

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**RADYAL TABAN FONKSİYONLU YAPAY SİNİR
AĞI KULLANARAK ZEKİ BİR İMZA TANIMA
SİSTEMİ TASARIMI**

YÜKSEK LİSANS TEZİ

Armağan Ebru TEMİZ

**Enstitü Anabilim Dalı : ELEKTRONİK VE BİLGİSAYAR
EĞİTİMİ**
Tez Danışmanı : Prof. Dr. Raşit KÖKER

Temmuz 2015

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**RADYAL TABAN FONKSİYONLU YAPAY SİNİR
AĞI KULLANARAK ZEKİ BİR İMZA TANIMA
SİSTEMİ TASARIMI**

YÜKSEK LİSANS TEZİ

Armağan Ebru TEMİZ

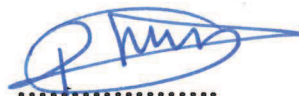
Enstitü Anabilim Dalı

:

**ELEKTRONİK VE BİLGİSAYAR
EĞİTİMİ**

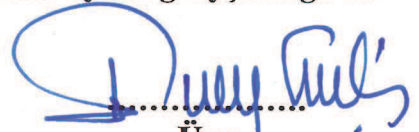
Bu tez 13/07/2015 tarihinde aşağıdaki jüri tarafından oybirliği/oyçokluğu ile kabul edilmiştir.

Prof. Dr. Abdülhak PERİKOĞLU
Jüri Başkanı



Üye

Prof. Dr. Rasit KÖKER



Üye

BEYAN

Tez içindeki tüm verilerin akademik kurallar çerçevesinde tarafımdan elde edildiğini, görsel ve yazılı tüm bilgi ve sonuçların akademik ve etik kurallara uygun şekilde sunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, tezde yer alan verilerin bu üniversite veya başka bir üniversitede herhangi bir tez çalışmasında kullanılmadığını beyan ederim.

Armağan Ebru TEMİZ

13.07.2015

ÖNSÖZ

Gizlilik arz eden bir durumda, bir cihaza ya da belirli kişi ya da kişilerin ulaşılması istendiğinde güvenlik sistemlerinin önemi büyüktür. Bu gibi durumlarda güvenlik sistemlerinin doğru bir şekilde tanıma gerçekleştirmesi gerekmektedir. En çok kullanılan güvenlik araçlarından biri de şifrelemedir. Ancak şifrelerin kırılabilmesi ve kötü amaçlı kişilerin eline geçmesi, istenmeyen durumlar meydana getirebilir. Bu gibi olumsuz durumlarla karşılaşmamak için, bir bireyin kişisel bir nitelik ya da davranışını analiz ederek kimliğini açıklayan biyolojik verileri doğrulama birimi olan biyometrik sistemler geliştirilmiştir. İmza tanıma en popüler biyometrik sistemler arasında yer almaktadır. Birey birçok durumda kendini temsi eden imzasını atmaktadır. Bu yüzden imzaların kimin tarafından atıldığına bilinmesi oldukça önemlidir. Bu durum, imza tanıma ve doğrulama sistemlerinin geliştirilmesi gerekliliğini doğurmuştur.

“Bu çalışma Sakarya Üniversitesi Bilimsel Araştırma Projeleri Komisyonu Tarafından Desteklenmiştir. Proje Numarası:2013-50-01-007”

İÇİNDEKİLER

| | |
|--|-----|
| ÖNSÖZ | i |
| İÇİNDEKİLER | ii |
| SİMGELER VE KISALTMALAR LİSTESİ | v |
| ŞEKİLLER LİSTESİ | vii |
| TABLolar LİSTESİ | x |
| ÖZET. | xi |
| SUMMARY | xii |
| BÖLÜM 1. | |
| GİRİŞ | 1 |
| BÖLÜM 2. | |
| GÖRÜNTÜ İŞLEME VE ÖZELLİK ÇIKARIMI | 11 |
| 2.1. Biyometrik Kimlik Tanıma | 11 |
| 2.1.1. Görüntünün oluşturulması | 14 |
| 2.1.2. Görüntüyü sayısallaştırma | 15 |
| 2.1.3. Nicemleme | 15 |
| 2.1.4. Görüntü bölümlenme | 18 |
| 2.1.4.1. Görüntü eşikleme | 18 |
| 2.1.5. Görüntü iyileştirme | 20 |
| 2.1.5.1. Morfolojik işlemler | 20 |
| 2.2. Özellik Çıkarımı | 21 |
| 2.2.1. Global özellikler | 23 |
| 2.2.1.1. Alan | 23 |
| 2.2.1.2. Genişlik | 23 |
| 2.2.1.3. Yükseklik | 23 |

| | |
|--|----|
| 2.2.1.4. Yükseklik/Genişlik oranı..... | 24 |
| 2.2.1.5. Ağırlık merkezi | 24 |
| 2.2.1.6. Görüntünün ağırlık merkezlerinden bölünmesi | 25 |
| 2.2.1.7. Boyut merkezi:..... | 25 |
| 2.2.1.8. Yoğunluk:..... | 26 |
| 2.2.1.9. Merkezler arası görelî yatay fark | 26 |
| 2.2.1.10. Merkezler arası görelî dikey fark..... | 26 |
| 2.2.2. Moment sabitleri..... | 27 |
| 2.2.2.1. Ölçeklendirilmiş sabit momenler | 30 |
| 2.2.2.2. Dönme sabit momentleri..... | 30 |
| 2.2.3. Dikey özellik çıkararak blok algoritma | 31 |
| | |
| BÖLÜM 3. | |
| YAPAY SİNİR AĞI TABANLI SINIFLANDIRMA | 32 |
| 3.1. Şablon Eşleştirme..... | 32 |
| 3.2. Spektrum Analiz Yaklaşımı..... | 32 |
| 3.3. Yapısal Yaklaşım | 33 |
| 3.4. İstatiksel Yaklaşım | 33 |
| 3.4.1. Klasik sınıflandırma..... | 34 |
| 3.5. Yapay Sinir Ağları | 36 |
| 3.5.1. Yapay sinir ağları yapısı | 37 |
| 3.5.2. Yapay sinir ağları genel özellikleri | 41 |
| 3.6. Radyal Taban Fonksiyonlu Yapay Sinir Ağı | 41 |
| | |
| BÖLÜM 4. | |
| İMZA TANIMA UYGULAMASI | 49 |
| 4.1. İmzaların Veritabanına Alınması..... | 49 |
| 4.2. Ön İşlemler | 54 |
| 4.3. Özellik Çıkarma | 57 |
| 4.4. Radyal Tabanlı Fonksiyonlu Yapay Sinir Ağı..... | 61 |
| 4.5. Analiz Çalışmaları..... | 63 |

| | |
|----------------------------|----|
| BÖLÜM 5. | |
| SONUÇLAR VE ÖNERİLER | 73 |
| KAYNAKLAR..... | 75 |
| EKLER..... | 82 |
| ÖZGEÇMİŞ | 92 |

SİMGELER VE KISALTMALAR LİSTESİ

| | |
|------------|---|
| μ_{pq} | : P ve q noktasındaki merkezi moment |
| \bar{x} | : X düzlemindeki ağırlık merkezi |
| \bar{y} | : Y düzlemindeki ağırlık merkezi |
| AER | : Ortalama hata oranı(average error rate) |
| Agmerx | : Görüntünün x koordinat düzleminde ağırlık merkezi |
| Agmery | : Görüntünün y koordinat düzleminde ağırlık merkezi |
| f(x,y) | : Görüntü |
| FAR | : Yanlış kabul oranı (false acceptance rate) |
| FRR | : Yanlış ret oranı (false rejection rate) |
| G | : Gri düzey |
| GPDS | : Dijital sinyal işleme grubu (the digital signal processing group) |
| I(x,y) | : Gri seviyeli görüntü |
| K | : Piksel yoğunluğunda kullanılan bit sayısı |
| M,N | : Dizin boyutları |
| MatLab | : Matris laboratuvarı programı (matrix laboratory program) |
| M_{ij} | : İ ve j noktalarındaki moment |
| n_{ij} | : Ölçeklendirilmiş sabit moment |
| NTSC | : Ulusal televizyon standardı komitesi (national television system committee) |
| RGB | : Kırmızı, yeşil, mavi (red,green,blue) |
| x_1 | : Görüntünün x düzlemindeki minimum noktası |
| x_2 | : Görüntünün x düzlemindeki maksimum noktası |
| x^p | : P derecesindeki x noktası |
| y_1 | : Görüntünün y düzlemindeki minimum noktası |
| y_2 | : Görüntünün y düzlemindeki maksimum noktası |

YIQ : Y değeri parlaklık miktarını r ve q bileşenleri renk değerlerini gösteren amerika renkli televizyonlarda kullanılan renk modeli
y^q : Q derecesindeki y noktası
YSA : Yapay sinir ağıları

ŞEKİLLER LİSTESİ

| | |
|---|----|
| Şekil 2.1. Görüntü işleme aşamaları | 13 |
| Şekil 2.2. Görüntü işleme teknikleri sınıflandırması | 14 |
| Şekil 2.3. Dijital görüntü elde etme aşaması (a) Enerji kaynağı (b) Görüntülenecekobje (c) Görüntü sistemi (d) Sayısal görüntü | 15 |
| Şekil 2.4. a) Eşit karelere bölünerek koordinatları belirlenmiş görüntü b) Örneklenmiş ve nicelenmiş görüntü sonucu | 16 |
| Şekil 2.5. A.T. adlı kişinin imzasının 256 gri düzeyinde, griye dönüştürülmüş hali . | 18 |
| Şekil 2.6. A.T. isimli kişinin imzasının siyah beyaza dönüştürülmüş hali | 20 |
| Şekil 2.7. Morfolojik iskelet çıkarma işlemi uygulanmış A.T. adlı kişinin imzası.... | 21 |
| Şekil 2.8. Şekil temsilcisi ve tanımlayıcısı | 22 |
| Şekil 2.9. A.T. adlı kişinin imzasının genişliğinin bulunması | 23 |
| Şekil 2.10. A.T. adlı kişinin imzasının yüksekliğinin bulunması | 24 |
| Şekil 2.11. A.T. adlı kişinin imzasının ağırlık merkezi | 25 |
| Şekil 2.12. A.T. adlı kişinin imzasının boyut merkezi..... | 26 |
| Şekil 2.13. A.T. adlı kişinin imzasının ağırlık merkezi ve boyut merkezi arasındaki fark | 27 |
| Şekil 3.1. Klasik sınıflayıcı (a) Sinir ağı sınıflayıcısı (b)..... | 35 |
| Şekil 3.2. Biyolojik sinir sistemi yapısı | 38 |
| Şekil 3.3. Biyolojik bir hücre | 38 |
| Şekil 3.4. Yapay sinir ağı hücresi | 39 |
| Şekil 3.5. Çok katmanlı geriye yayımlı sinir ağları | 40 |
| Şekil 3.6. Radyal taban fonksiyonlu sinir ağı | 43 |
| Şekil 3.7. İki boyutlu desenler ve kümeleme (a) klasik kümeleme (b) homojen analiz ile kümeleme..... | 45 |
| Şekil 3.8. Gauss genişliğinin kümelemeye etkisi..... | 46 |

| | |
|---|----|
| Şekil 4.1. İmza örnekleri a) A.C.T. adlı kişiye ait imza örnekleri b) A.S.T. adlı kişiye ait imza örnekleri c) A.T. adlı kişiye ait imza örnekleri d) S.T. adlı kişiye ait imza örnekleri | 50 |
| Şekil 4.2. İmza tanıma uygulaması arayüzü | 51 |
| Şekil 4.3. İmza tanıma uygulaması ilk çalıştırıldığındaki arayüzü | 52 |
| Şekil 4.4. İmza tanıma uygulamasındaki giriş menüsü..... | 52 |
| Şekil 4.5. İmza tanıma uygulamasındaki ön_işlemler menüsü | 53 |
| Şekil 4.6. İmza tanıma uygulamasındaki kontrol menüsü | 53 |
| Şekil 4.7. Test edilmesi istenen imza açıldıktan sonra imza tanıma uygulaması arayüzü | 54 |
| Şekil 4.8. Alınan görüntü örneği | 54 |
| Şekil 4.9. A.T. adlı kişinin imzasının 8 bitlik gri düzeyinde griye çevrilmiş hali | 55 |
| Şekil 4.10. Eşik değeri girilmediğinde karşılaşılan hata mesajı | 55 |
| Şekil 4.11. Eşik değeri 100 olan A.T. adlı kişinin imzasının ikili görüntüsü | 55 |
| Şekil 4.12. Eşik değeri 150 olan A.T. adlı kişinin imzasının ikili görüntüsü | 56 |
| Şekil 4.13. Eşik değeri 250 olan A.T. adlı kişinin imzasının ikili görüntüsü | 56 |
| Şekil 4.14. Otsu metoduyla elde edilmiş ikili görüntü..... | 56 |
| Şekil 4.15. Morfolojik iskelet çıkarma uygulanmış imza | 57 |
| Şekil 4.16. Özellikler çıkarılırken kullanılan imza | 57 |
| Şekil 4.17. Özellikler çıkarıldıktan sonra imza tanıma uygulamasının görürüsü ... | 60 |
| Şekil 4.18. Yayılım değeri veya maksimum sinir sayısı girilmediğinde karşılaşılan hata mesajı | 60 |
| Şekil 4.19. İmza tanıma için oluşturulan radyal taban fonksiyonlu sinir ağı yapısı .. | 62 |
| Şekil 4.20. Ağın eğitilmesi ile performans epochs grafiği..... | 62 |
| Şekil 4.21. İmza tanıma uygulaması performans analiz bölümü | 63 |
| Şekil 4.22. Yayılım değeri 1 alındığında değişik maksimum sinir sayılarındaki performans değerleri | 65 |
| Şekil 4.23. Yayılım değeri 5 alındığında değişik maksimum sinir sayılarındaki performans değerleri | 66 |
| Şekil 4.24. Yayılım değeri 10 alındığında değişik maksimum sinir sayılarındaki performans değerleri | 67 |

| | |
|---|----|
| Şekil 4.25. Yayılım değeri 15 alındığında değişik maksimum sinir sayılarındaki performans değerleri | 68 |
| Şekil 4.26. Yayılım değeri 20 alındığında değişik maksimum sinir sayılarındaki performans değerleri | 69 |
| Şekil 4.27. Yayılım değeri 25 alındığında değişik maksimum sinir sayılarındaki performans değerleri | 70 |
| Şekil 4.28. Yayılım değeri 1, 5, 10, 15, 20, 25 olduğunda her bir değeri için maksimum sinir sayısı (50, 100, 150, 225, 230, 235, 250, 300, 400, 550) olduğundaki performans grafiği..... | 71 |
| Şekil 4.29. Maksimum sinir sayısı 50, 100, 150, 225, 230, 235, 250, 300, 400, 550 olduğunda her bir değeri için yayılım değeri (1, 5, 10, 15, 20, 25) performans grafiği | 72 |

TABLolar LİSTESİ

| | |
|--|----|
| Tablo 2.1. Gri düzeyi 256 olarak belirlenmiş [8,8] boyutlarındaki bir görüntü ve piksel yoğunlukları | 17 |
| Tablo 3.1. Sinir sistemiyle YSA sistemi karşılaştırması..... | 39 |
| Tablo 4.1. 8 kişinin 1. imzalarına ait 0-1 arası normalize edilmiş 30 tane özellik | 59 |
| Tablo 4.2. Yayılım değeri 1 alındığında maksimum sinir sayısı değiştirilerek elde edilen performans tablosu | 64 |
| Tablo 4.3. Yayılım değeri 5 alındığında maksimum sinir sayısı değiştirilerek elde edilen performans tablosu | 65 |
| Tablo 4.4. Yayılım değeri 10 alındığında maksimum sinir sayısı değiştirilerek elde edilen performans tablosu | 66 |
| Tablo 4.5. Yayılım değeri 15 alındığında maksimum sinir sayısı değiştirilerek elde edilen performans tablosu | 67 |
| Tablo 4.6. Yayılım değeri 20 alındığında maksimum sinir sayısı değiştirilerek elde edilen performans tablosu | 68 |
| Tablo 4.7. Yayılım değeri 25 alındığında maksimum sinir sayısı değiştirilerek elde edilen performans tablosu | 69 |

ÖZET

Anahtar kelimeler: Matlab, yapay sinir ağırları, radyal taban fonksiyonu, çevirim dışı imza tanıma

Biyometrik sistemler bir bireyin kişisel bir nitelik ya da davranışını analiz ederek kimliğini açıklayan biyolojik verileri doğrulama bilimidir. İnsanları birbirinden ayırt edebilme şansını bize sunduğundan dolayı biyometri bir kimlik doğrulama sistemi olarak da kullanılmaktadır. En popüler biyometrik sistemlerden biri de imza tanıma ve doğrulamadır. Bu çalışmada çevirim dışı imza tanıma sistemi için bir uygulama gerçekleştirilmiştir. Çevirim dışı olarak 24 ayrı kişiden 36 tane, yani toplamda 864 tane imza toplanmıştır. 36 tane imzanın 26 tanesi yani toplamda 624 tane imza eğitim için, 10 tanesi yani 240 tane imza test için ayrılmıştır.

İmza tanıma uygulaması boyunca, ilkönce imza görüntüleri bir tarayıcı yardımıyla 450X250 boyutlarında alınmıştır. Bu görüntüler gri seviyeli görüntüye çevrilmiştir. Ondan sonra Otsu otomatik eşik seçme metoduyla ikili görüntüye çevrilmiştir. Bundan sonra, kenar inceltme metoduyla, ikili imza görüntüleri inceltirilmiştir. İmza görüntülerinin özellikleri bundan sonra imzanın çevresindeki gereksiz boşluklar çıkarılarak, sahip oldukları boyutlarda bulunmuştur. İmzaların yoğunluk, genişlik, yükseklik, genişlik yükseklik oranı, x eksenindeki ağırlık merkezi, y eksenindeki ağırlık merkezi, genişliğinin x eksenindeki orta noktası, yüksekliğinin y eksenindeki orta noktası, x eksenindeki ağırlık merkezi ile genişliğinin orta noktası arasındaki fark, y eksenindeki ağırlık merkezi ile yüksekliğinin orta noktası arasındaki fark özellik çıkarma metotları kullanılmıştır. Ve bundan başka imzalar ağırlık merkezlerinden 4 eşit parçaya bölünmüştür. Ve bu her bir parça tekrar ağırlık merkezlerinden 4 eşit parçaya bölünmüştür.

İmzaların sınıflandırılması radyal taban fonksiyonlu sinir ağırlı tasarımları ve kullanılmıştır. Tasarımları RBF sinir ağırlı, imza sınıflarına ait özelliklere dayalı 30 öz giriş ve 24 çıkış kullanılmıştır. Çalışmada 91.6667 % sınıflandırma başarımları gözlenmiştir. Doğrulama işlemi gerçekleştirilmemiştir. Sinir sayılarının maksimum sayısı ve yayılım değeri analiz edilmiştir. Yayılım değeri 1, 5, 10, 15, 20, 25 ve maksimum sinir sayısı 50,150, 225, 230, 235, 250, 300, 400, 550 olarak değiştirilmiştir. Yayılım değeri 1 ve maksimum sinir sayısı 225 veya 235 olduğunda en iyi performansa ulaşılmıştır. Bu çalışmaların hepsi tablo, grafik ve şekillerle gösterilmiştir.

DESIGN OF AN INTELLIGENT SIGNATURE RECOGNITION SYSTEM BY USING RADIAL BASIS FUNCTION NEURAL NETWORK

SUMMARY

Keywords: Matlab, artificial neural networks, radial basis function, offline signature recognition

Biometric systems are being verified that analyzing personal character or behavior so describing identify. Biometrics is used as an authentication system because of providing to distinguish between people. One of the most popular biometric systems is signature recognition and verification systems.

During the study, 864 signatures were collected offline. These signatures were taken from 24 different people. 36 signatures were collected from each person. 26 signatures have been used for the training process and other remaining signatures have been used for testing. During the implementation of the signature recognition, firstly the signature images have been taken to computer by using a scanner as 450x250 images. These images have been firstly converted to gray level image. Then, these images have been converted to binary images by using Otsu automatic threshold selection method. After that edge thinning operation has been applied to the binary signature images. Signature features of images have been found in the size that they have by removing unnecessary spaces around after the signature. Signature's density, width, height, ratio of width and height, center of gravity at x axis, center of gravity at y axis, midpoint of width, midpoint of height, difference between center of gravity at x and midpoint of width, difference between center of gravity at y and midpoint of height feature extraction methods have been used. And also, the signatures have been divided into mainly 4 pieces based on the geometric centroid of the signature image. Then, each part has been again divided into 4 pieces based on their centroid.

For the classification of the signatures a radial bases neural network (RBFNN) has been designed and used. Designed RBF neural network has 30 inputs based on used features and 24 outputs belonging to signature classes. 91.6667 % classification performance have been observed during the study. Verification process has not been implemented. Effect of the maximum number of neurons and spread values has analyzed. Spread values have changed 1, 5, 10, 15, 20, 25 and maximum number of neurons has been changed 50, 150, 225, 230, 235, 250, 300, 400, 550. When spread is 1 and maximum number of neurons has been 225 or 235 that the best performance has obtained. All of these have showed with tables, graphics and shapes.

BÖLÜM 1. GİRİŞ

Gizlilik arz eden bir durumda, bir cihaza ya da yalnızca belirli kişi ya da kişilerin ulaşılması istendiğinde güvenlik sistemlerinin önemi büyüktür. Bu durumlarda güvenlik sistemlerinin doğru kişi ya da kişileri tanınması gerekmektedir. En çok kullanılan güvenlik araçlarından biri de şifrelemedir. Ancak şifrelerin kırılabilmesi ve kötü amaçlı kişilerin eline geçmesi, istenmeyen sonuçlar doğurabilmektedir. Bu gibi olumsuz durumlar biyometrik sistemlerin geliştirilmesini sağlamıştır [1].

Biyometrik tanıma, kişiye ait fiziksel veya davranışsal tepkilerin verileştirilerek tanımlanması ve karşılaştırılması anlamına gelmektedir. Bu özellikler her kişi için ayrıdır. Bu nedenden dolayı güvenlik gerektiren sistemlerde şifre, manyetik kart, vs gibi sistemler asla biyometrik sistemler kadar güvenli değildir [2]. Elle atılmış imza, parmak izi, iris, retina ve yüz tanıma ve doğrulama, biyometrik tanıma yöntemlerinden en bilinenleridir. Biyometrik sistemlerin çalışma prensibi birbirine benzer. Kayıtlar toplanır ve bu kayıtlar gerektiğinde ilgili kişinin kaydıyla karşılaştırılır ve sonuca varılır. Kayıtlar veritabanında saklanırken ihtiyaç dâhilinde çeşitli ön işlemlerden geçirilerek gereksiz bilgilerden arındırılır. Daha sonra kayıtların özellikleri çıkarılır. Biyometrik sistemlerde, özellik çıkarma en önemli işlemdir. Çünkü sistemin performansını doğrudan etkiler [3]. Bu özellikler daha sonra sınıflandırma aşamasına kullanılırlar [4].

Bir kişi herhangi bir yazıyı okuduğunda, okuduğunu belirtmek, şartları kabul etmek, gibi nedenlerden dolayı, kendini belirtmek amacıyla, her zaman aynı biçimde yazdığı yazı ve işaretler olarak tanımlanan imza kullanırlar. Kişiler hayatları boyunca binlerce kez imza kullanmışlardır. Fakat kişinin imzasının taklit edilmesi büyük bir sorundur. Kişiyi oldukça zor durumda bırakabilir, sahip olmadığı borçlar ya da

işlemediği suçlar üzerine kalabilir. Yani imzanın gerçekten o kişi tarafından atılıp atılmadığının belirlenmesi oldukça önemli, çözülmesi gerekli bir sorundur [4].

İmza tanıma içinde formüle edilmesi çok zor kurallar içerir. İmza tanıma, atılan imzanın kime ait olduğunun bulunması ve böylece sahte imzaların da ortaya çıkmasını sağlar [5]. İmza tanıma sistemleri çevrimiçi ve çevirimdışı yapılabilir. Çevrimiçi sistemlerde imza işlemi boyunca imzalar o anda dijital tabletlerden, bu iş için yapılmış kalemlerle veya diğer özel donanımlarla gerçekleştirilir [6]. Bunun sonucunda dinamik özellikler ortaya çıkar. Dinamik özellikler, dokunmatik kalemin her noktadaki basıncı, kalem hızı, yapılan vuruş sayısı, imza atış hızı gibi özelliklerdir. Çevirimdışı sistemler imzanın sabit özelliklerine dayanır. Bu sistemlerde kâğıt üzerine atılmış bir imza bu iş için yeterli olacaktır. Bu imzalar bir tarayıcı ya da bir kamera yardımıyla dijital ortama atılabilir. İmza doğrulama tipik bir örüntü tanımadır. Finansal zorluklardan dolayı, çevirimdışı sistemler çevrimiçi sistemlere göre daha yararlı ve kullanışlı görülmektedir [5]. Çevrimiçi ya da çevirimdışı olarak elde edilen görüntülerin biyometrik sisteme sokmadan önce iyileştirme çalışmaları yapılmalıdır. Çünkü görüntünün gürültülerden ve gereksiz kısımlarından arındırılması, sistemin daha hızlı ve dolayısıyla daha kısa zamanda çalışmasını sağlar [7].

Bir kişinin imzası her zaman tutarlı olmayan bir niteliğe sahiptir. Tek bir kişi tarafından tekrarlansa da, belirli bir ölçüde değişir. Biyometrik nitelikli olan imza, eğer taklit edilebiliyorsa, o zaman araştırmacı sistemini tasarlarken, birçok şeyle mücadele etmek zorunda kalır. Bunlar içsel ve kişiler arası değişimlerdir. Bu konuda birçok araştırma yapılmış ve bunlar aşağıda listelenmiştir.

Baltzakis ve Papamarkos [8], çevirimdışı imza doğrulamada iki tane sinir ağı sınıflandırıcı kullanmışlardır. Bu sistem genel, ızgara, metin özelliklerini içerir. Bu özellik kümesinin her biri özel iki bölümlü algılayıcı için geliştirilir. Birinci bölümdeki sınıflandırıcı, sinir ağlarının sonucundaki kararı ve elde edilen üç özellik kümesinin öklid mesafesini birleştirir. Birinci aşamadaki sınıflandırıcının sonuçlarının dönütleri ile ikinci bölümdeki radyal taban fonksiyonlu sinir ağı ile son

kararlar verilir. Sistemde üç tane alt gruba ayrılmış 160 tane özellik elde edilmiştir. FAR=%9,81 ve FRR=%3'tür.

Shashi Kumar ve ark. [9], yapay sinir ağlarını kullanarak global ve ızgara özelliklerinin birleşmesine dayanan çevrimdışı imza doğrulama sistemini tanıttı. Global ve ızgara özellikler, yapay sinir ağları sınıflandırmada kullanılmak üzere güçlü bir özellik kümesi yaratmak için birleştirilmiştir. FAR=%4,56 ve FRR=%7,51'dir.

Basavaraj ve Sudhaker Samuel [7], çalışmalarında yeni bir çevrimdışı imza tanıma sistemi önermişlerdir. Önermiş oldukları sistem, statik imzadan vuruş hızı çıkartılarak dinamik özelliklerin elde edilmesine dayanmaktadır. İmzanın vuruşu yavaştan hızlıya değişmiyorsa ya da tam tersi hızlıdan yavaşta değişmiyorsa, normal bir şekilde devam ediyorsa, bir hızdan diğerine yumuşak geçiş vardır. Bu metotta hız seviyeleri hızlı, çok hızlı, yavaş, çok yavaş olarak belirtilir. Buna göre imzanın yoğunluğu direk olarak imzanın vuruş hızıyla alakalıdır. Yani daha hızlı vuruş imzanın daha yoğun olması demektir. Bundan sonra, birçok deney yapılarak sahte imzalardan, gerçek imzaları ayırt edebilme yeteneği, önerilen şemada gösterilmiştir. Bu yöntemin başarısında FAR=%13,78 ve FRR=%14,25'dir.

Prashanth ve Raja [10], açısal özellikler temelli, çevrimdışı imza doğrulama sistemini önermiştir. Taranmış imza görüntüsünün iskeleti çıkarılır ve tam imza alanı ön işlemlerle elde edilir. İlk aşamada imzanın merkezi kullanılarak 128 tane parçaya bölünür. Her bloktaki siyah piksellerin sayısının sayılmasıyla ve her bir bloktaki açısal özelliğin belirlenmesiyle 128 açısal özellik oluşturulur. İkinci kısımda, imza parçalara bölünür. Bu imzanın dört köşesinin her birinin bölünmesiyle olur. Böylece 40 tane açısal özellik elde edilir. Toplamda 168 tane özellik imza doğrulamasında kullanılır. Eşik değeri, orijinal ve sahte imzayı karşılaştırmak üzere ayarlanır. FAR=%4,995 ve FRR=%8,5 bulunmuştur.

Abdala ve Yousif [11], iki yapay sinir ağı sınıflandırıcı bir sistem önermişlerdir. Bu sistem üç bölümden oluşmaktadır. Birinci bölüm ön işleme bölümü, ikinci bölüm

global, metin ve ızgara özelliklerinin çıkarıldığı, özellik çıkarma bölümü, son kısım da sinir ağı (sınıflandırıcı) bölümüdür. Sınıflandırıcı bölümü de iki tane sınıflandırıcı içerir. İlk sınıflandırıcı üç geriye yayımlı yapay sinir ağı ve ikinci sınıflandırıcı iki radyal tabanlı fonksiyon sinir ağı içermektedir. İmzanın sahte ya da gerçek olduğuna dair son karar ikinci sınıflandırıcıdan alınır. Eğer ilk sınıflandırıcıda iki tane geriye yayımlı yapay sinir ağı kullanılırsa sistemin imzayı tanıma oranı %95.95'dir. Bu sınıflandırıcıda üç tane geriye yayımlı yapay sinir ağı kullanılırsa, bu oran %99.31'e ulaşır.

Vargas ve ark. [12] tarafından, çevrimdışı imza doğrulama sistemi için elle atılmış statik imzadan basınç dağılımlarını gösteren özellikler analiz edilmiştir. Sahte septiral katsayılarına dayalı çevrimdışı imza doğrulama sistemi gerçekleştirilmiştir. Bu teknikte gri düzeydeki görüntüler ve onların histogramı hesaplanır. Sahte sepstral katsayılarının hesaplanması için "spektrum" kullanılır. Son olarak eşsiz en küçük faz dizisi tahmin edilebilir ve bu imza doğrulamada özellik faktörü olarak kullanılır. Sahte katsayıların en uygun sayısı, en iyi sistem performansı olarak tahmin edilebilir. Analiz edilen sistemin sağlamlığı basit sahte imzalarla bir LS-SVM (Least Squares-Support Vector Machine) modeli kullanılarak test edilmiştir. Deneyler 100 kişiden alınan imzalarla gerçekleştirilmiştir. EER %6,20'dir.

Fasquel ve Bruynooghe [13], bazı istatiksel sınıflandırıcıları birleştirerek bir gerçek zamanlı, otomatik imza doğrulama sistemi önerdi. Bu yöntem birçok istatiksel sınıflandırıcının birleştirilmesiyle oluşturulmuştur. İmza doğrulama sistemi imzanın dönüştürülmesi, karakterize edilmesi ve sınıflandırılması olarak üç adımdan oluşuyordu. Birinci adımda kimlik belirleme ve dörtlü Gabor dönüşümü kullanılarak orijinal imzalar dönüştürülür. İkinci adımda öğrenme veritabanında benzer şekilde dönüştürülmüş imzalar ile dönüşüme uğramış imzalar birbirleriyle ilişkilendirilir. Ve üçüncü adımda her dönüşümdeki ilgili kararların birleştirilmesiyle, imzaların kime ait olduğu doğrulanır. Önerilen sistem, denemeler sonucunda imzaların %62,4 oranında reddedilmesini sağlamıştır. Orijinal imzaların %99'u tanınmıştır. FAR=%2,56 ve FRR=%1,43 olarak bulunmuştur.

