem(eps2(:,sayi))
    k=k+1;
end

```

```

dogru=0;
for sayi=1:48
    kkisi(sayi)=find(eps2(:,sayi)==min(eps2(:,sayi)));
    if kkisi(sayi)==double((int8(sayi-1)/3)+1)
        dogru=dogru+1;
    end
end

```

```

figure
stem(kkisi)

```

```

for sayi=1:16
    kkkisi(sayi)=find(eps3(:,sayi)==min(eps3(:,sayi)));
end

```

ÖZGEÇMİŞ

Serkan TÜZÜN, 1982 yılında İstanbul'da doğdu. İlk ve orta öğrenimini İstanbul'da Bahçelievler İlköğretim Okulu ve Osman Ülkümen YDA Lisesi'nden sırasıyla 1996 ve 2000 yılında mezun olarak tamamladı. Aynı yıl İstanbul Üniversitesi, Mühendislik Fakültesi, Elektronik Mühendisliği Bölümü'nde lisans öğrenimine başladı. 2005 yılında lisans öğrenimini tamamladı. 2005-2006 Akademik yılında İstanbul Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik-Elektronik Mühendisliği Ana Bilim Dalı'nda Yüksek Lisans öğrenimine başladı. 21 Şubat 2006 tarihinde İstanbul Üniversitesi, Döner Sermaye İşletme Müdürlüğü'ne bağlı Biyomedikal ve Klinik Mühendisliği Birimi'nde göreve başladı. 30 Kasım 2008 tarihinde Diyarbakır, 7. Kolordu, Aile Destek Birimi'nde İdari İşler Kısım Amiri vazifesiyle yerine getirdiği askerlik hizmetini Mu. Tğm. olarak tamamladı. 28 Eylül 2009 tarihinden beri Biyomedikal ve Klinik Mühendisliği Birimi'ndeki görevine devam etmektedir.