

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

**NEGATİF BAĞLANTILI ÖĞRENME ALGORİTMALI
YAPAY SİNİR AĞLARI İLE MOBİL CİHAZLARDA
OPTİK KARAKTER TANIMA UYGULAMASI**

YÜKSEK LİSANS TEZİ

Bilg. Müh. Burcu KIR

Enstitü Anabilim Dalı : BİLGİSAYAR VE BİLİŞİM MÜHENDİSLİĞİ
Tez Danışmanı : Yrd. Doç. Dr. Ali GÜLBAĞ

Eylül 2012

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ


**NEGATİF BAĞLANTILI ÖĞRENME ALGORİTMALI
YAPAY SINIR AĞLARI İLE MOBİL CİHAZLARDA
OPTİK KARAKTER TANIMA UYGULAMASI**

YÜKSEK LİSANS TEZİ


Bilg. Müh. Burcu KIR

Enstitü Anabilim Dah : BİLGİSAYAR VE BİLİŞİM MÜHENDİSLİĞİ

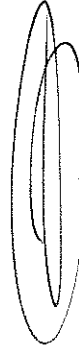
Bu tez 11 / 09 /2012 tarihinde aşağıdaki jüri tarafından Oybirliği ile kabul edilmiştir.


Yrd.Doc.Dr. ALI GÜLBAĞ

Jüri Başkanı


Prof. Dr. Nejat Yılmaz

Üye


Prof. Dr. I. Mustafa

Üye

CEDMSE

ÖNSÖZ

Bu çalışma, Sakarya Üniversitesi Fen bilimleri Enstitüsü Bilgisayar ve Bilişim Mühendisliği Anabilim Dalı Yüksek Lisans Tezi olarak hazırlanan “Negatif Bağlantılı Öğrenme Algoritmaları Yapay Sinir Ağları İle Mobil Cihazlarda Optik Karakter Tanıma Uygulaması” isimli tezi içermektedir.

Yüksek lisans ders ve tez aşamasındaki çalışmalarım süresince her türlü destek ve yardımlarını esirgemeyen değerli hocalarım Yrd. Doç. Dr. Ali GÜLBAĞ ve Doç. Dr. Cemil ÖZ’e teşekkürü bir borç bilirim.

Hayatımın her döneminde bana kararlı, azimli ve çalışkan olmayı aşıl原因, daima ve her koşulda yanımda olan aileme de teşekkür ederim.

İÇİNDEKİLER

| | |
|---|------|
| TEŞEKKÜR..... | ii |
| İÇİNDEKİLER | iii |
| SİMGELER VE KISALTMALAR LİSTESİ..... | vi |
| ŞEKİLLER LİSTESİ | viii |
| TABLolar LİSTESİ..... | x |
| ÖZET..... | xi |
| SUMMARY..... | xii |
| BÖLÜM 1. | |
| GİRİŞ..... | 1 |
| BÖLÜM 2. | |
| MOBİL CİHAZLAR ile GÖRÜNTÜ İŞLEME | 4 |
| 2.1. Mobil Cihazlar ve Teknoloji..... | 4 |
| 2.1.1. Mobil cihazların tarihçesi..... | 4 |
| 2.1.2. İşletim sistemi ve mobil cihaz donanımı..... | 6 |
| 2.2. Görüntü İşleme..... | 7 |
| 2.2.1. Görüntü işlemeye giriş..... | 7 |
| 2.2.2. Piksel tanımı..... | 9 |
| 2.2.3. Resmi tarama..... | 10 |
| 2.2.4. Gri resme dönüştürme..... | 10 |
| 2.2.5. Ön işleme aşamaları..... | 12 |
| 2.2.6. Eşikleme..... | 12 |
| 2.2.7. Gürültü temizleme..... | 12 |

| | |
|--|-----------|
| 2.2.8. Geniřletme..... | 13 |
| 2.2.9. İnceltme..... | 14 |
| 2.2.10. Normalize etme..... | 14 |
| 2.2.11. Bölütleme (Segmentasyon)..... | 14 |
| 2.2.12. Özellik çıkartma..... | 15 |
| 2.2.13. Karakter tanıma..... | 15 |
| 2.3. Mobil Cihazlarda Görüntü İşleme..... | 16 |
| 2.4. Optik Karakter Tanıma..... | 17 |
| 2.4.1. Optik karakter tanımanın tarihçesi..... | 18 |
| 2.4.2. OCR kullanım alanları..... | 20 |
| 2.4.3. OCR sisteminin genel yapısı..... | 21 |
| | |
| BÖLÜM 3. | |
| YAPAY SİNİR AĞLARI..... | 24 |
| 3.1. Yapay Sinir Ağları Giriş..... | 24 |
| 3.2. Yapay Sinir Ağları Tarihçe..... | 25 |
| 3.3. YSA' nın Genel Tanımları ve Özellikleri..... | 29 |
| 3.4. Yapay Sinir Hücresinin Yapısı..... | 30 |
| 3.5. Yapay Sinir Ağının Yapısı..... | 32 |
| 3.6. Yapay Sinir Ağlarının Avantajları ve Dezavantajları..... | 34 |
| 3.7. Uygulama Alanları..... | 34 |
| | |
| BÖLÜM 4. | |
| BİRLEŐİK YAPAY SİNİR AĞLARI..... | 37 |
| 4.1. Birleşik Yapay Sinir Ağı Tanımı ve Yapısı..... | 37 |
| 4.2. Birleşik Yapay Sinir Ağların Eğitilmesi ve Test Edilmesi..... | 39 |
| 4.3. Çıkış Modülü (Ortak Karar Verme Modülü)..... | 39 |
| 4.4. Negatif Bağlantılı (İlişkili) Öğrenme..... | 40 |
| 4.4.1. Negatif bağlantılı öğrenmenin yararları..... | 44 |
| 4.4.2. Negatif bağlantılı öğrenme ile diğer yöntemlerin karşılaştırılması..... | 45 |

| | |
|--|----|
| BÖLÜM 5. | |
| NEGATİF BAĞLANTILI ÖĞRENME ALGORİTMALI YAPAY SİNİR AĞLARI ile MOBİL CİHAZLARDA OPTİK KARAKTER TANIMA UYGULAMASI..... | 47 |
| 5.1. İşlem Basamakları..... | 47 |
| 5.2. Karakter Tanıma..... | 49 |
| 5.3. Uygulamanın Çalışması..... | 52 |
| | |
| BÖLÜM 6. | |
| SONUÇLAR ve ÖNERİLER..... | 56 |
| | |
| KAYNAKLAR..... | 58 |
| ÖZGEÇMİŞ..... | 64 |

SİMGELER VE KISALTMALAR LİSTESİ

| | |
|--------------|--|
| G_i | : Ağa verilen girdi |
| W_i | : Ağırlık değerleri |
| d | : Skaler büyüklük |
| $d(n)$ | : n. skaler büyüklük |
| N | : Eğitim setinin boyutu (eğitim seti örneği) |
| M | : Yapay sinir ağı topluluğundaki bireysel sinir ağı sayısı |
| $F_i(n)$ | : n. eğitimdeki i. ağıın çıktısı |
| $F(n)$ | : n. eğitimdeki topluluğun çıktısı |
| D | : Eğitim seti |
| E_i | : Hata fonksiyonu |
| $E_i(n)$ | : n. eğitimdeki i. ağıının hata fonksiyonu |
| p_i | : İlişkilendirilmiş hata fonksiyonu (ceza fonksiyonu) |
| $P_i(n)$ | : n. eğitimdeki ilişkilendirilmiş hata fonksiyonu |
| λ | : Hata kuvveti katsayısı |
| $E_{ens}(n)$ | : n. eğitim için deneysel risk fonksiyonu |
| ADALINE | : Adaptif lineer eleman |
| AMPS | : İleri mobil telefon sistemi (Advanced mobile phone service) |
| ART | : Adaptif rezonans teorisini |
| cHTML | : Kompakt metin işaretleme dili |
| DOS | : Disk işletim sistemi |
| DSP | : Sayısal sinyal işleme (Digital signal processing) |
| ECG | : Elektrokardiyogram |
| EEG | : Beyin akım grafiği |
| FPGA | : Sahada programlanabilir kapı dizisi (Field programmable gate Arrays) |
| GPRS | : Paket anahtarlamalı radyo hizmetleri |
| GRNN | : Genel regrasyon ağları |

| | |
|----------|---|
| GSM | : Mobil iletişim için küresel sistem (Global system for mobile communication) |
| HDML | : El cihazları işaretleme dili (Handheld device markup language) |
| HMM | : Saklı markov modelleri |
| HTML | : Metin işaretleme dili (Hyper text markup language) |
| HTTP | : Bağlantılı metin aktarım protokolü |
| IP | : İnternet kuralı |
| MADALINE | : Çoklu adaptif lineer eleman |
| ME | : Uzman birleştirme mimarisi |
| MR | : Manyetik rezonans |
| NCL | : Negatif ilişkili öğrenme |
| OCR | : Optik karakter tanıma (Optical character recognition) |
| PDA | : Kişisel sayısal yardımcı (Personal digital assistant) |
| PNN | : Olasılıksal ağlar |
| RAM | : Rastgele erişimli hafıza |
| RBF | : Radyal tabanlı fonksiyon ağı |
| RGB | : Kırmızı-yeşil-mavi (red-green-blue) |
| SOM | : Özdüzenleyici haritalar |
| TCP | : Aktarım denetim kuralı |
| WAP | : Kablosuz uygulama protokolü (Wireless application protocol) |
| YSA | : Yapay sinir ağları |

ŞEKİLLER LİSTESİ

| | | |
|-------------|---|----|
| Şekil 2.1. | 2000 yılında kablosuz internet kullanımı..... | 6 |
| Şekil 2.2. | Dijital görüntünün yapısı..... | 9 |
| Şekil 2.3. | Zayıf kalitede taranmış yazının görüntü örneği..... | 10 |
| Şekil 2.4. | Orjinal resmin görüntüsü..... | 11 |
| Şekil 2.5. | 256 bit değerindeki resmin görüntüsü..... | 11 |
| Şekil 2.6. | 16 bit değerindeki resmin görüntüsü..... | 11 |
| Şekil 2.7. | 4 bit değerindeki resmin görüntüsü..... | 11 |
| Şekil 2.8. | 2 bit değerindeki resmin görüntüsü..... | 11 |
| Şekil 2.9. | Resmin gürültülü hali..... | 13 |
| Şekil 2.10. | Resmin gürültüden arındırılmış hali..... | 13 |
| Şekil 2.11. | Maske örneği | 13 |
| Şekil 2.12. | Karakter tanıma örneği..... | 16 |
| Şekil 2.13. | OCR sisteminin genel yapısı..... | 22 |
| Şekil 3.1. | Yapay sinir hücresinin yapısı..... | 31 |
| Şekil 3.2. | YSA'da katmanların birbirleri ile ilişkisi..... | 33 |
| Şekil 3.3. | Yapay sinir örneği..... | 33 |
| Şekil 4.1. | Grup halinde olan yapay sinir ağı modeli..... | 38 |
| Şekil 5.1. | Çalışmanın algoritması..... | 48 |
| Şekil 5.2. | Karakterlerin matrissel gösterimi..... | 49 |
| Şekil 5.3. | Grup halinde olan yapay sinir ağı modeli..... | 50 |
| Şekil 5.4. | Yapay sinir ağının hedeflenen çıkış değerleri..... | 50 |
| Şekil 5.5. | Hedeflenen değerler matrisinde hangi satırın hangi karaktere ait olduğunun gösterimi..... | 51 |
| Şekil 5.6. | Windows Mobile 5.0 Pocket PC R2 Emulator..... | 52 |
| Şekil 5.7. | Mobil uygulamanın çalışması..... | 53 |

| | | |
|------------|--|----|
| Şekil 5.8. | Mobil uygulamada resmin seçilmesi..... | 54 |
| Şekil 5.9. | Masaüstü uygulamada karakter tanıma..... | 54 |

TABLolar LİSTESİ

| | | |
|------------|--|----|
| Tablo 3.1. | YSA kronolojik sırası..... | 28 |
| Tablo 3.2. | YSA'nın avantajları ve dezavantajları | 34 |
| Tablo 5.1. | Bazı test sonuçlarının şekillerle gösterimi..... | 54 |
| Tablo 5.2. | Başarı oranları..... | 55 |

ÖZET

Anahtar kelimeler: Yapay Sinir Ağları, Negatif Bağlantılı Öğrenme, Görüntü İşleme, Optik Karakter Tanıma, Mobil Cihazlar

Örüntü tanımanın amacı nesnelere kategorize etmek ya da sınıflandırmaktır. Bu nesnelere, uygulamaya göre görüntü, ses ya da sınıflandırılması istenen herhangi bir işaret olabilir ve genel olarak örüntü diye adlandırılır. Bu çalışma cep telefonu kamerası ile alınan resmin içerisindeki metin karakterlerini belirlemekte ve bu metindeki kelimenin tanınmasını sağlamaktadır. Karakterlerin sınıflandırılmasında, negatif bağlantılı öğrenme metodu kullanan birleşik yapay sinir ağı kullanılmıştır.

Bu tez çalışmasında cep telefonu kamerası ile çekilen resim, görüntü işleme tekniklerinden geçirilerek karakterlerine ayrıştırılmaktadır. Elde edilen karakter matris dizileri, negatif bağlantılı öğrenme metodu ile eğitilmiş yapay sinir ağı topluluğuna giriş olarak sırasıyla verilmektedir. Ağ çıkışında giriş matris dizilerine karşılık düşen harf elde edilmekte ve elde edilen harfler kelime oluşturacak şekilde sırası ile birleştirilmektedir.

Proje C Sharp dili kullanılarak Visual Studio 2008 ortamında tasarlanmıştır.

THE APPLICATION OF OPTICAL CHARACTER RECOGNITION FOR MOBILE DEVICE VIA ARTIFICIAL NEURAL NETWORKS WITH NEGATIVE CORRELATION LEARNING ALGORITHM

SUMMARY

Key Words: Artificial Neural Network, Negative Corelation Learning, Image Processing, Optical Character Recognition

The aim of pattern recognition is to categorize or classify objects. These objects might be images, sound, or any sign which are to be classified, and are generally called as patterns. This work identifies the characters inside a word picture which is taken by a cell phone camera and it provides a recognition of the word. Optic character recognition has been realized by using ensemble artificial neural network classification. Negative correlation learning method have been used in artificial neural network.

In the study, first, the word picture taken by a cell phone camera has been gone through various image proccessing techniques, and then letter pictures have been obtained by separating the word image into characters. Being binarizated, the letter pictures have been assigned. Matrix data is given to artificial neural network ensemble trained by negative correlation learning in order and the classification processing is made depending on output data.

The project is designed on Visual Studio 2008 platform by using C Sharp.

BÖLÜM 1. GİRİŞ

Cep telefonu, 1990'lı yıllarda insan yaşamına girmiş, o tarihten sonra da insan hayatındaki önemi artarak devam etmiştir. Günümüzde cep telefonları, ofisimizde ve evlerimizde kullandığımız birçok elektronik cihaz, bilgisayar ve kameraların yerine kullanılmaktadır. Bu cep telefonları, "Akıllı telefonlar" olarak isimlendirilmekte, teknolojik açıdan da mobil cihazlar olarak ifade edilmektedir. Mobil cihazlar, insanların ses, görüntü ya da elektronik verilerinin herhangi bir kısıtlamaya bağlı kalmadan rahatça kullanılabilmesine, daha az maliyetle aktarılabilmesine ya da taşınabilmesine olanak sağlar [1].

Günümüzde mobil teknolojilerin erişebilirliğinin artması sayesinde zamandan ve mekandan bağımsız olarak bilgiye erişim sağlanmıştır. Her geçen yıl bilginin olağanüstü miktarda artmasından dolayı, bireylerce bu bilgilerin tamamının öğrenilmesini zorlaştırmaktadır. İnsanların bilgiye ihtiyaç duyduğu zamanda ve yerde erişebilme ihtiyacı mobil teknolojilere olan gereksinimi zorunlu hale getirmiştir. Dijital görüntülerin, gündelik hayatın her alanında olduğu gibi mobil cihazlarda da kullanımı göz ardı edilemeyecek kadar büyüktür. Mobil cihazlar ile yapılabilecek sayısız uygulamalarla birlikte görüntü işleme uygulamalarının da dünyadaki satışları artmıştır [2].

Optik Karakter Tanıma (OCR), taranmış ya da kamera ile çekilmiş görüntülerin, el yazılarının veya çıktısı alınmış metin bilgilerinin elektronik ortama dönüştürülmesi ve sayısallaştırılması işlemidir [3]. OCR ile basılı karakterlerin, el yazısının ya da işaret bilgilerinin kolaylıkla okunup bilgisayar ortamına aktarılması sağlanabilmektedir. Bu sayede kullanıcı ile bilgisayar arasındaki iletişim ve etkileşimi arttırmaktadır [4]. OCR, yıllardır ofislerimizde ve evlerimizde OCR sitemleriyle ve bilgisayarlarla başarılı bir şekilde kullanılmaktadır. İlk modern bilgisayar destekli, 23 harfi tanıyabilen OCR sistemi GİSMO, 1951'de David

Shepard tarafından Amerika'da geliştirilmiştir. 1950'li yılların ortasından başlayıp günümüze kadar donanım cihazı ve yazılım paketi olarak kullanılan ticari OCR sistemleri gelişmeye devam etmektedir. OCR sistemleri tek bir parçadan değil, çeşitli farklı sistemlerin bir araya gelerek çalışmasıyla oluşmaktadır [5].

Mobil cihazların yaygınlaşması ile birlikte, bu cihazlar içinde OCR sistemlerine ihtiyaç duyulmaya başlanmıştır. Mobil cihaz teknolojileri, yazılım ve donanım teknolojisi ile ev ve ofislerimizde kullandığımız bilgisayar sistemleri yazılım ve donanım teknolojilerinden farklıdır. OCR sisteminin mobil cihazın sınırlı yazılım ve donanım kaynaklarını kullanacak şekilde yeniden düzenlenmesi gerekmektedir. Günümüzde mobil cihazlar üzerinde çok sayıda optik karakter tanıma çalışması yürütülmektedir [6-8].

Yapay sinir ağları, insan beyninin biyolojik sinir yapısının çalışma prensiplerinden yola çıkılarak geliştirilmiştir. YSA'lar gerçek dünyaya ait ilişkileri tahmin edebilir, sınıflandırabilir ya da tanıyabilirler [9]. 1980 yılların başından beri, mühendislik problemlerinin çözümünde yaygın olarak YSA kullanılmaktadır [10]. Ev ve iş bilgisayarlarımızda kullandığımız OCR sistemlerinde YSA başarılı bir şekilde uygulanmaktadır [4]. YSA'ların daha iyi sonuçlar vermesi için yapılan çalışmalar sonucunda çok sayıda YSA modeli ortaya çıkmıştır [11]. NCL eğitim modeli YSA eğitiminde daha iyi sonuç üretmek için birçok çalışmada kullanılmıştır [12,13].

Yapılan araştırmalar göstermektedir son yıllarda özellikle mobil telefon ya da diğer cihazların kameralarından alınan görüntülerdeki kelime ya da metinleri okuyan ve farklı dillere çeviren ya da çeşitli başka çeşitli amaçlar için kullanan [5] çalışmalar bulunmaktadır. Bu çalışmada da mobil cihazlar için OCR yazılımı gerçekleştirilmiş ve görüntüdeki metin bilgisinin tanınması sağlanmıştır. Windows Mobile 5.0 Pocket PC R2 Emulator üzerinde yapılan uygulamada, ilk önce cep telefonu ile çekilen kelime resimleri önce örüntü tanıma işleminden geçirilerek karakterlerine ayrılmış ve daha sonra önceden negatif bağlantılı öğrenme metodu ile eğitilmiş yapay sinir ağı ağırlıkları kullanılarak, karakterlerin tanınması sağlanmıştır.

Tez ařađıdaki řekilde b6l6mlere ayrılmıřtır.