Julio ve Rogelio [14], en uygun özellik temsil edilmesine dayanan ve sinir ağı odaklı bulanık mantığa dayalı yeni bir çevrimiçi imza doğrulama sistemini tanıttı. Bir kişinin bir referans imza modelini oluşturmak için, orijinal imzalardan dinamik ve şekil özellikleri çıkartıldı. Daha sonra her ayırt edici özellik için, genetik uygunluk kullanılarak bir ortalama prototip ve tutarlılık fonksiyonu hesaplandı. En uygun özelliğin temsili kavramından türetilen bu işlemle, $FRR=1.05$ ve $FAR=0.27$ olarak bulunur.

Nguyen ve ark. [15], imzanın projeksiyonları ve sınırının temel alındığı temel özelliklerin çıkarılması ile imza doğrulama sistemi için SVM (Support Vector Machine) kullanımını önermiştir. İlk global özellik toplam enerjiden türetilir ve yazar bunu onların imzalarını oluşturmak için kullanır. İkinci özellik olarak imzanın anahtar vuruşları, yani her ikisi de yatayda veya her ikisi de dikeyde olmak şartıyla siyah piksellerin en yüksek indekse sahip olduğu yerler arasındaki mesafe bulunur. Bunun için imzanın yatay ve dikeydeki projeksiyonlarının bilgisi kullanılır. Ve bir de imzanın genişlik, yüksekliği özellikleri bulunur. Modifiye Yön Özelliği (MDF) ve oran özelliği ile bu özelliklerin kombinasyonu çevrimdışı imza doğrulama sorunu için umut verici sonuçlar ortaya koydu. 12 orijinal örnekler ve kamuya açık veritabanından alınan 400 rastgele sahte, Destek Vektör Makinesi (DVM) sınıflandırıcı kullanılarak %17,25 hata oranı (AER) elde edilmiştir. $FAR=0,08$ olarak bulunmuştur.

Pansare ve Bhatia [16], çıkarılan farklı özellik kümelerinin sinir ağıyla kullanılmasını tanıttılar. Onlar imzalardan geometrik özellik kümesi çıkardılar. Bu özellikler, kütle merkezi, imzanın alanı üç boyut yüzey özellikleri, 6 kat yüzey özellikleri, vb. $FAR=14,66$ ve $FRR=20$ olarak bulunmuştur.

Han ve Sethi [17], imza görüntülerinden oluşan veritabanının içeriği ve imza görüntülerinin tanınması ve düzeltilmesine yönelik bir çalışmayı, geometrik ve topolojik özellikleri kullanarak yapmışlardır. Bu çalışmada, imza görüntü veritabanında arama yapmak ve diziler üzerinde düzenleme yapmak için yerel ilişkisel indeksleme şeması kullanılmıştır.

Yedekçođlu [18], karıştırlmıř örneklerden, tamamen birbirinden eşsiz olan imzaların bir düzen içinde olmasını reddederek, imza tanımının ilk şeklinin kullanılabilir olduğunu söylemiştir.

Bajaj ve Chaudhury [19], imza sınıflandırmasında global özelliklerin üç farklı çeşidini kullanmışlardır. Sınıflandırma için ileri beslemeli yapay sinir ađı sınıflandırıcısı kullanmışlardır. Sınıflandırma için çıkarılan özellikler izdüşüm momentleri ve üst ve alt zarf özellikleridir. Bu özellikleri ile üç tane sınıflandırıcı elde edilir. Bu sınıflandırıcıların çıkışı ilişkisel şema kullanılarak birleştirilir. Deney sonuçları sınıflandırıcıların kombinasyonunun tanıma oranını arttırdığını göstermiştir.

Mercimek ve ark. [20], esnek bir tanıma sistemi geliřtirmişlerdir. İncelenen 3 boyutlu görüntüler üzerinde, yüksek bir sınıflandırma için, iyi özellikler hesaplanmıştır. Nesne tanımada, nesnenin yöneliminden, boyutu ve pozisyonundan bağımsız olarak, lineer olmayan sabit moment fonksiyonları sayesinde, özellik vektörleri alınmıştır.

Yılmaz ve Yanıkođlu [21], çevirimdışı imza tanıma sistemini, imzaların yerel histogram bilgilerine dayalı olarak tanıtmaktadırlar. İmzalar kartezyen ve kutupsal koordinat sistemlerinin her ikisi de kullanılarak bölgelere ayrılmıştır ve her bölge için yönelimli degradeler ve lokal ikili desen histogram özellikleri hesaplanmıştır. Sınıflandırıcı olarak Destek Vektör Makinesi sınıflandırıcısı kullanılmıştır. Eğitimde GPDS-160 imza veritabanından alınan imzalarla, herhangi bir yetenekli imza kullanılmadan yetenekli imza testindeki EER=15,41'dir.

Mirzaei ve ark. [22], yeni bir çevirim-dışı imza tanıma sistemi olan modüler sinir ađları (Modüler Neural Networks-MNN) ve bulanık çıkarımlı sistem üzerinde çalışmışlardır. MNN üç tane modülden oluşmaktadır. Her modül giriş olarak bir farklı görüntü özelliklerini kullanılır. Bu özellikler, kenar belirleme, curvelet dönüşüm ve Hough dönüşümüdür. Mamdani bulanık çıkarım sistemi ise her modülden gelen çıkışları birleřtirmek için kullanılır. 30 kişiden elde edilen ve 60

tanelik küçük bir test setine sahip sistem bu mimari ile %96,6 bir tanıma oranına ulaşmıştır.

Pérez-Hernández ve ark. [23], basit adaptif çevirim-dışı imza tanıma metodunu tanıtmışlardır. Bu yöntemde, imzanın önemli vuruşlarından çıkarılan özellikler kullanılmıştır.

Şenol ve Yıldırım [24], bu çalışmada iki bilinen yapay sinir ağı mimarisi olan, Çok Katmanlı Perseptron (Multi Layer Perceptron-MLP) ve Radyal Taban Fonksiyonlu Sinir Ağı (Radial Basis Function Neural Network-RBFNN) ve bir melez sinir ağı olan Konik Kesit Fonksiyonlu Sinir Ağı (Conic Section Function Neural Network-CSFNN) imza tanıma için önermişlerdir. Bu sistem toplamda 8 kişi için eğitim için toplanan 200 imzadan ve test için toplanan 56 imzadan oluşturulmuştur. Tanıma oranları MLP için, %92,143, RBF için %98,21 ve CSFNN için %96,43 bulunmuştur.

Çikoğlu [2], yapay sinir ağı kullanılarak imza tanıma sistemi tasarlamıştır. Bunun için imzalar, önce normalize edilmiş, daha sonra belli bir eşik değeri belirlenerek imzalar gürültüden arındırılmış ve daha sonra yapay sinir ağlarında eğitmek üzere özellikler elde edilmiştir. Bu özellikler; imza yoğunluğu, imzanın merkezler arası görelî yatay farkı, imzanın merkezler arası görelî dikey farkı, imzanın genişliği, imzanın yüksekliği olmak üzere 5 özelliğe bakılmıştır. Bu beş özellik 3 farklı eşik değeri hesaplanarak 15 özellik haline getirilmiştir.

Öz [25], görüntü işleme, moment sabiti yöntemi ve yapay sinir ağlarını kullanarak, çevirimdışı imza tanıma ve doğrulama sistemi tasarlamıştır. İki ayrı sinir ağı tasarlamıştır. Birini tanıma, diğerini doğrulama için kullanmıştır.

Chadha ve ark. [26], imza tanıma uygulaması gerçekleştirmişlerdir. Özellik çıkarmak için DCT dönüşümünü kullanmışlardır. Sınıflandırma için radyal fonksiyonlu yapay sinir ağı kullanmışlardır. Yöntem 200 örnek için yaklaşık %80 tanıma sağlamıştır.

Azzopardi ve Camilleri [27], radyal taban fonksiyonlu yapay sinir ağının çevirimdışı imza doğrulamasına olan etkisini incelemişlerdir. Özellik kümesinde global, ızgara ve metin özellikleri kullanılmıştır. Bu özelliklerden bir sürü kombinasyonlar oluşturulmuş ve bunlar üzerinde birçok deney yapılmıştır. En iyi kombinasyon 592 elemanlı global ve ızgara özellikli vektörle elde edilmiştir. Ortalama hata oranı %2,04, yanlış ret oranı (FRR) %1,58 ve yanlış kabul oranı (FAR) %2,5 elde edilmiştir.

Patil ve ark. [28], çevirim-dışı imza tanıma ve doğrulama sistemini incelemişlerdir. İmzalardan özellikler çıkarmadan önce ölçekleme ve renk normalizasyonu ön işlemlerinden geçirmişlerdir. Bundan sonra imzalardan global özellikler elde etmişlerdir. Öklid mesafesi modeliyle sınıflandırma yapmışlardır. %89 başarı elde etmişlerdir.

Yingyong ve ark. [29], bu çalışmada, imza görüntülerin genel geometrik ve yerel ızgara özellikleri çıkarılarak algoritma elde etmişlerdir. Bu özellikleri, birçok ölçekli doğrulama fonksiyonuna sahip olacak şekilde bir araya getirmişlerdir. Bu çok ölçekli doğrulama fonksiyonunu istatistiksel yöntemler kullanılarak değerlendirmişlerdir. Sonuçlar çok ölçekli doğrulama fonksiyonu kullanmanın, sadece genel geometrik ya da yerel ızgara özelliği kullanmadan, düşük doğrulama hata oranı ve yüksek güvenilirlik nedeniyle daha iyi olduğunu göstermiştir. Bu sistemin doğrulaması %90'dan fazla ve ustalıkla veya basit olarak atılmış sahte imzaları reddetmektedir.

Mohammadzade ve Ghonodi [30], çalışmalarında çevirim-dışı imza tanıma sistemini incelemişlerdir. Çalışmalarını üç bölüme ayırmışlardır. Birinci bölümde imza görüntülerini iyileştirmek için çeşitli ön işlemlerden geçirmişlerdir. İkinci bölümde imzaların özelliklerini çıkarmışlardır. Çıkardıkları özellikler genel ve geometrik özelliklerdir. Ve son bölümde destek vektör makineleri sınıflandırıcısı kullanarak imzaları sınıflandırmışlardır. İmzalar iki ayrı gruptan alınmıştır. Bir tanesi çalışanlar ve bir diğeri üniversite öğrencileridir. Çalışanların imzalarından elde edilen başarı oranı %92 ve üniversite öğrencilerinden elde edilen başarı oranı, %94,4'tür.

Sthapak ve ark. [31], çalışmalarında yapay sinir ağı tabanlı imza tanıma ve doğrulama sisteminin incelemiştir. Tarayıcıyla aldıkları imzaları ölçekleme, gürültü giderme, arka planı kaldırma, normalizasyon ve inceltme ön işlemlerinden geçirmişlerdir. Daha sonra imzanın genel özellikleri, maske özelliği ve metinsel özelliği çıkarılmıştır. Sınıflandırma için yapay sinir ağı kullanmışlardır ve %100 tanıma oranı sergilemişlerdir.

Kaewkonga ve ark. [32], çalışmalarında çevirim-dışı imza tanıma sisteminde Hough dönüşümünü kullanarak vuruş hattının tespit etmişlerdir. Hough dönüşümü, imza iskeletinden eşsiz karakteristik bir özelliği olan parametrelili Hough alanı çıkarır. Bu yöntemin performansını değerlendirmek için geri beslemeli yapay sinir ağı kullanmışlardır. Bu sistemi farklı insanlardan alınan 70 imza ile test etmişlerdir. İmza tanıma oranı %95,24'dir.

Bu tezde yapay sinir ağlarıyla çevirim dışı imza tanıma anlatılmıştır. Öncelikle imzalar toplanmış ve uygun ön işlemlerden geçirildikten sonra özellik vektörleri çıkarılmıştır. Ve son olarak da bu imzaların kime ait olduğu tanınmıştır. Bunu yaparken Matlab yazılımından faydalanılmıştır. Büyük ölçüde matlab hazır fonksiyonlarından yararlanılmıştır.

İkinci bölümde birtakım özellik çıkarma ve görüntü işleme tekniklerinden bahsedilmiştir. İmza tanıma akış şeması gösterilmiş ve açıklanmıştır. Görüntü işleme tekniklerinden, görüntünün oluşması, bölümlenmesi, iyileşmesi ve analizi yöntemlerinden uygulamada kullanılan kısımlarına değinilmiştir. Bundan sonra imzanın özelliklerinin çıkarılması yöntemlerinden bahsedilmiştir.

Üçüncü bölümde görüntüleri sınıflandırmak için kullanılan çeşitli sınıflandırma yöntemleri anlatılmıştır. Uygulamada kullanılan yapay sinir ağı tabanlı sınıflandırma yöntemlerinden radyal taban fonksiyonlu sinir ağı üzerinde ayrıntılı olarak durulmuştur.

Dördüncü bölümde bu radyal taban fonksiyonlu sinir ağı ile yapılan imza tanıma uygulaması ayrıntılı olarak anlatılmıştır. Bunun için tarayıcı yardımıyla bilgisayar ortamına atılan 450X250 boyutlarındaki imzalar tek tek kaydedilerek, veritabanı oluşturulmuştur. İmza görüntüleri çeşitli ön işlemlerden geçirilmiş ve her imzadan 30 tane özellik çıkarılmıştır. Bu özellikler her imzanın kendi yatay ve dikey sınırları içerisinde çıkarılmıştır. İmzalar radyal taban fonksiyonlu sinir ağıyla eğitildikten sonra, test için ayrılan imzalar üzerinde denenmiştir. Bunların sonuçları, çeşitli şekiller, tablolar ve grafiklerle verilmiştir.

Son bölümde ise, elde edilen sonuçlar yazılmış, bunlar üzerinde tartışılmış ve ileride yapılacak çalışmalar için önerilerde bulunulmuştur.

BÖLÜM 2. GÖRÜNTÜ İŞLEME VE ÖZELLİK ÇIKARIMI

2.1. Biyometrik Kimlik Tanıma

Görüntü işleme, gerçek yaşamdaki görüntülerin dijital (sayısal) hale getirilerek ve çeşitli işlemlerden geçirilerek, o görüntünün ve özelliklerinin değiştirilmesi sonucu yeni bir görüntünün elde edilmesini ifade eder [33].

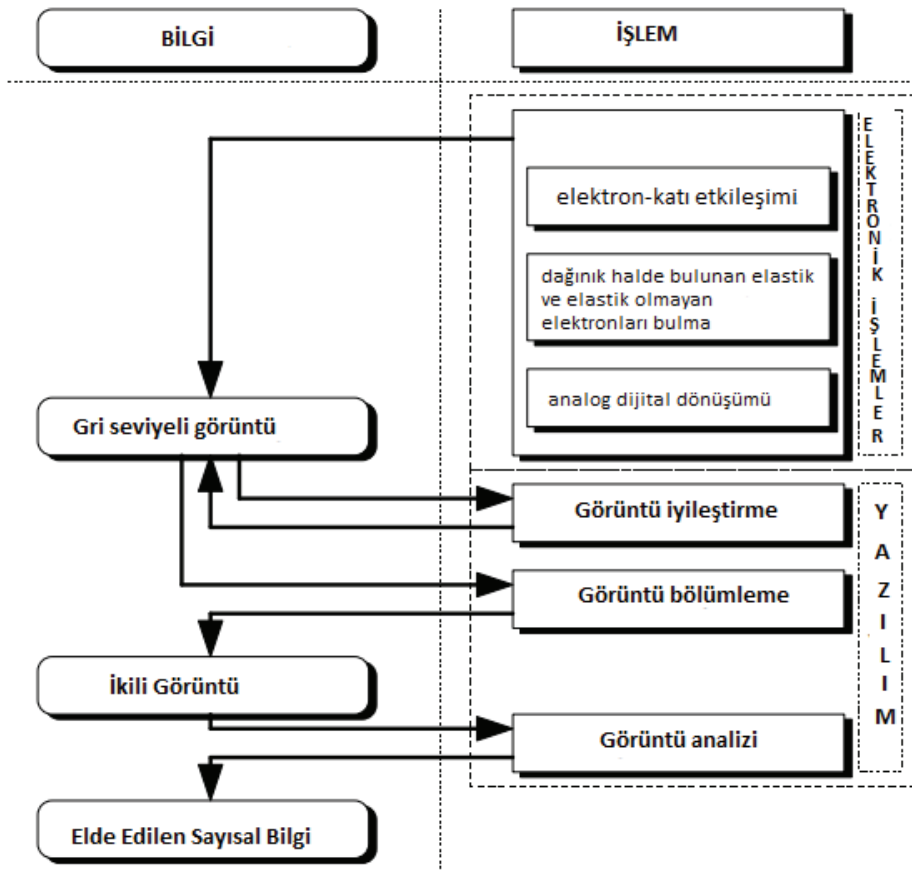
Biyometrik sistemler bir bireyin kişisel bir nitelik ya da davranışını analiz ederek, kimliğini açıklayan biyolojik verileri doğrulama bilimidir. İnsanları birbirinden ayırt edebilme şansını bize sunduğundan dolayı, biyometri bir kimlik doğrulama sistemi olarak da kullanılmaktadır [34].

İnternet ve bilgisayar teknolojilerinin etkin kullanılmaya başlanması ile birlikte, bazı kişisel bilgilere veya kurum ve kuruluşlara ait gizli verilere, yetkili olmayan girişlerin yasaklanması gibi zorunlulukları doğurmuştur. Aynı zamanda bu işlemin doğru, güvenilir ve çok kısa sürede gerçekleştirilebilmesi çok önemlidir. Bu sistemler genel olarak kullanıcıdan bir bilgi, şifre ve pin numarası ya da başka bir yöntemle giriş bilgisi istemektedir. Bu durumda hem zaman kaybı olmakta, hem de giriş bilgilerinin güvenliğini sağlamak ve izinsiz girişleri engellemek oldukça zordur. Bunun yerine biyometrik teknolojiler kişileri doğrudan tanıdıkları için, yüksek güvenlik uygulamalarının vazgeçilmez unsuru olarak kullanılabilirler. Çünkü biyometrik tanıma sistemleri bir bireyin sadece kendisine has özelliklerinin kanıtlanmasına olanak sağlar. Bu sistemler unutulması veya başkası tarafından kullanılması söz konusu olmayan bir kimlik onaylama yoludur. Bu sayede kimlik, pasaport, ehliyet gibi kartların yerini tamamen alacak bir sistem geliştirilebilir. Biyometrik sistemler yüz, parmak izi, retina, iris, el geometrisi, imza, ses, koku tanıma, imza tanıma gibi yapılar ile açıklanabilir. Görüntü işleme biyometrik

sistemlerin görüntü tabanlı olanlarında kullanılmaktadır. Yüz tanıma, parmak izi tanıma, imza tanıma, el geometrisi gibi [35].

Biyometrik özelliklerin kişiyi ayırt edebilmesi için özelliklerin doğru olması gerekir. Fakat parmak izi tanıma sistemlerinde, parmak izi sensörünün yağlanması, ya da tanımda kullanılacak biyometrik kısmın (parmak, iris retina, yüz, vb.) gibi zarar görmesi ya da imza tanıma sistemlerinde, atılan imzanın yıllar içinde değişikliğe uğraması sonucu biyometrik sistemlerin hatalı sonuçlar verebilir. Sisteme kayıtlı olan kişilerin tanınmaması sonucunda, Yanlış Reddetme Oranı (False Rejection Rate-FRR), sisteme kayıtlı olmayan kişinin sistemle karşılaştırılması sonucu sistemde olduğunun kabulü sonuçlanmasında, Yanlış Kabul Oranı (False Acceptance Rate-FAR) ve FRR ve FAR hata oranlarının ortalaması sonucu Ortalama Hata Oranı (Average Error Rate-AER) ortaya çıkar [1,4]. Birçok biyometrik sistemde verilerin elde edilmesi aşamasında görüntü işleme algoritmaları kullanılır.

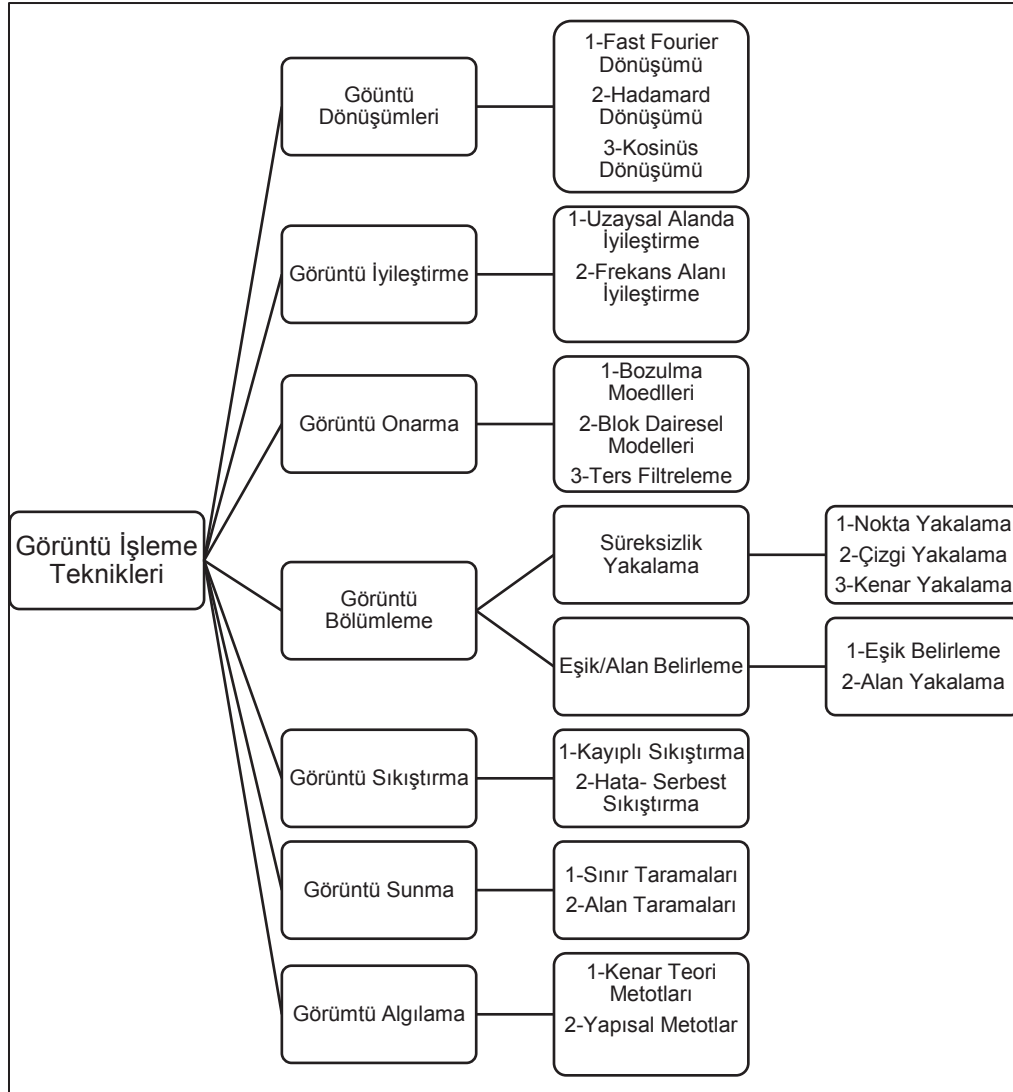
Sayısal görüntü işleme aşamaları; görüntüyü oluşturma ve elde etme, görüntüyü sayısallaştırma, görüntü iyileştirme, görüntü analizidir. Görüntü işlemenin ilk kısmı “görüntü elde etme” aşamasıdır. Bu aşama görüntünün bilgisayar ortamına aktarılması işlemidir. İkinci aşama olan “görüntü sayısallaştırma” ise, görüntünün bilgisayarda nasıl saklanacağını ifade eder. Görüntü M satır ve N sütunlu bir matris olarak ifade edilebilir. Bir diğer aşama olan “görüntü iyileştirme” ise görüntünün kalitesinin artırılması veya görüntünün bir başka forma (hangi form daha uygunsa) dönüştürülmesi işlemidir. Bundan sonraki kısım “görüntü bölüme, bir görüntüyü birçok piksel kümesine ayırması işlemidir. Görüntünün bölümeleme amaç basitleştirmek ve/veya görüntünün analiz edilmesini kolaylaştırmaktır. ”Görüntü analiz etme” kısmında ise görüntüden sayısal bilgiler çıkarılır. Görüntü analiz işlemleri, görüntü işlemede son bölümdür ve genellikle görüntü bölümeleme işlemleri tamamlandıktan sonra ikili görüntüler üzerine yapılır. Görüntü analizi verilen bir imgeden otomatik ya da yarı otomatik yöntemler ile veri, bilgi ya da bazı işlemlerin elde edilmesi ile ilgilidir. Görüntü işleme aşamalarını Şekil 2.1'de daha açık olarak görebiliriz [36].



Şekil 2.1. Görüntü işleme aşamaları

Çalışmanın bundan sonraki kısmında görüntü elde edilerek, sayısallaştırılacak, görüntü sayısallaştırıldıktan sonra nicemleme ile görüntü 8 bitlik düzeyde griye çevrilecektir. Görüntü bölümlenme aşamasında eşikleme yöntemi kullanılacaktır. Görüntü eşikleme yöntemlerinden, eşik değerini otomatik olarak bulan Otsu metodu kullanılacaktır. Bundan sonra görüntü iyileştirme için morfolojik işlemler hakkında genel bir bilgi ve özel olarak morfolojik bir işlem olan iskelet çıkarma yöntemi kullanılacaktır. Bu çalışma imza görüntüsünü incelemekte olduğu için kullanılan teknikler imza çalışmalarında kullanılan tekniklerdir.

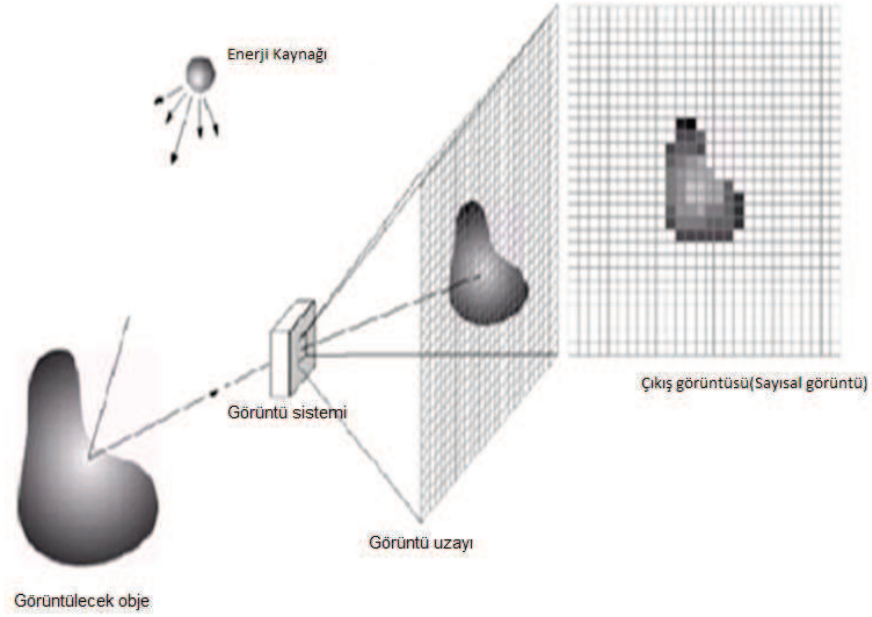
Görüntü işleme konusunda yapılan çalışmalarda birçok teknik kullanılmıştır. Bu teknikleri Şekil 2.2'deki gibi sınıflandırabiliriz [37].



Şekil 2.2. Görüntü işleme teknikleri sınıflandırması

2.1.1. Görüntünün oluşturulması

Görüntünün nasıl oluştuğu Şekil 2.3'te gösterilmiştir.



Şekil 2.3. Dijital görüntü elde etme aşaması (a) Enerji kaynağı (b) Görüntülenecek obje (c) Görüntü sistemi (d) Görüntü uzayı (e) Sayısal görüntü

2.1.2. Görüntüyü sayısallaştırma

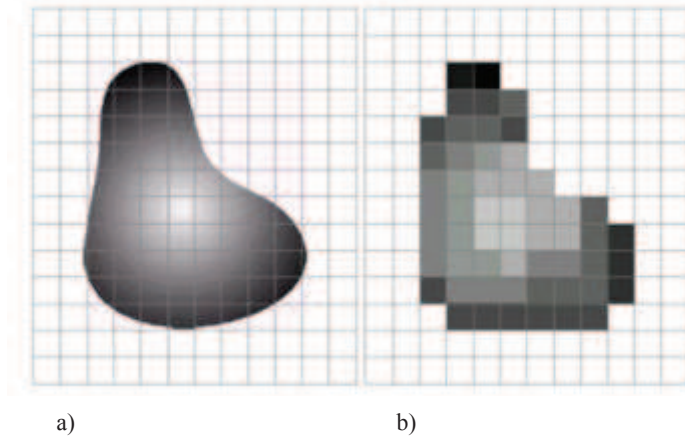
İki boyutlu ışığın yoğunluk değerine göre oluşturulmuş fonksiyondur. Koordinatlar x,y ise bunun geri döndürdüğü değer (fonksiyon) $f(x,y)$, (x,y) koordinatlardaki ışıklık değeri olarak adlandırılır. Görüntülerin işlenebilmesi için analog biçimden sayısal şekle dönüştürülmelidir [38].

2.1.3. Nicemleme

Uzaysal koordinatlar ve genlik değerlerinin sayısallaştırılması işlemi, $f(x,y)$ fonksiyonu ile ifade edecek olursak, burada görüntü birimlerinin uzaysal koordinatlarını belirten (x,y) atama işlemi örnekleme, her birine atanacak genlik değerlerini belirleme işlemi ise nicemleme olarak adlandırılır [38].

Örnekleme ve nicemleme sonucu iki boyutlu bir matris elde edilir. Her bir (x,y) koordinatının alacağı değer nicemleme işlemiyle belirlenir. Şekil 2.4'te bununla ilgili örnek bir görüntü gösterilmiştir.

Bundan da anlaşılacağı gibi, iki değerden söz ediyoruz. Birincisi, görüntünün her bir birimine karşılık gelen bir koordinat, ikincisi ise her bir koordinat değerine karşılık gelen piksel yoğunluğudur. Tek bir renk yoğunluğuyla belirlenmiş düzeyi belirtmek için gri düzeyi (gray level) terimi kullanılır [39].