- B6l6m 2’de mobil cihazlarda g6r6nt6 iřleme tekniklerinin neler olduđuna deđinilmiř olup, optik karakter tanımının ortaya ıkıřı anlatılmıřtır. Buna ek olarak, mobil cihazlardan ve g6n6m6zdeki teknolojilerden bahsedilmiřtir.
- B6l6m 3’de yapay sinir ađlarına deđinilmiřtir.
- B6l6m 4’de birleřik yapay sinir ađları anlatılmıřtır.
- B6l6m 5’de yapılan alıřma ve iřleyiřin nasıl olduđu, bu alıřmanın algoritması řekiller yardımıyla anlatılmaya alıřılmıřtır.
- B6l6m 6’da yapılan alıřmanın sonuları tartıřılmıř olup, gelecekte bu alıřmaya neler eklenebileceđinden bahsedilmiřtir.

BÖLÜM 2. MOBİL CİHAZLAR ile GÖRÜNTÜ İŞLEME TEKNİKLERİ

2.1. Mobil Cihazlar ve Teknoloji

Mobil cihazlar, insanların ses, görüntü ya da yazılı verilerinin herhangi bir kısıtlamaya bağlı kalmadan rahatça kullanılabilmesine, daha az maliyetle aktarılabilmesine ya da taşınabilmesine olanak sağlayan mobil teknoloji cihazlarıdır. Mobil cihazlar çalışırken tüm parçaları üzerinde taşırlar ve gücünü de üzerinde bulunan güç kaynağından alırlar. Radyo teknolojisiyle birlikte mobil cihazların kullanımının başladığı kabul edilmektedir. Daha sonra teknolojik olarak uygulama sırasına göre çağrı cihazları, analog ve sayısal cep telefonları, masaüstü ve dizüstü bilgisayarlar, cep bilgisayarları, internete girebilen ve üzerlerinde işletim sistemi bulunan cep telefonları ve cep bilgisayarlarıdır [1].

Günümüzde mobil teknolojilerin erişebilirliğinin artması sayesinde zamandan ve mekandan bağımsız olarak bilgiye erişim sağlanmıştır [14]. Her geçen yıl bilginin olağanüstü miktarda artması, bireylerce bu bilgilerin tamamının edinilmesi zorlaştırmaktadır. İnsanların bilgiye ihtiyaç duyduğu zamanda ve yerde erişebilme ihtiyacı mobil teknolojilere olan gereksinimi zorunlu hale getirmiştir.

2.1.1. Mobil cihazların tarihçesi

1890'lı yıllarda radyo teknolojisinin ortaya çıkması ve geliştirilmesi mobil cihazların başlangıcı kabul edilir. Bilgisayarların tarihi ise 1950'li yıllara dayanmaktadır. Bilgisayar tarihinin ilk zamanlarında, bilgisayarlar ağırlığı çok fazla olan cihazlardı. O zamanlarda bu cihazların taşınabilir (mobil) olması hayal bile edilemiyordu [15]. Radyo teknolojisinin geliştirilmesi mobil cihazların başlangıcı kabul edilir. 1965 yılından itibaren mikroişlemci teknolojisinin gelişimi ile sayısal devrelerin

küçülmesinden ve azalan güç gereksinimlerinin avantajlarından yararlanılarak cep telefonları ve PDA (Personal Digital Assistant – Kişisel Sayısal Yardımcı) üretildi.

Analog mantıkla çalışan ilk taşınabilir telefonlar 1947’de taşıtlarda kullanılmaya başlandı. Cihaz sayıları arttıkça sinyal aralıkları yetersiz kalmaya başladı ve sistemi iyileştirmek amacı ile 1980’li yıllarda HÜCRE (CELL) teknolojisi geliştirildi ve bu teknoloji Advanced Mobile Phone Service (AMPS) adı ile ticari olarak kullanılmaya başlandı.

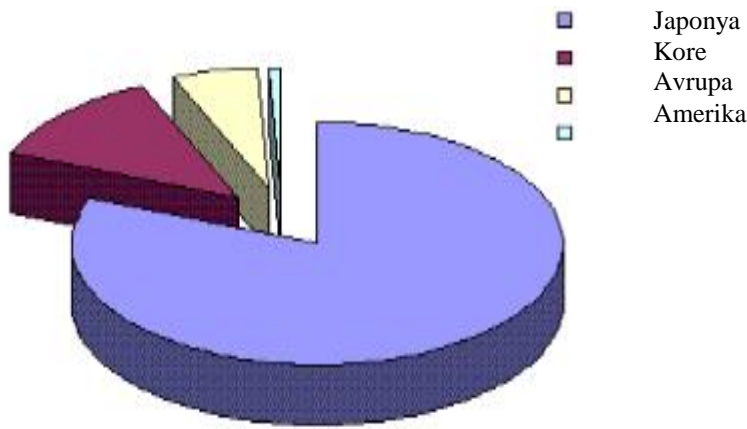
Hücreli telefon kullanıcılarının sayısının artması ve ülkeler arasında kullanılan servislerin uyumsuzluğu nedeniyle sayısal teknolojilerin kullanılması konusunda çalışmalara başlandı. 1990 yılında hücreli iletişimde sayısal teknolojiye geçildi ve GSM (Global Sistem for Mobile Communication) teknolojisi kullanılmaya başlandı. Başlarda kablosuz ağlar üzerinden veri iletişimi yapabilecek cihazlar geliştirilmek için çalışmalar yapıldı. Ancak cep telefonlarının pil tüketim sürelerinin kısıtlı olması, yüksek güçlü işletim sistemlerinin olmaması, yerel hafızalarının düşük olması ve görüntüleme yeteneklerinin az olmasından dolayı internet sayfalarında kullanılan HTML (Hyper Text Markup Language - Metin işaretleme dili) mobil web iletişimi için uygun değildi. Bu nedenle cep telefonlarında HTML yerine kullanılmak üzere HDML (Handheld Device Markup Language-El cihazları işaretleme dili) adıyla yeni bir işaret dili ve ağ kaynaklarını verimli kullanabilecek yeni bir protokol yaratıldı [15].

Nokia, Ericsson ve Motorola kendi veri iletişimi konsepti olan WAP (Wireless Application Protocol- Kablosuz Uygulama Protokolü) FORUM topluluğunu oluşturdu. WAP protokolü, TCP/IP ve HTTP’ den farklı olarak, parazitsiz, kaliteli ve kablosuz ağlarda verimli ve güvenli veri transferi yapabilecek şekilde geliştirildi.

Japonya'nın en büyük GSM şirketlerinden biri olan NTT DoCoMo Şubat 1999 yılında i - Mode ("i" harfi "interaktif", "information" ve "internet") teknolojisini tanıttı ve bu hizmet için bugüne kadar 45 milyonun üzerinde müşteri kazanabildi. WAP'a benzeyen ve internet içeriklerini cep telefonuna sunan i - Mode sayfaları cHTML (compact HTML) dilinde programlanmaktadır. HTML'nin bir türü olan

cHTML mobil telefonları tarafından desteklenen özel fonksiyonlara sahiptir. cHTML internette kullanılan HTML'ye benzediğinden telefonda animasyonlu ve renkli içerikler izlenebilmektedir. GPRS'de (General Packet Radio Service - Paket Anahtarlamalı Radyo Hizmetleri) olduğu gibi i – Mode da kullanım süresine göre değil veri alışverişine göre ücretlendirilmektedir.

Şekil 2.1'de 2000 yılındaki bazı ülkeler için kablosuz internet kullanım oranları verilmiştir.



Şekil 2.1. 2000 yılında kablosuz internet kullanımı

2.1.2. İşletim sistemi ve mobil cihaz donanımı

İşletim sistemi, bilgisayar üzerinde çalışan ve bilgisayarda bulunan tüm donanım kaynaklarını yöneten ve çeşitli uygulama yazılımları için yaygın servisleri sağlayan bir yazılımdır. İşletim sistemi, uygulama kodları genellikle direkt donanım tarafından yürütülmesine rağmen, girdi - çıktı, bellek atama gibi donanım fonksiyonları için uygulama programlarıyla bilgisayar donanımı arasında aracılık görevi yapar. İşletim sistemi giriş, çıkış ve bellek ayırma gibi donanım işlevleri için bilgisayar donanımı ve uygulama programları arasında bir bağlantıdır. Bir işletim sistemi temel olarak, kullanıcı ara yüzü (kabuk), sistem fonksiyonları ve çekirdekten oluşur. İşletim sistemleri bilgisayar, video oyun konsolları, cep telefonları ve web sunucularında bulunabilir.

Başlıca İşletim sistemleri DOS, GNU/Linux, Mac OS, Windows ve Unix' tir. Ayrıca Mobil cihazlar için; PALM OS, Symbian OS, Windows Mobil, J2ME (Java 2 Mikro Sürüm) işletim sistemleri kullanılır.

Günümüzde cep telefonları ve cep bilgisayarlarında donanım oldukça gelişmiş ve hatta dizüstü bilgisayar donanımına yaklaşmıştır. Zaman içerisinde bir masaüstü donanımına yaklaşması da beklenmektedir. Mobil cihazlarda bellek kaynak kullanımının sınırlı olması, işlemcinin yavaş çalışması ve hafıza kapasitesinin az olması mobil cihazları masaüstü cihazlardan ayıran temel özelliklerdendir. Mobil cihazlarda kullanıcı verileri genellikle RAM ortamında tutulur. Bu nedenle de mobil cihazlarda güç yönetimi oldukça önemli bir konu olmuştur.

Mobil cihazların donanım kapasiteleri ve yazılımsal altyapıları kişisel bilgisayarlardan farklıdır. Geliştirilecek bir uygulamanın mobil cihaz üzerinde çalışabilmesi ancak mobil cihazın donanım kapasitesine ve yazılım kapasitesine uygun ve bunu desteklenmesiyle mümkün olabilmektedir.

Örnek verecek olursak eğer, yaygın olarak Windows Mobile İşletim Sistemi kullanan mobil bir cihaz, ortalama olarak 600 MHz hızında bir işlemciye, 128 MB – 256 MB RAM ve 4 GB ile 32 GB arasında değişen bir depolama kapasitesine sahiptir. Teknolojinin gelişmesi ile birlikte bu hızların artacağı da aşîkârdır. Belirtilen donanım kapasiteleri görüntü işleme gibi yoğun veri işlemeyi gerektiren uygulamalarının gerçekleştirilmesi için yeterli olmaktadır.

2.2. Görüntü İşleme

2.2.1. Görüntü işlemeye giriş

Görüntü, $f(x, y)$ gibi iki değişkenli bir fonksiyon olarak tanımlanabilir. Burada x ve y değişkenlerinin uzaysal koordinatlar olduğunu düşünülürse, herhangi bir (x, y) koordinatındaki f ' nin genliğine, görüntünün o noktadaki yoğunluğu denir. x, y ve f 'nin genlik değerleri sonlu ve ayrık sayılar ise, görüntüye sayısal görüntü denir.

Sayısal görüntü, kırmızı (R), yeşil (G) ve mavi (B) piksel bileşenleri ile gösterilebilir.

Görüntü işleme esnasında, gerçek yaşamdaki görüntülerin, dijital ortama aktarılması ve resim haline getirilmesi daha sonra da bu resmin özelliklerinin ve görüntüsünün işlenerek görüntüsünün değiştirilmesi sonucunda yeni bir resim oluşur. Kaydedilmiş olan elektronik görüntü verilerinin elektronik ortamda kullanıcının amacına uygun bir şekilde değiştirmeye, düzeltmeye yönelik yapılan bilgisayar çalışmasına görüntü işleme denilebilir. Diğer bir deyişle, görüntü işleme durağan veya durağan olmayan bir görüntüden alınan anlık görüntü üstünde yapılan işlemlerin bütününe denir [15].

Görüntü işlerken bilmemiz gereken en önemli kavram pikseldir. Bir görüntü piksellerden oluşur. Piksellerin kendi başlarına en ve boy değerleri yoktur ancak, eldeki görüntünün bir satırda bulunan piksel sayıları o görüntünün genişliğini, satır sayısı da boyunu bize verir. Görüntü resimlerinin elektronik ortama aktarılması için veri formatlarının sayısallaştırılması gerekmektedir. Görüntüler analog ortamdan sayısal ortamlara geçirilirken genellikle bozulurlar. Görüntü işleme ise bu bozuklukları düzeltmek için ya da daha kullanışlı bir hale getirmek için kullanılmaktadır.

Son yıllarda dijital veri işleme inanılmaz bir hızla gelişmesini sürdürmektedir. Bu gelişme bilgisayar ve elektronik teknolojilerindeki gelişmelere paralel olarak meydana gelmiştir. Bilgisayarların boyutlarının giderek küçülmesi, bellek kapasitelerinin ve veri işleme hızlarının artışı görüntü işleme teknolojilerindeki gelişmenin hızlanmasına sebep olmuştur. Bilişim teknolojisindeki gelişmeleri de eklemek gerekirse günümüzde dünyadaki gelişmeler internet aracılığı ile anında izlenebilmekte, dolayısıyla bilginin hızla yayılması olanaklı kılınabilmektedir.

Mikroişlemci, DSP ve FPGA gibi kontrol elemanlarının gelişmesi ile görüntü işleme uygulamaları son yıllarda önem kazanmıştır. Aktif olarak kullanıldığı alanlara tıp ve biyoloji (biyomedikal görüntülerin işlenmesi ya da nükleer manyetik rezonans ve ultrasonik taramada elde edilen tıbbi görüntülerin işlenmesi), coğrafi bilimler (Jeodezi, hava ve uydu görüntülerinden hava tahmini, yeryüzündeki doğal

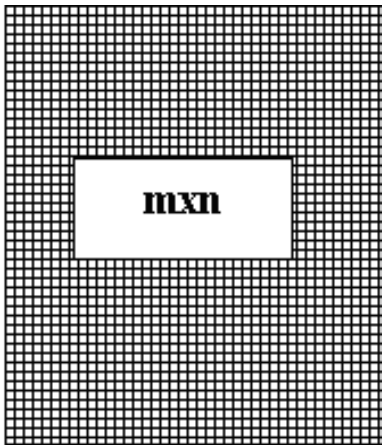
kaynakların izlenmesi, coğrafi haritaların çıkarılması, hava durumunun incelenmesi), uzay bilimleri (uzayın derinliklerinden elde edilen görüntülerin içerdiği cisimlerin anlaşılması ve analizi), savunma sanayi (Gece görüş, akıllı robot sistemleri, radar, sonar sistemleri), güvenlik sistemleri (İris ya da parmak izi tanıma, güvenlik-kamera uygulamaları) örnek gösterilebilir.

2.2.2. Piksel tanımı

Tüm sayısal görüntülerin en küçük parçası olan üçlü nokta grubuna piksel denir. Görüntünün temel bileşeni olan piksel noktalarının temel bir boyutları yoktur. Piksel noktalarının temel bir boyutları yoktur ve görüntünün temel bileşenidir. Sayısal görüntüler yan yana ve alt alta gelen piksel değerlerinden oluşmaktadırlar. Bir piksel değeri ise kırmızı (red - R), yeşil (green - G) ve mavi (blue - B) renk değerlerinin toplamından meydana gelir.

Görüntünün piksel değerlerinin belirli aralıklarda olması (0, 1, 2, ..., n), meydana gelen görüntünün niteliğini değiştirir. Örneğin "0" piksel değeri beyaz ve "n-1" piksel değeri de siyahı temsil ederse ve bu değerler arası gri tonlarını ifade etmektedir.

İkili resim (binary image) ise, resmin sadece 0 ve 1 değerlerinden oluşmasıdır. Dijital bir görüntünün yapısı Şekil 2.2'de verilmiştir.



Şekil 2.2. Dijital görüntünün yapısı

2.2.3. Resmi tarama

Tarama, gerçek nesnelerin bilgisayarların anlayabileceği forma dönüştürülmesi işlemine denilmektedir. Başka bir deyişle, istenilen metin bilgisi tarayıcıdan taranarak elde edilebilir ya da dijital kamera veya cep telefonu ile resmi çekilerek elde edilebilir. En son işlem olarak da metin bilgisi bilgisayar ortamına (sayısal ortama) geçirilir.

Tanıma işleminin kolaylığı ve doğruluğu açısından tarama kalitesinin yüksek olması önemlidir. Buna rağmen kaliteli bir görüntünün her zaman elde edilmesi zordur.



Şekil 2.3. Zayıf kalitede taranmış yazının görüntü örneği

Şekil 2.3'te görüldüğü gibi taranarak ya da resmi çekilerek elde edilen bir dokümanın görüntü kalitesi her zaman mükemmel olmayabilir. Ayrıca, nokta vuruşlu yazıcılar, çok gürültülü görüntüler oluşturabilmektedir. Eğer, iki karakter arasındaki boşluk bir piksel büyüklüğünden küçükse o zaman karakterler bitişik olarak görüntülenebilirler. Karakterleri sınıflandırmanın daha doğru yapılabilmesi için çözünürlük kalitesini arttırmak gerekmektedir [16].

2.2.4. Gri resme dönüştürme

Karakter tanıma uygulamasında görüntülerin renkli olması çok fazla gereksiz ayrıntıya yol açabilir. Gri tonlu görüntüler ise sadece renk tonu bilgisi taşırlar. Bit sayısı bu renk tonu sayısına göre değişiklik gösterir. Örneğin 8 bitlik bir görüntü 0-255 arasında toplam 256 farklı renk tonundan oluşur. Karakter tanıma işleminde

kullanılacak örnekler, beyaz zemin üzerine siyah karakterler olduğunda tanıma işleminin sonucunun daha iyi olduğu gözlemlenmektedir. Bu nedenle de renkli resmin gri forma dönüştürülmesi uygulanması gereken ilk ön işlem adımıdır. Resimler piksellerden, bir piksel değeri ise kırmızı (red-R), yeşil (green-G) ve mavi (blue-B) renk değerlerinin toplamından oluşmaktadır. 0-255 arası değer alan bu RGB değerlerinde “0” değerine yaklaştıkça rengin en koyu halini almakta, 255 e yaklaştıkça da rengin en açık tonuna yaklaşmaktadır.

Gri değer aralıkları: $G=\{0,1,2,\dots,255\}$ şeklinde ifade edilir. Bunun anlamı şudur: Bir gri tonlu görüntüde 256 tane farklı gri ton değeri daha doğrusu gri değer bulunabilir. Burada 256 gri değer bir byte olarak tanımlanabilir. 0 gri değeri kural olarak beyaz renge, 255 gri değeri ise siyaha karşılık gelir. Bu değerler arasında ise gri tonlar oluşur. 8 bitlik kırmızı, yeşil ve mavi değerleri kullanılarak, her bir pikselin ortalama gri değeri aşağıda verilen formül ile hesaplanır.

$$\text{Gri}(i, j) = \frac{k(i, j) + y(i, j) + m(i, j)}{3} \quad (1)$$

Burada, i satır numarasıdır, j ise sütun numarasıdır; k, kırmızı değeri; y, yeşil değeri, m ise mavi değerini gösterir. Gri ise ilgili pikselin gri değeridir. Test görüntüleri sırası ile Şekil 2.4, Şekil 2.5, Şekil 2.6, Şekil 2.7 ve Şekil 2.8’de bir resmin bit değerlerindeki değişiminin görüntüsü gösterilmiştir.



Şekil 2.4. Orijinal resmin görüntüsü



Şekil 2.5. 256 bit değerindeki resmin görüntüsü



Şekil 2.6. 16 bit değerindeki resmin görüntüsü



Şekil 2.7. 4 bit değerindeki resmin görüntüsü



Şekil 2.8. 2 bit değerindeki resmin görüntüsü

2.2.5. Ön işleme aşamaları

Görüntü tarandıktan ve sayısallaştırıldıktan sonra karakter tanıma işlemi için, genel olarak gürültüleri temizlemek için görüntünün ön işlem aşamalarından geçirilmesi gerekmektedir. Sayısal görüntüde ciddi bozulmalara, sonuç olarak da belirsiz özelliklere ve zayıf tanıma oranlarına yol açabilecek istenmeyen bir gürültüyü silmek için ya da karakterleri yumuşatmak için ön işlemler kullanılabilir. Bu ön işlem aşamalarının içerisinde karakterlerin çerçevesi, eşikleme, genişletme, normalize etme, inceltme (iskelet çıkarma) ... gibi adımlar bulunmaktadır.

2.2.6. Eşikleme

Eşikleme işlemi resmin ikili görüntüsünü elde etmek için kullanılır. İstenmeyen arka plan resimleri, bitişik karakterler ve gürültüler karakter tanıma için sorun teşkil etmektedirler. İkili seviyeye indirgenen görüntü, bilgi karmaşıklığını azaltmak için bazı bilgilerin budanmasına rağmen, nesnenin şekli, pozisyonu ve piksel sayısı gibi gerekli bilgileri içerir. Görüntüdeki piksel değerleri 0-255 arası değerlere sahiptirler. Bu piksellerin içerisinde, kameradan kaynaklanan gürültü olabilir. Çoğu karakter tanıma sistemleri gri seviyelerini siyah ve beyaz (0,1) olmak üzere iki seviyeye indirirler. Bir eşik değerine göre bu indirgemeye ikili hale getirme adı verilir [17]. Arka planın değerini '0' değeri verilen piksel, karakter resimlerinin değerini ise '1' değeri verilen piksel temsil etmektedir. Belirli bir sınırın altında kalan piksel '0', o sınırın üzerinde olan piksel ise '1' değerini almaktadır.

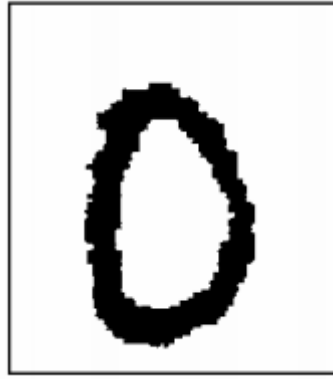
2.2.7. Gürültü temizleme

Görüntüler ikili düzene indirildikten sonra oluşan gürültülerin giderilmesi görüntünün sonraki ön işlem adımlarına katkı sağlamaktadır. Gürültülerin genellikle oluşma nedeni görüntünün arka planında bulunan kalem hatalarından ya da görüntünün çözünürlüğünün düşük olmasından meydana gelmektedir. Gürültü piksellerinin giderilmesi amacıyla kullanılan yöntemlerden birisi 5x5 boyutlarında bir filtre elemanı kullanmaktır. Maskeleye yöntemi ile gerçekleştirilen filtreleme işleminde 5x5 maskeleye elemanı, karakter alanı içerisindeki bütün piksel

noktalarına uygulanır [18]. Maske eleman değerleri 1 olarak seçilir. Komşuluk sayısı eşik değerinin altında olan piksellerin silinmesi sonucu gürültüyü oluşturan pikseller silinmektedir. Şekil 2.9 ve 2.10'da gürültülü bir yazı görüntüsünün üzerinde filtreleme işleminin sonucu görülmektedir.



Şekil 2.9. Resmin gürültülü hali



Şekil 2.10. Resmin gürültüden arındırılmış hali

2.2.8. Genişletme

Resim içerisinde oluşabilecek kopuklukları ve küçük boşlukları gidermek amacıyla kullanılır. Genelde inceltme ile birlikte kullanılır. Bunun amacı inceltme sonucunda oluşacak sonucun hatalı olmasını engellemektir. Örnek olarak Şekil 2.11'de dört yönde komşulukları ele alınarak merkezdeki boş pikselin doldurulmasına yardımcı olan maske gösterebilir.

| | | | | | | | |
|--|---|---|---|--|---|---|---|
| | | 1 | | | | 1 | |
| | 1 | 0 | 1 | | 1 | 1 | 1 |
| | | 1 | | | | 1 | |

Şekil 2.11. Maske örneği

2.2.9. İnceltme

İnceltme karakter tanıma işleminin daha kolay ve hızlı bir hale getirmek için kullanılır. Ayrıca görüntü işlenirken hataların en aza indirilmesini sağlar. İnceltme işlemi için Zhang-Suen [19,20], Stentiford inceltme algoritmaları örnek gösterilebilir.

2.2.10. Normalize etme

Normalizasyon genelde karakter geometrisindeki değişim olarak kabul edilebilir. Karakterin pozisyonu, konumu ya da boyutundaki değişim olarak kabul edilebilir. Örneğin 12x7 ve 14x8 boyutlarında bir karakterin 20x20 bir alana taşınması bir normalizasyon (ölçekleme) işlemidir [21,22].

2.2.11. Bölütleme (Segmentasyon)

Metin halinde bulunan karakterler üzerinde işlem ya da modelleme yapabilmek için karakterlerin hem dökümandan hem de birbirlerinden ayrılması gerekmektedir. Bu aşamada önce her bir satır daha sonra da her bir karakter birbirinden ayrılır. Karakterleri birbirinden ayırırken genelde görüntünün renk, desen, hareket gibi çeşitli özelliklerinden yararlanılabilir. Bölütleme aşamasında yatay ya da dikey histogram eğrilerin bulunması (projection), bölgesel komşuluklara göre tarama (region growing) ... gibi bilinen segmentasyon metotları bulunmaktadır. Genel olarak ise segmentasyon aşamasında piksel histogramları kullanılır. Giriş imgesindeki aynı rengi temsil eden pikseller toplanır ve toplam sayılara göre histogramlar oluşturulur. Bu histogramlara dayanarak 0 ve 1 arasındaki sınır değerleri belirlenir ve bu sınır da genelde histogramdan iki zirve arasındaki aralıktan seçilir [4]. Bu aşama görüntü işleme aşamalarında en zor olan bölümlerden biridir. Çünkü karakterler birbirine bitişik, aralarındaki mesafe az ya da düzensiz olabilir. Eğer sistem harfler konusunda yeterince ayrıncılığa sahip değilse bu harflerin ayırt edilebilmesi zordur [23].

2.2.12. Özellik Çıkartma

OCR uygulamalarında en önemli özelliklerden biri özellik çıkarma aşamasıdır. Özellik çıkarma metotlarında farklı teknikler mevcuttur. Bu teknikler karakterlerin doğru bir şekilde hangi sınıfa ait olduğunu belirlemesini sağlar. Özellik çıkarımında bir tür boyut daraltma durumuna gidilir. Genellikle karakter tanıma çalışmalarında histogram özelliklerinden yararlanılır [24].

Histogramların özellikleri, gri tonlu görüntü üzerinden, histogram fonksiyonu kullanılarak elde edilirler. Bir görüntüde yer alan gri tonlara ait piksel seviyesi histogram fonksiyonun oluşmasını sağlar. Gri tonlara ait piksel değerlerinin oluşma sıklığını tanımlayan bir olasılık dağılım fonksiyonu oluşmaktadır [25].

Bir görüntüde ortalama değer, standart sapma görüntüdeki gri değerlerin dağılımına ilişkin basit ölçütlerdir. Aynı şekilde görüntü histogramı da görüntüde piksellerin gri değerlerine ilişkin bağıl sıklık ölçütünü oluşturur.

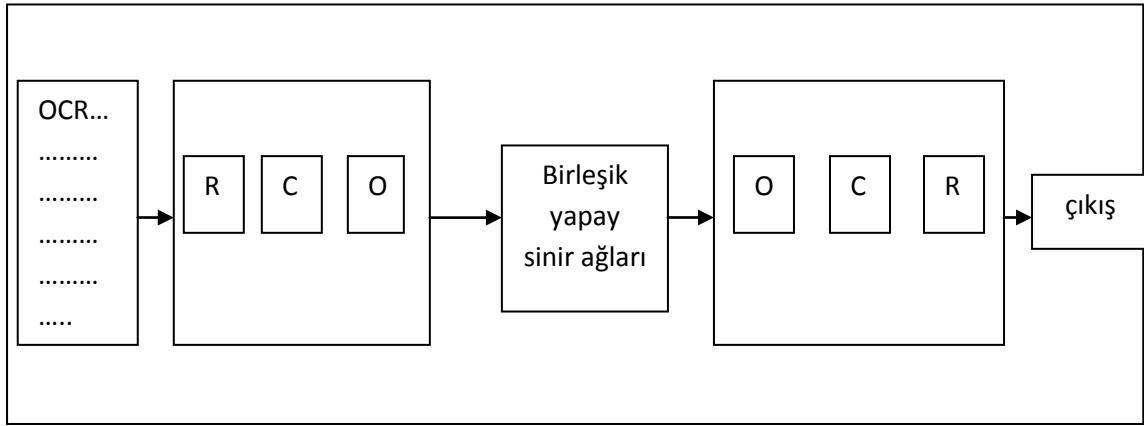
2.2.13. Karakter Tanıma

Örüntü tanımanın bir uygulaması olan karakter tanıma alanında uzun yıllardır çalışmalar sürmektedir. Araştırmacılar insanlar gibi örüntüleri de ayırt etmek için çeşitli çalışmalar yapmışlardır. Karakterler, semboller, resimler, üç boyutlu fiziksel objeler örüntü tanıma konularına örnek gösterilebilirler.

Karakter tanıma işlemi formun taranması, el yazısı ve makine çıktısı ile basılmış öğelerin formdan ayrılması işlemlerinden oluşmaktadır. Karakter tanıma, basılı karakter ve el yazısı tanıma olmak üzere iki ayrı kategoride incelenebilir. Karakterler, her yazı sisteminde farklılıklar gösterdiklerinden her biri için farklı karakter tanıma sistemine ihtiyaç duyarlar.

Bu çalışma kapsamında, normalize edilmiş karakter görüntüleri bir matriste tutulmaktadır. Karakter görüntüsüne ait matristen çıkarılan şekiller veri tabanında bulunan karakter matrislerinin şekilsel özellikleri karşılaştırılarak hangi karakter

olduđuna karar verilir. Yüksek gürültü ve hata toleransından dolayı karakterlerin sınıflandırılması ve tanınması için genellikle yapay sinir ađları kullanılır. Bu çalışmada daha iyi bir tanıma işlemi gerçekleştirmek için Birleşik Yapay Sinir Ađları kullanılmıştır. İkili matris formuna dönüştürülen her bir karakter resmi öncelikle bir diziye atılır ve tüm karakterler sırasıyla, eğitilmiş Yapay Sinir Ađı ağırlıkları işleme konularak tanınır ve harf resimleri yazıya çevrilir. Şekil 2.12’de basit bir şekilde karakter tanıma örneđi gösterilmiştir.



Şekil 2.12. Karakter tanıma örneđi

2.3. Mobil Cihazlarda Görüntü İşleme

Günümüz kişisel bilgisayarlarda görüntü işleme uygulamaları önemli bir yere sahiptir. Teknolojinin gelişmesine paralel olarak mobil cihazların kullanımı da yaygınlaşmaktadır. 3G teknolojinin de yaygınlaşması ile birlikte mobil cihazlarda internet erişim hızı da gittikçe artmaktadır. Kişisel bilgisayarlarla yapılabilen işlemlerin büyük bir bölümü mobil cihazlarla da yapılabilir hale gelmiştir. Mobil cihazların işlemci güçleri ve depolama kapasiteleri arttığı için bu cihazların işlem yapabilme kabiliyetleri neredeyse kişisel bilgisayarlarla aynı seviyeye gelmiş durumdadır. Dolayısıyla görüntü dosyaları gibi yoğun veri içeren uygulamaların çalıştırılabilmesi için gereken altyapı oluşturulmuştur.

Günlük hayatta kullanılan mobil cihazlarla birlikte gelen görüntü işleme uygulamaları sınırlı görevleri yerine getirebilmekte ve eğlence amaçlı kullanımın

ötesine geçmemektedir. Mobil cihaz kamerası ile elde edilen bir görüntü kolaylıkla işlenebilmekte ve görüntü işlemede kullanılan bütün teknikler kolaylıkla uygulanabilecek düzeye gelmiştir. Bu görevlerin yerine getirilmesi için de ekstra yazılımlara duyulan ihtiyaç her geçen gün artmaktadır. Dijital görüntüler, gündelik hayatın her alanında olduğu gibi mobil cihazlarda kullanımı en üst seviyede olacağı göz ardı edilemez. Ülkemizde ve dünyada satışlarında büyük bir artış yaşayan mobil cihazlar ile yapılacak olan sayısız uygulamaların yanında görüntü işleme uygulamaları en büyük yere sahiptir. Örneğin, nesne tanıma yazılımların temellerinde görüntü işleme teknikleri yer almaktadır.

Kişisel bilgisayarların donanımsal alt yapısı, görüntü işleme algoritmalarının gerçekleştirilmesinde son derece başarılıdır. Fakat mobil cihazların donanımsal alt yapısı kişisel bilgisayarlara nazaran daha düşük seviyede olduğundan bu tekniklerin gerçekleştirilebilmesi için en iyi algoritmaların kullanılması gerekmektedir.

2.4. Optik Karakter Tanıma

Optik Karakter Tanıma (OCR), taranmış ya da kamera ile çekilmiş görüntülerin, el yazılarının veya çıktısı alınmış metin bilgilerinin elektronik ortama dönüştürülmesi ve sayısallaştırılması işlemidir. OCR ile basılı karakterlerin, el yazısının ya da işaret bilgilerinin kolaylıkla okunup bilgisayar ortamına aktarılması sağlanabilmektedir. Bu sayede kullanıcı ile bilgisayar arasındaki iletişim ve etkileşim arttırmaktadır [4]. Optik karakter tanıma sistemlerinde, sistemin tanınması dışarıdan verilecek olan girdilere göre belirlenmektedir. Bunlar çevrimiçi tanıma sistemleri ve çevrimdışı tanıma sistemleridir [26].

Çevrimiçi tanıma sistemleri karakterlerin yazılması anında çalışan ve işleyen bir tanıma yöntemidir. Yaygın kullanımında karakterler dijital bir ekrana elle ya da çubukla yazılır ve bilgisayar sistemi, çubuğun üzerinden geçtiği piksel noktalarını boyar. Bu yöntemde önemli olan bilgiler yazının yazılma hızı, harflerin yönü, harflerin sırası ve kullanılan aletin hassasiyetidir. Günümüzde, taşınabilir el bilgisayarları ve elektronik tabletler gibi sistemlerde bu yöntem kullanılmaktadır.

Çevrimdışı tanıma sistemleri ise, kağıt üzerindeki yazıların tanınabilmesi için kullanılan yöntemlerdir. Öncelikle belgenin ya da dokümanın sayısallaştırılması gerekmektedir. İlk aşama belgenin analizi ile başlar. Sonraki aşamalarda ön işleme, dilimleme, öznitelik çıkarımı, harflerin tanınması ve son işleme aşamasıdır. OCR sistemleri tek bir parçadan değil, çeşitli farklı sistemlerin bir araya gelerek birlikte çalışmasıyla oluşur [5].

OCR sistemleri, günümüzde artık daha hızlı, daha ucuz ve daha güvenilir bir sistem olmuşlardır. Bugün kullanılan OCR sistemleri ile artık daha fazla yazı tipi tanınabilmektedir.

OCR sistemleri üzerine yapılan fazla sayıda araştırmalar ve daha ucuz elektronik bileşenler, bu sistemlerin yolunu açmışlardır. Bu sistemlere olan talep hala devam etmektedir. Bu nedenle, araştırmacılar bu sistemleri daha da hızlandıracak ve hata oranlarını düşürecek yeni yöntemlere ihtiyaç duymaktadırlar.

2.4.1. Optik karakter tanımının tarihçesi

1920'li yıllarda OCR teknolojisi alanında ilk patent sahibi Alman bilim adamı Gustav Tauschek'tir. Bu yıllarda optik ve mekanik teknolojiler kullanılarak, karakter maskeleri ve ışık kullanılarak da karakterler tanınmıştır. Sistemde tanınacak karakter ve şablon üst üste geldiği takdirde ışık geçmez ve böylece fotoğraf dedektör ile algılanamazdı. Bu prensibin kullanımı günümüzde de kullanılmaktadır [4].

İlk modern bilgisayar destekli, 23 harfi tanıyabilen OCR sistemi GİSMO, 1951'de David Shepard tarafından Amerika'da geliştirilmiştir [27]. 1950'li yılların ortasından başlayıp günümüze kadar donanım cihazı ve yazılım paketi olarak ticari OCR sistemleri geliştirilmeye devam etmektedir. OCR üzerine yapılan çalışmalar oldukça fazladır [24,28]. OCR ile ilgili çalışmalar günümüzde de devam etmektedir. Karmaşık zeminlerde ve çevre koşulların da, video çerçeveleri üzerindeki yazıların tanınması v.b. konularda çalışmalar devam etmektedir. Basılı ve el yazılarının bilgisayar ortamına dönüştürülmesini hızlandırması OCR'ı bilgisayar bilimcileri, mühendisler ve farklı disiplinlerdeki pek çok kişinin ilgisini çeken bir konu haline

getirmiştir. Bu tarz yetenekleri sayesinde OCR, insan-bilgisayar etkileşimi, örüntü tanıma, görüntü işleme, metin-konuşma çevrimi, dil işleme, metin madenciliği, yapay zekâ gibi pek çok bilgisayar uygulama alanına yayılmıştır. Ayrıca, OCR ile donatılmış bilgisayar sistemlerinin hızlı veri girişi ve metin işlemeyi izin verme, bazı olası insan hatalarını azaltma ve hızlı bilgi bulmayı mümkün kılma gibi avantajları vardır.

1999 yılında, Gorski ve arkadaşları, Fransa, İngiltere ve Amerika’da düzenlenen el yazısı veya basılı çekleri işlemek için bir çek tanıma sistemini gerçekleştirdiler [29]. Ni ise 2007 yılında, Amerikan posta servislerindeki adresleri sıralamak için posta kodlarını okuyan çok katmanlı geri-beslemeli yapay sinir ağı kullanan bir OCR sistemi geliştirdi [30]. Mani ve Srinivasan 1997 yılında yapay sinir ağlarını kullanan bir OCR sistemi önerdi [31]. Ahmed 1994 yılında, daha sonra Alshebeili ve arkadaşları ise 1997 yılında Arap alfabesini tanıyan bir OCR sistemi geliştirdi [32]. Leung ve daha sonra Zhang ve arkadaşları 2002 yılında, el yazısı Çin karakterlerini tanıyan çalışmalar geliştirdiler [33,34]. Inoue ve arkadaşları ise 1998 yılında matematiksel formüller içeren Japonca dokümanlara ait taranmış sayfa görüntüsündeki Japonca metinleri ve matematiksel formülleri ayrı ayrı tanıyan bir OCR sistemi geliştirdiler [35]. Bu sistemleri takiben Singh ve arkadaşları YSA yardımı ile Devanagari (Hint) alfabesini tanıyan bir optik karakter tanıma sistemi geliştirdi [36].

Son yıllarda, mobil telefon ve diğer cihazların kameralarından alınan görüntülerdeki kelime ve metinleri okuyan ve farklı dillerde çevirilerini yapan çalışmalar bulunmaktadır. Koga ve arkadaşları mobil bir cihaz kullanılarak resmi çekilmiş Japonca karakterleri tanıyan bir OCR sistemi geliştirdi [7]. Park ve Kwon, masaüstü OCR sistemlerinin mobil platformlar üzerinde çalışabilecek gömülü versiyonunu geliştirdi [28]. Rodriguez ve arkadaşları afiş ve tabelalardaki cep telefonu kamerası kullanılarak çekilmiş JPEG formatındaki resimlerindeki metinleri İngilizceden İspanyolcaya çeviren bir mobil uygulama geliştirdi [8].

Genel olarak incelendiğinde; İngiliz, Çin, Arap.. vb. alfabeleri için çok sayıda optik karakter tanıma çalışması bulunmaktadır [37-39]. Ancak, Türkçe de bu dillerin

aksine karakter tanıma çalışmaları daha sınırlı sayıdadır. Bunun nedenleri ise karakter tanıma çalışmalarına geç başlanması ve alfabemizde bazı özel karakterlerin (ö,ü,ş... vb.) bulunmasıdır [40].

2.4.2. OCR kullanım alanları

Genel amaçlı kullanılan OCR Sistemlerinin hedefi, taranmış ya da fotoğrafı çekilmiş resim formundaki bir dökümanı kendisine karşılık gelen tekrardan kullanılabilir sembolik bir gösterime dönüştürmektir. Bu güne kadar birçok alanda OCR teknolojileri kullanılmıştır. Bu teknolojinin kullanımının artmasının başlıca sebebi binlerce sayfalık verinin birkaç dakika içerisinde bilgisayarlara ya da veri tabanlarına aktarabilmesidir. Ayrıca çok sayıda ticari ve bilimsel amaçlı OCR sistemleri geliştirilmiştir. Bunlara örnek olarak AIOCR, OmniPage, Fine Reader gösterilebilir.