Şekil 2.4. a) Eşit karelere bölünerek, koordinatları belirlenmiş görüntü b) Örneklenmiş ve nicelenmiş görüntü sonucu

Görüntünün boyutlarını $[i,j]$ ile ifade edersek, görüntüyü $I[i,j]$ ile ifade edebiliriz [39]. Görüntü sayısallaştırılırken, dizinin boyunun gri düzeyinin belirlenmesi gerekir. Bunu yaparken genelde 2'nin katları seçilir. M,N dizinin boyları ve k piksel yoğunluğunda kullanılan bit sayısı ve G ise Gri düzeyidir. Denklem 2.1'de belirtilmiştir. Örnek olarak $[8,8]$ boyutunda 256 gri düzeyli bir görüntünün yoğunlukları Tablo 2.1'de belirtilmiştir

$$M=2^n \quad N=2^m \quad G=2^k \quad (2.1)$$

Tablo 2.1. Gri düzeyi 256 olarak belirlenmiş [8,8] boyutlarında bir görüntü ve piksel yoğunlukları

| | | | | | | | |
|-----|-----|-----|----|-----|-----|----|-----|
| 50 | 220 | 95 | 97 | 126 | 225 | 42 | 40 |
| 70 | 210 | 91 | 42 | 50 | 227 | 45 | 112 |
| 83 | 200 | 87 | 35 | 78 | 245 | 66 | 125 |
| 0 | 174 | 83 | 38 | 58 | 201 | 88 | 128 |
| 255 | 183 | 100 | 22 | 115 | 108 | 23 | 198 |
| 11 | 47 | 102 | 18 | 89 | 35 | 28 | 190 |
| 25 | 48 | 57 | 16 | 82 | 95 | 27 | 128 |
| 3 | 1 | 45 | 4 | 8 | 16 | 22 | 88 |

İzin verilen en büyük gri düzeyi 256 seçildiğinde, 0 en koyu rengi, 255 ise en açık rengi ifade eder. Renkli bir görüntünün gri bir görüntüye çevrilmesi için RGB kanallarının her biri seçilen bit sayısına göre gri seviyeye dönüştürülür. Bunun için RGB imgesi YIQ renk uzayına dönüştürülür. Y değeri parlaklık miktarıdır. I ve Q bileşenleri renk değerleri taşımaktadır. O yüzden dönüşüm işlemlerinde bu değerler sıfırlanır. YIQ'nun RGB cinsinden değeri Denklem 2.2'de gösterilmiştir.

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.299 & -0.587 & 0.378 \\ 0.578 & -0.587 & -0.114 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2.2)$$

Matlab 'da rgb2gray fonksiyonu NTSC'nin (Ulusal Televizyon Standardı Komitesi) belirlediği renk uzayı olan YIQ renk uzayını kullanır. Denklem 2.3'te Y değerinin nasıl bulunduğu gösterilmiştir [38].

$$Y = 0.299R + 0.587G + 0.114B \quad (2.3)$$

R=Y, G=Y, B=Y olur ve artık yeni görüntümüz 3 boyutlu bir matrisle değil 2 boyutlu bir matrisle ifade edilmektedir [40].

Biz bu çalışmada 8 bitlik sayısallaştırma işlemi kullandık. Bunun anlamı Beyaz=255 ve Siyah=0. Ve bu iki değer arasındaki değerler de gri tonlarıdır. Taranmış ham resmin, 8 bitlik sayısallaştırılmış hali Şekil 2.5'te gösterilmiştir.



Şekil 2.5. A.T. adlı kişinin imzasının 256 gri düzeyinde, griye dönüştürülmüş hali

2.1.4. Görüntü bölümlenme

Görüntü bölümlenme, çıktı olarak sadece yeni bir görüntü üretmekle kalmayıp, aynı zamanda görüntüye ilişkin sayısal bazı yeni bilgiler de üretebilmektedir. Görüntü bölümlenme görüntüyü birbiriyle çakışmayan, fakat görüntünün tümünü alt görüntü gruplarına ayırma işlemidir. Bu gruplandırma işlemi görüntünün belirli bir veya birden fazla özelliği dikkate alınarak gerçekleştirilebilmektedir.

Bölümlenme işleminde kullanılan özelliğin seçimi uygulamaya bağlı olarak değişiklik gösterebilir. Bölümlenmede kullanılan en temel özellikler gri görüntülerde parlaklık, renkli görüntülerde renk bileşenleri olarak sıralanabilir. Görüntülerdeki ayrıtlar ve doku özellikleri de bölümlenme açısından oldukça yararlı bilgiler taşıyabilmektedir [41].

2.1.4.1. Görüntü eşikleme

Basit bölümlenme yöntemlerinden bir tanesi görüntünün sadece parlaklık bilgisinin göz önüne alınarak bölümlenme işleminin gerçekleştirildiği eşikleme işlemidir. Eşikleme işlemi genelde imgenin birden fazla özelliğini kullanarak bölümlenmeyi gerçekleştiren yöntemlere göre daha hızlı sonuç vermekte ve pek çok kere, otomatik hedef tanıma, imza tanıma, karakter tanıma ve endüstriyel ortamların kalite kontrol,

gibi yapay görme uygulamalarının pek çoğunda kullanılabilir [42]. Bu çalışmada görüntü eşikleme yöntemlerinden Otsu metodu kullanılmıştır.

Otsu; otomatik eşik seçme metotlarından biridir. Ve görüntüyü otomatik hesapladığı eşik değerine göre 0 ve 1'lere çevirir [43]. Bu metot olası tüm eşik değerlerinin ve eşik her tarafındaki (ön plandaki veya arka plandaki) piksel seviyelerinin hesaplanmasını içerir. Bu yöntemin amacı ön plandaki ve arka plandaki yayılım değerlerinin toplamının en küçük olduğu eşik değerini bulmaktır. Bu eşik değeri aynı sınıftaki değişiklikleri (varyans) minimize edecek şekilde farklı sınıflardaki değişiklik (varyans) maksimize edecek şekilde çalışır [44].

Eğer karanlık arka plandaki aydınlık nesnelere ilgilenecekse o zaman seçilen eşik değerine göre algoritma;

Eğer $f(x,y) \geq \text{esik}$ ise $f(x,y)=1$

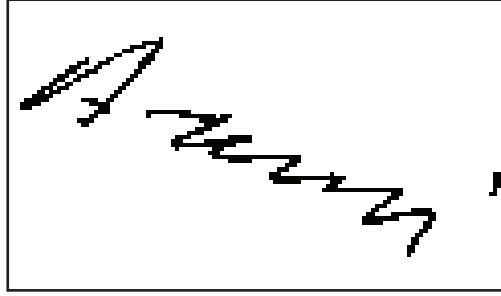
Değilse $f(x,y)=0$

Eğer aydınlık arka plandaki karanlık nesnelere ilgilenecekse o zaman seçilen eşik değerine göre algoritma;

Eğer $f(x,y) < \text{esik}$ ise $f(x,y)=1$

Değilse $f(x,y)=0$

Eğer görüntü çok yüksek karşıtığa sahip bir imge ise bu eşik değeri 128 olarak seçilebilir [45]. Şekil 2.6'da Otsu metoduyla siyah beyaza dönüştürülmüş imza görüntüsü görülmektedir.



Şekil 2.6. A.T.isimli kişinin imzasının siyah beyaza dönüştürülmüş hali

2.1.5. Görüntü iyileştirme

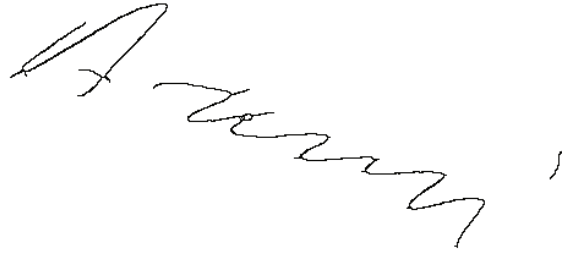
Görüntü iyileştirme, görüntünün kalitesinin arttırılması veya görüntünün bir başka forma(hangi form daha uygunsa) dönüştürülmesi işlemidir. Kesin bir görüntü iyileştirme tekniği yoktur. Bu durum insanın algısına göre değişir. Görüntü iyileştirme tekniği, bir uygulamanın sonraki aşamaları için ön işlemlerinde kullanıldığı zaman, sayısal ölçümler, seçilmiş uygun görüntü iyileştirme tekniğine göre ayarlanabilir [46].

2.1.5.1. Morfolojik işlemler

Morfoloji, canlıların şekil ve yapılarıyla ilgilenen bir bilim dalıdır. Morfoloji şekil veya obje tabanlı kullanılan bir görüntü işleme tekniğidir. Morfolojik metotlar yapısal bir eleman kullanarak giriş görüntüsüyle aynı boyutta çıkış görüntüsü elde edilmesini sağlarlar. Giriş görüntüsündeki her bir pikselin, kendisine karşılık gelen komşu piksellerle karşılaştırılması işlemine dayanır. Komşu piksel ve şeklini seçerek giriş elamanına göre uygun bir morfolojik bir işlem oluşturulabilir. En çok ikili görüntüler üzerinde kullanılırlar. İkili görüntülerin analizinde ve iyileştirilmesinde kullanılırlar.

Görüntü işlemede temel iki tane morfolojik işlem vardır. Bunlar yayma ve aşındırmadır. Ve diğer morfolojik işlemler için, bu iki temel morfolojik işlem kullanılmalıdır. Örneğin; açma (opening), kapama (closing) [38].

Bu çalışmada iskelet çıkarma morfolojik işlemine değinilecektir. Matlab iskelet çıkarma işlemini çok kolaylıkla bwmorph adlı komutun 'skel' parametresini N kadar yaparak bu işlemi gerçekleştirmektedir. Bu N parametresi sonsuz seçildiğinde, görüntü daha fazla incelemeyene kadar devam eder. Şekil 2.7'de İmza görüntüsün en incelebildiği noktaya kadar incelerek iskeleti çıkarılmıştır.



Şekil 2.7. Morfolojik iskelet çıkarma işlemi uygulanmış A.T. adlı kişinin imzası

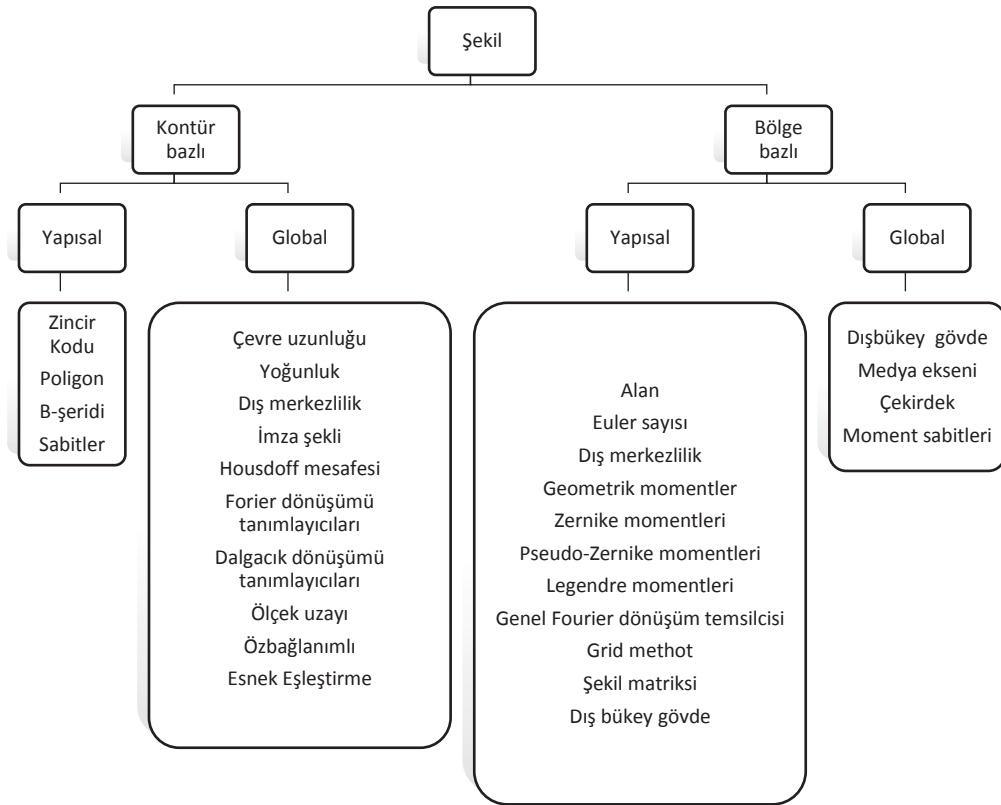
2.2. Özellik Çıkarımı

Bir dijital görüntü sayısallaştırıldığında, bir görüntüden çok daha fazla görüntü bölünerek daha fazla görüntü elde edilebilir. Gerektiğinde etkili araçlar sayesinde görüntüden istenildiği gibi yararlanılabilir. Görüntünün şekli görüntünün içeriğini açıklarken en büyük özelliklerden biridir. Fakat görüntüyü temsil etmek ve tanımlamak da zor bir görevdir. Görüntüden, imza şekli, imza histogramı, şekil değişmezleri, momentleri, eğrilik, şekil bağlam, şekil matrisi, spektral özellikleri, bunun gibi özellikler bulunabilir.

Şekiller düzgün alınabilirse o zaman veritabanındaki benzer şekillere benzetilebilir. Algısal olarak bakıldığında benzer şekiller, döndürülmüş, çevrilmiş, ölçeklenmiş şekillerdir ve affine olarak değiştirilmiş şekillerdir. Algılayıcı ayrıca resmi etkileyen gürültü, çeşitli şekillerde bozulmuş şekiller, kusurlu şekilleri insan tarafından karşılaştırma yapılırken tolere edebilmelidir. Bu sağlamlık için gereklidir. Eğer şekil temsilcisi iyi karakteristik özellikleri için aşamalı olarak kalitesizse, o zaman yüksek düzeyde verimlilikte eşleştirme yapılmalıdır. Çünkü kalitesiz eşleştirmelerde ancak

birbirine hiç benzemeyen şekiller elimine edilir ve ince seviyeli eşleştirmelerde şekillerin detayları eşleştirilebilir.

Şekil temsilcisi ve tanımlayıcısı teknikler genellikle 2 grupta toplanır. Kenar bazlı ve bölge bazlı metotlar. Yani bu özellikler sadece şeklin kenarları baz alınarak yapılırsa kenar bazlı bütün şekil bölgesi alınıp yapılırsa bölge bazlı olur. Her sınıf da kendi içinde yapısal yaklaşımlar ve global yaklaşımlar olmak üzere ikiye ayrılmıştır. Bu alt sınıf şeklin bütün şekli temsil etmediğine veya parçalar halinde temsil edilip edilmediğine dayanır. Bu yaklaşımlar uzay domeyni ve dönüşüm domeyni olarak ayrılır. Bütün hiyerarşi Şekil 2.8’de gösterilmiştir [47].



Şekil 2.8. Şekil temsilcisi ve tanımlayıcısı teknikler

Biyometrik sistemin uygulamadaki başarısı, kullanılan eğitim sistemlerinin tutarlılığına bağlıdır. Özellik çıkarma yöntemleri kullanılarak, sistemlerin çeşitli eğitim setleri oluşturulur. Bütün imza görüntüleri için kurallar kümesi elde etmek amacıyla, örneklerin gerekli kümesi oluşturulmalıdır. Özellik çıkarma, şablon tanıma veya sınıflandırma sistemlerinde en önemli işlemlerden biridir. Özellik çıkarma tekniğinin başarısı, sınıflandırma sonucunu direkt etkiler [48,49].

Bundan sonra imza tanımada kullanılan özellik çıkarma yöntemlerinden bahsedilecektir.

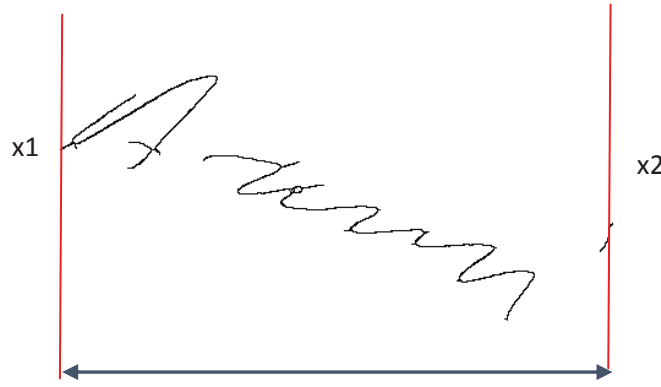
2.2.1. Global özellikler

2.2.1.1. Alan

İmzadaki piksel sayısıdır. Bu özellik imzanın yoğunluğu hakkında bilgi verir. Bu çalışma için imzanın sadece siyah yani 0 alan piksellerinin sayıyı hesaplanacaktır [28].

2.2.1.2. Genişlik

Genişlik, yatay izdüşümde ikili görüntülerde birden fazla piksel içeren noktalar baz alınarak baştan sona kadar olan mesafedir [28]. Genişlik imzanın global özelliklerinden biridir. İmzanın genişliği Şekil 2.9'da gösterilmiştir.

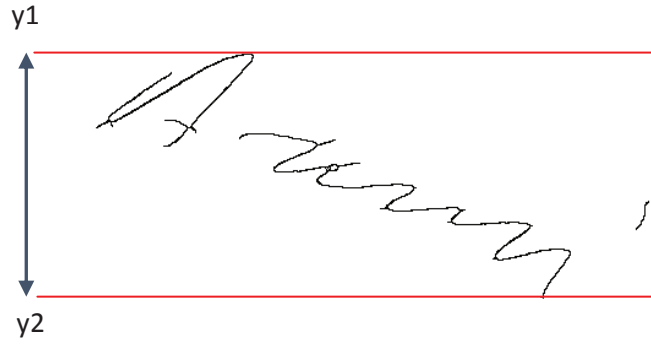


Şekil 2.9. A.T. adlı kişinin imzasının genişliğinin bulunması

$$\text{genislik} = x_2 - x_1 \quad (2.4)$$

2.2.1.3. Yükseklik

Yükseklik dikey izdüşümde ikili görüntülerde birden fazla piksel içeren noktalar baz alınarak baştan sona kadar olan mesafedir [28]. Yükseklik imzanın global özelliklerinden biridir. İmzanın yüksekliği Şekil 2.10'da gösterilmiştir.



Şekil 2.10. A.T. adlı kişinin imzasının yüksekliğinin bulunması

$$\text{yukseklık}=\text{y2}-\text{y1} \quad (2.5)$$

2.2.1.4. Yükseklik/genişlik oranı

Görüntünün yüksekliğinin genişliğine bölünmesiyle elde edilen orandır [28].

$$\text{oran}=\text{x2}-\text{x1}/\text{y2}-\text{y1} \quad (2.6)$$

2.2.1.5. Ağırlık merkezi

İlgili düzlemdeki siyah piksellerin ağırlıklı toplamının, bütün siyah piksellerin toplamına bölünmesiyle elde edilir [28]. Şekil 2.11'de imzanın ağırlık merkezi bulunmuştur.

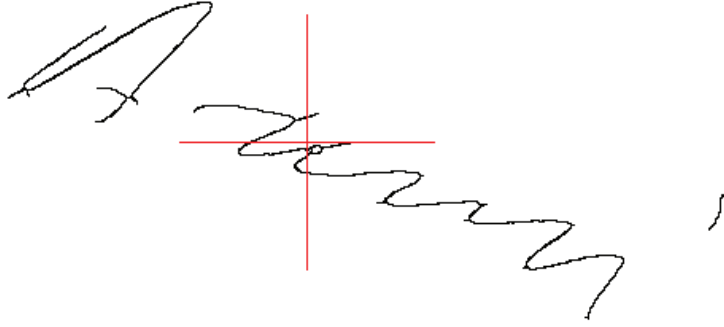
$$\text{agmerx}=\text{sutun}/\text{butun} \quad (2.7)$$

$$\text{agmery}=\text{satır}/\text{butun} \quad (2.8)$$

sutun= Siyah olan piksellerin sütun olarak ağırlıklı toplamını,

satır= Siyah olan piksellerin satır olarak ağırlıklı toplamını,

butun= bütün siyah piksellerin toplamı.



Şekil 2.11. A.T. adlı kişinin imzasının ağırlık merkezi

2.2.1.6. Görüntünün ağırlık merkezlerinden bölünmesi

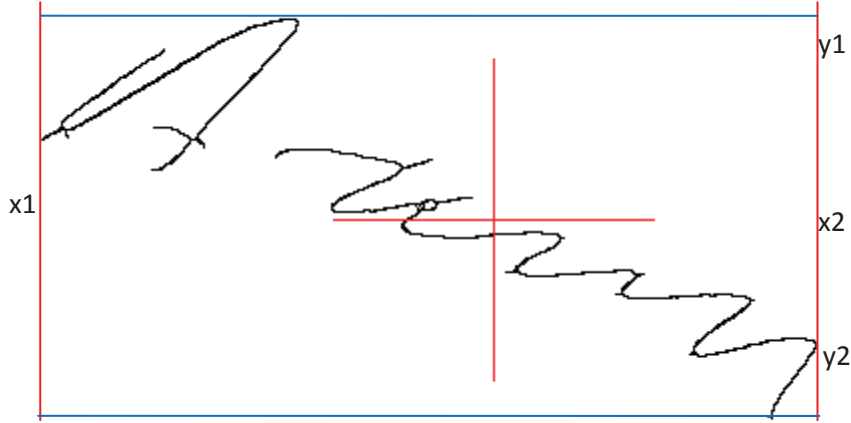
Görüntünün ağırlık merkezlerinden 4 eşit parçaya ayrılarak her parçadaki siyah piksel sayısının hesaplanmasıdır. Ve ayrılan her 4 parça tekrar ağırlık merkezlerinden dörde bölünerek her parçadaki siyah piksellerin hesaplanmasıdır [28].

2.2.1.7. Boyut merkezi

İmzanın bulunduğu alan, siyah piksellerin olduğu alanda siyah noktaların başlangıç (x_1, y_1) ve bitiş noktaları (x_2, y_2) noktalarının aritmetik ortalamasının hesaplanmasıdır [2]. Şekil 2.12'de imzanın genişlik ve yüksekliklerinin orta noktaları, yani boyut merkezleri bulunmuştur.

$$\text{Boyut_merkezi_x} = (x_1 + x_2) / 2 \quad (2.9)$$

$$\text{Boyut_merkezi_y} = (y_1 + y_2) / 2 \quad (2.10)$$



Şekil 2.12. A.T. adlı kişinin imzasının boyut merkezi

2.2.1.8. Yoğunluk

Yoğunluğu, görüntünün alanını, boyutlarının çarpımına bölerek hesaplayabiliriz [2].

$$\text{yogunluk} = \text{alan} / (\text{genislik} * \text{yukseklık}) \quad (2.11)$$

2.2.1.9. Merkezler arası görelî yatay fark

Görüntünün ağırlık merkezinin x ekseninin, boyut merkezinin x ekseninin arasındaki mutlak farkının, genişliğe oranı ile hesaplayabiliriz [2].

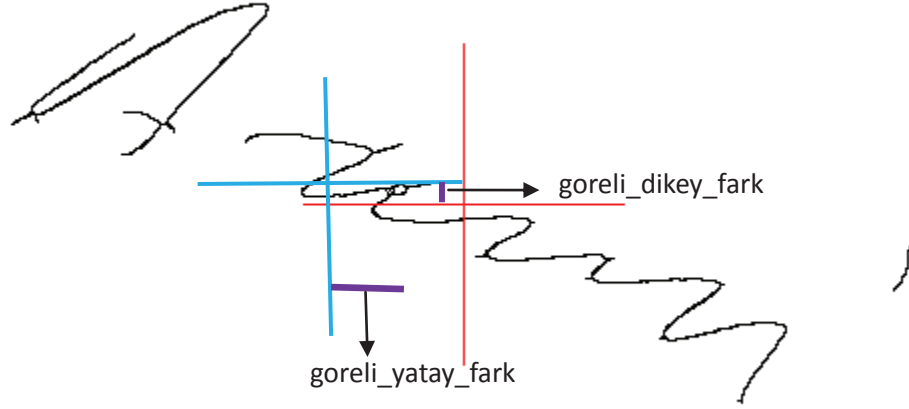
$$\text{goreli_yatay_fark} = \text{abs}(\text{agmerx} - \text{boyut_merkezi_x}) / \text{genislik} \quad (2.12)$$

2.2.1.10. Merkezler arası görelî dikey fark

Görüntünün ağırlık merkezinin y ekseninin, boyut merkezinin y ekseninin arasındaki mutlak farkının, yüksekliğe oranı ile hesaplayabiliriz [2].

Şekil 2.13'te ağırlık merkezleri ile boyut merkezleri arasındaki farklar gösterilmiştir.

$$\text{goreli_dikey_fark} = \text{abs}(\text{agmery} - \text{boyut_merkezi_y}) / \text{yukseklık} \quad (2.13)$$



Şekil 2.13. A.T. adlı kişinin imzasının ağırlık merkezi ve boyut merkezi arasındaki fark

2.2.2. Moment sabitleri

Moment sabitleri ikili görüntülerdeki bağlantılı bölgelerin özellikleridir. Bu özellikler, çeviri, dönme ve ölçü sabitleridir. Bu yöntem, bölgenin özellik kümelerini, şekil sınıflandırması ve parça tanımayı kullanarak kolaylıkla hesapladığından, oldukça kullanışlıdır. Bu özellikler Hu tarafından 1962 yılında sunulmuştur. Bu sabitlerin hesaplanması, şeklin veya nesnenin ağırlık merkezine bağlıdır. Daha önce de belirtildiği gibi gürültünün yok edilmesi, moment sabitli uygulamalarda oldukça önemlidir [50-52]. Moment sabitleri değerleri imza tanıma için uygulandığında, her imza örneği için hesaplanır. Bu sabitleri 7 sayı temsil etmektedir. Diğer bir ifadeyle, her imza örneği 7 sayıyla (I1,I2,I3...vb) tanımlanır. Bu yedi sabit, tanıma sisteminde, imzaların yön değişimine karşı duyarlıdır. Bu imzalar, insanlardan farklı yönlerde ve boyutlarda, günlük yaşamlarından toplanmıştır [48].

Derecesi (p+q) olan 2 boyutlu sürekli fonksiyon f(x,y) momenti için Denklem 2.14'te tanımlandığı gibidir [53].

$$M_{ij} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x^p y^q dx dy \quad (2.14)$$

$p, q=0,1,2$ için piksel yoğunluğu $I(x,y)$ olan gri seviyedeki görüntüye uyarlandığında, görüntü momenti M_{ij} (Denkem 2.15)'teki gibi olur [53].

$$M_{ij} = \sum_x \sum_y x^i y^j I(x,y) \quad (2.15)$$

Bazı durumlarda, görüntü olası yoğunluk fonksiyonu olarak göze alındığında, bu şekilde, yani, Denklem 2.15'in bölünmesiyle elde edilebilir.

$$M_{ij} = \sum_x \sum_y I(x,y) \quad (2.16)$$

Bu eşsiz teoremdede eğer, $f(x,y)$ parçalı sürekli ise ve hiçbir 0 değerine sahip değilse, bütün durum momentleri vardır ve moment dizisi (M_{pq}), eşsiz olarak $f(x,y)$ ile belirtilir. Buna karşıt olarak (M_{pq}) $f(x,y)$ 'yi eşsiz olarak ifade eder. Uygulamada görüntü birkaç taban dizi momentleriyle özetlenmiştir [51].

Moment yoluyla elde edilen basit görüntü özellikleri şunları içerir,

- Alan (ikili görüntü için) veya gri düzeylerin toplamı (gri tondaki görüntüler) için: M_{00}
- M_{10} x'lerin toplamı, M_{01} y'lerin toplamı
- Ağırlık merkezi $\{ \bar{x}, \bar{y} \} = \{ M_{10}/M_{00}, M_{01}/M_{00} \}$

n. momentin anlamı veya herhangi bir n. merkezi moment değeri x Denklem 2.17'de açıklanmıştır [53].

$$\mu_n = \int_{-\infty}^{+\infty} (x - \mu)^n f(x) dx \quad (2.17)$$

2 boyutlu olarak merkezi momentler Denklem 2.18'deki gibi olur;

$$\mu_{pq} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} (x - \bar{x})^p (y - \bar{y})^q f(x, y) dx dy \quad (2.18)$$

Burada ağırlık merkezindeki elemanlar;

$$\bar{x} = \frac{M_{10}}{M_{00}}, \bar{y} = \frac{M_{01}}{M_{00}} \quad (2.19)$$

Eğer $f(x,y)$ bir sayısal görüntü ise, o zaman Denklem 2.18'deki eşitlik Denklem 2.20'deki gibi olur.

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad (2.20)$$

Sırasıyla 3 tane merkezi momentler, Denklem 2.21'deki gibi bulunur.

$$\begin{aligned} \mu_{00} &= M_{00}, \\ \mu_{01} &= 0, \\ \mu_{10} &= 0, \\ \mu_{11} &= M_{11} - \bar{x}M_{01} = M_{11} - \bar{y}M_{10}, \\ M_{20} &= M_{20} - \bar{x}M_{10}, \\ M_{02} &= M_{02} - \bar{y}M_{01}, \\ \mu_{21} &= M_{21} - 2\bar{x}M_{11} - \bar{y}M_{20} + 2\bar{x}^2 M_{01}, \\ \mu_{12} &= M_{12} - 2\bar{y}M_{11} - \bar{x}M_{02} + 2\bar{y}^2 M_{10}, \\ M_{30} &= M_{30} - 3\bar{x}M_{20} + 2\bar{x}^2 M_{10}, \\ M_{03} &= M_{03} - 3\bar{y}M_{02} + 2\bar{y}^2 M_{10}. \end{aligned} \quad (2.21)$$

Genel olarak merkezi momentler Denklem 2.22'deki gibi gösterilebilir.

$$M_{pq} = \sum_m \sum_n \binom{p}{m} \binom{q}{n} (-\bar{x})^{p-m} (-\bar{y})^{q-n} M_{mn} \quad (2.22)$$

Merkez momentler ötelendiğinde değişmezler.

2.2.2.1. Ölçeklendirilmiş sabit momentler

Momentler $n_{ij}, i+j \geq 2$ olduğunda, 00. derecedeki merkezi momente karşılık gelen bölünme derecesindeki değişim ve ötelemenin her ikisi de sabit olarak ifade edilir. Denklem 2.23'te gösterilmiştir.

$$n_{ij} \frac{\mu_{ij}}{\mu_{00}^{1+\frac{i+j}{2}}} \quad (2.23)$$

Öteleme, ölçek değişiklikleri ve yönlendirme esnasından değişmeyen, sabit momentler, tabii ki hesaplanabilirler. Çok sıklıkla kullanılan Hu'nun sabit momentler kümesi [48,51].

2.2.2.2. Dönme sabit momentleri

$$\begin{aligned} I_1 &= n_{20} + n_{02}, \\ I_2 &= (n_{20} - n_{02})^2 + 4n_{11}^2, \\ I_3 &= (n_{30} - 3n_{12})^2 + (3n_{21} - n_{03})^2 \\ I_4 &= (n_{30} - n_{12})^2 + (n_{21} - n_{03})^2 \\ I_5 &= (n_{30} - 3n_{12})[(n_{30} + n_{12})(n_{30} + n_{12})^2 - (3n_{21} + n_{03})^2] + \\ & (3n_{21} - n_{03})(n_{21} + n_{03})[3(n_{30} + n_{12})^2 - (n_{21} + n_{03})^2] \\ I_6 &= (n_{20} - n_{02})[(n_{03} + n_{12})^2 - (n_{21} + n_{03})^2] + 4n_{11}(n_{30} + n_{12})(n_{21} + n_{03}) \\ I_7 &= (3n_{21} - n_{03})(n_{30} + n_{12})[(n_{30} + n_{12})^2 - 3(n_{21} + n_{03})^2] - \\ & (n_{30} - 3n_{12})(n_{21} + n_{03})[3(n_{30} + n_{12})^2 - (n_{21} + n_{03})^2] \end{aligned} \quad (2.24)$$

1.si olan I_1 momentin eylemsizliği etrafındaki imajın ağırlık merkezidir ki, bu ağırlık merkezinde piksellerin şiddetleri, fiziksel yoğunluğa benzer. Sonuncu I_7 , düz eğim sabiti, aynı görüntülerin aksine aynadaki görüntülerin ayırt edilebilmesi için olanak sağlar [51].