OCR teknolojisinin sıklıkla kullanıldığı alanlar aşağıdaki gibi örneklendirilebilir.

Bankacılık: Bankacılıkta insan müdahalesi olmadan kontrol işlemi için OCR sistemi kullanılmaktadır. Yeni bir makine eklendiğinde bunun onayı için yada üzerinde yazı yazan mektup yada değerli belgelerin tarandıktan sonra işlem yapılır. Bu teknoloji, neredeyse basılı kontroller için mükemmeldir. Özellikle çekler elden verildiğinde imza ve yazı kontrolü yapmada kolaylık sağlar. Böylece kısa sürede doğrulanır ve zaman kaybı azalır.

Hukuk Sektörü: Hukuk sektöründe, önemli bir işlem olan kağıt dokümanların tanınip dijital ortama aktarılması için kullanılmaktadır. Bu işlemin amacı kağıtların bulunduğu kullanım alanlarından tasarruf etmek ve gerek kağıt dosyaların gerekse kutuların kalabalığını ortadan kaldırmaktır. Bu nedenle taranan belgeler dijital bir veri tabanına aktarılmıştır. OCR ile belgeler arasında metin aranabilir, böylece zamandan da tasarruf sağlanır.

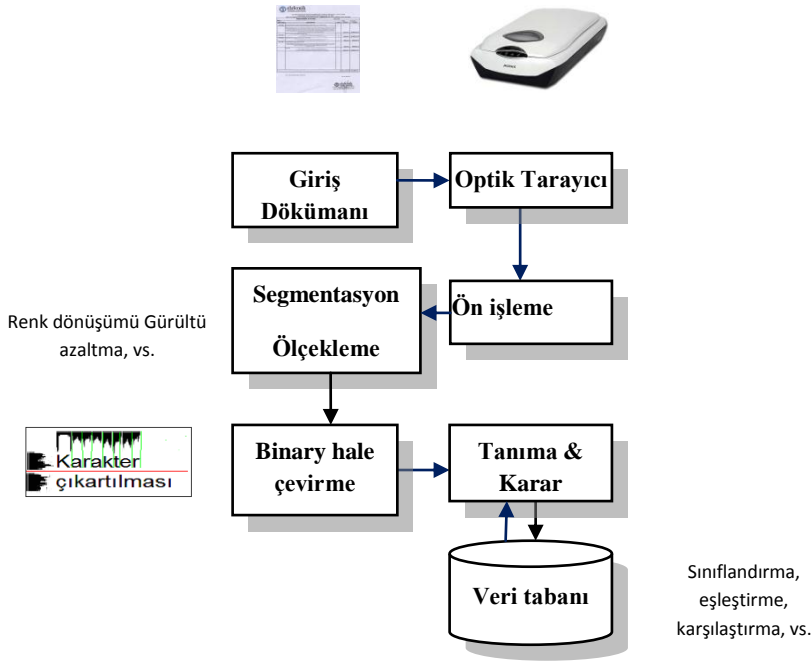
Sağlık ve Personel Kayıt: Sağlık sektöründe evraklar ve raporlar arttıkça OCR teknolojisinin kullanımında artış görülmüştür. Sağlık çalışanları her zaman sigorta

formları dahil her hasta için geniş hacimli formların yanı sıra, genel sağlık formları ile uğraşmak zorundadır. Tüm bu bilgilerin düzenli kontrolü için gerektiğinde ulaşılabilir olması için bir elektronik veri tabanı kullanılması kullanıcıya kolaylık sağlamaktadır. OCR tarafından desteklenen form işleme araçları, formlardan bilgi elde etmek ve veri tabanlarına eklemektedir, Böylece her hastanın verileri hemen ve düzenli bir şekilde kaydedilir. Sonuç olarak, OCR sistemleri sağlık personelinin her hastaya en iyi hizmeti sunmasına yardımcı olur olur.

Sipariş İşlemleri: Sipariş sırasında özellikle barkod tanıma kolaylık sağlamaktadır. Barkodlu ürünlerle bilgilerin ilişkilendirilmedi açısından gerekli olmaktadır. Barkodlar sayesinde bilgisayara otomatik veri girişi hızlı bir şekilde sağlanmaktadır. Ya da barkod bilgisi üzerinden tarama yapılabilmektedir.

2.4.3. OCR sisteminin genel yapısı

Optik Karakter Tanıma işlemi, basılı belgelerin okunarak bilgisayar metinlerine dönüştürülmesi işlemidir. Şekil 2.13’de basit bir OCR sisteminin genel yapısı gösterilmektedir. Optik bir cihaz aracılığı ile taranan basılı veya el yazısı dokümana ait karakterler kümesi sayısal görüntüye dönüştürülür. Karakter şekillerinin karmaşıklığı, gürültü, kelime haznesinin kapsamı, sayısallaştırılmış karakterin çözünürlüğüne etki eden matris boyutu, karakter tanımanın hız ve doğruluğunu etkileyen faktörler arasındadır. Basılı veya el yazısı verilerin bulunduğu bölgeyi belirleyerek başlayan tipik bir OCR sisteminde sonraki işlem adımları sırasıyla belge tarama, ön işleme, bölütleme (segmentasyon), siyah-beyaz (binary) hale dönüştürme ve karakter tanımadır.



Şekil 2.13. OCR sisteminin genel yapısı

Fotoğraf ve taranmış dokümanlar genellikle RGB renkli resimlerdir. RGB değerlerinin kullanılması, görüntü analizi açısından karmaşıklığın artmasına neden olmaktadır. Bu nedenle renkli resimden gri seviye resme transfer karmaşıklığı sorunlarının çözümü için ve bir standartlaşma sağlamak amacı ile dönüşüm yapılmaktadır. Resim gri renk formatına dönüştükten sonra yazının zeminden ayrılması için eşikleme (ikili resim) yapılması gerekir. Böylece yazının arka planı çıkarılabilir. Elde edilen ikili görüntüde bir takım gürültüler olması tanıma kalitesini düşürmektedir. Bu olumsuzluğu gidermek için filtreleme ve gürültü azaltıcı algoritmalar uygulanır.

Resim üzerindeki satırların bulunması için yatay satırlara ayırma (yatay projeksiyon) gerçekleştirilir böylece resim satırları içeren küçük resimlere dönüşür. Daha sonra dikey satırlara ayırma (dikey projeksiyon) yapılarak karakterler bölünür. Elde edilen karakterler farklı boyutlarda olabileceğinden dolayı sınıflandırma algoritmasında kullanımı problem teşkil edebilir. Bu nedenle sınıflandırmadan önce görüntüye ölçeklendirme yapılır.

OCR uygulamalarında, özellik çıkartma önemlidir ve her bir özellik çıkarmanın, kendine özgü güçlü ve zayıf yanları bulunan farklı teknikler mevcuttur. Karakterleri doğru bir şekilde hangi sınıfa ait olduğunu belirleyebilen özellikleri çıkartmak en uygun olan yöntemdir. Karakterler doğru parçalarından oluştuğuna göre, karakterleri tanımak için pek çok farklı özellik çıkartılabilir. Son adımda karakter görüntüsüne ait özellik vektörleri YSA, HMM vb. sınıflandırma algoritmaları ile sınıflandırılır.

BÖLÜM 3. YAPAY SİNİR AĞLARI

3.1. Yapay Sinir Ağlarına Giriş

Bu bölüm, Yapay Sinir Ağları teknolojisine ait temel bilgileri kapsamaktadır. Günümüzde teknolojinin kullanımının artması ve teknolojik gelişmelerin hızlanması, insanların kendisini tanımaya yönelik çalışmalarında önemli aşamalar kaydetmesini sağlamıştır. Yapay Zeka (Artificial Intelligence) kavramının ortaya çıkması ile insanın en önemli özelliklerinden olan düşünebilme ve öğrenebilme yetenekleri en önemli araştırma konuları durumuna gelmiştir. Yapay Zeka terimi Stanford Üniversitesinde Profesör olan John McCarthy tarafından ortaya atılmıştır. Özellikle son zamanlarda, teknoloji kullanımının hızla yaygınlaşması sonucunda yapay zeka üzerine yapılan çalışmalarda artış görülmektedir.

Yapay Sinir Ağları (YSA) teknolojisi, yapay zeka çalışmaları kapsamında ortaya çıkan ve yapay zeka çalışmalarına destek sağlamakta olan farklı alanlardan bir tanesidir. Yapay zeka alanının bir alt dalını oluşturan yapay sinir ağı teknolojisi aynı zamanda öğrenebilen sistemlerin temelini oluşturmaktadır. YSA'lar, insan beyninin temel işlem elemanı olan nöronunu şekilsel ve işlevsel olarak basit bir şekilde taklit etmek üzere oluşturulan programlardır diyebiliriz. Bu programlar, insan beyninin özelliklerinden olan öğrenme yolu ile yeni bilgiler oluşturma, türetme ve keşfedebilme gibi yetenekleri herhangi bir yardım almadan otomatik olarak gerçekleştirmek amacı ile çalışmaktadır. Bu yeteneklerin geleneksel programlama yöntemleri ile gerçekleştirme oldukça zor olacaktır. Yapay sinir ağlarının, bu nedenle, programlanması zor olan olaylar için geliştirilmiş, sabit olmayan bilgi işleme ile ilgilenen bir bilim dalı olduğu söylenebilmektedir [10].

Bu bölümde, YSA hakkında kısaca temel bilgiler anlatılmıştır. Bundan sonraki kısımda yani 3.2’de YSA’ nın tarihçesine yer verilmekte ve daha sonraki kısım 3.3’de YSA’ nın genel tanımları ve özellikleri anlatılmaktadır. Diğer kısımlarda ise sırası ile 3.4’te Yapay Sinir Hücresi ve Elemanlarının Yapısı hakkında ve 3.5’te YSA’ nın yapısı hakkında kısaca bilgiler verilmiştir. Ve son olarak da 3.6’daki kısımda YSA’ nın avantaj ve dezavantajları tablo halinde verilmiş olup, 3.7’deki kısımda da YSA’ ların uygulama alanlarına değinilmektedir.

3.2. Yapay Sinir Ağları Tarihçe

YSA’nın tarihçesi, insanların nöro-biyoloji konusuna ilgi duymasıyla ve elde ettikleri bilgileri bilgisayar bilimine uygulamaları ile başlamaktadır. YSA ile ilgili çalışmalar genelde 1970 yılı öncesi ve sonrası diye ikiye ayrılmaktadır [41]. Bunun nedeni, 1970 yılında YSA biliminin tarihinde önemli bir dönüm noktasına gelinmesi ve o zamana kadar olmaz diye düşünülen birçok sorunun çözülüp, yeni gelişmelerin başlamış olmasıdır.

İnsan beyninin fonksiyonları ve nasıl çalıştığı uzun yıllar araştırılan bir konu olmuştur. 1940’dan önceki yıllarda da bazı bilim adamlarının yapay sinir ağı kavramı üzerinde çalıştıkları bilinmektedir. 1980’li yıllarda beynin fonksiyonları hakkında bilgi veren ilk eserler yayınlanmaya başlanmış [10].

1943 yılında Hebb [42], McCulloch [43] ve Pitts gibi bilim adamları yapılan araştırmaları mühendislik alanlarına kaydırmaya ve günümüzdeki YSA’ nın temellerini oluşturmaya başlamışlardır. Bu bilim adamları ilk yapay sinir hücresinin yapısını oluşturmuşlardır ve yapay sinir hücreleri ile her türlü mantıksal ifadeyi formüle etmenin mümkün olduğunu üzerine araştırma yapmışlardır. Hücrelerin birbirleri ile paralel çalışması gerektiği fikrini ortaya atarak öğrenme kurallarını belirlemeye başlamışlardır.

1949 yılında Donald Hebb “Hebbian öğrenme kuramını” geliştirmiştir. Bu kuram, yapay hücrelerden oluşan bir yapay sinir ağının değerlerini değiştiren bir öğrenme

kuralıdır [42]. Ayrıca bu kuramın işleyişi; “A hücrenin bir aksonu, B hücreni uyaracak ve tekrarlı veya sürekli olarak tetikleyecek kadar yakında ise, hücrelerde B’ yi tetikleyen A’nın etkinliğini artıracak bir büyüme işlemi veya metabolik değişiklik olur” şeklindedir [43]. “Hebbian öğrenme” kuralı denilen bu kural günümüzde de birçok öğrenme kuralının temelini oluşturmaktadır. İlk neuro - bilgisayar 1951 yılında üretilmiştir [10].

Marvin Minsky, Hebb kuramı ile birlikte makroskobik zeka kavramını ortaya atmış ve uzman sistemlerin doğmasını bu şekilde gerçekleştirmiştir. “Yapay Zeka” kavramı ilk olarak 1956 yılında “ortaya atılmıştır. Bu kavram, araştırmacılar ve bilimle ilgilenen insanlar tarafından kabul görmüş ve araştırılmaya başlanmıştır. Bu bilim adamları, öğrenme ve zeka konusunu bilgisayar simülasyonlarında nasıl kullanabileceklerini 1956’ da düzenlenen ilk yapay zeka konferansında tartışmışlardır. İlk yapay zeka çalışmaları da YSA’ na pek değinmemiştir. Herkesin ilgi odağı olan konular yapay zeka ve nöro-bilgisayar olmuştur. Bundan dolayı YSA popülerliğini yitirmeye başlamıştır. 1960’lı yıllarda Grosberg, Kahonen, Widrow, Fukushima vb. gibi bilim adamları ilginin tekrar YSA üzerinde olmasını sağlamak için tekrar konunun üzerine gitmeye başladılar [44].

Silikon teknolojinin geliştirilmesi ile bu çalışmalar 1960’lı yıllarda oldukça önemli gelişmelere neden oldu. Özellikle Rosenblatt tarafından geliştirilen algılayıcı model (perceptron) YSA tarihinde önemli bir gelişmeye öncülük etmiştir. Çünkü bu model, daha sonraları geliştirilecek ve YSA’nda devrim niteliğinde olacak olan çok katmanlı algılayıcıların temelini oluşturmaktadır. Benzer şekilde Widrow ve Hoff “ADALINE” (Adaptive Linear Neuron) modelini ortaya attılar [45]. 1959’ da Bernard Widrow, “ADALINE” (Adaptive Linear Neuron) olarak adlandırılan bir adaptif lineer elemanı geliştirmiştir. Bu adaptif model, Rosenblatt’ın algılayıcı modeli ile aynı niteliklere sahip bir model olup sadece öğrenme algoritması daha da gelişmiş bir modeldir. Adaline ve iki tabakalı biçimi olan “MADALINE” (Multiple Adaline); ses tanıma, hava tahmini, karakter tanıma ve adaptif kontrol gibi çok çeşitli uygulamalar için kullanılmıştır. Widrow, telefon hatları üzerinde yansımaları elemine etmeye yarayan adaptif filtreleri geliştirmede, adaptif lineer eleman

algoritmasını kullanmıştır. Bu çalışma ile birlikte YSA ilk defa gerçek bir probleme uygulanmıştır.

1969-1982 yılları YSA için karanlık bir dönemi temsil etmektedir. Çünkü 1969 yılında Minsky ve Papert' in yapmış olduğu çalışma ile yapının XOR lojik problemlerine çözüm getiremediği ispatlanmıştır. Bu gelişme YSA' na olan ilgiyi azaltmış ve karanlık dönemin başlangıcı olmuştur. 1974 yılında Werbos, Geriye Yayılım (Back Propagation) Algoritmasını tanıtmıştır. Bu gelişme, yapay sinir ağlarının yeniden bir canlanma dönemine geçişinin ilk adımı olarak sayılabilmektedir. Willshaw ve Malsburg 1976 yılında Öz denetimli Harita (Self Organized Feature Map) adında kümeleme algoritmasını kurmuştur. 1982 yılında Kohonen, öz denetimli harita kuramını beyindeki oluşumların karşılaştırmalı haritasını çıkarabilmek için ortaya koymuştur [46]. Hopfield, 1982 yılında moleküler biyolojiden beyin kuramcılığına geçiş yapan bir model geliştirmiştir. Bu model ile yapay sinir ağlarının genelleştirilebileceğini ve çözülmesi zor olan problemlere çözüm üretilebileceğini göstermiştir [47,48]. Ayrıca günümüzde, kendi adıyla anılan bir ağ yapısı mevcuttur ve bu yapı birçok alanda uygulanmaktadır.

1986 yılında, Rumelhart, çok katmanlı algılayıcı tipli geri yayılım algoritması denilen bir eğitim algoritması geliştirmiştir. İlk YSA konferansı 1987 yılında yapılmış ve sonrasında bu algoritmalarla ilgili uygulamalar artmaya başlamıştır. Carpenter ise Adaptif Rezonans Teorisini (ART) geliştirmiştir. Bu teori, öğretmensiz öğrenme konusunda zamanının geliştirilmiş olduğu en karmaşık yapay sinir ağı olmuştur [49].

Çok katmanlı algılayıcıların ortaya çıkması, yapay sinir ağlarının tarihsel gelişimi açısından oldukça önemli bir rol oynamaktadır. Tek katmanlı algılayıcının çözemediği XOR problemi çok katmanlı algılayıcıların bulunması ile çözülmüştür ve YSA'nın çalışmadığını söyleyen bütün tezler çürütülmüştür. Aynı zamanlarda, Parker ve Werbos tarafından da çok katmanlı algılayıcı ile ilgili olarak bazı çalışmalarda yürütülmüştür [50,51]. Çok katmanlı algılayıcılar sadece XOR problemine çözüm üretmekle kalmamıştır, aynı zamanda Hopfield ve Boltzman makinelerinin sınırlamalarını da çözmüştür [52].

Broomhead ve Lowe Radyal 1988’de, tabanlı fonksiyonlar modelini geliştirmişlerdir. Özellikle filtreleme problemlerine oldukça başarılı sonuçlar üretildi. Specht daha sonra bu ağların daha gelişmiş şekli olan Probabilistik Ağlar (PNN) [53] ve Genel Regrasyon Ağlarını geliştirdi [54]. Bu zamana kadar yapılan çalışmaların bazıları kronolojik olarak aşağıdaki tablodaki gibi listelenebilir [10,55]:

Tablo 3.1. YSA kronolojik sırası

| | |
|-------------|--|
| 1890-1920 | İnsan beyninin yapısının ve fonksiyonlarının incelenmesi ve bununla ilgili ilk yayının çıkarılmıştır. İnsan beyninin bileşenlerinin belirli bir düzenek ile sinir hücrelerinden (nöronlar) oluştuğu fikrinin benimsenmeye başlanmıştır. |
| 1940-1950 | Yapay sinir hücrelerine dayalı hesaplama teorilerin ortaya çıkmış ve eşik değer mantığına göre çalışan mantıksal devreler geliştirilmiştir. Biyolojik olarak mümkün olabilen öğrenme metodunun bilgisayarlar tarafından gerçekleştirilebilecek hale getirilmiştir. |
| 1949-1956 | Hebb Kuralının ortaya çıkışı ve “Yapay Zeka” kavramının ortaya çıkması [42] |
| 1956 - 1962 | Adaline ve Widrow öğrenme algoritmalarının ve Tek katmanlı algılayıcının geliştirilmesi ve Rochester, Holland, Habbit ve Duda ilk bilgisayar simülasyonunu Hebb’in öğrenme varsayımlarına dayalı sinir teorisine göre gerçekleştirdi. |
| 1967 - 1969 | Bazı gelişmiş öğrenme algoritmalarının geliştirilmesi [56] |
| 1969 | XOR problemlerinin ortaya çıkışı ve buna bağlı olarak tek katmanlı algılayıcıların XOR gibi problemleri çözme yeteneklerinin olmadığına gösterilmesi nedeniyle duraklama döneminin başlaması gerçekleşmiştir. |
| 1969 - 1972 | Doğrusal ilişkilendiricilerin geliştirilmesi |
| 1974 | Geriye yayılım modelinin (Çok katmanlı algılayıcının) ilk çalışmalarının geliştirilmesi |
| 1974 | Öğretmensiz öğrenmenin geliştirilmesi |
| 1976 | Öz denetimli harita yönteminin oluşması |
| 1978 | ART modelinin geliştirilmesi |
| 1982 | Kohonen öğrenmesi, SOM modeli geliştirilmesi ve Hopfield ağlarının geliştirilmesi sağlanmış ve çok katmanlı algılayıcının geliştirilmesi için çalışmalar başlamıştır. |
| 1985 | Genelleştirilmiş Delta Kuralının oluşması |
| 1986 | Geri yayılım algoritması |
| 1987 | YSA sempozyumu |
| 1988 | RBF ve PNN modellerinin geliştirilmesi |
| 1991 | GRNN modelinin geliştirilmesi |

3.3. YSA' nın Genel Tanımları ve Özellikleri

Yapay sinir ağı modellemesi, insan gibi canlı organizmalarda bulunan biyolojik sinir yapısından esinlenerek yapılmıştır. Kısaca, yapay sinir ağları insan beyninden esinlenerek geliştirilmiş olan bağlantılar aracılığı ile birbirine bağlanan ve her biri kendi belleğine sahip olan işlemci elemanlardan oluşan paralel ve dağıtılmış bilgi işleme yapılarına denilir [52].