2.2.3. Dikey özellik çıkarıcı blok algoritması

Bu bölümde özellik çıkarıcı algoritmalarından olan, dikey özellik çıkarıcı algoritması anlatılmıştır. Bu algoritması, farklı insanları temsil eden imzaların özellik vektörünü, aynı insanların imzalarının içindeki varyasyon seviyesine rağmen, açıklamaktadır [54].

Özellik çıkarıcının nasıl olduğu adım adım anlatılmıştır.

- Sınırlanmış imza görüntülerinin, ağırlık merkezi tanımlanmıştır. Ağırlık merkezi \bar{x} ve \bar{y} noktalarından, 2 parçaya ayrılır.
- Blok 1 ve blok 2'nin ağırlık merkezleri \bar{x} ve \bar{y} noktalarından ikiye bölünerek 4 tane küçük parçaya ayrılmış olsun.
- Bu imzaların her parçasını tekrar ağırlık merkezinden \bar{x} ve \bar{y} noktalarından 2'ye ayrılarak 8 tane küçük imza parçacıkları haline gelsin.
- Bu 8 parçanın gene ağırlık merkezinden, \bar{x} ve \bar{y} noktalarından ikiye ayrılarak 16 tane küçük imza parçası olsun.
- Her bir imza parçasının 3 tane özelliğini çıkarılsın. Piksellerin ağırlık merkezine olan sapması hesaplanarak city-blok mesafesini hesaplınsın. İmza görüntüsünün blok boyutu hesaplınsın.
- 16 bloğun 3 özelliğini birleştirerek 48 tane özellik vektörü elde edilsin.

BÖLÜM 3. YAPAY SİNİR AĞI TABANLI SINIFLANDIRMA

Bu bölümde yapay sinir ağları ile imza sınıflandırmaya yönelik bilgiler verilmiştir. Çevirim dışı imza tanımaya yönelik başlıca tekniklerden, şablon eşleştirme, istatistiksel yaklaşım, yapısal analiz yaklaşımı, spektrum analiz yaklaşımı ve sinir ağı yaklaşımı açıklanmıştır [55].

3.1. Şablon Eşleştirme

Bu sınıflandırma yöntemi basit, eski ama katı bir yaklaşımdır. Eldeki görüntü önceden saklanmış örüntü örnekleriyle karşılaştırılarak yapılır. Bu yaklaşım piksel seviyesinde ve yüksek seviyede gerçekleşir. Bu yaklaşımının sertliği nedeniyle birçok dezavantajı vardır. İmzaların bozulması durumunda veya bunun gibi birçok durumda şablon eşleştirme yaklaşımında başarısızlık olabilir. Orijinal imzadan sahteleri tespit edilebilir. Ama orijinal imzalarla, çok becerikli bir şekilde atılmış sahteler ayırt edilmeyebilir. Şablon eşleştirme yöntemi farklı özelliklere bağlı olarak, grafik eşleştirme, vuruş analizi ve geometrik özellik çıkarımı gibi çeşitli şekillerde kategorize edilebilir [1].

3.2. Spektrum Analiz Yaklaşımı

Bu yöntemin ilk aşamasında, spektrum analizi yörünge matrisinin bir sürümünün bir başka matris içine veri dönüşümüdür. Görüntünün bütün olası yerleri, kare bir pencere üzerine yerleştirilir [56]. Çoklu çözünürlük biçimindeki imza, basitçe eğrilik-tabanlı olarak ayrıştırılır. Bu yaklaşım uzun olarak yazılmış imzalar için kullanılır [1].

3.3. Yapısal Yaklaşım

Bu yaklaşım, grafik, dizi, ağaç eşleştirme teknikleri ve onların diğer tekniklerle birlikte kullanılması ile ilgilidir. Bu yaklaşım orijinal ve sahte imzaların belirlenmesinde iyi bir performans gösterir. Bu yaklaşımın en büyük dezavantajı, daha fazla doğruluk için daha büyük veri kümelerinin kullanılıyor olmasıdır [1].

3.4. İstatiksel Yaklaşım

Bu yaklaşımda özellikleri yönünden temsil edilen her görüntü d boyutlu uzayda nokta olarak görüntülenir. Farklı kategorilere ait her görüntü vektörü, d boyutlu özellik uzayında, küçük ve ayrık bölgeler işgal eder. Farklı sınıflara ayırmak için karar sınıfları ayarlanır. Özellik setinin etkinliği, farklı sınıflara ayırmak için, ne kadar iyi örüntü alınırsa o kadar iyi belirlenir. Hidden Markow Model (HMM) Bayes örüntü tanımada yaygın olarak kullanılan istatiksel bir yaklaşımdır. Gelişi güzel atılmış sahte imzaların yanı sıra, ustalıkla ve iyi taklit edilmiş imzaları da gerçeklerinden ayırabilir [1].

İstatiksel sınıflandırma aşamalarını aşağıdaki gibi gösterebiliriz.

- Model Oluşturma Ayağı: Önceden bilinen sınıfların tanımlanması,
- Verilen her örneğin bilinen bir sınıfa ait olduğu kabul edilir,
- Bu örnek seti “Öğrenme Kümesi (Seti)” olarak adlandırılır,
- Bulunan model “Karar Ağacı”, “Sınıflandırma Kuralları” veya “Matematiksel Formül” olarak ifade edilir.
- Modeli Kullanma: Sınıfları bilinmeyen verilerin sınıf tahmin,
- Modelin doğruluk derecesini kestir,
- Test kümesinin bilinen sınıf etiketleri tahmin sonucu elde edilen sınıflandırma sonuçları ile karşılaştırılır,
- Doğruluk derecesi test setindeki tahmin başarı oranıdır.
- Test kümesi öğrenme kümesinden bağımsızdır fakat aynı dağılımdan geldiği kabul edilir.

- Eğer modelin doğruluk derecesi kabul edilebilir ise modeli kabullen ve yeni noktaları(verileri) sınıflandırmak için uygula [57].

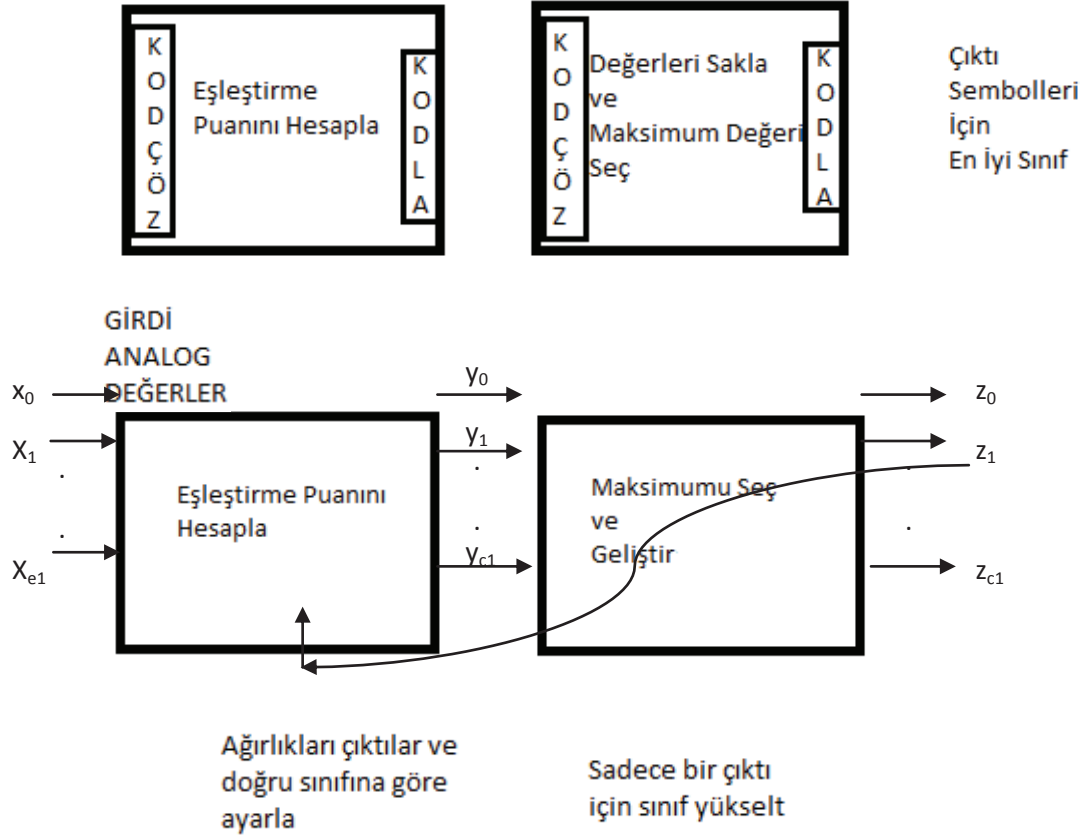
Şimdi istatistiksel sınıflandırma yöntemlerinden klasik sınıflandırmayı inceleyeceğiz.

3.4.1. Klasik sınıflandırma

Klasik sınıflandırma en basit olarak iki bloktan meydana gelir. Birinci blokta her bir sınıf için skorlar hesaplanırken ikinci blokta en büyük skora sahip skor belirlenir. İlk bloğa giren girişler, giriş elemanlarının e değerlerini veren semboller, aritmetik ve sembolik işlemler için gereklidir. Dış sembolik şekilden, iç tanımlamaya kodlanır ve sırasıyla girilir. Bu algoritmada, sınıfların her birisi için eşleştirme puan değeri hesaplanır. Yani her sınıfa ait örnek dokuya, giriş eşleştirmesine ne kadar yakın olduğu görülür. Örnek örüntü her sınıfın en iyi şekilde tanımlandığı örüntüdür. Birçok olasılıksal bir model, örnek örüntüden, giriş örüntüsünün genelleştirilmesini modellemek için kullanılır. Ve eşleştirme skoru, giriş örüntülerinin mümkün örneklerinden her birisinden üretilen olasılığı tanımlar. Bu durumda güçlü varsayımlar, giriş elementlerinin dağılımı ile yapılır. Dağılımların parametreleri, örüntü seti ihtiva eden örnekleri üzerinden hesaplanabilir. "Multivariate Gaussian" algoritması eşleştirme puanlarını hesaplamak için kolay bir algoritmadır. Şekil 3.1.'deki eşleştirme puanını hesapla bloğu ile sembolik olarak eşleştirme puanları kodlandırır ve sırasıyla sınıflayıcının ikinci bloğuna gönderir. Burada yeniden kodlandırılır ve en büyük puana sahip olan sınıf seçilir.

Aşağıdaki Şekil 3.1'de klasik ve yapay sınıflandırıcı yapısı görülmektedir.

Öğrenme sabitinden
hesaplanan parametreler



Şekil 3.1. Klasik sınıflayıcı (a) Sinir ağı sınıflayıcısı(b)

Sinir ağı sınıflayıcı da girdi değerleri girdi bağlantılarına karşı paralel olarak beslenir. Bu girdi değerleri ikili ya da sürekli olabilir. İlk blokta eşleştirme puanları hesaplanır ve analog çıkış hatları üzerinden bu puanlar paralel olarak ikinci bloğa çıkış olarak gönderilirler. Bu değerlerden en büyüğü seçilir ve geliştirilir. İkinci blok mümkün olan her c sınıfı için bir çıktıya sahiptir. Sınıflandırma tamamlandıktan sonra, seçilen çıkışın sınıfı güçlü ve diğer çıkışlar zayıf olurlar. Buradan da anlaşılacağı gibi klasik sınıflandırmanın girdisiyle çıktısı arası geçişler yapay sinir ağları gibi, paralel değil, seridir. Blok içerisindeki hesaplamalar, sırasıyla yapılarak, buna ek olarak da parametreler eğitilerek veriden tahmin edilir. Bundan sonra da parametreler sabitleştirilir.

Sinir ağı sınıflandırıcılarda, girdi ve çıktılar paralel yapıdadırlar. İç hesaplamaların teorik olarak paralel yapıldığı varsayılır [58].

3.5. Yapay Sinir Ağları

Yapay sinir ağları kavramı beynin taklit edilmesi fikri ile ortaya çıkmıştır. 1940'lı yılların başlarında YSA'nın gelişim yılı olarak anılır. Bu fikir McCulloch ve Pitts [59] tarafından ilk olarak ortaya atılmıştır. Bu bilim adamları birkaç hücre modelini geliştirdiler ve birkaç hücrenin ara bağlaşımını incelemişlerdir. Hebb [60], ilk öğrenme kuralını önermiştir. Rosenbatt [61], 1957'de perceptron modelini ve öğrenme kuralını geliştirmiştir. 1960-1962 yılları arasında Widrow ve Holf tarafından Adaline ve Lms kuralı geliştirilmiştir. 1969 yılında Minsky ve Papert [62], algılayıcıların kesin analizini yapmışlar, algılayıcıların karmaşık mantık fonksiyonları için kullanılmayacağını ispatlamıştır. Bu ispattan sonra yapay sinir ağları ile ilgili çalışmalar 1960-1980 arasında bir durgunluk dönemi yaşamıştır. Ancak bazı bilim adamları çalışmalarına devam etmişlerdir. (Taylor, Grossberg, Fukushima, Amari, Kohonen) 1982 yılında Hopfield çalışmalarıyla YSA'nın birçok problemi çözebilecek nitelikte olduğunu göstermiştir. Doğrusal olmayan Hopfield ağını geliştirmiş optimizasyon problemlerini çözebilmek için geliştirmiştir. 1982-1984 yılları arasında Kohonen öz düzenlemeli haritayı yapmıştır. Ayrıca kendi adıyla anılan eğitimcisz öğrenen bir ağ geliştirmiştir. 1986 yılından Rumelhart geriye yayılım ağını tekrar gündeme getirmiştir. 1988 yılında ise Chua ve Yang hücrel sinir ağlarını geliştirmişlerdir. Böylece Yapay sinir ağları adında bir bilim dalı ortaya çıkmıştır [63].

Yapay sinir ağlarını beyni oluşturan hücrelerin matematiksel olarak modellenmesi olarak tanımlayabiliriz. Ya da basit işlem birimlerinden oluşan, çeşitli denemeler sonucunda verilen eğitimle bilgiyi öğrenip bunu saklayan paralel olarak dağıtılmış bir sinir ağıdır diyebiliriz [64].

3.5.1. Yapay sinir ağı yapısı

İnsan beyni kompleks, paralel ve doğrusal olmayan bir yapıya sahiptir. YSA'da eğitim yoluyla öğrendiği bilgiyi saklayan ve bunu kullanan paralel dağıtık bir yapıdır. YSA görüntü tanıma, algılama, öğrendiklerinden yeni bilgi türetebilme keşfedebilme gibi beynin üstün niteliklerini makinelere/bilgisayarlara kısmen de olsa kazandırmaya çalışır. Çünkü bu gibi özelliklerde insan beyni, çok daha yeteneklidir. Örnek verecek olursak yarasanın sonar sisteminin etkinliğidir. Bir yarasa sonarı hedefin uzaklığı, büyüklüğü hedefin hızı, çeşitli uzuvlarının büyüklüğü, yeryüzünden yüksekliği, gökyüzünde bulunduğu yer ve bunun dışında hedefin belirgin özelliklerini tahmin ve başarı oranını radar veya solar mühendislik hesaplarıyla yakalanmayacak imkânsızlıktadır. Ya da önceden gördüğümüz bir insanı tanımamız oldukça 100-200 ms gibi çok kısa bir zaman alırken, bilgisayar bize göre çok basit olan bu işlemi çok uzun zamanda yapmaktadır. İnsan beyninin bu üstünlüğü sahip olduğu kompleks yapıdan gelmektedir. İnsan beyninin yaklaşık sinir sayısı 100 milyarlar ve ara bağlantı (synapses) sayısı 60 Trilyonlar mertebesinde. Bir biyolojik sinirin fonksiyonları, yapay sinir ağlarının yapısında bulunan mantık kapılarının fonksiyonlarından çok daha karmaşıktır. Beynin sakladığı yüksek seviyedeki bilgiyi nasıl tanımladığı bilinmemekle beraber, yüksek yoğunluklu ama yapay sinir ağlarına kıyasla oldukça yavaş üniteler kullanıldığı bilinmektedir. Beynin temelini oluşturan sinirler, yapay sinir ağlarının temelini oluşturan silikon mantık kapılarından daha yavaştır. Sinirler bilgiyi milisaniyeler süresince işlerken, silikon mantık kapıları nano saniyeler civarında, belki de daha düşük sürede işlerler. Yapay sinir ağları beyne iki yönüyle benzer. Birincisi yapay sinir ağları da beyin gibi bilgiyi öğrenme yoluyla elde eder. İkincisi ise beynin sinaps'larında olduğu gibi sinaptik ağırlıklar olarak bilinen sinirler arası bağlantılar bilgiyi saklamak için kullanılırlar [64].

Yapay sinir ağları tarafından model olarak alınan insan beyni nelerden oluşmaktadır. Bunları inceleyelim.

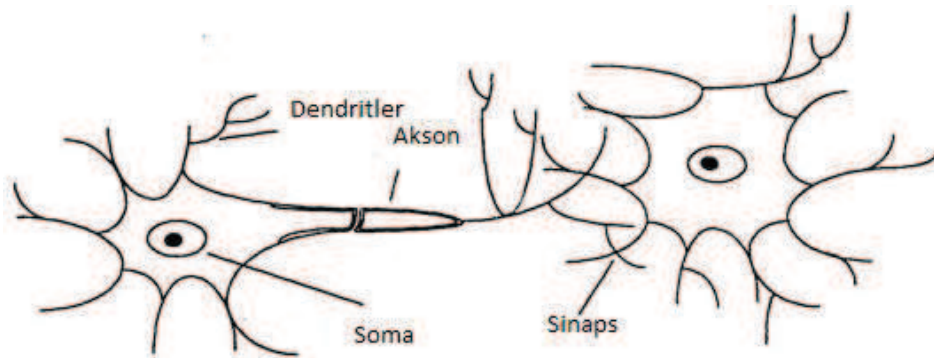
Biyolojik sinir sistemi dışarıdan gelen uyarıları alarak bunu alıp yorumlayan ve buna uygun bir tepki veren beynin bulunduğu üç katmanlı bir sistemdir denilebilir. Beyin

burada beyin merkezi sinir ağını oluşturmaktadır. Daha ayrıntılı olarak söyleyecek olursak beyin alıcı (receptor) sinirler tarafından alınıp elektriksel sinyallere çevrilen uyarıları alarak bunlara uygun tepkiler üretip bunu tepki sinirlerine ileten ve bunu alıcı ile tepki sinirleri arasında ileri geri besleme yoluyla yapan Merkezi Sinir Ağı'dır diyebiliriz [64]. Şekil 3.2'de biyolojik bir sinirin yapısı görülmektedir.



Şekil 3.2. Biyolojik sinir sistemi yapısı

Sinir hücreleri merkezi sinir sisteminin temel elemanıdır. Nöronlardan oluşur. Nöronlar ise, sinaps, dendrit, hücre gövdesi (soma), akson olmak üzere dört tane bölümden oluşmaktadır. Aşağıdaki Şekil 3.3'te biyolojik bir hücrenin yapısı görülmektedir.



Şekil 3.3. Biyolojik bir hücre

Nöronlar elektrik devrelerine benzer şekilde iletim yaparak beynin işlevlerini yerine getirmesinde ana elemandır [64].

Sinapslar gelen sinyali işleyen yapılardır. Sinapslar pre-sinaptik elektrik sinyallerini kimyasal sinyallere çeviren daha sonra da kimyasal sinyalleri post-sinaptik elektrik sinyallerine çevirerek dendritlere ileten biyokimyasal bir yapılardır. Yani sinapslar sinirler arasındaki bilgi akışını sağlar. Sinapslar bilgi geçiş noktalarıdır da diyebiliriz.

Dendritler bağı olduğu duyu organlarından veya nörondan gelen sinyalleri hücre gövdesine iletir [64].

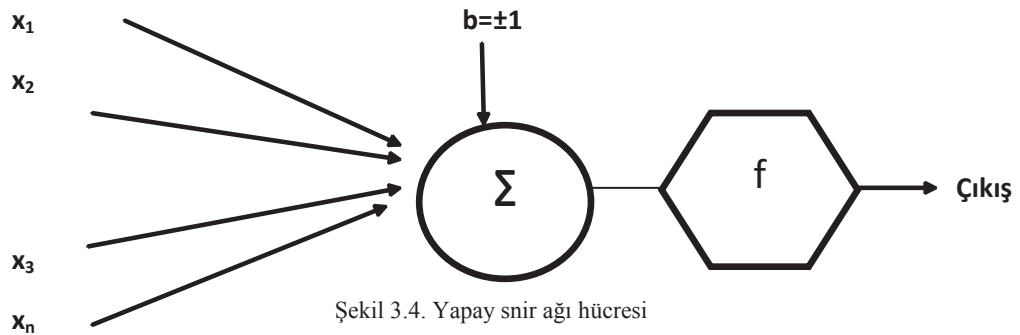
Hücre gövdesi past-sinaptik sinyalleri aksonlara doğru iletilirler. Hücre gövdesi aksonlardan diğer hücrelerin sinapslarına doğru aktarılan çıkış sinyali üretirler. Akson, dendritlerden gelen verileri hücrenin genel durumuna ve verilerin yaratacağı toplam etkiye göre aksonlar aracılığıyla diğer nörona iletilirler [64].

Aşağıdaki Tablo 3.1'de İnsan Sinir Sistemi ile Yapay Sinir Ağları Sistemi karşılaştırılmaktadır.

Tablo 3.1. Sinir sistemiyle YSA sistemi karşılaştırması

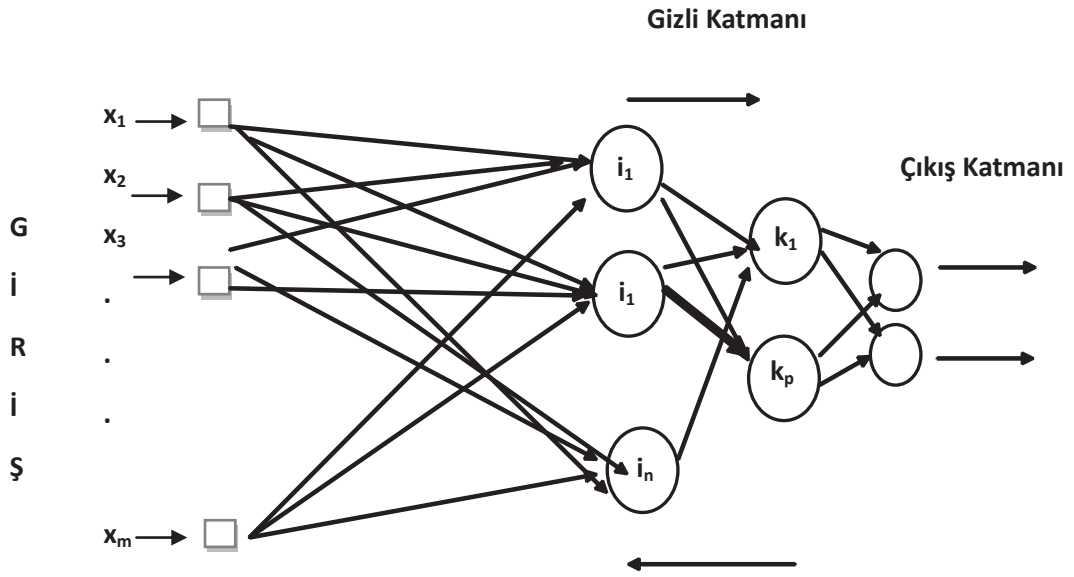
| Sinir Sistemi | YSA Sistemi Karşılığı |
|---------------|----------------------------|
| Sinir | Yapay sinir(işlem elemanı) |
| Sinaps | Ağırlık |
| Dendrit | Toplama Fonksiyonu |
| Hücre gövdesi | Transfer fonksiyonu |
| Aksonlar | Eleman çıkışı |

Aşağıdaki Şekil 3.4'te Yapay Sinir Ağı hücresi açıklanmaktadır.



x_1, x_2, \dots, x_n 'ler hücrelerin girdileridir. Alınan girdiler değişik ağırlık oranlarda $+1$ değerindeki bias değerleriyle toplanarak aktivasyon fonksiyonuyla işle yapılarak

çıkış oluşturulur. Bu çıkış sistem için istenilen çıkışla karşılaştırılır. Ve bir hata oranı bulunur. Bu hata oranına göre yapay sinir ağlarının girdilerinin ağırlıklarını (w_1, w_2, \dots, w_n) günceller. Böylece daha doğru sonuç için döngü oluşturulur. Ayrıca ağırlıkların öğrenmesi bu ağırlıkların güncellenmesiyle meydana gelir. Aşağıdaki Şekil 3.5'te multilayer perceptron (çok katmanlı geriye yayılım algoritması) gösterilmiştir.



Şekil 3.5. Çok katmanlı geriye yayımlı yapay sinir ağları

Perceptron beyin işlevlerini modelleyebilmek için ortaya çıkan tek katmanlı eğitilebilir ve tek çıkışa sahip ağ modelidir. Tek katmanlı perceptr'lar doğrusal problemlerinde kullanılırken çok katmanlı perceptron'lar lineer olmayan problemlerde kullanılabilir. Şekilde birçok sinir hücresinin birbirine bağlandığı ileri yönlü çok katmanlı yapay sinir ağı modeli görülmektedir. Giriş ve çıkış arasındaki katman veya katmanlar gizli katman olarak adlandırılır. Bununla birlikte YSA'da kaç tane gizli katman olacağı ve her bir gizli katmanda kaç nöron olacağı şimdiye kadar belirlenememiştir. Probleme göre değişen bu durum deneme yanılama yönüyle bulunabilir. Yapay sinir ağları kullanılan ağ modeline karşı değişik karakteristik özellikler gösterebilmesine rağmen temel bazı özellikler gösterebilmektedir.

3.5.2. Yapay sinir ağıları genel özellikleri

- Non-linear olabilme
- Genelleme Yeteneği
- Donanımsal Gerçeklenme

Bu özellikleri incelersek 1. Özellik sistemin paralelliği ve toplamsal işlevin yapısal olarak dağılımıdır. Yapay sinir ağıları birçok nörondan meydana gelir ve bu nöronlar eş zamanlı çalışarak karmaşık işlevleri yerine getirebilir. Bu süreç içinde bu nöronlarından herhangi biri bozulsada sistem güven içerisinde görevini yerine getirebilmektedir. 2. özellik genelleme yeteneğidir. Diğer bir deyişle ağı yapısının eğitim için kullanılan nümerik bilgilerden eşleştirmeyi sağlayan ağırlık değerlerinden en özgü olanlarını çıkarması böylelikle eğitim için kullanılmayan girdiler için de çeşitli yanıtlar üretebilmesidir. 3. özellik ise Sayısal olarak tasarlanan yapay sinir ağlarının donanımsal olarak gerçekleştirilebilmesidir [64].

Bu özelliklerden genelleme yeteneğine bakacak olursak elimizde imzalar var. Bu imzaları eğitimde tanımladığımızı varsayalım. Ağımıza imza olmayan başka bir girdi verirsek genelleme yeteneğinden dolayı bir imzaya benzetmeye çalışacak. Ve ağıımız yanlış sonuç verecek. O yüzden ağıımızı ne kadar çok veriyle eğitirsek o kadar doğru sonuçlar elde ederiz. YSA'nın kolaylıkla donanımsal olarak gerçekleştirilmesi günlük hayatta daha fazla göreceğimiz anlamına gelmektedir. Başlıca sınıflandırma, modelleme ve tahmin uygulamaları olmak üzere pek çok alanda kullanılmaktadır [65].

3.6. Radyal Taban Fonksiyonlu Yapay Sinir Ağı

Radyal Tabanlı Fonksiyon Ağı, aktivasyon fonksiyonu radyal tabanlı bir fonksiyon olan özel bir sinir ağıdır. RBF ağıları eğri uyurma zaman serisi tahmini, kontrol ve sınıflandırma problemlerinde oldukça popülerdir. Radyal Taban Fonksiyonlu Yapay Sinir Ağı diğer sinir ağlarından birkaç tane farklı ayırt edici özelliğe sahip bir sinir ağıdır. Çünkü daha kompakt topolojili ve daha hızlı öğrenme hızına sahip

olduğundan, büyük ilgi görmüş ve bazı bilim ve mühendislik alanlarında yaygın olarak uygulanmıştır [66-70].

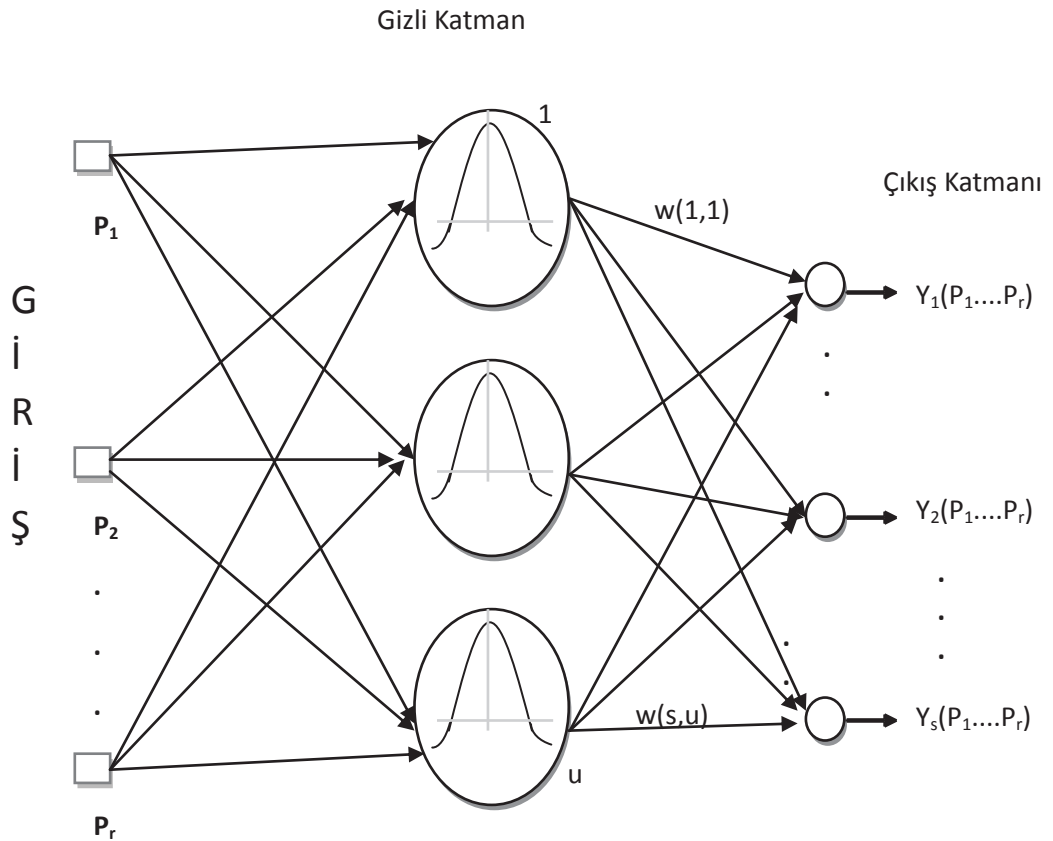
Radyal Tabanlı Fonksiyonlu Yapay Sinir Ağları birçok YSA uygulamasında çoğunlukla geri yayımlı, ileri beslemeli çok katmanlı algılayıcı ağları kullanılmasına rağmen bu tip ağlar eğitim sürelerinin uzun olması, yerel optimuma takılma, kaç ara katman kullanılacağı ve bu ara katmanlardaki nöron sayısının belli olmaması gibi dezavantajlara sahiptir [71]. Radyal Taban Fonksiyonlu (RTF) Yapay Sinir Ağı güçlü, hızlı öğrenme ve kendi kendine organize olabilen sinir ağıdır. Özellikle linear olmayan problemlerdeki, sınıflandırma ve öğrenme hızının daha yüksek olması nedeniyle geriye beslemeli parçeteron ağlardan daha iyidir [72,73].