Diğer bir tanımlama ise, yapay sinir ağları deneysel bilgiyi saklayan ve onu kullanmak için işe yarar hale getiren paralel dağılmış işlemcilerden oluşur. Bu tanımlamayı Haykin 1994'te yapmış olup, ayrıca YSA' ların iki yönden de beyine benzediğini belirtmiştir. Bu iki yön;

1. Bilgi, ağ yapısı tarafından bir öğrenme işlemi yoluyla kazanılır.
2. Nöronlar arasında bulunan ve sinaptik ağırlıklar olarak adlandırılan ağırlıklar bilgiyi depolamak için kullanılır [57].

Gerçek sinir ağlarıyla benzerlik oranları düşüktür. Gerçek beyin fonksiyonları hakkındaki bilgiler sınırlı olsa bile işleyişi ve fonksiyonel yapısı hakkında taklit örnek alınmaya yetecek kadar bilgi birikimi bulunmaktadır [21].

Yapay sinir ağları, bilgisayar sistemlerinin herhangi bir yardım almadan, otomatik olarak insan beyninin özelliklerinden olan öğrenme yolunu taklit ederek, yeni bilgiler oluşturabilmesi, yeni bilgiler türetebilmesi ve yeni bilgiler keşfedebilmesi gibi yetenekleri uygulamaya geçirmesidir [10]. Bu tür yetenekleri uygulamaya geçirirken, ağırlıklı bağlantılar aracılığıyla birbirine bağlanan ve her biri kendi belleğine sahip işlem elemanlarını kullanırlar [58]. Normal programlama yöntemleriyle programlanması çok zor veya mümkün olmayan olayları çözebilmek oldukça zor olmakta veya mümkün olmamaktadır. Bu nedenle, yapay sinir ağlarının, geliştirilmiş adaptif bilgi işleme ile ilgilenen bir bilgisayar bilim dalı olduğu söylenebilmektedir [10].

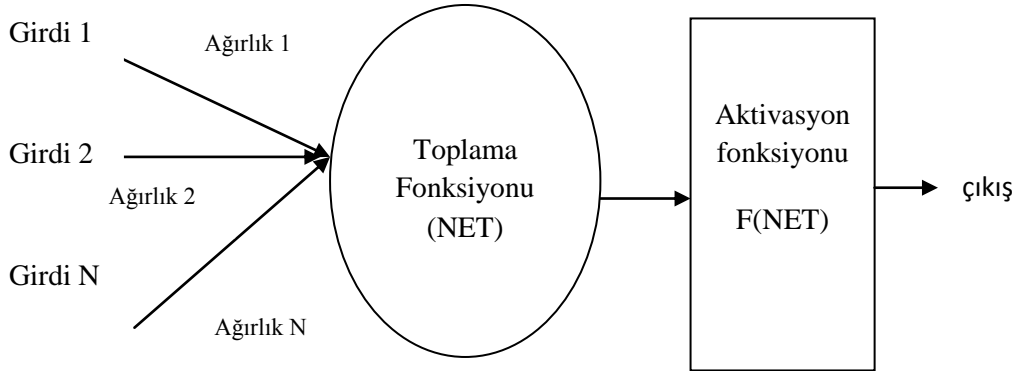
YSA genel olarak biyolojik sinir ağlarının yapısını örnek alarak benzer şekilde çalışmayı hedeflemektedir. Böylece canlıların sinir sisteminin modellenmesi sayesinde YSA biyolojik sinir sisteminin bilgi işleme üstünlüklerine sahip olmuştur. YSA' nın hesaplama gücünü, paralel dağılmış yapısından, öğrenme ve genelleme yeteneğinden aldığı söylenebilmektedir. Yapay sinir ağları, uygulanan ağ modeline göre değişik karakteristik özellikler göstermelerine karşın temel birkaç ortak özelliğe sahiptirler. Bunlar [55],

- Doğrusal olmayan yapıları modellenmesi
- Paralel dağılmış yapıya sahip olması
- Hata toleransına sahip olması
- Farklı problemler için uyarlanabilme yeteneği
- Genelleme yeteneği
- Öğrenme yeteneği

olarak örneklendirilebilmektedir.

3.4. Yapay Sinir Hücresinin Yapısı

Beynin bütün davranışlarını tam ve bütün olarak modelleyebilmek için fiziksel bileşenlerinin doğru olarak belirlenmesi ve modellenmesi gerekmektedir. Bu amaç ile çeşitli yapay hücre ve ağ modeli gerçekleştirilmiştir. Yapay sinir ağlarının çalışmasını sağlayan en küçük temel bilgi işleme birimi yapay sinir hücreleridir. Geliştirilen hücre modellerinde bazı farklılıklar olmakla beraber genel özellikleri aynıdır. Genel olarak bir yapay sinir hücre modelinde girdi, ağırlık, toplama fonksiyonu, aktivasyon fonksiyonu ve çıktı birimi bulunmaktadır. Şekil 3.1'de genel olarak yapay sinir hücresinin yapısı gösterilmiştir. Geliştirilen hücre modelleri birbirlerinden bazı farklılıklarla ayrılmakla beraber, genel özellikleri aynıdır.



Şekil 3.1. Yapay sinir hücresinin yapısı

Girdiler, nöronlara gelen verilerdir. Yani yapay sinir hücresinin girişlerini oluştururlar. Girdiler yapay sinir hücresine bir diğer hücreden gelebileceği gibi direkt olarak dış dünyadan da gelebilmektedirler. Yapay sinir hücresine gelen bilgiler, girdiler üzerinden çekirdeğe ulaşmadan önce geldikleri bağlantıların ağırlığıyla çarpılarak çekirdeğe iletilir. Bu sayede girdilerin üretilecek çıktı üzerindeki etkisi ayarlanabilmektedir.

Ağırlıklar, yapay sinir hücresine gelen bilginin önemini ve hücre üzerindeki etkisini gösterir. Ağırlıkların değerleri pozitif, negatif veya sıfır olabilir. Ağırlığı sıfır olan girdilerin çıkış üzerinde herhangi bir etkisi olmamaktadır. Ağırlıklar değişken ve ya sabit değerler olabilirler.

Toplama fonksiyonu, bir hücreye gelen net girdiyi hesaplayan bir fonksiyondur ve genellikle net girdi, girişlerin ilgili ağırlıkla çarpımlarının toplamıdır. Birleştirme fonksiyonu, ağ yapısına göre maksimum alan, minimum alan ya da çarpım fonksiyonu olabilir. Bu fonksiyon,

$$NET = \sum_{i=1}^n G_i W_i$$

şeklinde formülize edilir.

Aktivasyon fonksiyonundan çıkan NET toplam hücrenin çıktısını oluşturmak üzere aktivasyon fonksiyonuna iletilir. Aktivasyon fonksiyonu genellikle probleme uygun seçilmekle birlikte çoğunlukla doğrusal olmayan bir fonksiyon seçilir. Aktivasyon fonksiyonları sabit parametrelili ya da uyarlanabilir parametrelili seçilebilir. Aktivasyon fonksiyonu seçilirken dikkat edilmesi gereken bir diğer nokta ise fonksiyonun türevinin kolay hesaplanabilir olmasıdır. Geri beslemeli ağlarda aktivasyon fonksiyonunun türevi de kullanıldığı için hesaplamaların yavaşlamaması için türevi kolay hesaplanabilir bir fonksiyon seçilmesinde fayda vardır. Günümüzde en çok kullanılan çok katmanlı algılayıcı modellerde genel olarak sigmoid aktivasyon fonksiyonu kullanılır. Bu fonksiyon aşağıdaki formül ile gösterilmektedir.

$$F(\text{NET}) = \frac{1}{1 + e^{-\text{NET}}}$$

Çıktılar, aktivasyon fonksiyonundan çıkan değer nöronun çıktı değeri olmaktadır. Bu değer ister yapay sinir ağının çıktısı olarak dış dünyaya verilir ister tekrardan ağın içinde kullanılabilir. Nöronun bir çıktısı olmasına rağmen bu çıktı istenilen sayıda nörona bağlı olabilir. Bir hücrenin birden fazla girdisi olmasına rağmen tek bir çıktısı vardır.

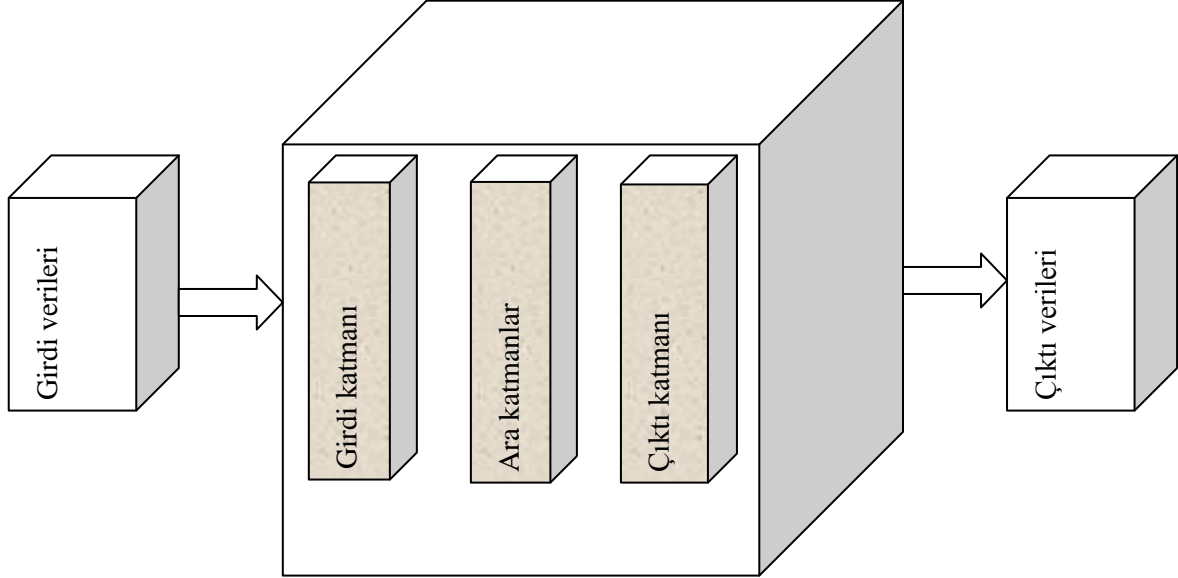
3.5. Yapay Sinir Ağının Yapısı

Yapay sinir hücrelerinin bir araya gelmesi ile birlikte oluşan yapıya “Yapay Sinir Ağı” yapısı denmektedir. Sinir hücrelerinin bir araya gelmesi rastgele olan bir işlem değildir. Tek katmanlı ağlar olmasına rağmen, çoğu uygulamalar da yapay sinir hücreleri 3 katman halinde oluşurlar. İlk katman nöronları gruplandırır. İkinci katman, katmanlar arası bağlantıları gruplandırma işlemini yapar. Ve son olarak da üçüncü katman toplama ve transfer fonksiyonlarını gruplandırır.

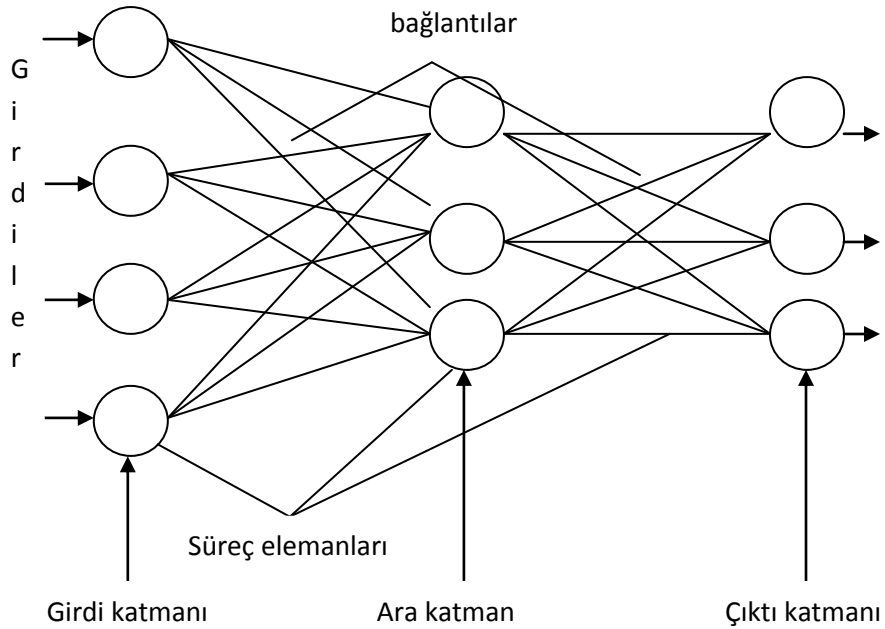
Girdi katmanı: Bu katmandaki elemanlar, dışarıdan aldıkları bilgileri işleyerek ara katmana iletirler.

Ara katmanlar: Girdi katmanından gelen bilgiler işlenerek çıktı katmanına gönderilirler.

Çıktı katmanı: Bu katman, ara katmandan gelen bilgileri işleyerek ağın girdi katman seti için üretmesi gereken çıktıyı üretir.



Şekil 3.2. YSA'da katmanların birbirleri ile ilişkisi



Şekil 3.3. Yapay sinir örneği

Şekil 3.2' de YSA' ya ait katmanlar arası ilişki ve Şekil 3.3'de yapay sinir ağı örneği gösterilmiştir.

3.6. Yapay Sinir Ağlarının Avantajları ve Dezavantajları

Tablo 3.2. YSA'nın avantajları ve dezavantajları

| Avantajları | Dezavantajları |
|---|--|
| Görülmemiş örnekler hakkında bilgi üretebilirler. | Donanım bağımlı çalışmalarda esneklik bulunmamaktadır. |
| Algılamaya yönelik olaylarda kullanılabilirler. | Probleme uygun ağ yapısı genellikle deneme yanılma yolu ile yapılmaktadır. |
| Örüntü tamamlama gerçekleştirebilirler | Ağda bulunan parametre değerlerinin belirlenmesinin bir kuralının olmaması, çözüm bulmayı geciktirebileceği gibi sonuçların yanlış olmasını da sağlayabilmektedir. |
| Kendi kendilerine öğrenme yetenekleri vardır. Farklı öğrenme algoritmalarıyla öğrenebilirler. | Sadece nümerik bilgiler ile çalışmaktadırlar. Farklı sistemlere uyarlanabilmesi zor olabilir. |
| Eksik bilgi ile çalışabilirler. | Ağın eğitiminin bitmesi hata değerine göre olmaktadır. Bitiş için geliştirilmiş bir yöntem yoktur. |
| Hata toleransına sahiptirler. | Bazı durumlarda ağın davranışları açıklanamaz. Ağın sonucunun nedeni her zaman belirgin olmayabilmektedir. |
| Dağıtık bellek yapısında bulunabilirler. | - |
| Belirsiz ve tam olmayan bilgileri işleyebilirler. | - |

3.7. Uygulama Alanları

Yapay sinir ağlarının, eksik verilerle çalışması ve normal olmayan verileri işleme gibi özelliklerinin bulunması, YSA'nın birçok alanda, özellikle çok sayıda veriyi işleme alanında yaygın olarak kullanılmasını sağlamaktadır. Günümüzde endüstriyel ve sosyal hayatta görülen çok sayıda örnekte yapay sinir ağlarına rastlanılabilmektedir. Sınıflandırma, modelleme ve tahmin uygulamaları yapay sinir ağlarının günümüzde en başta kullanılan alanlarına girmektedirler.

Başarılı uygulamalar incelendiğinde, YSA'nın çok boyutlu, gürültülü, karmaşık, kesin olmayan, eksik, kusurlu, hata olasılığı yüksek duyarlı verilerinin olması ve problemi çözmek için matematiksel modelin ve algoritmaların bulunmadığı, sadece

örneklerin var olduğu durumlarda yaygın olarak kullanıldıkları görülmektedir. Yapay sinir ağları pek çok sektörde değişik uygulama alanlarında kendini göstermiştir. Bunlardan bazıları aşağıdaki gibidir [10,55]:

Mühendislik Alanı:

- örüntü tanıma ve sınıflandırma yapma
- basılı karakter tanıma
- veri madenciliği
- sinyal filtreleme
- veri sıkıştırma
- optimizasyon,
- muhtemel fonksiyon kestirimleri
- zaman serileri analizleri
- doğrusal olmayan sinyal işleme ve sistemleri modelleme
- beyin modellemesi çalışmaları
- robotlar
- görsel sistemler ve uzaktan kumandalı sistemler (Parmak izi tanıma, kredi kartı hileleri saptama, retina tarama, yüz eşleştirme...)

Bankacılık Alanı:

- kredi uygulamaları geliştirilmesi,
- müşteri analizi ve kredi müracaat değerlendirilmesi
- bütçe yatırım tahminleri

Askeri ve Savunma alanı :

- silah yönlendirme
- radar ve sensör sonar sistemleri
- uçuş simülasyonları ve otomatik pilot uygulamaları
- trafik yoğunluğunun kontrolü

Üretim Alanı:

- üretim işlem kontrolü, ürün dizaynı, makine yıpranmalarının tespiti
- ürünlerin dayanıklılık analizi ve kalite kontrolü

Sağlık Alanı:

- kan hücrelerinin reaksiyonları ve kan analizlerini sınıflandırma
- kanserlerin erken teşhis ve tedavisi

- EEG, ECG, MR gibi cihazların kalite artırımı
- kan analizlerinin sınıflandırması

Yukarıda belirtilen örnekler daha da arttırılabilir. Görüldüğü gibi YSA günlük hayatımızda farkında olmadığımız pek çok alanda kullanılmaktadır. Gün geçtikçe uygulama alanları genişlemekte ve gelişmektedir.

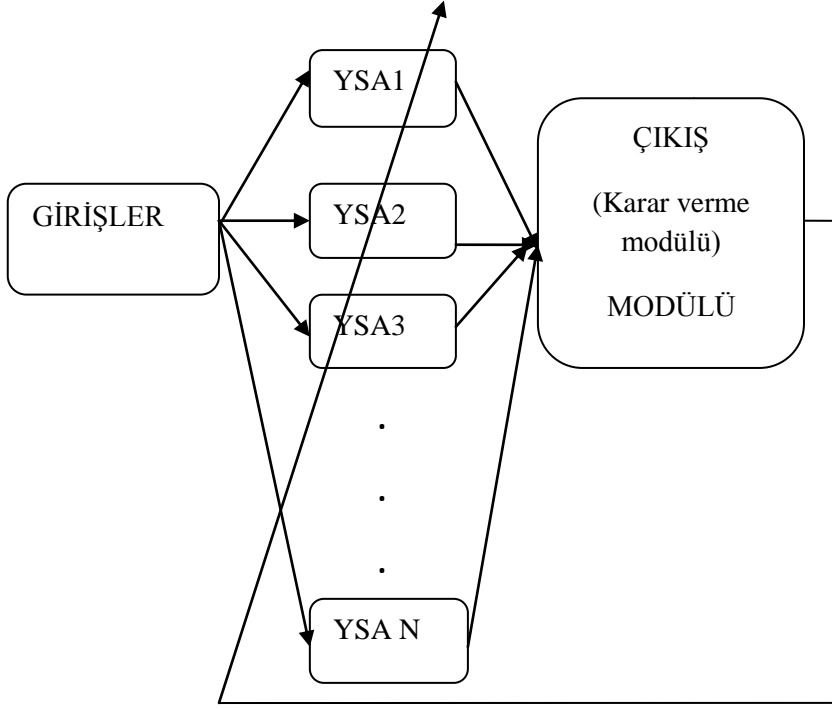
BÖLÜM 4. BİRLEŞİK YAPAY SİNİR AĞLARI

4.1. Birleşik Yapay Sinir Ağı Tanımı ve Yapısı

Birleşik yapay sinir ağı, birden fazla yapay sinir ağının birlikte kullanılarak aynı probleme çözüm üretmek üzere geliştirilen sistemlere denilmektedir [59]. Literatürde birleşik yapay sinir ağları ile ilgili çeşitli (Bagging [60], Boosting [61]... v.b.) metotlar bulunmaktadır.

Birleşik ağların çalışma mantığında, her bir ağ, o anda gerçekleşen olayın ya da problem çözümünün farklı bir yönünü öğrenebilmekte ve her bir bireysel ağların kararları (çıkışları) bir araya getirilerek ortak bir karar oluşturulmaktadır. Sonuçta, birleşik ağın performansı sistemi oluşturan ağların her birisinden daha yüksek olmaktadır. En az iki ağın bir araya gelmesi birleşik bir ağ yapısı oluşturmak için yeterli olmaktadır. Fakat verilen kararların daha objektif olarak, ortak bir karara dönüştürülmesi için en az 3 adet yapay sinir ağının kullanılması önerilmektedir. Yine de böyle bir zorunluluk yoktur [62,63]. Bazı durumlarda ise, ağa sunulan bir örnek bir ağ tarafından tanınmaz iken diğer ağ tarafından tanınmaktadır, bu da ağda karmaşaya neden olacaktır. Böyle durumlarda üçüncü bir ağın kararı önemli olmaktadır. Üçüncü ağın kararı hangi ağa yakın ise birleşik ağın kararı da o yönde olacaktır.

Birleşik yapay sinir ağlarında, birden çok yapay sinir ağı bir araya gelerek aynı problemi çözmeye çalışmakta ve sonuçlarını Çıkış Modülüne göndermektedir. Burada her bir ağdan gelen bütün kararlar incelenmektedir ve sonuçta birleşik ağın tek kararı oluşturulmaktadır. En son da bu karar dış dünyaya iletilmektedir. Aşağıdaki Şekil 4.1'de grup halinde bulunan yapay sinir ağı modeli verilmiştir.



Şekil 4.1. Grup halinde olan yapay sinir ağı modeli

Şekil 4.1’de gösterilen ağların her birinin aynı yapay sinir ağı olması gerekmez. Farklı yapay sinir ağı modelleri de aynı problemi çözebilir. Mesela şekilleri sınıflandırma probleminde çok katmanlı ağlar ve destekleyici öğrenme ağları bir araya gelerek bir birleşik ağ oluşturabilirler. Ya da bu duruma benzer bir şekilde, aynı yapay sinir ağı modelinden oluşan bir birleşik sistem oluşturulsa da bu ağların yapılarının eşit olması gerekmez. Yani herhangi bir bireysel sinir ağın ara katmanı iki tane olurken diğerinin beş tane olabilir. Tüm bunların yanında aynı problem üzerinde karar verebilmek için ağların girdi ve çıktılarının aynı olması gerekmektedir. Fakat yukarıda bahsedildiği gibi kullanılan ağların ara katman sayıları farklı olabilir.

Birleşik sistemlerin sadece yapay sinir ağlarından oluşması da gerekmez. İstatistiksel sistemler, uzman sistemler, klasik öğrenme sistemleri de yine birleşik sistemlerin bir parçası olabilirler. Burada önemli olan bu sistemlerinde aynı girdi hakkında fikir üretebilmeleridir. Birleşik ağların en bilinen özelliği hepsinin aynı örnek üzerinde karar verebilmeleridir. Ağların kendilerinin ürettikleri çıktılar bağımsız olarak kullanılmaz. Çıktıları sistemin çıktısını oluşturmak için girdi olarak kullanılmaktadır. Bununla birlikte ağların birbirlerine doğrudan bir ilişkisi

bulunmamaktadır. Her ağ girdilerini bağımsız olarak işlemekte ve diğer ağların kararlarını etkileyecek bir girişimde bulunmamaktadır. Kararların ortak değerlendirilmesinde ağların etkisi görülmemektedir [10].

4.2. Birleşik Yapay Sinir Ağların Eğitilmesi ve Test Edilmesi

Birleşik yapay sinir ağlarında her ağ birbirinden bağımsız eğitilmekte ve test edilmektedir. Eğitim ve test işlemlerinde herhangi bir özel algoritmaya gerek yoktur. Birleşik yapay sinir ağları eğitimini tamamlamış ve test edilmiş ağlardan oluşmaktadır. Eğitim ve test işlemi aşamalarında dikkat edilecek hususlar aşağıdaki gibi verilmiştir [10].

- 1- Eğer ağları eğitmek için aynı örnek seti kullanılırsa, bunlar ağlara farklı sıralarda sunulabilirler. Büyük bir örnek seti parçalara bölünerek her ağ farklı eğitim setinde eğitilebilir.
- 2- Birleşik sisteme ait olan ağırlıkları ve ağ parametreleri farklı olabilir.
- 3- Mümkün olduğu surece birbirinden bağımsız bütün ağları aynı test seti üzerinde eğitmekte fayda vardır. Böylece ağların öğrenip öğrenmedikleri birbirleri ile kıyaslanarak daha rahat belirlenebilir.

4.3. Çıkış Modülü (Ortak Karar Verme Modülü)

Çıkış Modülü problemin ortak olarak belirlenen son çözümünü hesaplamaktadır. Uygulamanın girdileri birleşik sistemin elemanı olan her ağa bağımsız olarak sunulmakta ve ağların çıktıları belirlenmektedir. Daha sonra bu çıktılar bir araya getirilerek birleşik sistemin ortak kararı belirlenmektedir. Ortak kararın verilmesi ağlar arasında sağlanan ‘oy birliği’ anlayışına dayanmaktadır. Çıkış modülü, programcının kendi sistemlerinin çalışma prensiplerine ve sistemin performans değerlendirmesine göre bu modülü kendi tecrübelerine göre istedikleri şekilde belirleyebilirler. Önemli olan farklı ağlardan gelen sonuçları tek bir sonuca indirgemektir [10].

4.4. Negatif Bağlantılı (İlişkili) Öğrenme

Gerçek dünyadaki birçok problem tek parça sistemler tarafından çözülmek için çok fazla karmaşık ve çok büyük olabilmektedir. Zor bir problemi tatmin edici şekilde çözmek için hem doğal sistemlerden hem de yapay sistemlerden ve de bunların birçok alt sisteminden oluşan yapılar mevcut olmuştur. Sinir ağlarının topluluklar üzerindeki başarısını sınıflandırmanın genelleştirmesine bağlı olarak geliştirilmesine dayanan tipik bir örneğe dayandığını söyleyebiliriz. Her şeye rağmen topluluklar için sinir ağları tasarlamak çok zor bir görevdir.

Yapay sinir ağı topluluğu tasarlamının birçok yolu vardır. Bunların çoğu iki aşamalı tasarım gerektirmektedir. İlk aşama bireysel ağlar oluşturmak, ikinci aşama ise oluşan bireysel ağları birleştirmektir. Genellikle bireysel ağlar birbirlerinden bağımsız eğitilir. Bu tür yapılarla uğraşmanın dezavantajlı yönlerinden birisi, bireysel ağların öğrenme sırasındaki birbiriyle etkileşimini kaybetmesidir. Bireysel tasarım aşamalarından birleşme aşamasında geri beslemeli yapı bulunmamaktadır. Bu yüzden bireysel ağların bağımsız yapısı tüm topluluk sistemi için çok fazla katkı sağlamamaktadır [64].

Hata çeşitliliği günümüzde çoklu sınıflandırılmış sistemlerde yaygın olarak kabul edilmektedir. Negatif ilişkili öğrenme, geri yayılım hata fonksiyonunun içinde düzenleme yapan bir toplu öğrenme tekniğidir. Buradaki düzenleme terimi hataların ilişki miktarını ölçmek içindir, eğitim sırasında bu hatalar minimize edilmektedirler [65].

Negatif ilişkili öğrenmenin temel mantığı, bir topluluktaki farklı bireysel ağların, farklı parçalarını ya da yönlerini birbirinden değişik verilerle eğiterek, tüm sistemin eğitimini daha verimli yapmaktır [66]. Bu yönüyle NCL, tek tek veya sırayla eğitilen bireysel ağların çalışmasından farklıdır. NCL' de, topluluktaki tüm bireysel ağlar aynı anda eğitilerek hata fonksiyonlarının birbiriyle ilişkilendirilmesi sağlanmaktadır. Tüm ağ aynı öğrenme işlemine tutularak, bireysel ağlar eğitilir ve bir araya getirilir. Standart eğitime ek olarak NCL' de hata fonksiyonları hesaplanırken, ceza fonksiyonlarını da işleme katmaktadır. Bu durum NCL' nin hata tahminlerini arttırmaktadır ve çıkıştaki fonksiyonun çeşitlenmesini sağlayıp,

sistemin doğru sonuç vermesindeki performansı arttırmaktadır. İşte bu sayede her bireysel eğitim tüm topluluk için en iyi sonucu verir [12]. Bu tür yapısıyla NCL bireysel tasarımlara ayırıp eğitim sürecinden geçiren diğer topluluk eğitimlerinden farklıdır.

NCL, aynı zamanda kapılanmış ağ oluşturan ve uzman ağlara bir numara veren Uzman Birleştime (Mixtures-of-experts- ME) mimarisinden de farklıdır. ME mimarisi negatif ilişkilendirilmiş bireysel ağlara karşı bir sapma da üretebilir. NCL' de ağ kapılara ayırmaya ihtiyaç yoktur. Tamamen farklı bir hata fonksiyonu kullanır. NCL' deki λ parametresi, sapma-varyans-kovaryans arasındaki dengeyi kurmak için kullanışlı bir yol sunar. ME mimarisinde böyle bir kontrol bulunmamaktadır [13].

Eğitim setimizin şu şekilde olduğunu farz edelim;

$$D = \{(x(1), d(1)), \dots, (x(N), d(N))\}$$

$\mathbf{x} \in \mathbb{R}^+$, d skaler bir büyüklük ve N eğitim setinin boyutu. Bu bölüm çıkışı, sinir ağların çıktılarının ortalamasını veren basit bir topluluğu formüle eder:

$$F(\mathbf{n}) = \frac{1}{M} \sum_{i=1}^M F_i(\mathbf{n})$$

M topluluktaki bireysel sinir ağların sayısı, $F_i(\mathbf{n})$ eğitimdeki i . ağın çıkışı, $F(\mathbf{n})$ eğitimdeki topluluğun çıkışı.

NCL, birden fazla çok katmanlı yapay sinir ağını geriye yayılım yöntemi kullanarak paralel olarak eğitmeye dayanır. Topluluktaki her bireysel ağın hata fonksiyonundaki hata teriminin ilişkisini tanımlar. Böylece tüm ağlar aynı anda eğitilir ve aynı D eğitim setiyle etkileştirilmiş olur. i ağı için hata fonksiyonu E_i , NCL' de ,

$$E_i = \frac{1}{N} \sum_{n=1}^N E_i(\mathbf{n})$$

$$= \frac{1}{N} \sum_{n=1}^N \frac{1}{2} [F_i(n) - d(n)]^2 + \frac{1}{N} \sum_{n=1}^N \lambda p_i(n) \quad (2)$$

ile tanımlanır. $E_i(n)$, n . eğitimdeki i ağıının hata fonksiyonu. Denklem 2' deki ilk kısım i ağıının deneysel risk fonksiyonu, 2. kısım p_i ilişkilendirilmiş hata fonksiyonu. Amaç her ağıın hatasını negatif korelasyonla azaltarak topluluğun gerisine yaymaktır.

λ parametresinin $0 \leq \lambda \leq 1$ aralığında kullanılması hatanın kuvvetini belirler. Hata fonksiyonu p_i ;

$$p_i(n) = (F_i(n) - F(n)) \sum_{j \neq i} (F_j(n) - F(n)) \quad (3)$$

$E_i(n)$ ' nin türevi;

$$\begin{aligned} \frac{\partial E_i(n)}{\partial F_i(n)} &= F_i(n) - d(n) + \lambda \frac{\partial p_i(n)}{\partial F_i(n)} \\ &= F_i(n) - d(n) + \lambda \sum_{j \neq i} (F_j(n) - F(n)) \\ &= F_i(n) - d(n) - \lambda (F_i(n) - F(n)) \\ &= (1 - \lambda) (F_i(n) - d(n)) + \lambda (F(n) - d(n)) \quad (4) \end{aligned}$$

$F(n)$ 'nin, $F_i(n)$ 'e bağlı sabit bir değere sahip olduğu varsayıldığında, ağıın düğümlerindeki ağırlıkların güncellenmesi için standart geriye yayılım algoritması kullanılmaktadır. Tüm bireysel ağıların ağırlıklarının her eğitim aşamasından sonra güncellendiği formül Denklem 4' de kullanılmıştır. Öğrenme işleminin, tüm

eğitiminde bir sefer tamamlamasına *devir* denir. NCL, Denklem 4' den de anlaşılacağı gibi standart geriye yayılım algoritmasının bir parçasını kullanmaktadır. Aslında tek değiştirilmiş kısmı i . ağı için ekstra bir $\lambda(F_i(n) - F(n))$ hesaplanmasıdır.

Denklem 2 ve Denklem 4' de şunları gözlemlenmektedir:

Eğitim işlemi boyunca, tüm bireysel ağlar, birbirlerinin hata fonksiyonu içindeki hata terimiyle etkileşim içindedir. Her i ağı sadece $F_i(n)$ ve $d(n)$ arasındaki farkı minimize etmekle kalmıyor aynı zamanda $F(n)$ ile $d(n)$ arasındaki farkı da minimize ediyor. Böylece NCL ile diğer ağlarda eğitim sırasında hatalar göz önünde bulundurulur.

- 1- $\lambda = 0$ için, bireysel ağların hata fonksiyonlarındaki hata terimlerinde korelasyon yoktur ve bireysel ağlar sadece bağımsız eğitilir. Bireysel ağlardaki, bağımsız eğitim NCL' nin özel bir durumudur.
- 2- $\lambda = 1$ için Denklem 4' den

$$\frac{\partial E_i(n)}{\partial F_i(n)} = F(n) - d(n) \quad (5)$$

elde edilir. n . eğitim için topluluğun deneysel risk fonksiyonu şöyle tanımlanmaktadır:

$$E_{ens}(n) = \frac{1}{2} \left(\frac{1}{M} \sum_{i=1}^M F_i(n) - d(n) \right)^2 \quad (6)$$

n . eğitimdeki F_i ' ye bağlı $E_{ens}(n)$ ' in türevi;

$$\frac{\partial E_{ens}(n)}{\partial F_i(n)} = \frac{1}{M} \left(\frac{1}{M} \sum_{i=1}^M F_i(n) - d(n) \right) = \frac{1}{M} (F(n) - d(n)) \quad (7)$$

Bu durumda;

$$\frac{\partial E_i(n)}{\partial F_i(n)} \propto \frac{\partial E_{ens}(n)}{\partial F_i(n)} \quad (8)$$

elde edilir. Topluluğun deneysel risk fonksiyonunun minimizasyonu, bireysel ağların hata fonksiyonlarının minimizasyonu ile elde edilir. Bu şekilde

topluluğun eğitiminde NCL' nin bağımsız ağlara yeni bir yol sağladığını görülmüş oldu.

Denklem 2' de toplam N tane eğitim örneği, F_i özgün bir çıkış ve d ise hedeftir. Gösterim kolaylığı için n endeksi ve noktalar toplamı için gerekliliğini kaldırarak, artık sadece tek bir noktadaki hata fonksiyonunu dikkate alınmaktadır. Hata;

$$p_i = (F_i - F) \sum_{j=i} (F_j - F) \quad (9)$$

F tüm grubun çıktısıdır. Bu önlemler ve hatalar belirleyiciler arasındaki ilişkiyi oluşturmaktadır. Aslında formülde sıfıra yakın atanmış değerlerin sıfır olduğu gerçeğini kullanılmıştır. $\sum_i (F_j - F) = 0$ denklemini şu şekilde yazılmaktadır:

$$p_i = (F_i - F) [-(F_i - F)] = -(F_i - F)^2 \quad (10)$$

Belirsizliğin tahmini hata fonksiyonunu vermektedir.

$$E_i = \frac{1}{2} (F_i - d)^2 - \lambda (F_i - F)^2 \quad (11)$$

İfadeden eşit ağırlıklarla hata toplulukları için belirsizlik ayrıştırması yapılırsa:

$$\frac{1}{2} (F_i - d)^2 = \frac{1}{M} \sum_i \left[\frac{1}{2} (F_i - d)^2 - \frac{1}{2} (F_i - F)^2 \right] \quad (12)$$

Denklem 11' de $\lambda = \frac{1}{2}$ olarak atarsak hata fonksiyonunu her ağın eğitimi için belirsiz ayrıştırma olarak verilen topluluğun hata katkısı olarak kullanabiliriz. Bu teorik bir metottur ve işe yarar, çünkü topluluğun hatası bir ağ yerine tüm ağa yayılmıştır.

4.4.1. Negatif bağlantılı öğrenmenin yararları

Negatif bağlantı öğrenmenin özellikle iki önemli faydası vardır. Bunlardan ilki, negatif bağlantılı olarak gürültü azaltma yöntemi, pozitif korelasyon gürültü

yöntemine göre daha hızlı ve bağımsız bir şekilde gürültüyü sıfıra doğru indirebilmesidir. Negatif ilişkili değişkenlerin ikinci faydası ise bir boşluğu doldurma konusundaki yeteneğinin pozitif korelasyonlu ya da bağımsız değişkenlere oranla daha iyi olmasıdır [12].

Boşluk doldurmanın yararı bağımsız bileşenler (temel fonksiyonların katsayıları) pozitif değerler ile sınırlı olduğu zaman görülmektedir. Boşluk doldurma yararı için gerekli olan negatif olmayan katsayı kısıtlamasına temel fonksiyonların negatif yönde birbirleriyle ilişkili oldukları yerde ihtiyaç meydana gelir. Tanım gereği temel fonksiyonlar için kullanılan negatif ilişkili fonksiyona kısaca “a” diyelim. Eğer temel fonksiyon a’ nın katsayılarına negatif değerler almasına izin verirsek, o zaman a ve -a modelleri arasında hiçbir anlamlı ayrım kalmaz. Bunun anlamı “a” nın olduğu yerde her zaman “-a” nında olması gerektiğidir. Bu şu demek bağımsız bileşenlerden oluşan bir set en iyi şartlar altında alan kapsayacaktır. Ancak, sadece negatif olmayan katsayılara izin verilir, yani artık negatif ilişkili temel fonksiyonlarda gerçek bir yarar var olmuş anlamına gelmektedir, ima edilenler daha fazla olmayacaktır. Bu negatif olmayan katsayı kısıtlaması yeni çalışmalarda giderek daha popüler olmaktadır. Doğal miktarların negatif olamaması, görüntüyü oluşturan bileşenlerin negatif değerler içermemesi, sinirsel atış hızlarının negatif olmaması gibi negatif olmayan kısıtlamalar, bu yaklaşımın zayıflığı olarak kabul edilmektedir [67,68].

4.4.2. Negatif bağlantılı öğrenme ile diğer yöntemlerin karşılaştırılması

- Negatif bağlantılı gürültü azaltma yöntemi pozitif bağlantılı gürültü yöntemine göre daha hızlı ve bağımsız bir şekilde gürültüyü sıfıra doğru indirgeyebilir.
- Boşluğu doldurma konusundaki yeteneğinin pozitif ilişkili ya da bağımsız değişkenlere oranla daha iyidir.
- NCL topluluk sinir ağları eğitiminde oldukça etkili bir yaklaşım olduğu gibi, ayrıca artımlı öğrenmede oldukça güçlü bir yöntem olma potansiyeline de sahiptir. Yeni bir eğitim verisi ağa sunulduğunda bireysel ağların bu verilere uyum sağlaması birçok farklı yoldan sağlanmış olur.