Birçok örüntü tanıma sistemlerinde, genellikle kullanılan yöntem, istatistiksel yaklaşımdır. Bu sayede karar verme teorisi, sınıflandırma tasarımında kullanılan giriş örüntülerinin istatistiğinden türetilmiştir. Bu paradigmanın, çeşitli örüntü tanıma problemlerinde başarılı olduğu kabul edilmesine rağmen, uygun özellik seçimi mümkün olmadıkça, istatistiksel bilgileri ifade etmek zordur. Ayrıca, bu yaklaşımda bir sınıflandırıcı tasarımı çok sezgisel bilgi gerektirir. Sinir ağı tabanlı değerler dizisi, istatistiksel ve yapısal tabanlı yaklaşımlara dayalı çeşitli sınıflandırıcıların uygulanmasında yeni bir yöntem olarak, onların öğrenme kabiliyetleri ve iyi genelleme yeteneklerinin avantajları nedeniyle birçok avantaja sahip olduğu kanıtlanmıştır. Genellikle söylenen şey, çok katmanlı ağların geriye yayımlı algoritmalarla birleştirildiğidir. Ancak bu konuda 2 büyük eleştiri vardır. Bir tanesi yoğun hesaplamalardan dolayı yavaş yakınsama hızı bir diğeri de, mutlak minimumu elde edilebileceğinin garantisinin olmamasıdır. Diğer bir yandan RBF Yapay Sinir Ağları son zamanlarda sinir ağlarındaki geniş uygulama alanları için oldukça ilgi görmüş ve dikkat çekmiştir. RBF ağının belirgin özellikleri şunlardır [74].

- Onlar evrensel tahmin edicilerdir.
- En iyi yakınsama özelliğine sahiptirler.
- Yerel olarak ayarlanmış sınırları nedeniyle yüksek öğrenme hızına sahiptirler.
- Diğer sinir ağlarına göre daha kompakt bir topolojiye sahiptirler.

Moody ve Darken [75], RBF Yapay Sinir Ağı, giriş boyutunun yeterince küçük olduğu gerçek değerli eşleştirmelerde, sürekli veya parçalı sürekli yaklaşımlarda öğrenme için en uygun olduğu üzerinde duruyorlar.

Radyal Taban Fonksiyonlu Yapay Sinir Ağı, giriş katmanı, ara katman ve çıkış katmanı olmak üzere üç katmandan oluşan ileri beslemeli bir ağ yapısıdır [76].



Şekil 3.6. Radyal taban fonksiyonlu sinir ağı

Radyal Tabanlı Yapay Sinir Ağı Şekil 3.6'da gösterilmiştir. R_r 'den R_s 'ye bir eşleştirme olarak kabul edilebilir. P_1, P_2, \dots, P_r radyal tabanlı sinir ağının girişleridir. u tane sinir kullanılmıştır. w 'ler ağırlık vektörleri, Y_1, Y_2, \dots, Y_s çıkış değerleridir.

$P \in R^r$ giriş vektörü olsun ve $C_i \in R^r$ ($1 \leq i \leq u$) giriş vektörlerinin prototipi olsun. Her RBF birimi için çıkış aşağıdaki Denklem 3.1'deki gibi olur.

$$R_i(P) = R_i(\|P - C_i\| \quad i=1, \dots, u \quad (3.1)$$

$\|\cdot\|$ Giriş uzayının Öklid normunu gösterir. Birçok olası radyal bazlı fonksiyon arasından, faktörü alınabilir olması nedeniyle gauss fonksiyonu tercih edilir. Bundan dolayı,

$$R_i(P) = \exp \left[-\frac{\|P - C_i\|^2}{\sigma_i^2} \right] \quad (3.2)$$

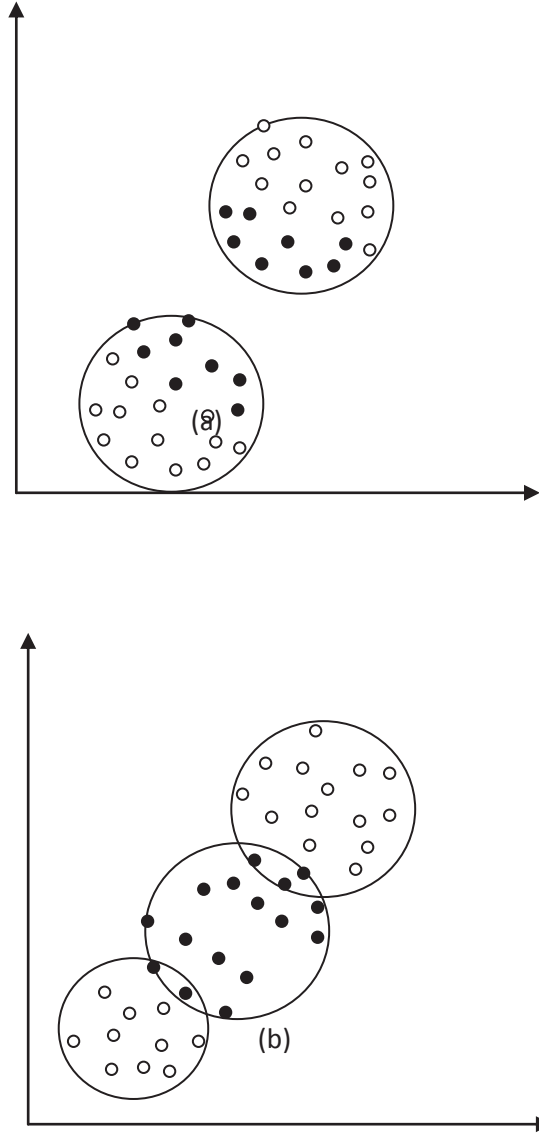
σ , RBF biriminin genişliği ve RBF sinir ağıının i . çıkışı $y_i(P)$,

$$y_i(P) = \sum_1^u R_i(P) \times w(j, i) \quad (3.3)$$

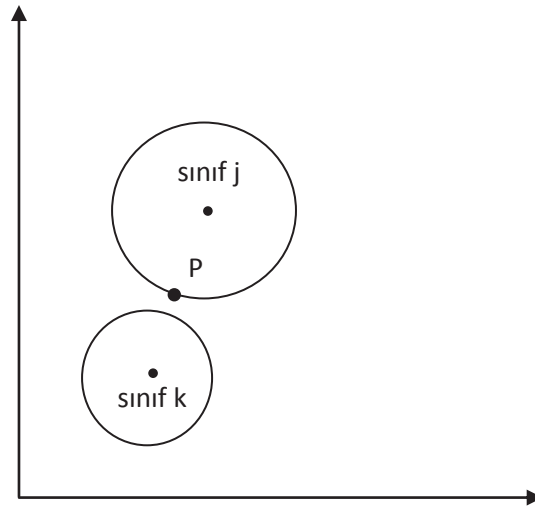
$R_0=1$ olduğu yerde, $w(j, i)$ ağırlık veya i . alıcı alanın j .gücüdür. Ve $w(j, 0)$ j . çıkışın bias (eğilim değeri)'dir.

Denklem 3.2 ve Denklem 3.3'te görüldüğü gibi RBF sinirsel sınıflandırıcının çıkışları doğrusal ayırma fonksiyonu ile tanımlanmıştır. Bu fonksiyonlar çıkış uzayında doğrusal karar sınırları (hiper düzlemler) üretirler. Sonuç olarak, RBF sinirsel sınıflandırıcının güçlü performansı, u RBF birimleri tarafından geliştirilen doğrusal olmayan dönüşümlerle oluşturulmuş u boyutlu uzaydaki sınıfların ayırma bağıdır. Cover'ın teoremine göre örüntülerin karmaşık örüntü sınıflandırma şeklindeki problemlerde, yüksek boyutlu lineer olmayan uzaydaki ayırma düşük boyutlu uzaya göre daha kolaydır. Gauss düğümlerinin sayısı $u \geq r$ olmalıdır. (r giriş uzayının boyutu) diğer bir yandan Gauss birimlerinin artması, özellikle küçük eğitim setleri olması durumunda yüksek uyum nedeniyle düşük genellemeye sebep olabilir. RBF gizli düğümlerinin uygun bir şekilde seçimi için eğitim örüntülerini analiz etmek önemlidir [74].

Geometrik olarak RBF sinir ağlarındaki temel fikir, hiper küreler şeklindeki alt uzayların birçok parçaya bölünmesidir. Buna göre RBF sinir ağlarında problemleri çözmek için oldukça yaygın olarak kullanılan mantıklı yaklaşımlar mevcuttur. Bunlar; k-ortalamlar kümelemesi, bulanık k-ortalamlar kümelemesi ve hiyerarşik kümelemedir [74].



Şekil 3.7. İki boyutlu desenler ve kümeleme (a) klasik kümeleme (b)homojen analiz ile kümeleme



Şekil 3.8. Gauss genişliğinin kümelemeye etkisi

Ancak bu gibi kümeleme yaklaşımların doğası gereği denetimsiz öğrenme algoritmaları olduğu bu nedenden dolayı da kullanılan örüntüler ile ilgili herhangi bir sınıf bilgisi olmadığı unutulmamalıdır. Açıklayıcı bir örnek olarak Şekil 3.7.'de gösterilen basit bir eğitim setini (x_k, y_k) düşünün. Siyah ve beyaz veri noktaları bağımlı değişken tarafından kontrol edilen ilgili değerleri yansıtır. y_k 'yı dikkate alınmadan k -ortalama kümesi basit olarak kullanırsa, iki belirgin küme Şekil 3.7(a)'daki elde edilir. Bu başlangıçtaki yanlış sınıflandırmayı beraberinde getirir. Kümelemenin sınırları sonraki öğrenmelerde değiştirilmiş olmasına rağmen, istenmeyen ve yüksek derecede baskın ortalama fenomenlere yol açabilir. Aynı zamanda da öğrenmeyi daha az etkili yapabilir. Homojen kümeler Şekil 3.7.(b)'de gösterildiği gibi korunmalıdır. Diğer bir deyişle eğitim setleri oluşturulurken sınıf bilgileri göz önüne alınarak oluşturulmalıdır. Eğitim örüntülerinin sınıf bilgisi göz önüne alınırken, eğitim desenlerinin sadece mesafesine bağlı olmadığı, bir de Gauss genişliğine bağlı olduğu vurgulanmalıdır. Şekil 3.8.'de gösterildiği gibi, P Öklid mesafesindeki sınıf k 'nin merkezine yakın, ama her sınıf için farklı bir Gauss genişlikleri seçilebilir ve böylece p noktası j sınıfında k sınıfına göre daha büyük sınıf üyelerine sahip olur [74].

İmza tanıma açısından bakıldığında, farklı sınıflar arasındaki uygun sınırlar RBF sinir ağları tarafından tahmin edilebilir. Aksine RBF sinir ağları açısından, sinir ağları

özellik hiper uzayından sınıflara eşleşme olarak düşünülmelidir. Her desen gerçek bir vektör tarafından temsil edilir ve her sınıf için uygun bir kod atanır. Ve bundan dolayı,

- Ağın giriş sayısı özelliklere eşit olmalıdır. (Giriş uzayının boyutu)
- Çıkış sayısı sınıflara eşit olmalıdır.

Gizli düğümleri seçmek zahmetli bir iştir. Mimari karmaşıklığı arttırmak veya azaltmak etrafında dönen farklı yaklaşımlar önerilmiştir. Birçok araştırmacının araştırmaları gizli birimlerin sayısının, eğitim örüntülerin geometrik özelliği kadar aktivasyon fonksiyonuna da bağlı olduğunu göstermiştir. Yine de bu konu RBF sinir ağları uygulamalarında açık bir konudur [74]. RBF sinir ağlarında gizli katman sayısının belirlenmesi çok önemlidir. Çünkü ağın kompleksliğini ve genelleme yeteneğini etkiler. Eğer gizli katmandaki nöron sayısı yetersiz ise, RBF ağı yeterince öğrenemez. Diğer bir yandan, nöron sayısı çok fazla ise, zayıf genelleme ve aşırı öğrenme durumu meydana gelebilir [77]. Gizli katmanın merkezlerinin pozisyonu ağın performansını önemli ölçüde etkiler [78]. Bu nedenle merkezlerin uygun olarak yerleşiminin belirlenmesi önemli bir konudur. Gizli katmanda, her nöron aktivasyon fonksiyonuna sahiptir. Gauss fonksiyonunun davranışını kontrol eden bir yayılma parametresi vardır ve Gauss fonksiyonu, en çok tercih edilen aktivasyon fonksiyonudur. RBF ağlarında eğitim işlemi, her nöronun yayılım parametrelerini uygun duruma getirmeyi içermektedir. Daha sonra gizli katman ve çıkış katmanı arasındaki ağırlıklar uygun seçilmelidir. En sonunda RBF ağı eğitime işleminde belirlenen bias değerleri her çıkışa eklenir [79].

Radyal bazlı sinir ağı matlabda yaklaşık olarak aşağıdaki fonksiyonu kullanır. newrb radyal bazlı ağ gizli nöronların sayısına ya da belirlenen hedef hata oranına ulaşıncaya kadar nöron ekler [80].

newrb(X,T,GOAL,SPREAD,MN,DF),

$X=R \times Q$ Q girişi vektör matrisi.

$T = S \times Q$ Q çıkış sınıf vektörü matrisi.

GOAL=Hata ortalamalarının karesi, varsayılan=0.0.

SPREAD=Radyal tabanlı fonksiyonların yayılımı, varsayılan=1.0

MN=Maksimum nöron sayısı, varsayılan=Q.

DF=Ekranında kaç tane nöron eklendiğinin gözükeceğinin sayısı

Radyal temelli fonksiyonun zorluğu için bir yayılma sabiti belirlemektir. Uygunsuz bir yayılma sabiti yetersiz uyum ya da aşırı uyuma neden olur [81,82]. Birçok araştırmacı, yayılım sabitini bulabilmek için birçok deneme yapmışlardır [83,84]. Ancak yayılım sabitini çok büyük veya sayılamayacak kadar büyük almak iyi bir sonuç vermez.

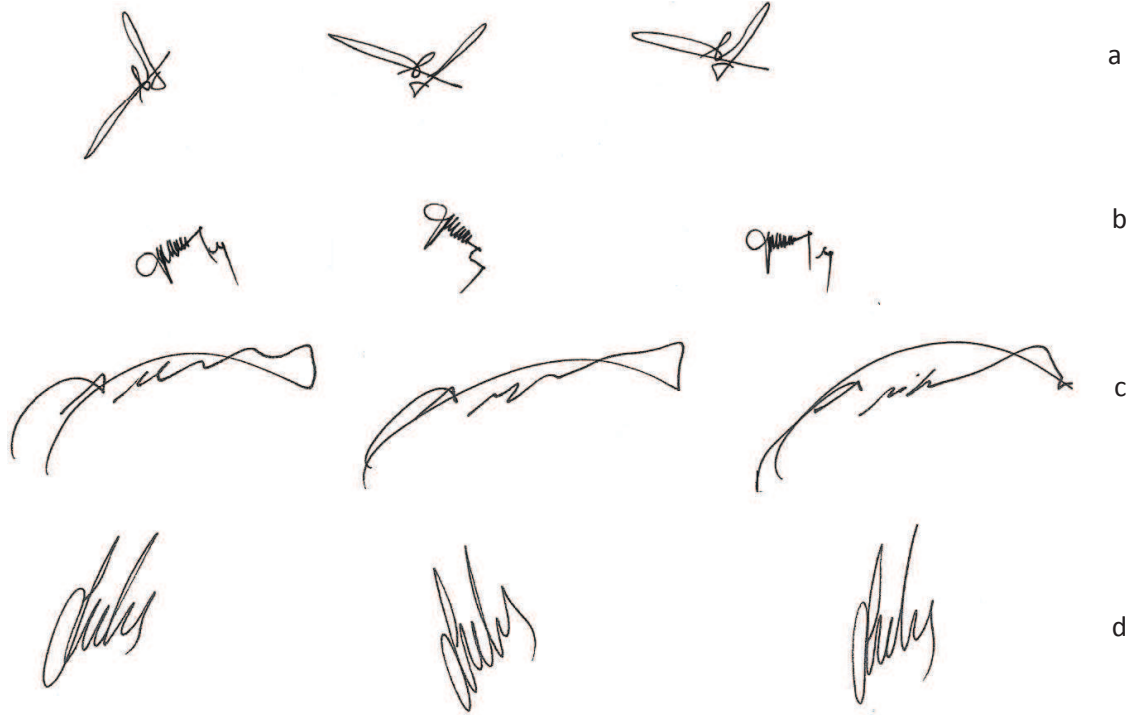
BÖLÜM 4. İMZA TANIMA UYGULAMASI

4.1. İmzaların Veritabanına Alınması

Bu tez çalışmasında İşlemci Intel(R) Core(TM) i5-5200U CPU @ 2.20GHz, 2201 Mhz, 2 Çekirdek, 4 Mantıksal İşlemcili 8GB RAM donanımlarına sahip, 64 bitlik Windows 8.1 Single Language işletim sistemi özelliklerine sahip bir diz üstü bilgisayar kullanılmıştır. Bu özellikler ileride anlatılacak olan özellik çıkarma, eğitim süresi veya performans analizi işlemlerinde, büyük önem taşımaktadır.

Bu çalışmada çevirimdışı imza tanıma uygulaması gerçekleştirilmiştir. Bunun için önce görüntüyle ilgili herhangi bir bilginin kaybolmaması açısından. bmp formatında imza görüntüleri alınmıştır. Bu uygulamada toplamda 24 kişiden imza alınmıştır. İmza kâğıdına her kişi için 2 tane kağıt kullanarak toplamda 36 tane imza toplanmıştır. Toplamda $36 \times 24 = 864$ tane imza elde edilmiştir. Bu imzaların her kişi için 26×24 tanesini=624 eğitim için, $10 \times 24 = 240$ test için kullanılmıştır. Uygulamanın ara yüzünü ve kodlama kısmının her ikisi de matlab R2012a yazılımında yapılmıştır.

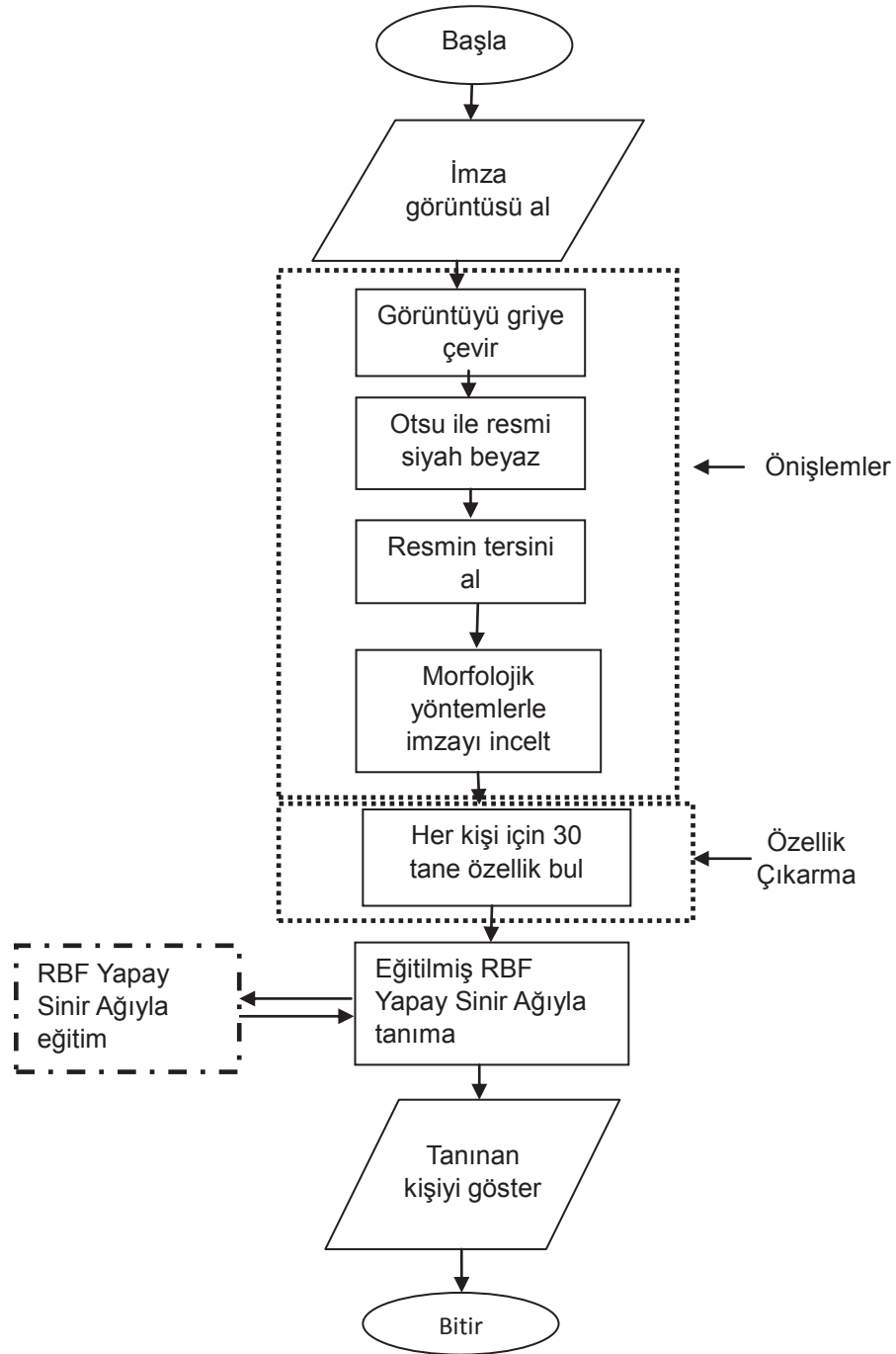
Şekil 4.1'de birkaç kişiye ait alınmış imza örneklerini görebilirsiniz.



Şekil 4.1. İmza örnekleri a) A.C.T. adlı kişiye ait imza örnekleri b) A.S.T. adlı kişiye ait imza örnekleri c) A.T. adlı kişiye ait imza örnekleri d) S.T. adlı kişiye ait imza örnekleri

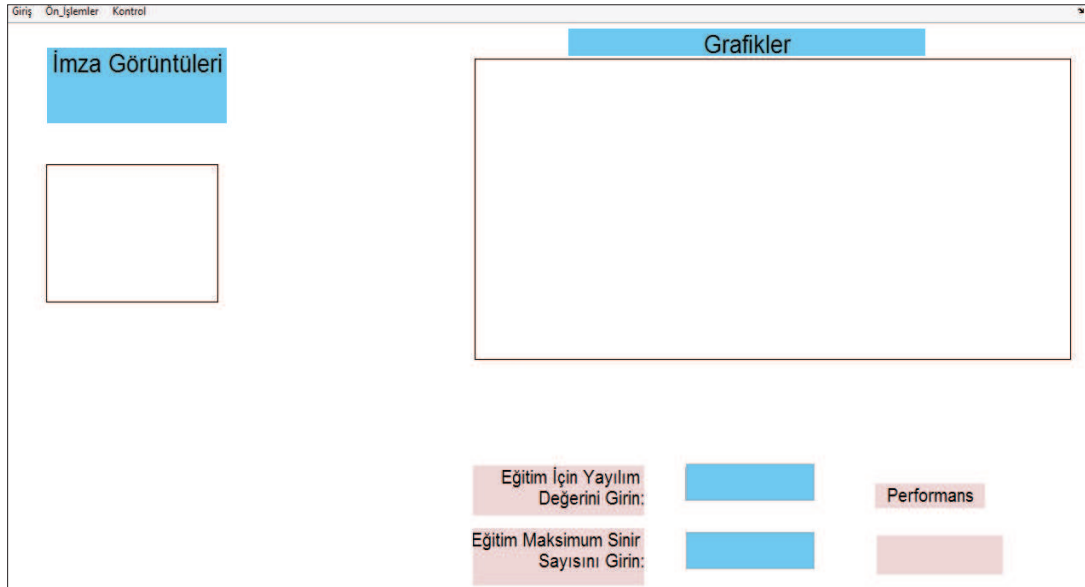
İmzalar önce 256 gri seviyesine dönüştürülmüş, daha sonra otsu metoduyla siyah beyaza çevrilmiştir. Siyah beyaza çevrilen görüntünün tersi alınarak, morfolojik işlemlerden iskelet çıkarma işlemine tabi tutulmuştur. İmzalar temiz bir şekilde alındığından ve sisteme aktarıldığından dolayı herhangi bir gürültü giderici bir filtre uygulanmamıştır. Ondan sonra imzanın başlangıç ve bitiş yerleri (siyah piksellerin başladığı ve bittiği yer) göz önüne alınarak özellikler bulunmuştur. Boyut her imzaya göre değişmektedir. Bunun yapılmasındaki amaç imzaların farklı boyutlarda olmasından dolayı özellik vektörlerinin daha anlamlı bir şekilde çıkarılmasının istenmesi ve imzaların çevresinde yer alan boşluklarla meydana gelebilecek zaman kaybının önlenmek istenmesidir. Daha sonra eğitim setindeki imzaların özellik vektörleri çıkarılmış ve aynı şekilde test imzalarının da özellik vektörleri çıkarılmıştır. İmzaların özellikleri çıkarılırken 0 ve 1 arasında normalize edilmiştir. Eğitim imzaları RBF sinir ağıyla eğitilmiş ve test imzalarıyla imzaların doğru tanınıp, tanınmadığı test edilmiştir.

Yapılan işlemlerin algoritmasını Şekil 4.2’de görebilirsiniz.



Şekil 4.2. İmza tanıma uygulama akış şeması

Şekil 4.3 ve Şekil 4.4 Şekil 4.5 ve Şekil 4.6'da imza tanıma uygulamasının resimlerle tanıtımı yapılmaktadır. Şekil 4.1.'de uygulamanın arayüzü, Şekil 4.2'de uygulamanın 'Giriş' menüsü, Şekil 4.3'de uygulamanın 'Ön İşlemler' menüsü, Şekil 4.4'te uygulamanın 'Kontrol' menüsü tanıtılmaktadır.



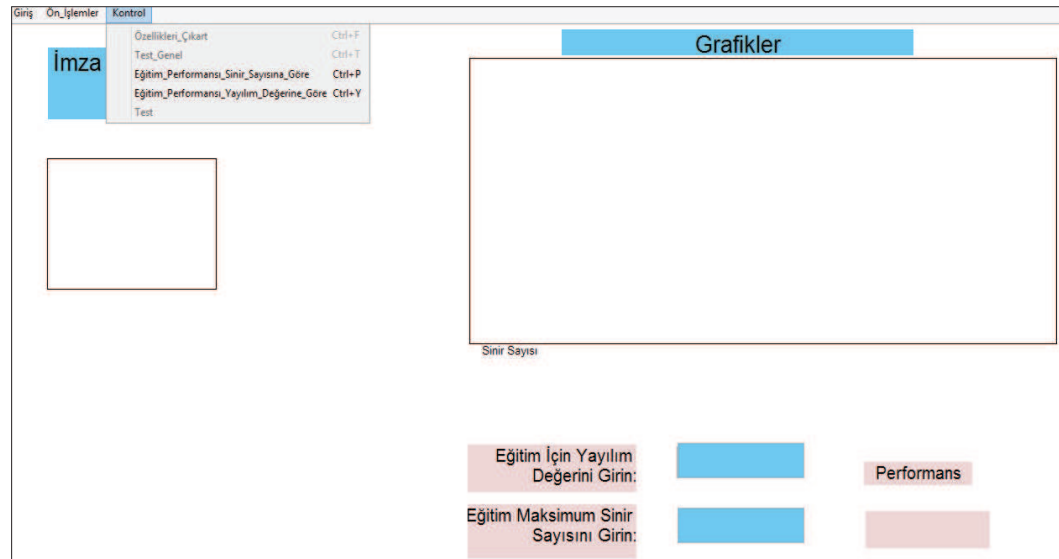
Şekil 4.3. İmza tanıma uygulaması ilk çalıştırıldığındaki arayüzü



Şekil 4.4. İmza tanıma uygulamasındaki giriş menüsü

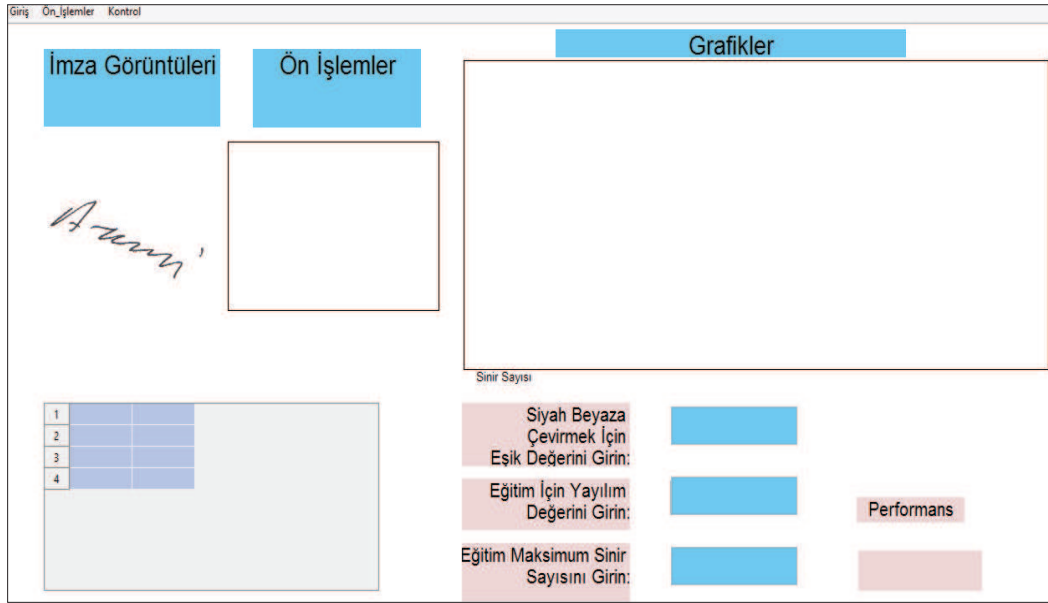


Şekil 4.5. İmza tanıma uygulamasındaki ön_işlemler menüsü



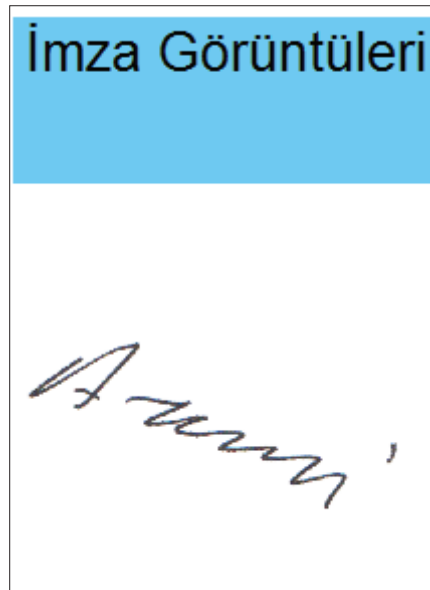
Şekil 4.6. İmza tanıma uygulamasındaki kontrol menüsü

Giriş menüsünden 'Dosya_Aç' seçeneği ile istenilen dosya seçildikten sonra birçok işlem aktif hale gelir. Uygulamada 'Dosya_Aç' seçeneği ile istenilen dosya seçildikten sonra uygulama Şekil 4.7'deki gibi olmuştur.



Şekil 4.7. Test edilmesi istenen dosya açıldıktan sonra imza tanıma uygulaması arayüzü

Alınan görüntü örneği Şekil 4.8'de gösterilmiştir.



Şekil 4.8. Alınan görüntü örneği

4.2. Ön İşlemler

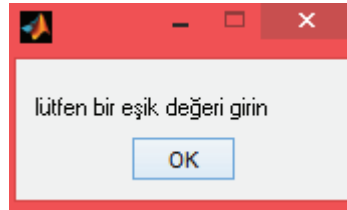
Birçok uygulamada görüntülerin griye çevrilmesi gerekmektedir. Uygulamamızdaki 'Dosya Aç ' seçeneği ile dosya seçildikten sonra 'Ön İşlemler' menüsündeki bütün

seenekler aktif olmaktadır. Buradan 'Griye_evir' seeneęi ile rnek grntmz 8 bitlik gri grntye evrilmiřtir. řekil 4.9'da gsterilmiřtir.



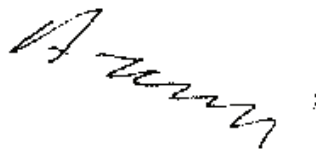
řekil 4.9. A.T. adlı kiřinin imzasının 8 bitlik gri dzeyinde griye evrilmiř hali

Grnt, iki řekilde siyah beyaza evrilmiřtir. Bunlardan bir tanesi 'n_İřlemler' mensnden yapılan Elle_Siyah_Beyaz seeneęinde, elle girilen eřik deęerine gre, grnty siyah beyaz yapılmıřtır. 'Siyah Beyaza evirmek İin Eřik Deęeri Girin' karıřındaki metin kutusu boř geildięinde, uygulama řekil 4.10'da grldę gibi hata mesajı vermektedir.



řekil 4.10. Eřik deęeri girilmedięinde karıřılařılan hata mesajı

Eřik deęerini farklı deęerler girildięinde elde edilen grntler řekil 4.11, řekil 4.12 ve řekil 4.13'te gsterilmiřtir.



řekil 4.11. Eřik deęeri 100 olan A.T. adlı kiřinin imzasının ikili grnts

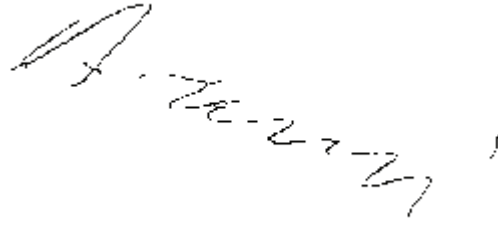
Şekil 4.12. Eşik değeri 150 olan A.T. adlı kişinin imzasının ikili görüntüsü

Şekil 4.13. Eşik değeri 250 olan A.T. adlı kişinin imzasının ikili görüntüsü

Herkesin kullandığı kalem, kalemi bastırma düzeyi gibi etkenler aynı olmadığından dolayı, eşik değerini otomatik olarak hesaplayan bir algoritmaya sahip otsu metodu kullanılmıştır. Bunun işlem uygulamanın 'Ön İşlemler' menüsündeki 'Otsu_Siyah_Bayaz_Yap' seçeneği ile yapılmıştır. Otsu metoduyla ikili görüntüye çevrilmiş imza görüntüsü Şekil 4.14'te gösterilmiştir.

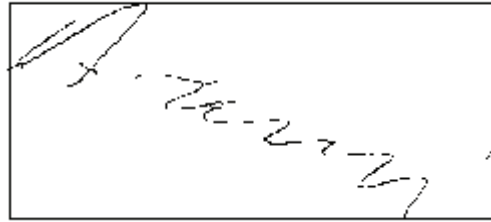
Şekil 4.14. Otsu metoduyla elde edilmiş ikili görüntü

Siyah beyaz çevrilen görüntünün, çeşitli morfolojik işlemlerden geçebilmesi için görüntünün tersi alınması gerekiyor. Bunun için görüntünün tersi alınmıştır. Tersine alınan imza görüntüsünün özelliklerini doğru bir şekilde çıkarmak için imza inceltilmiştir. Matlab'ın morfolojik olarak iskelet çıkarma metodu kullanılmıştır. İskeleti çıkarılmış bir imza görüntüsü Şekil 4.15'te gösterilmiştir.



Şekil 4.15. Morfolojik iskelet çıkarma uygulanmış imza

Özelliklerin düzgün ve doğru elde edilebilmesi için, imzanın çevresindeki gereksiz ayrıntılardan kurtulmak gerekir. Bunun için imzaları siyah noktaların başlangıç ve bitiş seviyesini kullanarak imzaların özellikleri çıkarılmıştır. İmzaların özellikleri normalize edilmiştir. Bu işlem uygulamanın 'Ön_İşlemler' menüsündeki 'İskeletini_Çıkart' seçeneği ile yapılmıştır. İskeleti çıkarılmış imza görüntüsü Şekil 4.16'da gösterilmektedir.



Şekil 4.16. Özellikler çıkartılırken kullanılan imza

4.3. Özellik Çıkarma

Özellik çıkarma işlemi, Şekil 4.6'da gösterildiği gibi, kontrol menüsünün altından yapılmaktadır. Dosya açtıktan sonra aktif olmaktadır. Daha sonra özellikler çıkarılmıştır. Sırasıyla çıkarılan özellikler şunlardır.

- İmzanın yoğunluğu
- genişlik
- yükseklik
- genişlik/yükseklik oranı
- x düzlemindeki ağırlık merkezi

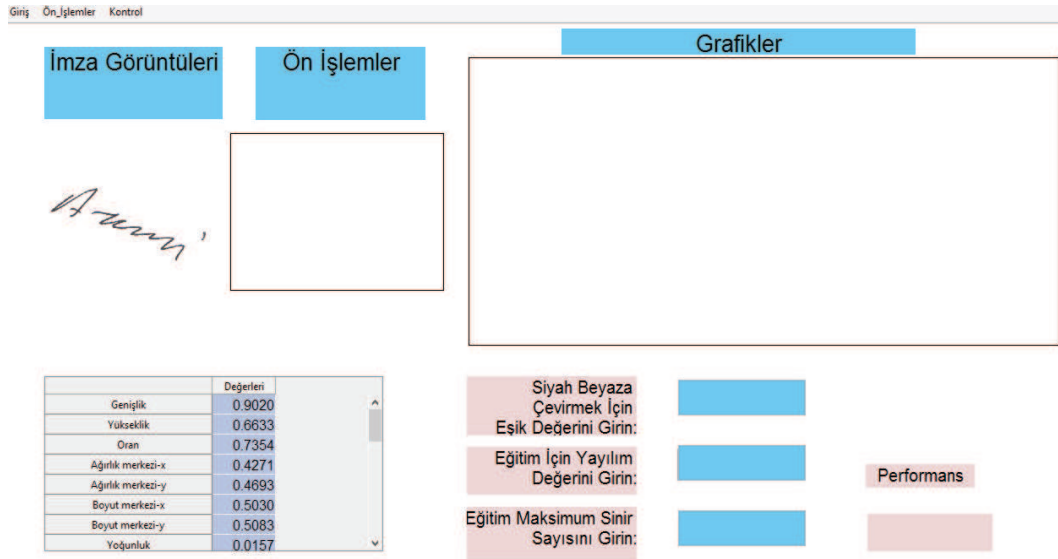
- y düzlemindeki ağırlık merkezi
- x düzlemi uzunluğunun merkezi
- y düzlemi uzunluğunun merkezi
- x düzlemindeki ağırlık merkezi ile x düzlemi uzunluğunun merkezi arasındaki fark
- y düzlemindeki ağırlık merkezi ile y düzlemi uzunluğunun merkezi arasındaki fark
- x ve y düzlemleri ağırlık merkezlerinden imzaların 4'e bölünmesi ve yoğunluklarının bulunması
- 4 e bölünen her parçanın x ve y düzlemleri ağırlık merkezlerinden 4'e bölünmesi ve toplamda 16 parçanın yoğunluklarının bulunması

Eğitilmek adına 8 kişinin 1, imzalarına ait özellikler kaydedilmiş olan veritabanından alınarak Tablo 4.1'de gösterilmiştir.

Tablo 4.1. 8 kişinin 1.imzalarına ait 0-1 arası normalize edilmiş 30 tane özellik

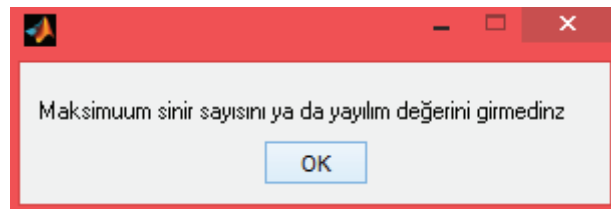
| | 1 kişi | 2 kişi | 3 kişi | 4 kişi | 5 kişi | 6 kişi | 7 kişi | 8 kişi |
|----------------------------|----------|----------|----------|----------|----------|----------|----------|----------|
| Genişlik | 0,006433 | 0,004044 | 0,013321 | 0,016554 | 0,00938 | 0,0132 | 0,016139 | 0,006784 |
| Yükseklik | 0,5 | 0,315956 | 0,683983 | 0,876623 | 0,902 | 0,91 | 0,655889 | 0,654919 |
| Oran | 0,48285 | 0,465374 | 0,715415 | 0,55303 | 0,663333 | 0,633333 | 0,549801 | 0,39314 |
| Ağırlık Merkezi-x | 0,965699 | 1,472909 | 1,045955 | 0,630864 | 0,735403 | 0,695971 | 0,838253 | 0,600288 |
| Ağırlık Merkezi -y | 0,473551 | 0,385943 | 0,385167 | 0,470774 | 0,427127 | 0,446133 | 0,426376 | 0,384429 |
| Boyut Merkezi-x | 0,410134 | 0,556247 | 0,612742 | 0,615674 | 0,469332 | 0,515988 | 0,446644 | 0,596922 |
| Boyut Merkezi-x | 0,458333 | 0,380727 | 0,419913 | 0,514069 | 0,503 | 0,501 | 0,427252 | 0,412628 |
| Yoğunluk | 0,383905 | 0,65928 | 0,551383 | 0,587121 | 0,508333 | 0,5 | 0,501992 | 0,621372 |
| Merkezi Farklar-x | 0,030436 | 0,01651 | 0,0508 | 0,049388 | 0,084116 | 0,060293 | 0,001335 | 0,043058 |
| Merkezi Farklar-y | 0,054321 | 0,221398 | 0,085767 | 0,051629 | 0,058796 | 0,025245 | 0,100668 | 0,062191 |
| 1.Parça Yoğunluk | 0,026776 | 0,084135 | 0,021371 | 0,026456 | 0,040069 | 0,00532 | 0,057318 | 0,018995 |
| 2.Parça Yoğunluk | 0,014646 | 0,039116 | 0,021789 | 0,020579 | 0,000468 | 0,033106 | 0,056818 | 0,042399 |
| 3.Parça Yoğunluk | 0,019044 | 0,006542 | 0,045271 | 0,058288 | 0,005149 | 0,052102 | 0,042374 | 0,037647 |
| 4.Parça Yoğunluk | 0,049442 | 0,02091 | 0,026165 | 0,038217 | 0,021104 | 0,003434 | 0,02972 | 0,008666 |
| 1.1.Parça Yoğunluk | 0,058079 | 0,074717 | 0,023604 | 0,008916 | 0,049465 | 0,108 | 0,045089 | 0,060504 |
| 1.2.Parça Yoğunluk | 0,008333 | 0,076923 | 0,027809 | 0,07958 | 0,033333 | 0,108 | 0,05625 | 0,045037 |
| 1.3.Parça Yoğunluk | 0,005842 | 0,074686 | 0,026667 | 0,047581 | 0,048849 | 0,134783 | 0,080256 | 0,051786 |
| 1.4.Parça Yoğunluk | 0,054087 | 0,108974 | 0,034194 | 0,069088 | 0,032091 | 0,130435 | 0,095881 | 0,081055 |
| 2.1. Parça Yoğunluk | 0,015873 | 0,096296 | 0,027854 | 0,02268 | 0,25 | 0,01047 | 0,062988 | 0,064688 |
| 2.2. Parça Yoğunluk | 0,130952 | 0,045045 | 0,005327 | 0,014063 | 0,6 | 0,049147 | 0,078671 | 0,020579 |
| 2.3. Parça Yoğunluk | 0,044005 | 0,071212 | 0,068452 | 0,015735 | 0,75 | 0,078431 | 0,064272 | 0,108333 |
| 2.4. Parça Yoğunluk | 0,087798 | 0,077396 | 0,055556 | 0,030428 | 0,4 | 0,002546 | 0,077734 | 0,029133 |
| 3.1. Parça Yoğunluk | 0,114919 | 0,257143 | 0,03007 | 0,036791 | 0,217778 | 0,020696 | 0,079238 | 0,012097 |
| 3.2. Parça Yoğunluk | 0,144213 | 0,142857 | 0,066455 | 0,163889 | 0,166667 | 0,083719 | 0,111688 | 0,09519 |
| 3.3. Parça Yoğunluk | 0 | 0,2 | 0,055936 | 0,054676 | 0,066667 | 0,074545 | 0,056133 | 0,040806 |
| 3.4. Parça Yoğunluk | 0,130852 | 0,15625 | 0,045894 | 0,016279 | 0,133333 | 0,03285 | 0,074725 | 0,019268 |
| 4.1. Parça Yoğunluk | 0,099599 | 0,062802 | 0,058442 | 0,167832 | 0,053839 | 0,095238 | 0,041868 | 0,020086 |
| 4.2. Parça Yoğunluk | 0,076541 | 0,08587 | 0,053854 | 0,099622 | 0,003121 | 0,107143 | 0,083333 | 0,102241 |
| 4.3. Parça Yoğunluk | 0,108182 | 0,011544 | 0,088648 | 0,039256 | 0,002809 | 0,144444 | 0,011432 | 0,106169 |
| 4.4. Parça Yoğunluk | 0,020964 | 0,050649 | 0,014752 | 0,028564 | 0,030256 | 0,1 | 0,027188 | 0,051821 |

Uygulamada özellik çıkarma işlemi şu şekilde yapılmaktadır. Bu işlem kontrol menüsündeki 'Özellikleri_Çıkart' seçeneği ile yapılmaktadır. Giriş menüsündeki 'Dosya_Aç' seçeneği ile herhangi bir imza dosyası seçildikten sonra aktif olacaktır. Bundan sonra açılan dosyadaki imzaya ait özellikler çıkarılacaktır. A.T. kişisine ait bir imzanın çıkarılan 30 tane özelliği satırlar halinde Şekil 4.17'deki şekilde gösterilmektedir.



Şekil 4.17. Özellikler çıkarıldıktan sonra imza tanıma uygulamasının görüntüsü

Özellikler çıkarıldıktan sonra ağ eğitilmiştir. Bu işle uygulamanın 'Giriş' menüsündeki 'Eğitim' seçeneği ile yapılmıştır. Eğitim için yayılım değeri veya sinir sayısı metin kutucuklarından herhangi biri ya da her ikisi de girilmediği takdirde Şekil 4.18'deki gösterilen hata mesajı alınmaktadır.



Şekil 4.18. Yayılım değeri veya sinir sayısı girilmediğinde karşılaşılan hata mesajı

Eğitim işlemi radyal taban fonksiyonlu sinir ağı ile yapılmıştır.

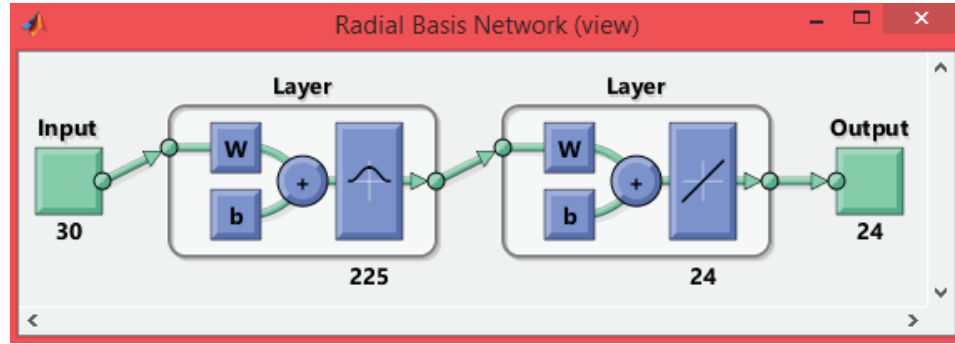
4.4. Radyal Tabanlı Fonksiyonlu Yapay Sinir Ağı

Radyal Taban Fonksiyonlu (RTF) Yapay Sinir Ağı güçlü, hızlı öğrenme ve kendi kendine organize olabilen sinir ağı olması ve özellikle lineer olmayan problemlerdeki, sınıflandırma ve öğrenme hızının daha yüksek olması nedeniyle tercih edilmiştir. Sinir ağlarının yapısal ve istatistiksel ve yapısal yaklaşımlara dayalı çeşitli sınıflandırıcıların uygulamasında öğrenme kabiliyetlerinin ve genelleme yeteneklerinin oldukça iyi olduğu gözlemlenmiştir. Genellikle bu tip sınıflandırma problemlerinde çok katmanlı, geriye yayımlı yapay sinir ağları tercih edilmektedir. Ancak bu tip ağların yakınsama hızının yavaş olması ve mutlak minimuma yakınsama yapıp yapamayacağının garantisi olmaması nedeniyle bu uygulamada Radyal Taban Fonksiyonlu Yapay Sinir Ağı kullanılmıştır.

Matlab'da `newrb(X,T,GOAL,SPREAD,MN,DF)` komutuyla bir radyal taban fonksiyonlu sinir ağı oluşturabiliriz.

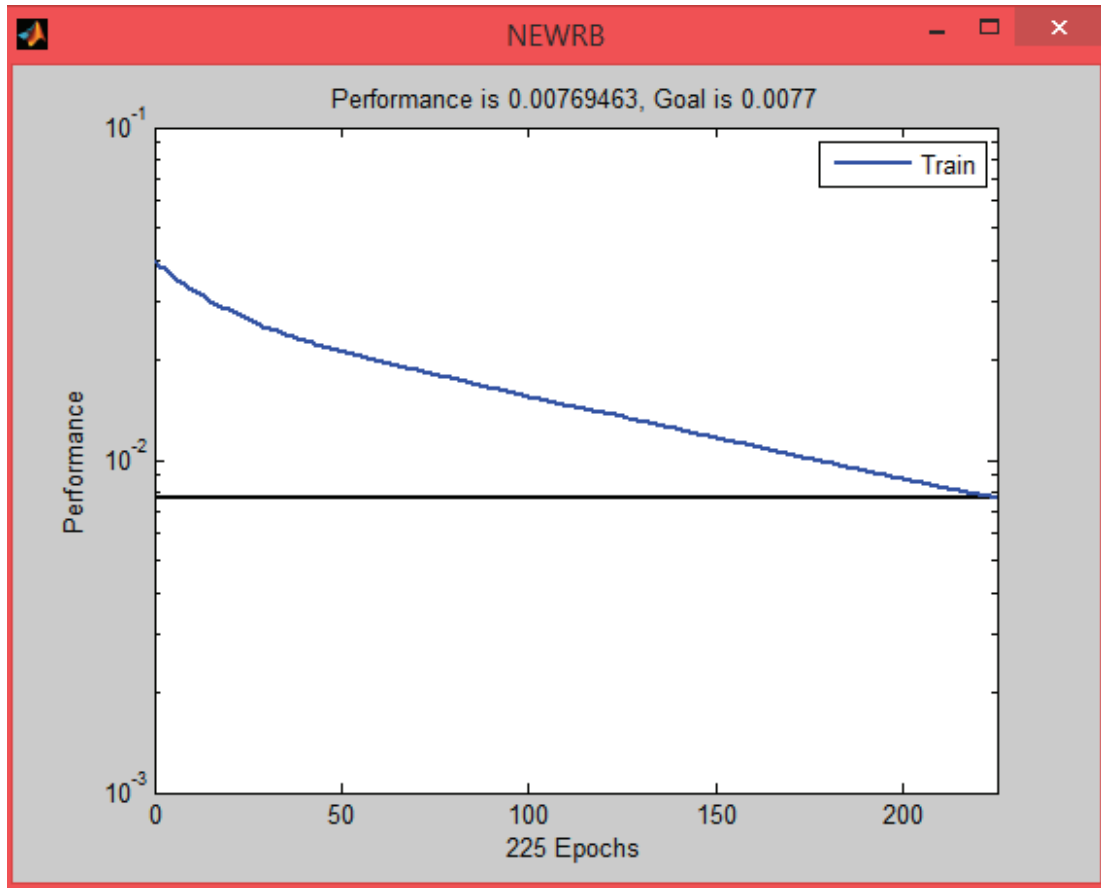
Burada sinir ağın girişi X, yani 30x624 boyutundadır. Yani 624 tane imzaya ait 30 tane özellik vardır. T yani hedef vektörü 24X624 boyutundadır. Bunun da anlamı her bir kişiye ait (26x24=624) 26 tane imza vardır. Toplamda ayırt edilmesi gereken kişi sayısı 24 tanedir. Bu sistemde her kişiye ait 30 tane özellik ile 24 kişi ayırt edilmiştir. Her kişinin imzasının sisteme tanıtılması için 26 tane imza kullanılmıştır. GOAL değeri yani, ağın ulaşması istenilen performans değeri 0,001 seçilmiştir. Spread yani ağın yayılım değeri ve MN yani eğin eğitiminde kullanılan maksimum sinir sayısı parametreleri değiştirilerek ağın performansı ölçülmüştür.

Aşağıdaki Şekil 4.19'da imza tanıma problemi için oluşturduğum maksimum sinir sayısı 225 seçilerek oluşturulmuş radyal taban fonksiyonlu sinir ağı gösterilmiştir.



Şekil 4.19. İmza tanıma için oluşturulan radyal taban fonksiyonlu sinir ağı yapısı

Aşağıdaki Şekil 4.20'de, Maksimum sinir sayısı 225 seçilerek eğitim sırasında performansın ağda kullanılan maksimum sinir sayısına(epoch) göre değişimi görülmektedir.



Şekil 4.20. Ağın eğitilmesi ile performans epochs grafiği

4.5. Analiz Çalışmaları

Ağ eğitildikten sonra genel olarak veritabanındaki tanıtılan bütün test verileri 'Kontrol' menüsündeki 'Test_Genel' seçeneği ile test edilerek performansı görülebilir. Veriler 'Test' seçeneği ile de teker teker test edilebilir.

Sistemin performansı eğitim yapılırken kullanılan yayılım ve maksimum sinir sayısı değerleri değiştirilerek test edilmiştir. Bu işlem uygulamanın Şekil 4.21'deki bölümünden yapılmaktadır. 'Eğitim İçin Yayılım Değerini Girin' karşısındaki metin kutusuna istenilen yayılım değeri girildikten ve 'Eğitim Maksimum Sinir Sayısını Girin' karşısındaki metin kutusuna maksimum sinir sayısı girildikten sonra 'Giriş' kısmındaki 'Eğitim' seçeneği ile ağ eğitilir. Ağın eğitiminde kullanılan değerler göre de imza tanıma sisteminin başarı performansı etkilenmektedir.

Grafikler

| | | |
|---|---|--|
| Siyah Beyaza Çevirmek İçin Eşik Değerini Girin: | <input style="width: 100%;" type="text"/> | |
| Eğitim İçin Yayılım Değerini Girin: | <input style="width: 100%;" type="text"/> | <input style="width: 100%;" type="button" value="Performans"/> |
| Eğitim Maksimum Sinir Sayısını Girin: | <input style="width: 100%;" type="text"/> | <input style="width: 100%;" type="button"/> |

Şekil 4.21. İmza tanıma uygulaması performans analiz bölümü

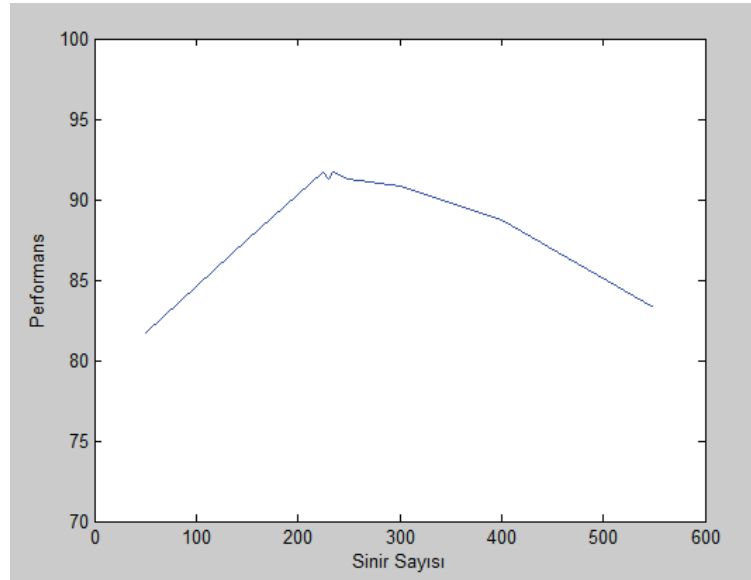
Aşağıdaki Tablo 4.2'de yayılım değeri 1 alındığında, maksimum sinir sayısı 50, 150, 225, 230, 235, 250, 300, 400, 550 alındığındaki bu değerleri gösteren tablo görülmektedir. Şekil 4.22'de ise bu verilere ait grafik gösterilmektedir.

Tablo 4.2. Yayılım değeri 1 alındığında maksimum sinir sayısı değiştirilerek elde edilen performans tablosu

| Yayılım Değeri | Maksimum Sinir Sayısı | Performans |
|----------------|-----------------------|------------|
| 1 | 50 | 81,6667 |
| 1 | 150 | 87,5 |
| 1 | 225 | 91,6667 |
| 1 | 230 | 91,25 |
| 1 | 235 | 91,6667 |
| 1 | 250 | 91,25 |
| 1 | 300 | 90,8333 |
| 1 | 400 | 88,75 |
| 1 | 550 | 83,3333 |

Tablodaki veriler kullanılarak matlab'da grafikler oluşturulmuştur. Yayılım değeri sabit tutularak maksimum sinir sayısının değişimine göre oluşturulan grafikler uygulamanın 'Kontrol' bölümündeki 'Eğitim_Performansı_Sinir Sayısına_Göre seçeneğinden yapılmıştır.

Grafiklerde performans değeri %70'den başlatılmıştır. Bunun nedeni ağın performansının bu değerden aşağıya düşmemesi ve performans dâhilindeki değerleri göstermektir.

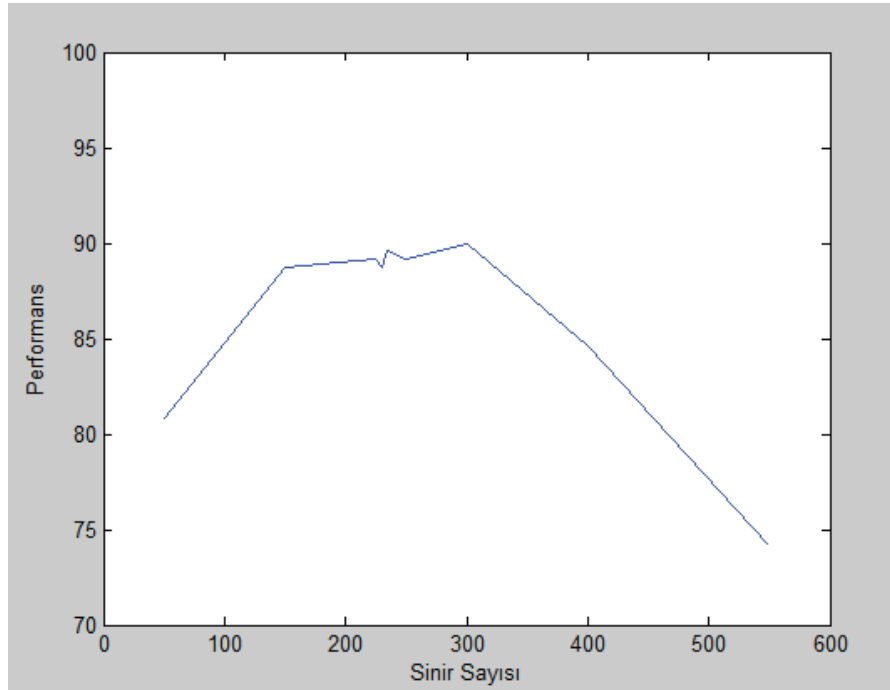


Şekil 4.22. Yayılım değeri 1 alındığındaki değişik maksimum sinir sayılarında performans grafiği

Aşağıdaki Tablo 4.3'te yayılım değeri 5 alındığında, maksimum sinir sayısı 50, 150, 225, 230, 235, 250, 300, 400, 550 alındığındaki bu değerleri gösteren tablo görülmektedir. Şekil 4.23'te ise bu verilere ait grafik gösterilmektedir.

Tablo 4.3. Yayılım değeri 5 alındığında maksimum sinir sayısı değiştirilerek elde edilen performans tablosu

| Yayılım Değeri | Maksimum Sinir Sayısı | Performans |
|----------------|-----------------------|------------|
| 5 | 50 | 80,8333 |
| 5 | 150 | 88,75 |
| 5 | 225 | 89,1667 |
| 5 | 230 | 88,75 |
| 5 | 235 | 89,5833 |
| 5 | 250 | 89,1667 |
| 5 | 300 | 90 |
| 5 | 400 | 84,5833 |
| 5 | 550 | 74,1667 |

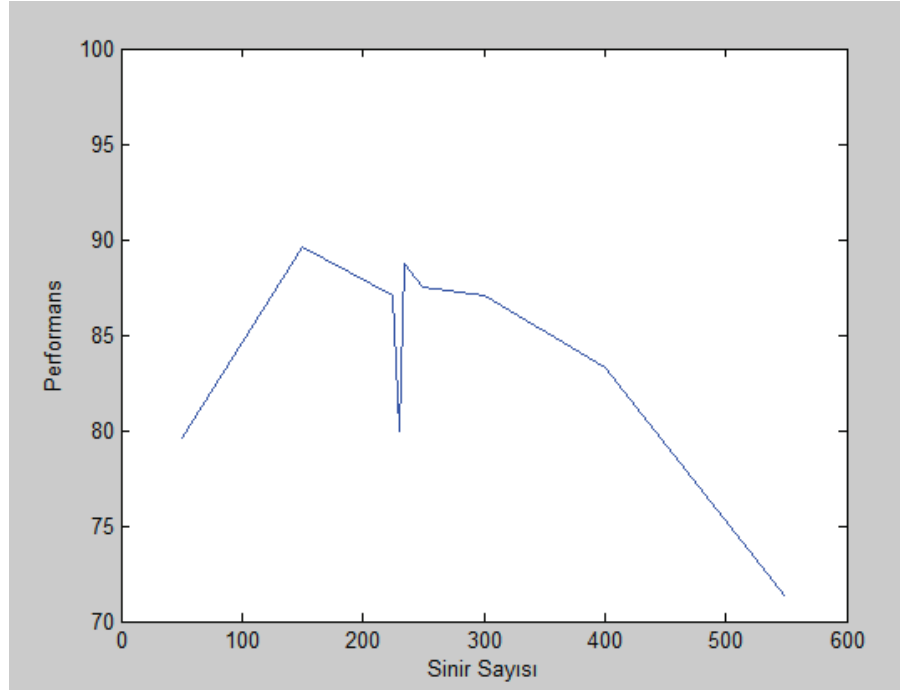


Şekil 4.23. Yayılım değeri 5 alındığındaki değişik maksimum sinir sayılarında performans grafiği

Aşağıdaki Tablo 4.4'te yayılım değeri 10 alındığında, maksimum sinir sayısı 50,150, 225, 230, 235, 250, 300, 400, 550 alındığındaki bu değerleri gösteren tablo görülmektedir. Şekil 4.24'te ise bu verilere ait grafik gösterilmektedir.