- NCL, tek tek veya sırayla eğitilen bireysel ağların çalışmasından farklıdır. NCL' de, topluluktaki tüm bireysel ağlar aynı anda eğitilerek hata fonksiyonlarının birbiriyle ilişkilendirilmesi sağlanmaktadır.
- NCL" de tüm ağ aynı öğrenme işlemine tutularak, bireysel ağlar eğitilir ve bir araya getirilir. Bu tür yapısıyla NCL bireysel tasarımlara ayırıp eğitim sürecinden geçiren diğer topluluk eğitimlerinden daha avantajlıdır.
- NCL" de ME mimarisine göre ağı kapılara ayırmaya ihtiyaç yoktur. ME mimarisinde negatif ilişkilendirilmiş bireysel ağlara karşı bir sapma da üretilir. NCL" de tamamen farklı bir hata fonksiyonu kullanılır. Sapma – varyas-kovaryans arasındaki dengeyi λ parametresini kullanarak sağlar.
- Doğal miktarların negatif olamaması, görüntüyü oluşturan bileşenlerin negatif değerler içermemesi, sinirsel atış hızlarının negatif olmaması gibi negatif olmayan kısıtlamalar, bu yaklaşımın diğer yöntemlere göre zayıflığı olarak kabul edilmektedir.

BÖLÜM 5. NEGATİF BAĞLANTILI ÖĞRENME ALGORİTMALI YAPAY SİNİR AĞLARI ile MOBİL CİHAZLARDA OPTİK KARAKTER TANIMA UYGULAMASI

Günümüzde, teknolojinin gelişmesine paralel olarak ve özellikle 3G teknolojinin de yaygınlaşması ile birlikte mobil cihazlar üzerinde görüntü işleme uygulamalarının kullanımı oldukça yaygın bir hale gelmiştir. Eskiye göre bu uygulamalar eğlence amaçlı kullanımın ötesine geçmeye başlamıştır.

OCR sistemleri, günümüzde artık daha hızlı, daha ucuz ve daha güvenilir bir sistem olmuşlardır. Bu teknolojinin kullanımının artmasının başlıca sebebi binlerce sayfalık verinin birkaç dakika içerisinde bilgisayarlara ya da veri tabanlarına aktarabilmesidir.

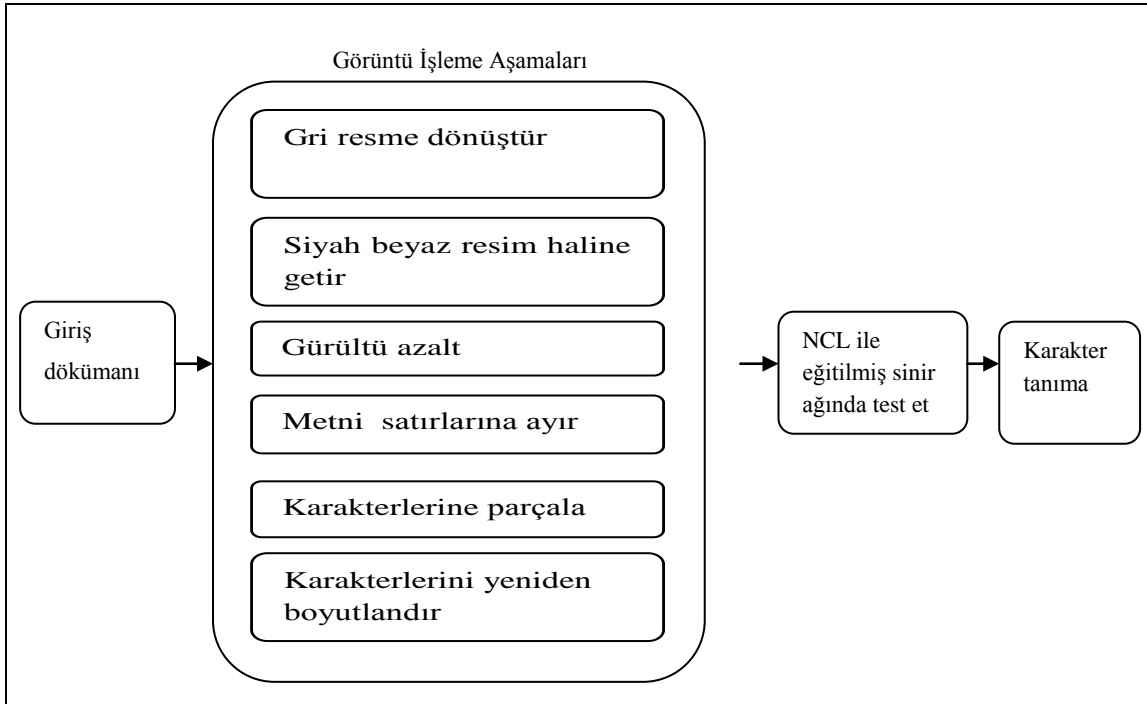
Mobil cihazların yaygınlaşması ile birlikte, bu cihazlar içinde OCR sistemlerine de ihtiyaç duyulmaya başlanmıştır. Bugün kullanılan OCR sistemleri ile artık daha fazla yazı çeşidi üzerinde daha iyi sonuçlar vermektedir [7,8].

Bu çalışmada, cep telefonu ile resmin çekilmesinden karakterlerin tanınmasına kadar olan kısımların işlem basamakları ve Şekil 5.1’de çalışmanın algoritması aşağıda verilmiştir.

5.1. İşlem Basamakları

- Program içerisine cep telefonu kamerası ile çekilen resim alınır.
- Alınan resim ilk önce gri hale dönüştürülür.
- Gri halde olan resim üzerinde eşikleme yapılarak resim ikili resim (binary resim) haline çevrilir. Çalışmada ikili görüntünün ikili resim hali bulunması için Otsu algoritması kullanılmıştır [17].
- Binary halde olan resimde daha rahat bir tarama yapabilmek için resimde bulunan gürültü azaltılır.

- Resim önce yatayda satırlarına ayrılır.
- Satırlarına ayrılan resim içerisinde resim dikey olarak bölünerek harflerine ayrılır.
- Tüm harfler yapay sinir ağı eğitimi için uygun bir hale getirilmek üzere boyutları tekrardan düzenlenir ve 20*20'lik matris boyutuna getirilirler.
- Boyutlandırma işlemi tamamlandıktan sonra her bir harf resmi bir matris şeklinde başka bir diziye atanır.
- Dizideki her bir harf resmi, sırayla birleşik YSA sistemine verilirler.
- Her bir bireysel ağda, eğitilmiş YSA ağırlıkları ile paralel olarak işleme koyulup, harf resimlerinin tanınması sağlanır ve daha sonra harf resimleri yazıya çevrilirler.
- Tanınan harfler çıkışta birleştirilir ve kelime metni olarak elde edilirler.



Şekil 5.1. Çalışmanın algoritması

5.2. Karakter Tanıma

Taranmış metinler ya da mobil cihaz kamerası tarafından alınan görüntüler (beyaz resim üzerine siyah yazı) görüntü işleme aşamalarından geçirildikten sonra

karakterlerine ayrıştırılmaktadır. Karakter resimlerinden 20*20 olacak şekilde karakter matrisleri elde edilmiştir. Bu matristeki bilgiler yapay sinir ağı eğitimi ve testi için dosyalanmıştır. Her bir bireysel ağ giriş katmanı 400 tane veri girişine, çıkış katmanında da 64 tane çıkışa sahip olmaktadır. Her bir ağ için gizli katman sayıları 50,100 ve 200 olarak belirlenmiştir. Çıkış katmanında 64 tane çıkış nöronumuzun olmasının sebebi ise her bir harfe şekline karşılık gelen bir karakteri temsil ettiği içindir. Karakter resim matrisi Şekil 5.2’de gösterilmiştir. Her bir bireysel ağın ağırlıkları birbirinden farklı ve rastgele verilmiştir. Ayrıca öğrenme katsayıları da birbirinden bağımsız olarak verilmiştir (0.1, 0.2, 0.5). Teorik olarak tüm ağ karakterleri tanıdığı anda eğitim sonlanır fakat pratiğe gelirse bunun için bir durdurma kriterinin belirlenmesi gerekmektedir. Çalışmada belirlenen durdurma kriteri 0.00001’dir. Yani karesel hata 0.00001’e düşünceye kadar eğitim devam etmektedir.

```

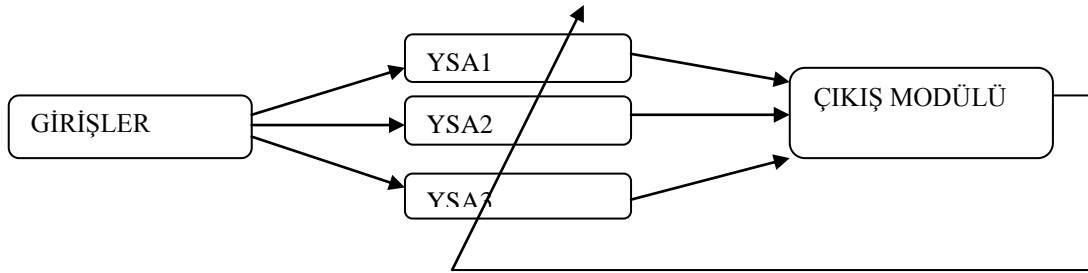
0000000000100000000 1111111111110000000 0000000000011100000 1111111111110000000
0000000000100000000 000111000000011110000 0000000111000001111 0001110000011110000
0000000001110000000 000111000000000111100 0000011000000000011 0001110000000111100
0000000001100000000 000111000000000111000 0000111000000000011 0001110000000001100
0000000001110000000 0001110000000001110 0001100000000000000 0001110000000001100
0000000010111000000 0001110000000001110 0011100000000000000 0001110000000001110
0000000010111000000 0001110000000001110 0111100000000000000 0001110000000001110
0000000110011100000 0001110000000001110 0111000000000000000 0001110000000001111
0000001000111000000 0001110000000001110 0111000000000000000 0001110000000001111
0000001000111000000 0001110000000001110 1111000000000000000 0001110000000001111
0000001000001100000 0001111111111110000 1111000000000000000 0001110000000001111
0000001000001110000 0001110000000001110 1111000000000000000 0001110000000001111
0000010000001100000 0001110000000001110 0111000000000000000 0001110000000001111
0000011111111110000 0001110000000001111 0111000000000000000 0001110000000001111
0000100000000110000 0001110000000001111 0111100000000000000 0001110000000001110
0000100000000111000 0001110000000001111 0111100000000000000 0001110000000001110
0000100000000111000 0001110000000001111 0011100000000000000 0001110000000001110
0001000000000110000 0001110000000001111 0001111000000000000 0001110000000001110
0011000000000111000 0001110000000001110 0000111000000000011 0001110000000001100
0111000000000111100 0011111111111110000 0000000111111110000 0011111111111100000

```

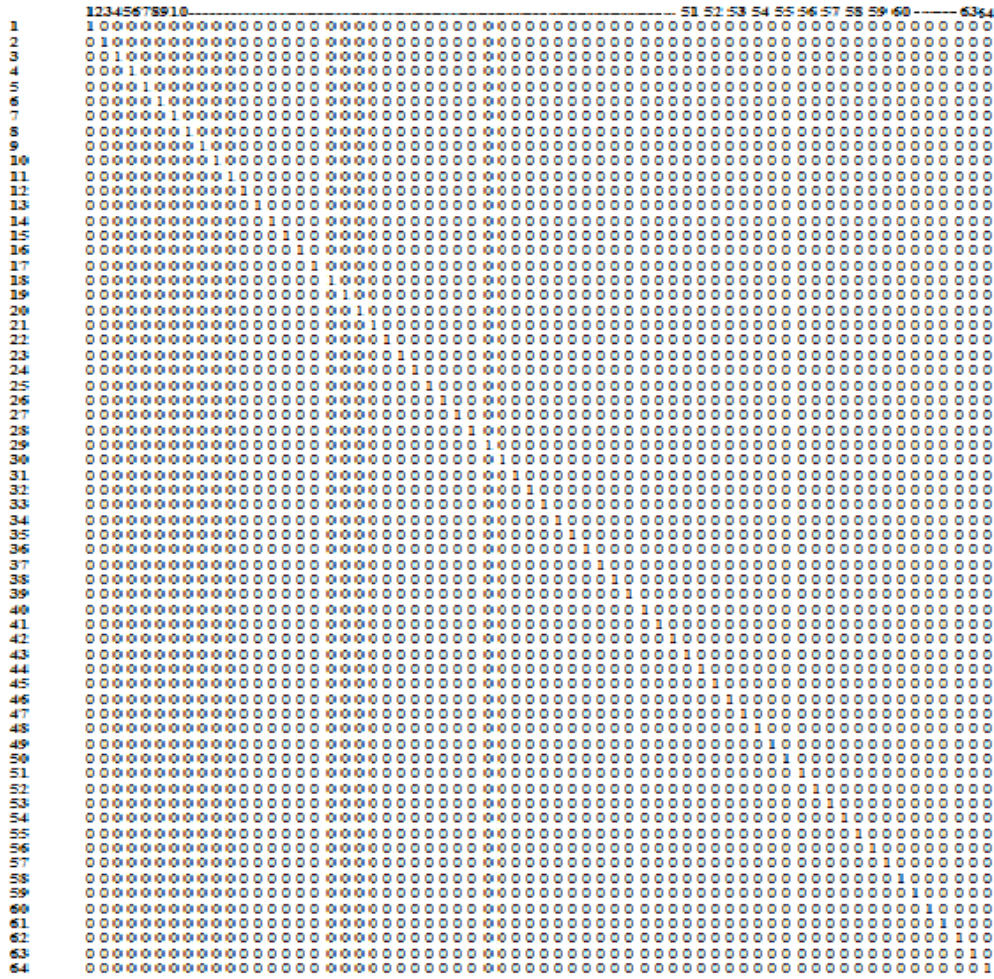
Şekil 5.2. Karakterlerin matrisel gösterimi

Ağların gruplandırılması için en az iki ağın bir araya gelmesi yeterlidir. Verilen kararların daha rahat ortak bir karara dönüştürülmesi için 3 adet yapay sinir ağının kullanılması önerilmiştir ancak böyle bir zorunluluk yoktur [51]. Bazı durumlarda ağa sunulan bir örnek bir ağ tarafından tanınmaz iken diğer ağ tarafından tanınmaktadır. Üçüncü bir ağın kararı bu durumda önemli olmaktadır. Bu ağın kararı hangi ağa yakın ise o zaman tüm ağın kararı o yönde olacaktır. Literatürde yapılan çalışmalarda genellikle üç ya da dört adet bireysel sinir ağı gruplandırılarak eğitim gerçekleştirilmiştir [13,65]. Bu çalışmada ise, üç tane bireysel yapay sinir ağı

kullanılarak karakter tanıma işlemi gerçekleştirilmiştir. Kullanılan yapay sinir ağı modeli şekilde verilmiştir.



Şekil 5.3. Grup halinde olan yapay sinir ağı modeli



Şekil 5.4. Yapay sinir ağının hedeflenen çıkış değerleri

Şekil 5.3'de görüldüğü gibi çalışmada 3 tane bağımsız yapay sinir ağı gruplandırılmıştır. Eğitimin sonucu için hedeflenen çıkış değerleri Şekil 5.4'de belirtilmiştir. Şekil 5.5'de, Şekil 5.3'de verilen hedeflene değerler matrisindeki her bir satırın hangi karaktere ait olduğu belirtilmiştir.

| |
|-------------------------|
| 1. satir "A" karakteri |
| 2. satir "B" karakteri |
| |
| 20. satir "P" karakteri |
| 21. satir "Q" karakteri |
| 22. satir "R" karakteri |
| 23. satir "S" karakteri |
| |
| 63. satir "y" karakteri |
| 64. satir "z" karakteri |

Şekil 5.5. Hedeflenen değerler matrisinde hangi satırın hangi karaktere ait olduğunun gösterimi

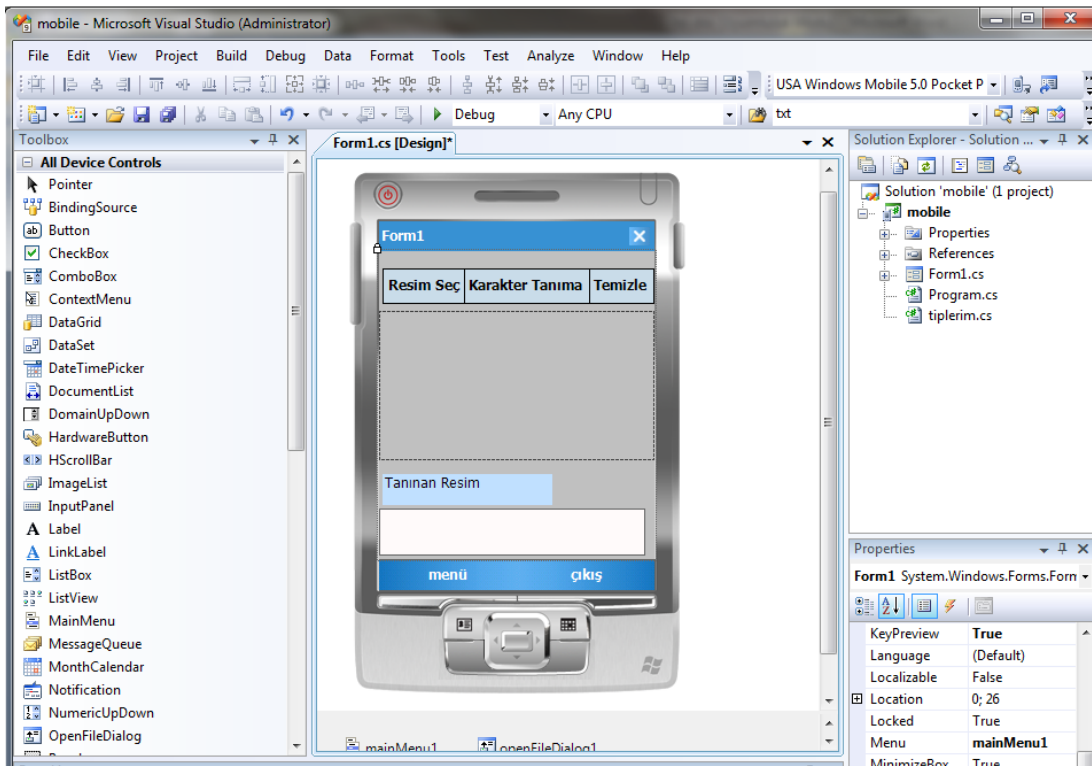
İlk eğitim gerçekleştirildikten sonra her bir bireysel ağın çıkışındaki hata (E_i) tespit edilmektedir. Ayrıca tüm sistemin çıkışındaki toplam hata (E) belirlenmekte ve minimize edilmektedir. Toplam hatanın elde edilmesi için tüm bireysel ağın çıkışlarının birbirleriyle bağlantılı hale getirilmesi gerekmektedir. Bunun için ilişkilendirilmiş hata fonksiyonu (penalty function - p_i),

$$p_i(n) = (F_i(n) - F(n)) \sum_{j \neq i} (F_j(n) - F(n))$$

şeklinde bulunur. Buradaki $F_i(n)$ n.eğitimdeki i. ağın çıktısını, $F(n)$ ise n. eğitimdeki tüm topluluğun çıktısını ifade eder. Tüm ağdaki toplam hata (E), bulunduktan sonra geri beslemeli öğrenme yöntemi (back propagation) ile azaltılır ve daha sonra elde edilen hata tüm ağa yayılır. Böylece aynı anda paralel olarak tüm ağda öğrenmenin ilk aşaması gerçekleşmiş olur. Öğrenme tamamlanıncaya kadar her bir ağdaki ağırlıklar geri yayılım yönteminden elde edilen sonuçlara göre yenilenir.