Tablo 4.4. Yayılım değeri 10 alındığında maksimum sinir sayısı değiştirilerek elde edilen performans tablosu

| Yayılım Değeri | Maksimum Sinir Sayısı | Performans |
|----------------|-----------------------|------------|
| 10 | 50 | 79,5833 |
| 10 | 150 | 89,5833 |
| 10 | 225 | 87,08333 |
| 10 | 230 | 80 |
| 10 | 235 | 88,75 |
| 10 | 250 | 87,5 |
| 10 | 300 | 87,08333 |
| 10 | 400 | 83,3333 |
| 10 | 550 | 71,25 |

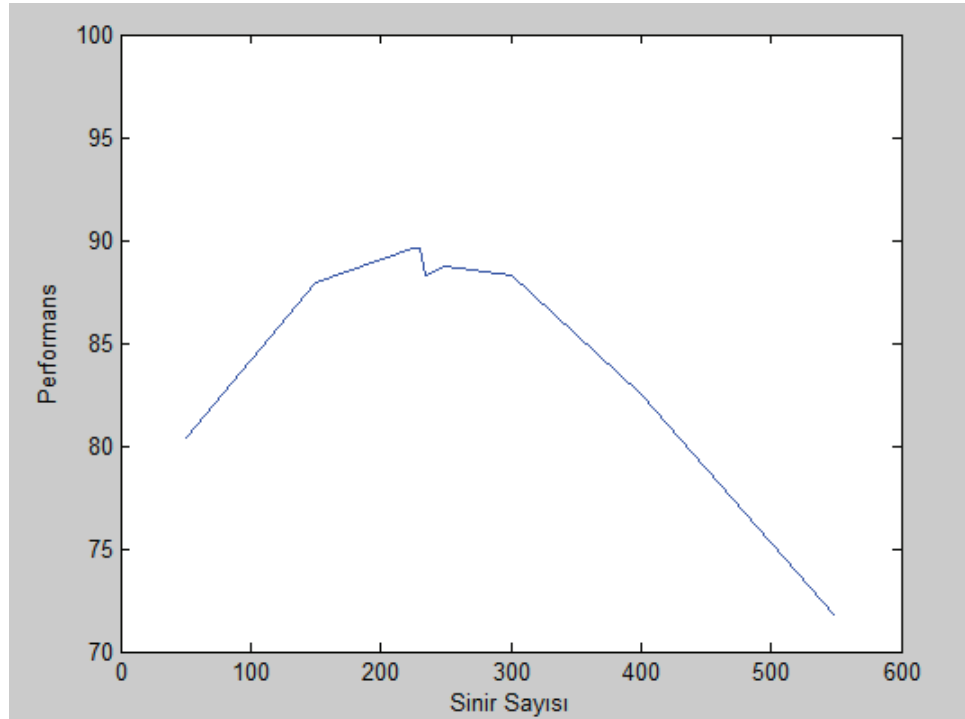


Şekil 4.24. Yayılım değeri 10 alındığındaki değişik maksimum sinir sayılarında performans grafiği

Aşağıdaki Tablo 4.5'te yayılım değeri 15 alındığında, maksimum sinir sayısı 50, 150, 225, 230, 235, 250, 300, 400, 550 alındığındaki bu değerleri gösteren tablo görülmektedir. Şekil 4.25'te ise bu verilere ait grafik gösterilmektedir.

Tablo 4.5. Yayılım değeri 15 alındığında maksimum sinir sayısı değiştirilerek elde edilen performans tablosu

| Yayılım Değeri | Maksimum Sinir Sayısı | Performans |
|----------------|-----------------------|------------|
| 15 | 50 | 80,4167 |
| 15 | 150 | 87,9167 |
| 15 | 225 | 89,5833 |
| 15 | 230 | 89,5833 |
| 15 | 235 | 88,3333 |
| 15 | 250 | 88,75 |
| 15 | 300 | 88,33333 |
| 15 | 400 | 82,5 |
| 15 | 550 | 71,6667 |

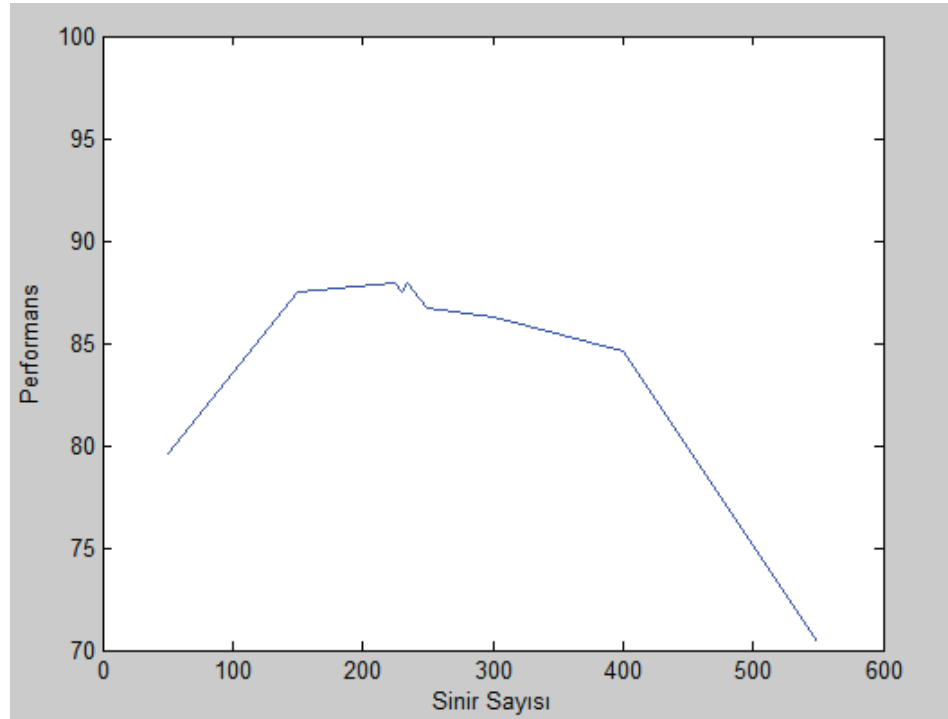


Şekil 4.25. Yayılm değeri 15 alındığındaki değişik maksimum sinir sayılarında performans grafiği

Aşağıdaki Tablo 4.6'da yayılım değeri 20 alındığında, maksimum sinir sayısı 50, 150, 225, 230, 235, 250, 300, 400, 550 alındığındaki bu değerleri gösteren tablo görülmektedir. Şekil 4.26'da ise bu verilere ait grafik gösterilmektedir.

Tablo 4.6. Yayılm değeri 20 alındığında maksimum sinir sayısı değiştirilerek elde edilen performans tablosu

| Yayılm Değeri | Maksimum Sinir Sayısı | Performans |
|---------------|-----------------------|------------|
| 20 | 50 | 79,5833 |
| 20 | 150 | 87,5 |
| 20 | 225 | 87,9167 |
| 20 | 230 | 87,5 |
| 20 | 235 | 87,9167 |
| 20 | 250 | 86,6667 |
| 20 | 300 | 86,25 |
| 20 | 400 | 84,5833 |
| 20 | 550 | 70,4167 |

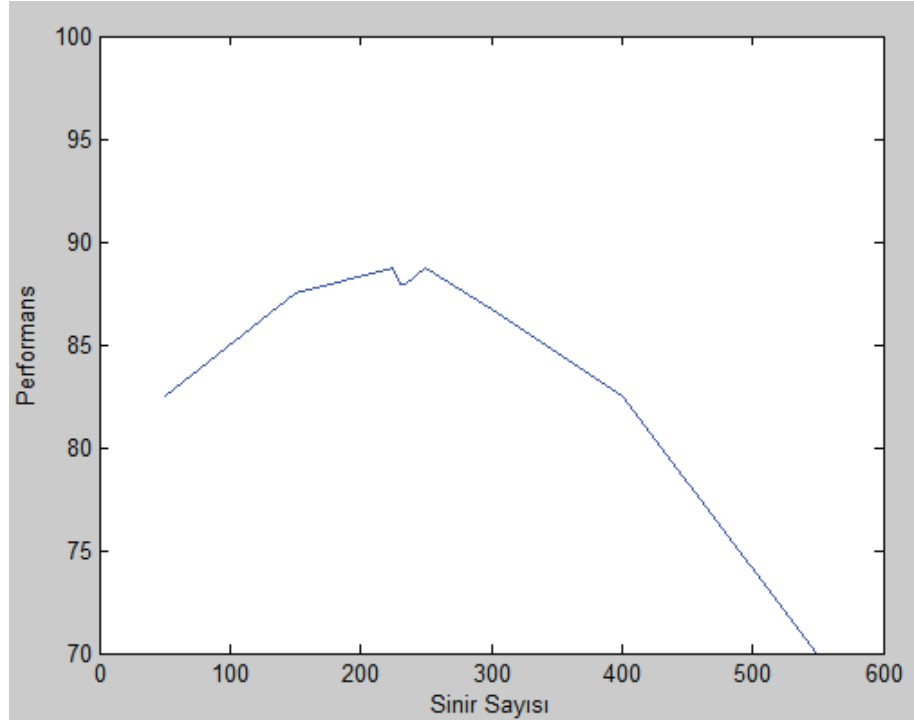


Şekil 4.26. Yayılım değeri 20 alındığındaki değişik maksimum sinir sayılarında performans grafiği

Aşağıdaki Tablo 4.7'de yayılım değeri 25 alındığında, maksimum sinir sayısı 50, 150, 225, 230, 235, 250, 300, 400, 550 alındığındaki bu değerleri gösteren tablo görülmektedir. Şekil 4.27'de ise bu verilere ait grafik gösterilmektedir.

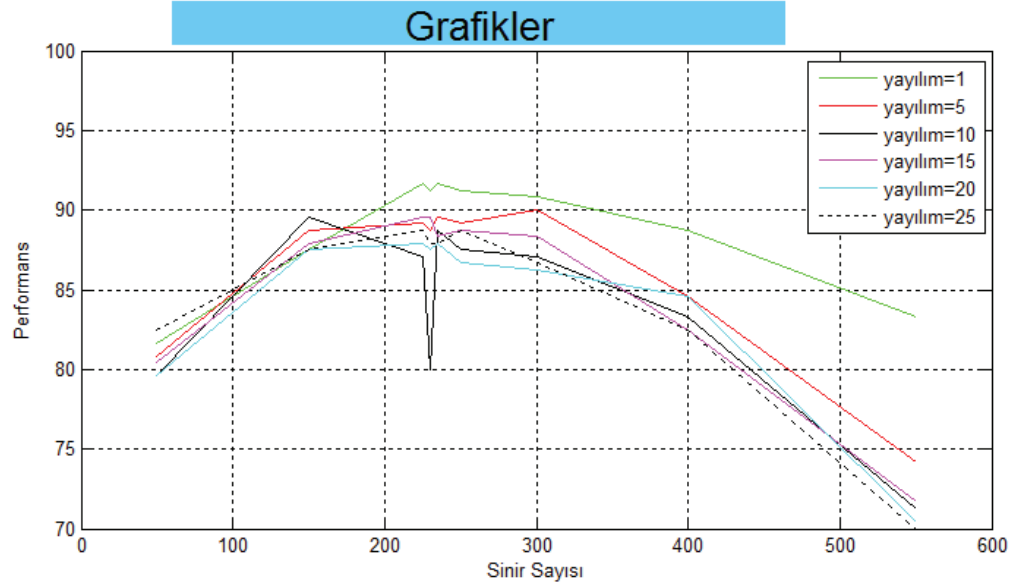
Tablo 4.7. Yayılım değeri 25 alındığında maksimum sinir sayısı değiştirilerek elde edilen performans tablosu

| Yayılım Değeri | Maksimum Sinir Sayısı | Performans |
|----------------|-----------------------|------------|
| 25 | 50 | 82,5 |
| 25 | 150 | 87,5 |
| 25 | 225 | 88,75 |
| 25 | 230 | 87,9167 |
| 25 | 235 | 87,9167 |
| 25 | 250 | 88,75 |
| 25 | 300 | 86,6667 |
| 25 | 400 | 82,5 |
| 25 | 550 | 70 |



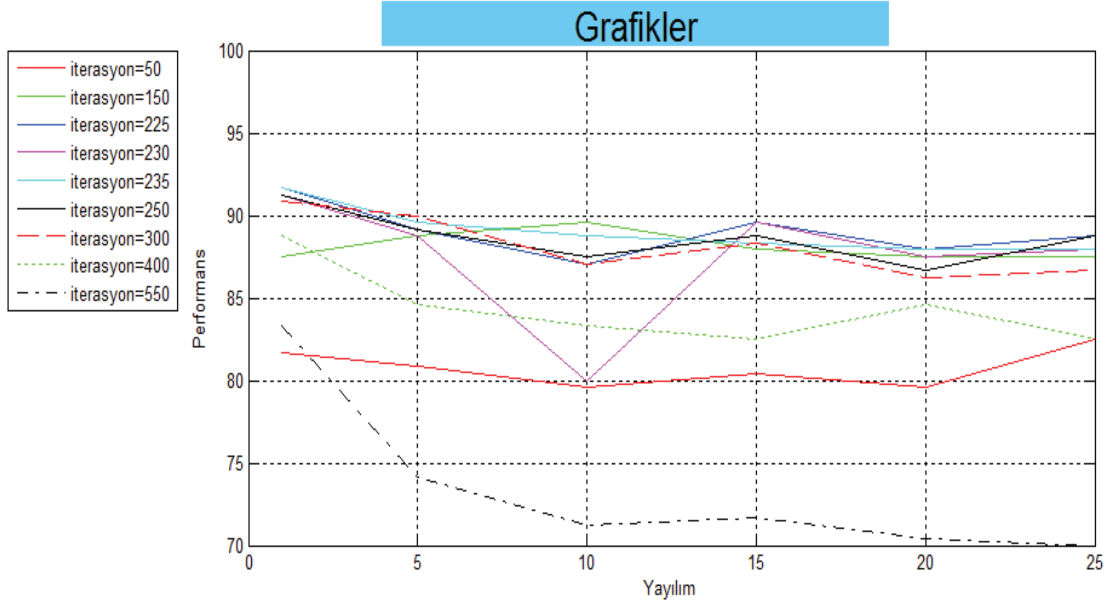
Şekil 4.27. Yayılım değeri 25 alındığındaki değişik maksimum sinir sayılarında performans grafiği

İmza tanıma uygulaması maksimum sinir sayısı ve yayılım değerine göre, ayrı ayrı belirlenen değerlerde toplu olarak grafik çizdirmektedir. Yayılım değeri uygulama boyunca 1, 5, 10, 15, 20, 25 olarak değiştirilmiştir. Yayılım değeri 1 olduğunda maksimum sinir sayısının değerlerine göre (50, 150, 225, 230, 235, 250, 300, 400, 550) performans grafiği ve aynı şekilde yayılım değeri 5, 10, 15, 20, 25 olduğundaki maksimum sinir sayısına göre performans grafiği çizdirilebilir. Bu işlem uygulamanın 'Kontrol' menüsünde 'Eğitim_Performansı_Sinir_Sayısına_Göre' bölümünden yapılabilir. Elde edilen grafik Şekil 4.28'de gösterilmiştir.



Şekil 4.28. Yayılım değeri 1, 5,10, 15,20, 25 olduğunda her bir değeri için maksimum sinir sayısı (50, 150, 225, 230, 235, 250, 300, 400, 550) olduğundaki performans grafiği

Grafiği bir başka şekilde de çizdirebiliriz. Bu sefer y eksenindeki maksimum sinir sayısı yerine yayılım değeri 1, 5, 10, 15, 20, 25 olduğunda her bir maksimum sinir sayısına göre (50, 150, 225, 230, 235, 250, 300, 400, 550) performans grafiği çizdirilebilir. Bu işlem uygulamanın 'Kontrol' menüsünde 'Eğitim_Performansı_Yayılım_Değerine_Göre' bölümünden yapılabilir. Elde edilen grafik Şekil 4.29'da gösterilmiştir.



Şekil 4.29. Maksimum sinir sayısı 50, 150, 225, 230, 235, 250, 300, 400, 550 olduğunda her bir değeri için yayılım değeri (1, 5, 10, 15, 20, 25) performans grafiği

BÖLÜM 5. SONUÇLAR VE ÖNERİLER

Sistem 864 adet imza kullanılmıştır. Bu imzalardan 624 adeti eğitimde 240 adeti test işlemlerinde kullanılmıştır. Toplanan imzalar sayısallaştırılarak veri tabanına kaydedilmiş ve imza tanıma işlemi gerçekleştirilmiştir. Doğrulama işlemi gerçekleştirilmemektedir.

İmza tanıma sistemi oldukça zor bir problemdir. Çünkü aynı kişiye ait imzalar bile kişinin ruhsal durumu, kullandığı kalem gibi çeşitli faktörlerden dolayı farklılıklar gösterebilir. Kişinin kendi imzasının bile zaman zaman birbirine benzemediği durumların olduğu düşünülürse imza tanıma probleminin zorluğu anlaşılacaktır.

Bu uygulamada imza tanıma sistemi tasarımının kodlama ve arayüz bölümlerinin her ikisi de matlab ortamında gerçekleştirilmiştir. Birçok hazır fonksiyondan yararlanılmıştır. Bunun için yorumlar matlab ortamındaki parametreler göz önüne alınarak yapılmıştır.

İmza tanıma sistemlerinde imzalar için veritabanı oluşturulur. Oluşturulan imzalar çeşitli ön işlemlerden geçirilerek özellik vektörleri çıkarılacak hale gelirler. İmza tanıma ya da doğrulama sistemlerinde en önemli aşama özellik çıkarma aşamasıdır. Çünkü düzgün ve yerinde çıkarılan özellikler imzaların tanınmasını doğrulanmasını kolaylaştıracaktır. Fakat her bulunan özellik de sistemin performansını arttırmayabilir, aksine sadece sistemin çalışma hızını düşürebilir. Bu yüzden özellikler çıkartılırken imzanın tanınması ve doğrulanmasını kolaylaştıracak özellikler çıkarılmalıdır. Özelliklerin çok değil, yeterli olması önemlidir.

Çıkarılan imzalar eğitilmek üzere radyal taban fonksiyonlu sinir ağına verilmiştir. Bu ağın tercih edilmesinin nedeni istenilen kolay ve kısa zamanda yakınsayabilmesi ve

genelleme yeteneğinin fazla olmasıdır. Matlab'da Radyal Tabanlı Yapay Sinir Ağı tasarlarken sistemin performansını etkileyen iki tane önemli özellik vardır. Bunlardan bir tanesi yayılım ve diğeri ağın maksimum sinir sayısıdır. Bunlar değiştirilerek sistemin performansına bakılmıştır. Sistemin en yüksek performansı % 91,6667 yayılım değeri bir alındığında ve maksimum sinir sayısı 225 olduğunda ulaşılmıştır. Aynı yayılım değeri ile maksimum sinir sayısı 230 olduğunda sistemin performansı biraz düşerek % 91,25 olmuştur. Sistemin maksimum sinir sayısı 235 olduğunda tekrar performans maksimum noktaya erişmiştir. En düşük performansa ise yayılım değeri 25 ve maksimum sinir sayısı 550 olduğunda ulaşılmıştır. Genele bakıldığında en düşük değerler yayılımdan bağımsız olarak maksimum sinir sayısı çok az ya da çok fazla seçildiğinde olmuştur. En iyi performans değerlerine ise 225, 235 gibi ara değerler seçildiğinde ulaşılmıştır. Performansın en yüksek olduğu değerlerde yayılım değerine bakıldığında ise 1 vermek yeterli olmuştur.

Buradan da anlaşılacağı gibi yayılım değeri ve maksimum sinir sayısının seçimi oldukça önemlidir. Doğru yayılım değeri ve doğru maksimum sinir sayısı ile sistem en iyi performansa ulaştırılabilir.

Sistemin genel performansına bakıldığında Chadha ve arkadaşları, özellik çıkarmak için DCT dönüşümünü ve sınıflandırma için radyal fonksiyonlu yapay sinir ağı kullandıkları çevrimdışı imza tanıma sistemlerinde, tasarladıkları 200 örnek için yaklaşık %80 tanıma sağladıkları çalışmalarından ve global özelliklerle çevrimdışı imza tanıma sistem gerçekleştiren Patil ve arkadaşlarının tasarladığı %89 tanıma gerçekleştiren sistemlerine göre daha iyi bir başarı elde etmiştir [26,28].

Tasarlanan imza tanıma uygulamasında sahte imzaların tespitine yönelik bir çalışma yapılmamıştır. Uygulamada sadece tanıma işlemi yapılmıştır. Bu tez çalışması sonrasında sahte imzaların anlaşılmasına yönelik olarak imza doğrulama sistemi geliştirilebilir. Sistemin hızını çok fazla düşürmeye sebep olmayan etkili özellikler eklenerek sistemin performansı artırılabilir.

KAYNAKLAR

- [1] Khan, S., Dhole, A., Review on offline signature recognition and verification techniques. *International Journal of Advanced Research Computer and Communication Engineering*, 3(6), pp. 6879-6882, 2014.
- [2] Çikoğlu, S., Yapay Sinir Ağlarıyla İmza Tanıma, Yüksek Lisans Tezi, Sakarya Üniversitesi, Fen Bilimleri Enstitüsü, 2003.
- [3] Pham, DT., Aksoy, MS, A new algorithm for inductive learning, *Journal of Systems Engineering*, 5, pp. 115-122, 1995.
- [4] Kaymaz, S., Çevrimdışı İmza Tanıma, Yüksek Lisans Tezi, Marmara Üniversitesi, Fen Bilimleri Enstitüsü, 2010.
- [5] Ramesh VE., Murty, MN., Off-line signature verification using genetically optimized weighted features. *Pattern Recognition*, 32, pp. 217-233, 1999.
- [6] Huang, K., Yan, H., Off-line signature verification based on geometric feature extraction and neural network classification. *Pattern Recognition*, Vol. 30(1), pp. 9-17, 1999.
- [7] Basavaraj, L, Sudhaker Samuel, RD., Off-line signature verification and recognition. An Approach Based on Four Speed Stroke Angle. *International Journal of Recent Trends in Engineering*, Vol 2(3), November 2009.
- [8] Baltzakis, H., Papamarkos, N., A new signature verification technique based on a two-stage neural network classifier. *Engineering Applications of Artificial Intelligence*, 14, 95-103, 2001.
- [9] Shashi Kumar, Dr., Raja RB., Chhotaray, RK., Pattanaik, S., Off-line signature verification based on fusion of grid and global features using neural networks. *International Journal of Engineering Science and Technology*, Vol. 2(12), 7035-7044, 2010.
- [10] Prashanth, CR., Raja KB., Off-line signature verification based on angular features. *International Journal of Modeling and Optimization*, Vol. 2(4), August 2012.

- [11] Mohammed, AA., Yousif, NA., Offline signature recognition and verification based on artificial neural network. Eng & Tech. Journal, Vol.27, No.7, 2009.
- [12] Vargas, JF., Ferrer, MA., Travieso, CM., Alanso, JB., Offline signature based on pseudo-cepstral coefficients. IEEE 10th International Conference on Document Analysis and Recognition, 2009.
- [13] Fasquel, JB., Bruynooghe, M., A hybrid opto-electronic method for real-time automatic verification of handwritten signatures. Digital Image Computing Techniques and Applications, Melbourne, Australia, 21-22 January 2002.
- [14] Julio, MR., Rogelio, AS., On-line signature verification based on optimal feature representation and neural network-driven fuzzy reasoning. IEEE Members, 2003.
- [15] Nguyen, V., Blumenstein M., Leedham, G., Global features for the off-line signature verification problem, 10th International Conference on Document Analysis and Recognition, 2009.
- [16] Pansare, A., Bhatia, S., Handwritten signature verification using neural network, International Journal of Applied Information Systems (IJ AIS) – ISSN : 2249-0868 Foundation of Computer Science FCS, New York, USA Volume 1–No.2, January 2012.
- [17] Han, K., Sethi, I.K., Handwritten signature retrieval and identification. Pattern Recognition Letters, 17, pp. 83-90, 1996.
- [18] Yedekcoğlu, OA., Off-line signature verification with thickened templates. COMCON 5-Proceedings of 5th International Conference on Advances in Communication and Control, Greece, pp. 131-142, 1995.
- [19] Bajaj, R., Chaudhury, S. Signature verification using multiply neural classifiers. Pattern Recognition, 30, pp. 1-7, 1997.
- [20] Mercimek, M., Gülez, K., Mumcu, TV., Real object recognition using moment invariants. Sadhana, 30(6), pp.765-775, 2005.
- [21] Yılmaz, MB, Yanikoğlu, B., Offline Signature Verification Using Classifier Combination of HOG and LBP Features. International Joint Conference on Biometrics, Washington, DC, Oct. 11-13, 2011, pp. 1-7.

- [22] Mirzaei,O., Irani, H., Puurreza, HR., Offline signature recognition using modular neural networks with fuzzy response integration. International Conference on Network and Electronics Engineering IPCSIT, 11, pp. 53-59, 2001.
- [23] Pérez-Hernández,A.,Sánchez,A.,Vélez,J.F., Simplified stroke-based approach for off-line signature recognition. Departamento de Informática, Estadística Telemática Universidad Rey Juan Carlos/Campus de Móstoles, Madrid, Spain.
- [24] Şenol, C.,Yıldırım,T., Standart ve hibrid yapılar kullanarak yapay sinir ağları ile imza tanıma. ELECO'2004 Elektrik-Elektronik ve Bilgisayar Mühendisliği Sempozyumu, 8-12 Aralık 2004.
- [25] Öz, C., Signature recognition and verification with artificial neural network using moment invariant method. Conference on Advances in Neural Networks, 2005.
- [26] Chadha, A., Satam, N., Wali, V., Biometric signature processing & recognition using radial basis function network. CIIT International Journal of Digital Image Processing, September 2013.
- [27] Azzopardi, G., Camilleri, KP., Offline handwritten signature verification using radial basis function neural networks. Methodology, pp.1–6, 2006.
- [28] Patil, Ms. P., Patil, Ms. A., Offline signature recognition using global features. International Journal of Emerging Technology and Advanced Engineering, ISSN 2250-2259, 3(1), January 2013.
- [29] Yingyong, Q., Bobby, RH., Signature verification using global and grid features, Pattern Recognition, 27(12), pp. 1621-1629, December 1994.
- [30] Mohammadzade, M., Ghonofi, A., Persian off-line signature recognition with structural and rotation invariant features using by one-against- all SVM classifier. Journal of Advances in Computer Research, 4(2), pp. 87-96, May 2013.
- [31] Sthapak, S., Khopade, M., Kashid, C., Artificial neural network based signature recognition & verification. International Journal of Emerging Technology and Advanced Engineering, 3(8), August 2013.

- [32] Kaewkongka, T., Chamnongthai, K., Thipakom, B., Off-line signature recognition using parameterized Hough transform. Proceedings of Fifth International Symposium on Signal Proceedings and its Applications, ISSPA '99, Brisbane, Australia, 22-25 August, 1999 Organized by the Signal Processing Research Centre OUT, Brisbane, Australia.
- [33] Urhan, O., Öztürk, S., İmge işleme, Ders Notları, Kocaeli Üniversitesi Elektronik ve Haberleşme Mühendisliği Bölümü, 2004.
- [34] Yıldırım, KS., Görüntü işleme, Ege Üniversitesi, Bilgisayar Mühendisliği Bölümü, 2003.
- [35] Daugman, JG., High confidence visual recognition of persons by a test of statistical independence, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 15, No. 11, pp. 1148-1161, 1993.
- [36] Shapiro, LG., Stockman, GC., Computer Vision, Paperback, 2001.
- [37] Karakoç, M., Görüntü İşleme Teknolojiler ve Uygulamaları, Ders Notları, Ege Üniversitesi, Bilgisayar Mühendisliği Bölümü, 2012.
- [38] Köker, A., Sayısal Görüntü İşleme, [Çevrimiçi]. <http://alikoer.name.tr/sayisal+goruntu+isleme>, 15 Kasım 2013.
- [39] Demir, S., Sayısal Görüntü Temelleri, Ders Notları, Hacettepe Üniversitesi, Fen Bilimleri Enstitüsü, 2010.
- [40] Yalım, B., Araç Plaka Tanıma Sistemi, Yüksek Lisans Tezi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, 2008.
- [41] Önal O., Development and use of digital image analysis techniques for analyzing sectional characteristics of some geomaterials, Doktora Tezi, Dokuz Eylül Üniversitesi, Fen Bilimleri Enstitüsü, 2008.
- [42] Cho, S., Haralick, R., YI, S., Improvement of Kittler and Illingworth's Minimum Error Thresholding, Pattern Recognition, Vol. 22, pp. 609-617, 1989.
- [43] Otsu, N., A threshold selection method from gray-level histograms, IEEE Transactions on Systems, Man, and Cybernetics, 9(1), pp. 62-66, 1979.
- [44] Çamaşırcıoğlu, E., Araç Plakası Algılama Ve Tanıma, Yüksek Lisans Tezi, Atılım Üniversitesi, Fen Bilimleri Enstitüsü, 2007.
- [45] Ozbay, S., Ercelebi, E., Automatic Vehicle Identification By Plate Recognition, Yüksek Lisans Tezi, Gaziantep Üniversitesi, 2005.

- [46] http://www.acikders.org.tr/pluginfile.php/638/mod_resource/content/0/Ders_Notlari/Unite7_Goruntu_Analizi.pdf, Erişim Tarihi;10.02.2014.
- [47] Zhang, D., Guojun, L., Review of shape representation and description techniques. Gippsland School of Computing and Info. Tech., Monash University, 37, pp. 1-19, 2004.
- [48] Köker, R., Lecturer, SC., Reliability signature recognition system using inductive learning. Automatic Control and Control Sciences, 38(5), pp. 35-44, 2004.
- [49] Abu-Rezq, AN., Tolba, A., Cooperative self-organizing maps for consistency checking and signature verification .Digital Signal Processing, 9, pp. 107-119, 1999.
- [50] Awcock, GJ., Thamos, R., Applied Image Processing. McGraw Hill, Inc., 1996.
- [51] Hu, M., Visual pattern recognition by moment invariants. IRE Trans. Inform. Theory, Vol. IT-8,pp. 179-187, 1962.
- [52] Reiss, TH., The revised fundamental theorem of moment invariants. IEEE Trans. On Pattern Analysis And Machine, 13(8), pp. 149-153, 2000.
- [53] Geric, G., Shape Analysis Moment Invariants, Ders Notları, Utah University, Bioenerji Bölümü, 2010.
- [54] Daramola, A., Badejo, B., Samuel I., Sokunbi T., Vertical off-line signature feature block for verification. Recent Advances in Circuits, Systems, Signal Processing and Communications, ISBN: 978-960-474-359-9.
- [55] Saikia H., Sarma KC., Approaches and issues in offline signature verification system, International Journal of Computer Applications (0975 – 8887) Volume 42(16), 2012.
- [56] Hassani, H., A brief introduction to singular spectrum analysis: Methodology and comparison. J. Data Sci, 5,pp. 239–257, 2007.
- [57] Demiriz, A., İstatiksel Sınıflandırma, Ders Notları, Sakarya Üniversitesi, Endüstri Mühendisliği Bölümü 2006.
- [58] Haşimoğlu, AS., Yapay Sinir Ağlarıyla İmza Tanıma, Doktora Tezi, Marmara Üniversitesi, Fen Bilimleri Enstitüsü, 1998.

- [59] Mcculloch, SW., Pitts, W., A logical calculus of the ideas immanent in nervous activity, *Bulletin Of Mathematical Biophysics.*, 30, pp. 115-133, 1943.
- [60] Hebb, DO., *The organization of behavior, A neuropsychological theory*, pp. 60-78, 1949.
- [61] Rosenblatt, F., *The perceptron a perceiving and recognizing automaton. Project Para*, Cornell Aeronautical Laboratory, 1957.
- [62] Minsk, M., Papert, SA., *An Introduction To Computational Geometry*, Paperback, 1987.
- [63] <http://www.yapay-zeka.org/modules/wiwimod/index.php?page=ANN>, Erişim Tarihi; 10.03.2015.
- [64] Temurtaş F., *Sinirsel Bulanık Sistemler Ve Uygulamaları, Ders Notları*, Sakarya Üniversitesi, Fen Bilimleri Enstitüsü, 2007.
- [65] http://www.mathworks.com/videos/using-neural-network-with-matlab-82469.html?form_seq=conf1218&confirmation_page&wfsid=6031560&refresh=true, Erişim Tarihi; 23.05.2014.
- [66] Devaraj, D., Yegnanarayana, B., Ramar, K., Radial basis function networks for fast contingency ranking. *Electric. Power Energy Syst*, 24, pp. 387–395, 2002.
- [67] Fu, X., Wang, L. Data dimensionality reduction with application to simplifying RBF network structure and improving classification performance. *IEEE Trans. Syst. Man Cybern, Part B* ,33, pp.399–409, 2003.
- [68] Han, M., XI, J., Efficient clustering of radial basis perceptron neural network for pattern recognition. *Pattern Recognition*, 37, pp. 2059–2067, 2004.
- [69] Du, JX., Zhai, CM., A hybrid learning algorithm combined with generalized GRLS approach for radial basis function neural networks. *Appl. Math. Comput.*, 208, pp. 908–915, 2008.
- [70] Oyang, YJ., Hwang, SC., Ou, YY., Chen, CY., Chen, ZW., Data classification with radial basis function networks based on a novel kernel density estimation algorithm. *IEEE Trans. Neural Netw.*, 16, 225–236, 2005.
- [71] Mahanty, RN., Gupta, PBD., Application of RBF neural network to fault classification and location in transmission lines, *IEE Proceedings of Gener. Transm. Distrib.*, 151, 2, 201-212, 2004.

- [72] Ding, S., Xu, L., Su, C., Zhu, H., Using genetic algorithms to optimize artificial neural networks, *Journal of Convergence Information Technology*, Vol. 5, No. 8, pp. 54- 62, 2010.
- [73] Wang, H., Xu, X., Applying RBF neural networks and genetic algorithms to nonlinear system optimization. In *Proceeding of the 2nd International Conference on Materials and Products Manufacturing Technology*, pp. 2457-2460, 2012.
- [74] Er, M.J., Face recognition with radial basis function (RBF) neural networks. *IEEE Trans Neural Network*, 13(3),pp. 697-710, 2002.
- [75] Moody, J., Darken, C.J., Fast learning in network of locally-tuned processing units, *Neural Comput.*, 1, pp. 281–294, 1989.
- [76] Sharma, M., Khann, K., Offline signature verification using supervised and unsupervised neural networks. *International Journal Computer Science and Mobile Computing*, 3(7), pp. 425-436, July 2014.
- [77] Liu, Y., Zheng, Q., Shi, Z., Chen, J., Training radial basis function networks with particle swarms. *Lect. Note. Comput. Sci.*,3173, pp. 317–322, 2004.
- [78] Simon, D., Training radial basis neural networks with the extended Kalman filter. *Neurocomputing*, 48, pp. 455–475, 2002.
- [79] Kurban, T., Beşdok, E., A comparison of RBF neural network training algorithms for inertial sensor based terrain classification. *Sensors*, 9, pp. 6312-6329,2009.
- [80] Fei, Z., Luo, D., He, Z., LI, B., Atmospheric environmental quality assessment RBF model based on the matlab. *Journal of Environmental Protection*, 3, pp. 689-693, 2012.
- [81] Liu, J., Qiu, L., Function approximation bases on radial basis neural network, *Popular Science*, No. 9, pp. 39-40, 2009.
- [82] Lu, S., Spread constant's effect on radial basis function neural network, *Silicon valley*, No. 18, pp. 100-102, 2011.
- [83] Su, Z., Lin, M., Application of RBF neural network on height fitting. *City Survey*, No. 3, pp. 65-67, 2006.
- [84] Li, J., Ma, C., Sun, G., Dynamic prediction of port container throughput based on RBF neural network. *Journal of Dalian Jiaotong University*, Vol. 29, No. 4, pp. 27-32, 2008.

EKLER

EK A: Kodlar

```
function tez_OpeningFcn(hObject, eventdata, handles, varargin)
```

```
handles.output = hObject;
```

```
guidata(hObject, handles);
```

```
set(handles.gricevir,'enable','off');  
set(handles.gricevir,'enable','off');  
set(handles.otsusiyahbeyaz,'enable','off');  
set(handles.ellesiyahbeyaz,'enable','off');  
set(handles.iskeletinicikar,'enable','off');  
set(handles.ozelcikart,'enable','off');  
set(handles.testg,'enable','off');  
set(handles.test,'enable','off');  
set(handles.axes2,'visible','off');  
set(handles.text7,'visible','off');  
set(handles.text1,'visible','off');  
set(handles.edit1,'visible','off');  
set(handles.uitable1,'visible','off');
```

```
% -----  
function ozelcikart_Callback(hObject, eventdata, handles)
```

```
load giris  
k=0;  
I3=rgb2gray(giris);%resimi griye ?eviriyor  
esik=graythresh(I3);  
siyahbeyaz=im2bw(I3,esik);  
I4=siyahbeyaz;  
I4=~I4;  
I5 = bwmorph(~I4, 'thicken',inf);  
I5=~I5;  
b=sum(I5);  
t=sum(b);  
satir=0;  
sutun=0;  
sutuna=0;  
sutunb=0;  
sutunc=0;  
sutund=0;  
satira=0;  
satirb=0;  
satirc=0;  
satird=0;  
[m n]=size(I5);  
for i=1:m  
for j=1:n
```

```
    [y x]=find(I5==1);  
    x1=min(x);  
    x2=max(x);  
    y1=min(y);  
    y2=max(y);
```

```
if I5(i,j)==1
```

```
    sutun=sutun+j*I5(i,j);  
    satir=satir+i*I5(i,j);  
end  
end  
end
```

```
agmerx=sutun/t;  
agmery=satir/t;  
genislik=x2-x1;  
yukseklilik=y2-y1;  
oran=genislik/yukseklilik;
```

```

boyut_merkezi_x=(x1+x2)/2;
boyut_merkezi_y=(y1+y2)/2;
yogunluk=t/(genislik*yuksekluk);

goreli_yatay_fark=abs(agmerx-boyut_merkezi_x)/genislik;
goreli_dikey_fark=abs(agmery-boyut_merkezi_y)/yuksekluk;

I1=I5(y1:int16(agmery),x1:int16(agmerx));
s=sum(I1);
t1=sum(s);
[m1 n1]=size(I1);
for i=1:m1
for j=1:n1
[y x]=find(I1==1);
x11=min(x);
x21=max(x);
y11=min(y);
y21=max(y);
if I1(i,j)==1

sutuna=sutuna+j*I1(i,j);
satira=satira+i*I1(i,j);
end
end
end;
agmera=sutuna/t1;
agmerb=satira/t1;
I1a=I1(y11:int16(agmerb),x11:int16(agmera));
s1a=sum(I1a);
t1a=sum(s1a);
[m1a n1a]=size(I1a);

I2a=I1(y11:int16(agmerb),int16(agmera):x21);
s2a=sum(I2a);
t2a=sum(s2a);
[m2a n2a]=size(I2a);
I3a=I1(int16(agmerb):y21,x11:int16(agmera));
s3a=sum(I3a);
t3a=sum(s3a);
[m3a n3a]=size(I3a);
I4a=I1(int16(agmerb):y21,int16(agmera):x21);
s4a=sum(I4a);
t4a=sum(s4a);
[m4a n4a]=size(I4a);
I2=I5(y1:int16(agmery),int16(agmerx):x2);
s=sum(I2);
t2=sum(s);
[m2 n2]=size(I2);
for i=1:m2
for j=1:n2
[y x]=find(I2==1);
x12=min(x);
x22=max(x);
y12=min(y);
y22=max(y);
if I2(i,j)==1

sutunb=sutunb+j*I2(i,j);
satirb=satirb+i*I2(i,j);
end
end
end;
agmerc=sutunb/t2;
agmerd=satirb/t2;
I1b=I2(y12:int16(agmerd),x12:int16(agmerc));
s1b=sum(I1b);
t1b=sum(s1b);
[m1b n1b]=size(I1b);
I2b=I2(y12:int16(agmerd),int16(agmerc):x22);
s2b=sum(I2b);
t2b=sum(s2b);
[m2b n2b]=size(I2b);
I3b=I2(int16(agmerd):y22,x12:int16(agmerc));
s3b=sum(I3b);
t3b=sum(s3b);
[m3b n3b]=size(I3b);
I4b=I2(int16(agmerd):y22,int16(agmerc):x22);
s4b=sum(I4b);
t4b=sum(s4b);
[m4b n4b]=size(I4b);
I3=I5(int16(agmery):y2,x1:int16(agmerx));

```

```

s=sum(I3);
t3=sum(s);
[m3 n3]=size(I3);
for i=1:m3
for j=1:n3
    [y x]=find(I3==1);
x13=min(x);
x23=max(x);
y13=min(y);
y23=max(y);
if I3(i,j)==1

sutunc=sutunc+j*I3(i,j);
satirc=satirc+i*I3(i,j);
end
end
end
agmere=sutunc/t3;
agmerf=satirc/t3;
I1c=I3(y13:int16(agmerf),x13:int16(agmere));
s1c=sum(I1c);
t1c=sum(s1c);
[m1c n1c]=size(I1c);
I2c=I3(y13:int16(agmerf),int16(agmere):x23);
s2c=sum(I2c);
t2c=sum(s2c);
[m2c n2c]=size(I2c);
I3c=I3(int16(agmerf):y23,x13:int16(agmere));
s3c=sum(I3c);
t3c=sum(s3c);
[m3c n3c]=size(I3c);
I4c=I3(int16(agmerf):y23,int16(agmere):x23);
s4c=sum(I4c);
t4c=sum(s4c);
[m4c n4c]=size(I4c);

I4=I5(int16(agmery):y2,int16(agmerx):x2);
s=sum(I4);
t4=sum(s);
[m4 n4]=size(I4);

for i=1:m4
for j=1:n4
    [y x]=find(I4==1);
x14=min(x);
x24=max(x);
y14=min(y);
y24=max(y);

if I4(i,j)==1

sutund=sutund+j*I4(i,j);
satird=satird+i*I4(i,j);
end
end
end
agmerg=sutund/t4;
agmerh=satird/t4;
I1d=I4(y14:int16(agmerh),x14:int16(agmerg));
s1d=sum(I1d);
t1d=sum(s1d);
[m1d n1d]=size(I1d);
I2d=I4(y14:int16(agmerh),int16(agmerg):x24);
s2d=sum(I2d);
t2d=sum(s2d);
[m2d n2d]=size(I2d);
I3d=I4(int16(agmerh):y24,x14:int16(agmerg));
s3d=sum(I3d);
t3d=sum(s3d);
[m3d n3d]=size(I3d);
I4d=I4(int16(agmerh):y24,int16(agmerg):x24);
s4d=sum(I4d);
t4d=sum(s4d);
[m4d n4d]=size(I4d);
k=k+1
Ibutun2(k,:)= [genislik/n yukseklik/m ((yukseklik/m)/(genislik/n)) agmerx/n agmery/m
boyut_merkezi_x/n boyut_merkezi_y/m yogunluk goreli_yatay_fark goreli_dikey_fark t1/(m1*n1)
t2/(m2*n2) t3/(m3*n3) t4/(m4*n4) t1a/(m1a*n1a) t2a/(m2a*n2a) t3a/(m3a*n3a) t4a/(m4a*n4a)
t1b/(m1b*n1b) t2b/(m2b*n2b) t3b/(m3b*n3b) t4b/(m4b*n4b) t1c/(m1c*n1c) t2c/(m2c*n2c) t3c/(m3c*n3c)
t4c/(m4c*n4c) t1d/(m1d*n1d) t2d/(m2d*n2d) t3d/(m3d*n3d) t4d/(m4d*n4d)];
Ibutun2yeni=Ibutun2';

```

```

set(handles.uitable1,'Data',Ibutun2yeni)
set(handles.uitable1,'RowName',{'Genişlik','Yükseklik','Oran','Ağırlık merkezi-x','Ağırlık
merkezi-y','Boyut merkezi-x','Boyut merkezi-y','Yoğunluk','Merkezi farklar x','Merkezi farklar
y','1.parça yoğunluk','2.parça yoğunluk','3.parça yoğunluk','2.4.parça yoğunluk','1.1.parça
yoğunluk','1.2.parça yoğunluk','1.3.parça yoğunluk','1.4.parça yoğunluk','2.1.parça
yoğunluk','2.2.parça yoğunluk','2.3.parça yoğunluk','4.parça yoğunluk','3.1.parça
yoğunluk','3.2.parça yoğunluk','3.3.parça yoğunluk','3.4.parça yoğunluk','4.1.parça
yoğunluk','4.2.parça yoğunluk','4.3.parça yoğunluk','4.4.parça yoğunluk'});
set(handles.uitable1,'ColumnName',{'Değerleri'});
save Ibutun2yeni
set(handles.test,'enable','on');
% -----
function testg_Callback(hObject, eventdata, handles)

load b1
load net12
folders=dir('C:\Users\hp\Documents\ebruyedek\EBRU HOCA YEDEKLER\EBRU HOCA D\imzalar\test\');
uzunluktest=length(folders);
sayac=0;
p=0;
for mmm=3:uzunluktest

    srcFiles =dir(strcat('C:\Users\hp\Documents\ebruyedek\EBRU HOCA YEDEKLER\EBRU HOCA
D\imzalar\test\',folders(mmm).name,'\*.bmp'));
    end
Rsayisitest= length(srcFiles);

for jj=1:Rsayisitest*(uzunluktest-2)

    XX(:,jj)=sim(net12, b1(:,jj));
    sayac=sayac+1;

    [deger,index]=max(XX(:,jj));

fprintf('%d Bu imza %d doğrulukla %s kişisine aittir\n',jj,deger, folders(index+2).name)

    if (sayac>=1 && sayac<=10)&&strcmp(folders(index+2).name,'01adnan_can')
        p=p+1;hata=1-deger;

    elseif (sayac>=11 && sayac<=20)&&strcmp(folders(index+2).name,'02adnan_sevki_temiz')
        p=p+1; hata=1-deger;

    elseif (sayac>=21 && sayac<=30)&&strcmp(folders(index+2).name,'03ahmet_ozdere')
        p=p+1; hata=1-deger;

    elseif (sayac>=31 && sayac<=40)&& strcmp(folders(index+2).name,'04armagan_ebru_temiz')
        p=p+1; hata=1-deger;

    elseif (sayac>=41 && sayac<=50)&&strcmp(folders(index+2).name,'05ayfer_temiz')
        p=p+1; hata=1-deger;

    elseif (sayac>=51 && sayac<=60)&& strcmp(folders(index+2).name,'06ayla_sacan');
        p=p+1; hata=1-deger;

    elseif (sayac>=61 && sayac<=70)&& strcmp(folders(index+2).name,'07ayse_alkis')
        p=p+1; hata=1-deger;

    elseif (sayac>=71 && sayac<=80)&& strcmp(folders(index+2).name,'08deniz_acar')
        p=p+1; hata=1-deger;

    elseif (sayac>=81 && sayac<=90)&& strcmp(folders(index+2).name,'09dursiye_aygul');
        p=p+1; hata=1-deger;

    elseif (sayac>=91 && sayac<=100)&& strcmp(folders(index+2).name,'10fatma_ceyda_gonenc');
        p=p+1; hata=1-deger;

    elseif (sayac>=101 && sayac<=110)&& strcmp(folders(index+2).name,'11fatma_omit_arslan');
        p=p+1; hata=1-deger;

```

```

elseif (sayac>=111 && sayac<=120) && strcmp(folders(index+2).name,'12kemal_mercan')
    p=p+1; hata=1-deger;

elseif (sayac>=121 && sayac<=130) && strcmp(folders(index+2).name,'13mahir_un')
    p=p+1;hata=1-deger;

elseif (sayac>=131 && sayac<=140) && strcmp(folders(index+2).name,'14merve_karaalioglu')
    p=p+1;hata=1-deger;

elseif (sayac>=141 && sayac<=150) && strcmp(folders(index+2).name,'15mustafa_sacan')
    p=p+1;hata=1-deger;

elseif (sayac>=151 && sayac<=160) && strcmp(folders(index+2).name,'16nesrin_karadas')
    p=p+1;hata=1-deger;

elseif (sayac>=161 && sayac<=170) && strcmp(folders(index+2).name,'17nihat_arda')
    p=p+1;hata=1-deger;

elseif (sayac>=171 && sayac<=180) && strcmp(folders(index+2).name,'18osman_soylu')
    p=p+1; hata=1-deger;

elseif (sayac>=181 && sayac<=190) && strcmp(folders(index+2).name,'19rasit_koker')
    p=p+1; hata=1-deger;

elseif (sayac>=191 && sayac<=200) && strcmp(folders(index+2).name,'20resat_odun')
    p=p+1; hata=1-deger;

elseif (sayac>=201 && sayac<=210) && strcmp(folders(index+2).name,'21sami_demirel')
    p=p+1; hata=1-deger;

elseif (sayac>=211 && sayac<=220) && strcmp(folders(index+2).name,'22selnur_toksoy')
    p=p+1; hata=1-deger;

elseif (sayac>=221 && sayac<=230) && strcmp(folders(index+2).name,'23yukse_altinel')
    p=p+1; hata=1-deger;

elseif (sayac>=231 && sayac<=240) && strcmp(folders(index+2).name,'24zekeriye_karaca')
    p=p+1; hata=1-deger;
end

deger1(jj,:)=deger;

index1(jj)=index;

end

yuzde = ( p/(Rsayisitest*(uzunluktest-2))*100;

set(handles.text5,'string',yuzde)

% -----
function egitimp_Callback(hObject, eventdata, handles)

)
set(handles.axes4,'Visible','off')
numi1 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfa2','B2:B10');
numi5 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfa2','B11:B19');
numi10 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfa2','B20:B28');
numi15 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfa2','B29:B37');
numi20 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfa2','B38:B46');
numi25 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfa2','B47:B55');

nump1 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfa2','C2:C10');
nump5 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfa2','C11:C19');
nump10 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfa2','C20:C28');
nump15 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfa2','C29:C37');
nump20 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfa2','C38:C46');

```

```

nump25 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfa2','C47:C55') ;

figure;
plot(numi1,nump1);
xlabel('Sindir Sayısı');
ylabel('Performans');
axis([0 600 70 100]);
figure;
plot(numi5,nump5);
xlabel('Sindir Sayısı');
ylabel('Performans');
axis([0 600 70 100]);
figure;
plot(numi10,nump10);
xlabel('Sindir Sayısı');
ylabel('Performans');
axis([0 600 70 100]);
figure;
plot(numi15,nump15);
xlabel('Sindir Sayısı');
ylabel('Performans');
axis([0 600 70 100]);
figure;
plot(numi20,nump20);
xlabel('Sindir Sayısı');
ylabel('Performans');
axis([0 600 70 100]);
figure;
plot(numi25,nump25);
xlabel('Sindir Sayısı');
ylabel('Performans');
axis([0 600 70 100]);

Ax = findall(0,'type','axes');
axis(Ax(10),[0 600 70 100])
axes(handles.axes3);

plot(numi1,nump1,'g',numi5,nump5,'r',numi10,nump10,'k',numi15,nump15,'m',numi20,nump20,'c',numi25,
nump25,'k:');
grid on;
xlabel('Sindir Sayısı');
ylabel('Performans');
legend('yayılım=1','yayılım=5','yayılım=10','yayılım=15','yayılım=20','yayılım=25')
axis([0 600 70 100]);
% -----
function gricevir_Callback(hObject, eventdata, handles)

load giris

I3=rgb2gray(giris);
axes(handles.axes2);
imshow(I3)
% -----
function otsusiyahbeyaz_Callback(hObject, eventdata, handles)

load giris
I1=rgb2gray(giris);
esik=graythresh(I1);
siyahbeyaz=im2bw(I1,esik);
I1=siyahbeyaz;
axes(handles.axes2);

imshow(I1)
% -----
function ellesiyahbeyaz_Callback(hObject, eventdata, handles)
load giris;

handles = guidata(gcf);
deger=get(handles.edit1,'String');

I3=rgb2gray(giris);
size(I3);
if isempty(deger)==1
    msgbox('lütfen bir eşik değeri girin')
else
user_entry = str2double(get(handles.edit1,'string'));

```

```

clear siyahbeyaz;

a=size(I3(:,1));
b=size(I3(1,:));

for i=1: a(1,1)
for j=1:b(1,2)

    piksel=I3(i,j);
    if (piksel)< user_entry

        % warndlg('Sistem kaynakları kritik seviyededir.', 'Tehlike', 'modal')

I3(i,j)=0;%burda ise 255 ise yani beyazsa siyah yap?yor ??nk? imzan?n arkas? siyah imza beyazd?
imzay? siyah arkaplan? beyaz yapmak i?in beyaz olan yerleri siyah siyah olan yerleri ikili renk
sisteminde beyaza ?evirdim.a?a??daki
%kodlar siyah beyaz kabul ediyor.
    else
I3(i,j)=255;
        end
    end
end
end

Ikili=double(I3);%double say?s??na ?evirmeyince baz? kodlar hata veriyor
for i=1:a(1,1)%yukar? kodda gri modunda siyah beyaza ?evirdik
for j=1:b(1,2)
if Ikili(i,j)==255
Ikili(i,j)=1;%burda ise 255 ise yani beyazsa siyah yap?yor ??nk? imzan?n arkas? siyah imza
beyazd? imzay? siyah arkaplan? beyaz yapmak i?in beyaz olan yerleri siyah siyah olan yerleri
ikili renk sisteminde beyaza ?evirdim.a?a??daki
%kodlar siyah beyaz kabul ediyor.
    else
Ikili(i,j)=0;
    end
end
end
end
siyahbeyaz=Ikili;
axes(handles.axes2);

imshow(siyahbeyaz);

end
% -----
function iskeletinicikar_Callback(hObject, eventdata, handles)

load giris
I2=rgb2gray(giris);
esik=graythresh(I2);
siyahbeyaz=im2bw(I2,esik);

I4=siyahbeyaz;
I4=~I4;
bw2 = bwmorph(~I4, 'thicken',inf);
    ince= bw2;
    axes(handles.axes2);

    imshow(ince);
% -----
function dosyaac_Callback(hObject, eventdata, handles)
folders=dir('C:\Users\hp\Documents\ebruyedek\EBRU HOCA YEDEKLER\EBRU HOCA D\imzalar\test\');
filename=imgetfile;

save folders
giris = imread(filename);
[pathstr,name,ext] = fileparts(filename);
save giris;
%handles.axes1 = axes;
axes(handles.axes1);
%axes(h1);
imshow(giris);
save name;
set(handles.gricevir,'enable','on');
set(handles.gricevir,'enable','on');
set(handles.otsusiyahbeyaz,'enable','on');

```

```

set(handles.ellesiyahbeyaz,'enable','on');
set(handles.iskeletinicikar,'enable','on');
set(handles.ozelcikart,'enable','on');
set(handles.axes2,'visible','on');
set(handles.text7,'visible','on');
set(handles.text1,'visible','on');
set(handles.edit1,'visible','on');
set(handles.uitable1,'visible','on');

function egitim_Callback(hObject, eventdata, handles)
load al
load target30
handles = guidata(gcf);
deger2=get(handles.edit5,'String');

deger3=get(handles.edit3,'String');
if isempty(deger2)==1 || isempty(deger3)==1
    msgbox('Maksimum sinir sayısını ya da yayılım değerini girmedinz')
else
    yayilim = str2double(get(handles.edit5,'string'));

dongu = str2double(get(handles.edit3,'string'));
eg = 0.0001; % sum-squared error goal
sc = 6; % spread constant
net12 = newrb(al,target30,eg,yayilim,dongu);
%ftdnn_net.divideFcn = ''
save net12
save al
set(handles.testg,'enable','on');

end

% -----
function egitimy_Callback(hObject, eventdata, handles)

set(handles.axes3,'visible','off');
numy50 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfal','A2:A7');

numy150 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfal','A8:A13') ;
numy225 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfal','A14:A19') ;
numy230 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfal','A20:A25') ;
numy235 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfal','A26:A31') ;
numy250 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfal','A32:A37') ;
numy300 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfal','A38:A43') ;
numy400 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfal','A44:A49') ;
numy550 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfal','A50:A55') ;

nump50 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfal','C2:C7') ;
nump150 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfal','C8:C13') ;
nump225 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfal','C14:C19') ;
nump230 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfal','C20:C25') ;
nump235 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfal','C26:C31') ;
nump250 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfal','C32:C37') ;
nump300 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfal','C38:C43') ;
nump400 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfal','C44:C49') ;
nump550 = xlsread('C:\Users\hp\Desktop\tezperformans.xlsx','Sayfal','C50:C55') ;
figure;
plot(numy50,nump50);
xlabel('Yayılım1');
ylabel('Performans');
axis([0 25 70 100]);
figure;
plot(numy150,nump150);
xlabel('Yayılım2');
ylabel('Performans');
axis([0 25 70 100]);
figure;
plot(numy225,nump225);
xlabel('Yayılım3');
ylabel('Performans');
axis([0 25 70 100]);
figure;
plot(numy230,nump230);

```



```

xlabel('Yayılım4');
ylabel('Performans');
axis([0 25 70 100]);
figure;
plot(numy235,nump235);
xlabel('Yayılım5');
ylabel('Performans');
axis([0 25 70 100]);
figure;
plot(numy250,nump250);
xlabel('Yayılım6');
ylabel('Performans');
axis([0 25 70 100]);
figure;
plot(numy300,nump300);
xlabel('Yayılım7');
ylabel('Performans');
axis([0 25 70 100]);
figure;
plot(numy400,nump400);
xlabel('Yayılım8');
ylabel('Performans');
axis([0 25 70 100]);
figure;
plot(numy550,nump550);
xlabel('Yayılım9');
ylabel('Performans');
axis([0 25 70 100]);

Ax = findall(0,'type','axes');
axis(Ax(13),[0 25 70 100])
axes(handles.axes4);
plot(numy50,nump50,'r',numy150,nump150,'g',numy225,nump225,'b',numy230,nump230,'m',numy235,nump235,'c',numy250,nump250,'k',numy300,nump300,'r--',numy400,nump400,'g:',numy550,nump550,'k-.');
grid on;
xlabel('Yayılım');
ylabel('Performans');
legend('iterasyon=50','iterasyon=150','iterasyon=225','iterasyon=230','iterasyon=235','iterasyon=250','iterasyon=300','iterasyon=400','iterasyon=550')
axis([0 25 70 100]);

% -----
function test_Callback(hObject, eventdata, handles)
load giris
load name
load net12
load Ibutun2yeni
folders=dir('C:\Users\hp\Documents\ebruyedek\EBRU HOCA YEDEKLER\EBRU HOCA D\imzalar\test\');
uzunluktest=length(folders);

testcikisi=sim(net12, Ibutun2yeni);
[deger,index]=max( testcikisi);

if strcmp(folders(index+2).name,'01adnan_can')
    msgbox('İmzası bulunan kişi Adnan Can''''dir')
elseif strcmp(folders(index+2).name,'02adnan_sevki_temiz')
    msgbox('İmzası bulunan kişi Adnan Şevki Temiz''''dir')

elseif strcmp(folders(index+2).name,'03ahmet_ozdere')

msgbox('İmzası bulunan kişi Ahmet Özdere''''dir')

elseif strcmp(folders(index+2).name,'04armagan_ebru_temiz')
    msgbox('İmzası bulunan kişi Armağan Ebru Temiz''''dir')

elseif strcmp(folders(index+2).name,'05ayfer_temiz')
    msgbox('İmzası bulunan kişi Ayfer Temiz''''dir')

elseif strcmp(folders(index+2).name,'06ayla_sacan');
    msgbox('İmzası bulunan kişi Ayla Saçan''''dr')

elseif strcmp(folders(index+2).name,'07ayse_alkis')
    msgbox('İmzası bulunan kişi Ayşe Alkış''''dir')
elseif strcmp(folders(index+2).name,'08deniz_acar')

```

```

        msgbox('İmzası bulunan kişi Deniz Acar''''dir')

elseif strcmp(folders(index+2).name,'09dursiye_aygul');
    msgbox('İmzası bulunan kişi Dursiye Aygül''''dür')

elseif strcmp(folders(index+2).name,'10fatma_veyda_gonenc');
    msgbox('İmzası bulunan kişi Fatma Ceyda Gönenc''''tir')

elseif strcmp(folders(index+2).name,'11fatma_omit_arslan');
    msgbox('İmzası bulunan kişi Fatma Ümit Arslan''''dır')

elseif strcmp(folders(index+2).name,'12kemal_mercan')
    msgbox('İmzası bulunan kişi Kemal Mercan''''dır')

elseif strcmp(folders(index+2).name,'13mahir_un')
    msgbox('İmzası bulunan kişi Mahir Ün''''dür')

elseif strcmp(folders(index+2).name,'14merve_karaalioglu')
    msgbox('İmzası bulunan kişi Merve Karaalioglu''''dur')

elseif strcmp(folders(index+2).name,'15mustafa_sacan')
    msgbox('İmzası bulunan kişi Mustafa Saçan''''dır')

elseif strcmp(folders(index+2).name,'16nesrin_karadas')
    msgbox('İmzası bulunan kişi Nesrin Karadaş''''dır')

elseif strcmp(folders(index+2).name,'17nihat_arda')
    msgbox('İmzası bulunan kişi Nihat Arda''''dır')

elseif strcmp(folders(index+2).name,'18osman_soylu')
    msgbox('İmzası bulunan kişi Osman Soylu''''dur')

elseif strcmp(folders(index+2).name,'19rasit_koker')
    msgbox('İmzası bulunan kişi Raşit Köker''''dir')

elseif strcmp(folders(index+2).name,'20resat_odun')
    msgbox('İmzası bulunan kişi Reşat Ödün''''dür')

elseif strcmp(folders(index+2).name,'21sami_demirel')
    msgbox('İmzası bulunan kişi Sami Demirel''''dir')

elseif strcmp(folders(index+2).name,'22selnur_toksoy')
    msgbox('İmzası bulunan kişi Selnur Toksoy''''dur')

elseif strcmp(folders(index+2).name,'23yuksel_altinel')
    msgbox('İmzası bulunan kişi Yuksel Altinel''''dir')

elseif strcmp(folders(index+2).name,'24zekeriye_karaca')
    msgbox('İmzası bulunan kişi Zekeriye Karaca''''dır')

end

```

ÖZGEÇMİŞ

Armağan Ebru Temiz 25.12.1982 tarihinde Alanya'da doğdu. İlkokulu Ankara'da bitirdi. Ortaokulu Ankara ve Rize Lise eğitimini ise Rize ve Kocaeli'de tamamladı. Lisede Kocaeli Anadolu Lisesinden 2001 yılında mezun oldu.2005 yılında Kocaeli Üniversitesi Teknik Eğitim Fakültesi Bilgisayar öğretmenliği bölümünü bitirdi. 2005 yılı Eylül ayında sözleşmeli olarak Eskişehir'de Bilgisayar öğretmenliği yaptı.2006 Şubat ayında kadrolu olarak Gölcük Endüstri Meslek Lisesi'nde bilgisayar öğretmeni olarak göreve başladı. Şu anda Sakarya'nın Sapanca ilçesi Kırkpınar Beldesinde Muazzez Sabri Gündoğar Ortaokulunda görev yapmaktadır.