5.3. Uygulamanın çalışması

Bu çalışmada programlama dili olarak c sharp (c#), uygulama geliştirme arabirimi (IDE) olarak da Visual Studio 2008 kullanılmıştır. Yapay sinir ağı eğitimi için ise Matlab R2010b kullanılmıştır. Visual Studio ile geliştirilen uygulamanın test edilebilmesi için Windows CE, Pocket PC Ve Smartphone olmak üzere 3 farklı emülatör bulunmaktadır. Bu çalışmada Windows Mobile 5.0 Pocket PC R2 Emulator kullanılmıştır.



Şekil 5.6. Windows Mobile 5.0 Pocket PC R2 Emulator

Uygulama emilatörü Şekil 5.6'de gösterilmiştir. Giriş verileri Times New Roman, Arial ve Comic sans yazı tipleri kullanılmıştır. Siyah beyaz hale getirilen harf resimleri YSA giriş matrisi boyutuna getirilmekte ve eğitilmiş sinir ağı grubuna test işlemi için verilmektedir. Sinir ağı için gerekli olan ağırlıklar random olarak belirlenmiş olup proje her çalıştığında yeniden belirlenmektedir.

Toplamda 32 küçük ve 32 büyük harf kullanılmıştır. Her üç yazı tipi içinde giriş verisi alındığından $64*3=192$ boyutlu bir dizi eğitim giriş dizisi olarak kullanılmıştır. Şekil 5.7, Şekil 5.8 ve Şekil 5.9'da mobil uygulamanın çalışması gösterilmektedir.



Şekil 5.7. Mobil uygulamanın çalışması



Şekil 5.8. Mobil uygulamada resmin seçilmesi



Şekil 5.9. Mobil uygulamada karakter tanıma

Tablo 5.1. Bazı test sonuçlarının şekillerle gösterimi

| Yazı tipi (Boyutu) | Orijinal resim | Elde Edilen Yazı |
|--------------------|---|--|
| Times (48) | <p>pijamalı hasta yağız şoföre çabucak güvendi x q w</p> | <p>pijamalı haSta yatız şoföre çabBcak g*vendi x q w</p> |
| Arial (48) | <p>PİJAMALI HASTA YAĞIZ ŞOFÖRE ÇABUCAK GÜVENDİ</p> | <p>I PİJAMALI * HASTAYAGIZ *OFORE Ç*BUC*K GUVENDİ</p> |
| Comic (48) | <p>PİJAMALI HASTA YAĞI ŞOFÖRE Q ÇABUCAK W GÜVENDİ Z X</p> | <p>PIJAMALI * HASTA YAGI *OFORE Q ** BUCA* W GUVENDİ Z X</p> |

Mobil uygulama test edilirken tüm test görüntü verilerinin boyutunun 480*260 pixel alınmıştır. Ayrıca resmin derinliği 72 pixel\inch'dir. Uygulamanın başarısı

hesaplanırken tanınabilen tüm harflerin oranının, tüm harflere oranı baz alınarak bulunmuştur. Bazı test sonuçlarının uygulama sonuçlarının görüntüleri Tablo 5.1’de verilmiştir. Tablo 5.2’de ise çalışmadan elde edilen sonuçların başarı yüzdeleri gösterilmiştir.

Tablo 5.2. Başarı oranları

| Yazı tipi (Boyutu) | Büyük/Küçük Harf | Başarı oranı |
|--------------------|------------------|--------------|
| Times (48) | Büyük Harf | %75 |
| Arial (48) | Büyük Harf | %78,125 |
| Comic Sans MS (48) | Büyük Harf | %81.25 |
| Times (48) | Küçük Harf | %84.37 |
| Arial (48) | Küçük Harf | %87.5 |
| Comic Sans MS (48) | Küçük Harf | %75 |

BÖLÜM 6. SONUÇLAR ve ÖNERİLER

Mobil cihazlar yazılım ve donanım teknolojilerindeki gelişmelerle birlikte, mobil cihazlardan beklentiler artmıştır. Bu beklentilerden bir tanesi de optik karakter tanımadır. Bu çalışmada gerçekleştirilen yazılım ile gerek basılı metinlerin gerekse tabela, v.b. metinlerin tanınması amaçlanmaktadır.

Literatürde yapılan araştırmalarda negatif bağlantılı öğrenme metodu kullanılarak yapılan sınıflandırma çalışmalarına bakıldığında %85' e yakın başarılı sonuçlar alınmıştır. Yapılan çalışmada negatif bağlantılı öğrenme metodu mobil cihazdan elde edilen veriler kullanılarak eğitilmiş ve başarılı bir şekilde test edilmiştir.

Bu çalışmada, üç bireysel yapay sinir ağı gruplandırılarak mobil cihazlar için optik karakter tanıma işlemi gerçekleştirilmiştir. Çalışmada, eğitim ve test setlerinde her bir karakterden üçer adet örnek kullanılmıştır. Test setinde kullanılan örnekler beyaz fon üzerine siyah yazı şeklinde kullanılmıştır. Yapay sinir ağı eğitimi için kullanılan üç adet bireysel sinir ağı gruplandırılarak, sinir ağı eğitimi Matlab R2010b üzerinde geliştirilmiştir. Sinir ağları eğitilirken her bir bireysel ağ için, birbirinden farklı gizli katmanlar oluşturulmuş ve her biri için farklı ağırlık değerleri rastgele olarak belirlenmiştir. Tablo 5.2'de verilen başarı oranlarına göre yapılan bu çalışma göstermektedir ki çözünürlüğü çok yüksek olmayan bir mobil cihaz üzerinde karakterler yaklaşık olarak %80 oranında tanınmıştır.

Elde edilen sonuçlarda %20 oranında kayıp oluşmuştur. Özellikle mobil cihazlardaki çözünürlüğün düşük olması ve görüntü işleme aşamalarında karakterlerin yeterince birbirinden ayıramaması bu kayıpların sebeplerindedir. Günümüz mobil cihazlardaki kameraların özelliklerinin kısıtlı olmasından dolayı bellek ve işlemci hızı bilgisayar sistemleri ile karşılaştırılamayacak kadar yavaştır. Bilgisayar üzerinde çok iyi bir performansla çalışan bir uygulama, mobil cihaz üzerinde aynı performansı gösterememektedir. Geliştirilen mobil karakter tanıma uygulaması içinde bu durum söz konusudur. Ayrıca kameraların çözünürlükleri karakter tanıma işlemi için önemli bir etkidir. Özellikle siyah beyaz formatına çevrilen resimde

kayıplar fazla olmaktadır. Çekilen resmin çözünürlüğü daha da iyileştirilebilirse bu kayıpların azaltılabileceği ve sınıflandırma başarısının da artırılabilceği düşünölmektedir.

Bunun için son yıllarda üzerinde çalışmalar devam eden süper çözünürlük yöntemi gibi bir yöntemin daha sonraki çalışmalarda kullanılması kayıpların azaltılmasını sağlayabilir.

Bu çalışmada kullanılan birleşik yapay sinir ağı yapısının içerisindeki ağların dizaynı eğitime başlamadan önce deneme yanılma yöntemiyle belirlenmektedir. Bu işlemin, dinamik bir şekilde ve probleme göre oluşturulması durumunda sistem daha iyi bir sonuç verebileceği ya da kelime yapısı ve istatistiksel özellikleri kullanılarak sınıflandırılmayan karakterler tahmin edilerek sistemin başarı oranı artırılabilir.

KAYNAKLAR

- [1] BOZTAŞ, A., Mobil cihazlar üzerindeki güncel programlama teknikleri ve karşılaştırılması, Yüksek Lisans Tezi, Beykent Üniversitesi, Fen Bilimleri Enstitüsü, Matematik Bilgisayar Anabilim Dalı, Bilgi Teknolojileri Bilim Dalı, 2006.
- [2] ATASEVER, V., ARSLAN, D., GÜVENOĞLU, E., Mobil cihazlarda online görüntü işleme yazılımının geliştirilmesi, Akademik Bilişim Konferansı, Muğla, 2010.
- [3] ŞEKERCİ, M., KANDEMİR, R., Birleşik ve eğik türkçe el yazısı tanımda k-nn sınıflama yöntemi ve sözlük kullanımı, Trakya University J Sci, 10, 1, pp. 97–102, 2009.
- [4] ÖZ, C., ÇİT, G., YAZGAN, H.R., Cep telefonu için bir optik karakter tanıma uygulaması. International Science & Technology Conference, Kuzey Kıbrıs Türk Cumhuriyeti, 2010.
- [5] AY, S., VAROL, A., Karakter tanıma için düzenli özellik çıkarma işleminin incelenmesi ve uygulanması. Bilimde Modern Yöntemler Sempozyumu (BMYS'08), Bildiriler Kitabı, Eskişehir Osmangazi Üniversitesi, Eskişehir, pp. 1063–1070, 15–17 Ekim 2008.
- [6] WATANABE, Y., SONO, K., YOKOMIZO, K., OKODA, Y., Translation camera on mobile phone (ICME'03), pp. 177–180, 2003.
- [7] KOGA, M., MINE, R., KAMEYAMA, T., TAKAHASHI, T., Camera-based kanji ocr for mobile-phones: practical issues, Proceedings of the Eight International Conferences on Document Analysis and Recognition (ICDAR'05), 2, pp. 635–639, 2005.
- [8] RODRIGUEZ, A., KIM, S., KIM, J.H., BLANKO-FERNANDEZ, Y., English to spanish translation of signboard images from mobile phone camera, IEEE SOUTHEASTCON '09, pp. 356–361, 2009.
- [9] ERDEM, O.A., UZUN, E., Yapay sinir ağları ile türkçe times new roman, arial ve el yazısı karakterler tanıma, Gazi Üniv. Müh. Fak. Der., 20, 1, pp. 13–19, 2005.
- [10] ÖZTEMEL, E., Yapay sinir ağları, papatya yayıncılık, İstanbul, 2003.

- [11] ALPAYDIN, E., Yapay öğrenme, Boğaziçi Üniversitesi Yayınevi, İstanbul, pp. 37-259, 2011.
- [12] ZEKE, S.H., KASABOV, N., Fast neural network ensembles via negative-correlation learning, IEEE TRANSACTIONS ON NEURAL NETWORKS, 16, 6, pp. 1707–1710, 2005.
- [13] LIU, Y., YAO, X., Simultaneous training of negatively correlated neural networks in an ensemble, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS PART B: CYBERNETICS, 29, 6, pp. 716–725, 1999.
- [14] BULUN, M., GÜLNAR, B., GÜRAN, S., Eğitimde mobil teknolojiler, The Turkish Online Journal of Educational Technology (TOJET), 3, 2, pp. 165–169, 2004.
- [15] ÇAMOĞLU, K., ATASEVER, V., Mobil programlama, Pusula Yayıncılık, İstanbul, 2010.
- [16] MULGAONKAR, P.G., CHEN, C., DECURTINS, J.L., Word recognition in a segmentation free approach to OCR, Proceedings of the Second International Conferences on Document Analysis and Recognition (ICDAR 1993), pp. 573–576, 1993.
- [17] OTSU, N., A threshold selection method from gray-level histograms, IEEE Transactions on Systems, Man, and Cybernetics, 9, 1, pp. 62–66, 1979.
- [18] PRATT, W.K., Digital image processing, Prentice Hall, 1991.
- [19] WANG, P.S.P., ZHANG, Y.Y., A fast and flexible thinning algorithm, IEEE Transactions on Computers, 38, 5, pp. 741–745, 1989.
- [20] LAM, L., SUEN, C.Y., An evaluation of parallel thinning algorithms for character recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence, 17, 9, pp. 914–919, 1995.
- [21] NABIYEV, V.V., Yapay zeka: insan-bilgisayar etkileşimi, Seçkin Yayıncılık, Ankara, 2012.
- [22] IŞIKDOĞAN, F., İçerik duyarlı görüntü ölçekleme, Seminer, Yıldız Teknik Üniversitesi, İstanbul, 25 Ekim 2010.
- [23] YANKIOĞLU, B., SANDON, P.A., Segmentation of off line handwriting using linear programming, Pattern Recognition, 31, 12, pp. 1825–1833, 1998.

- [24] LAINE, M., NEVALAINEN, O.S., A standalone ocr system for mobile cameraphones, IEEE 17th International Symposium on Personal, Indoor and Mobile Radio Communications, Department of Information Technology, Turku University, pp. 1–5, 2006.
- [25] ANIL, K.J., Fundamentals of digital image processing, Prentice Hall, 1989.
- [26] YANIKOĞLU, B., KHOLMATOV, A., Türkçe için geniş sözcük dağarcıklı doküman tanıma sistemi, 11. Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SIU'03), Sabancı Üniversitesi, İstanbul, 2003.
- [27] MUSAYEV, E., Bilgisayar destekli karakter tanıma sistemi tasarımı, Yüksek Lisans Tezi, İstanbul Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, 2004.
- [28] PARK, J., KWON, Y.B., An embedded OCR: A Practical case study of code porting for a mobile platform, Pattern Recognition (CCPR 2009), Chung-Ang University, Seoul, pp. 1–5, 2009.
- [29] GORSKI, N., ANISIMOV, V., AUGUSTIN, E., BARET, O., PRICE, D., SIMON, J.C., A2iA Check Reader: A family of bank check recognition systems, 5th International Conference on Document Analysis and Recognition (ICDAR'99), pp. 523–526, 1999.
- [30] NI, D.X., Application of neural networks to character recognition, Proceedings of Students/Faculty Research Day, Pace University, pp. 1–6, 2007.
- [31] MANI, N., SRINIVASAN, B., Application of artificial neural network model for optical character recognition, IEEE International Conference on Computational Cybernetics and Simulation, 3, pp. 2517–2520, 1997.
- [32] AHMED, O.A.W., Application of artificial neural networks to optical character recognition, Master's Thesis, King Fahd University Of Petroleum & Minerals, Saudi Arabia, June 1994.
- [33] LEUNG, C.H., SZE, L., Feature selection in the recognition of handwritten chinese characters, Engineering Applications of Artificial Intelligence, 10, 5, pp. 495–502, 1997.
- [34] ZHANG, L.X., ZHAO, Y.N., WANG, J.X., Feature selection in recognition of handwritten chinese characters, Proceedings of the 1st International Conference on Machine Learning and Cybernetics, 3, pp. 1158–1162, 2002.
- [35] INOUE, K., MIYAZAKI, R., SUZUKI, M., Optical recognition of printed mathematical documents, Proceedings of the 3th Asian Technology Conference in Mathematics, Springer-Verlag, pp. 280–289, 1998.

- [36] SINGH, R., YADAV, C.S., VERMA, P., YADAV, V., Optical character recognition (ocr) for printed devnagari script using artificial neural network, *International Journal of Computer Science & Communication*, 1, 1, pp. 91–95, 2010.
- [37] SIMON, J.C., Off-line cursive word recognition, *Proceedings of the IEEE*, 80, 7, pp. 1150–1161, 1992.
- [38] JUNG, M., SHIN, Y., and SRIHARI, S.N., Multifont classification using typographical attributes, *5th International Conference on Document Analysis and Recognition (ICDAR'99)*, pp. 353–356, 1999.
- [39] ARICA, N. and YARMAN VURAL, F.T., Optical character recognition for cursive handwriting, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24, 6, pp. 801–813, 2002.
- [40] KORKMAZ, S.U., KIRÇIÇEĞİ, G., ATALAY, V., A character recognizer for turkish language, *Seventh International Conference on Document Analysis and Recognition (ICDAR'03)*, pp. 1238–1241, 2003.
- [41] UĞURLU, B., Yapay sinir ağları, Çanakkale Onsekiz Mart Üniversitesi Bilgisayar Mühendisliği Bölümü, Ders Notları, İzmir.
- [42] HEBB, D.O., *The organization of behaviour: A Neuropsychological Theory*, Psychology Press, pp. 60-78, 2002.
- [43] MCCULLOCH, W.S., PITTS, W.A., A logical calculus of the ideas immanent in nervous activity, *Bulletin of Mathematics and Biophysics*, 52, pp. 115–133, 1990.
- [44] ŞAHİN, M., Karakter tanıma, Yüksek Lisans Tezi, Haliç Üniversitesi Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Bölümü, İstanbul, 2007.
- [45] WIDROW, B., HOFF, M.E., Adaptive switching circuits, *IRE WESTCON Convention, Record Part IV*, pp. 96–104, 1960.
- [46] KOHEBEB, T., Self – organized formation of topologically correct feature maps, *Biological Cybernetics*, 43, pp. 59–69, 1982.
- [47] HOPPFIELD, J.J., Neural networks and physical systems with emergent collective computational abilities, *Proceedings of the National Academy of Sciences*, 79, pp. 2554–2558, 1982.
- [48] HOPPFIELD, J.J., Neurons with graded response have collective computational properties like those of two state neurons, *Proc. Natl. Acad. Sci.*, 81, pp. 3088–3092, 1982.

- [49] CARPENTER, G.A., GROSSBERG, S.A., ART2: self-organization of stable category recognition codes for analog input patterns, *Applied Optics*, 26, 3, pp. 4919–4930, 1987.
- [50] PARKER, D.B., *Learning-logic*. M.I.T. Cen. Computational Res. Economics Management Sci., Cambridge, MA, 1985.
- [51] WEBROS, P.J., *Beyond regression: new tools for prediction and analysis in the behavioural sciences*. PhD dissertation, Committee on Appl. Math., Harvard Univ., Cambridge, M.A., 1974.
- [52] ELMAS, Ç., *Yapay sinir ağları*, Seçkin Yayıncılık, Ankara, 2003.
- [53] SPECHT, D.F., Probablistic neural networks for classification mapping, or associative memory, *IEEE Conference on Neural Networks*, San Diego, 1, pp. 525–532, 1988.
- [54] SPECHT, D.F., A general regression neural network, *IEEE Transaction on Neural Networks*, 2, 6, pp. 568–576, 1991.
- [55] HAMZAÇEBİ, C., *Yapay sinir ağları: tahmin amaçlı kullanımı matlab ve neurosolutions uygulamalı*, Ekin Kitapevi Yayınları, İstanbul, 2011.
- [56] AMARI, S.I., A Theory of adaptive pattern clasification, *IEEE Transactions on Electronic Computers*, 16, pp. 299–307, 1967.
- [57] HAYKIN, S., *Neural networks a comprehensive foundation*, Macmillian College Publishing Company, New York, 1994.
- [58] ELMAS, Ç., *Yapay zeka uygulamaları*, Seçkin Yayıncılık, Anakara, 2011.
- [59] HANSEN, L.K., SALAMON, P.K., Neural network ensembles, *IEEE Trans. Pattern Analysis and Machine Intelligence*, 12, 10, pp. 993–1001, 2001.
- [60] BREIMAN, L., Machine learning, *Bagging Predictors*, 24, 2, pp. 123–140, 1994.
- [61] SCHAPIRE, R.E., A brief introduction to boosting, *Proc. 16th Int'l Joint Conf. Artificial Intelligence*, pp. 1401–1406, 1999.
- [62] LIU, Y., YAO, X., ZHAO, Q., HIGUCHI, T., An experimental comparison of ensemble learning methods on decision boundaries, *Proceedings of the 2002 International Joint Conference on Neural Networks (IJCNN'02)*, 1, pp. 221–226. Los Alamitos, CA.: IEEE Computer Society Press, 2002.
- [63] CHAN, Z.S.H., KASABOV, N., A preliminary study on negative correlation learning via correlation- correlated data (NCDD), *Neural Processing Letters*, 21, pp. 207–214, 2005.

- [64] LIU, Y., YAO, X., Ensemble learning via negative correlation, Evolvable Systems Laboratory, Computer Science Division, Mbox 1501, Electrotechnical Laboratory, Japan, School of Computer Science, The University of Birmingham, UK, 1999.
- [65] BROWN, G., YAO, X., On effectiveness of negative correlation learning, Proceedings of First UK Workshop on Computational Intelligence, 2001.
- [66] LIU, Y., YAO, X., HIGUCHI, T., Evolutionary ensemble with negative correlation learning, IEEE Transactions on Evolutionary Computation, 4, 4, pp. 380–387, 2000.
- [67] DURRANT, S., Negative correlation in neural systems, Phd Thesis, Sussex University, Department of Information., Brighton, 2008.
- [68] DURRANT, S., KENDRICK, K., FENG, J., Algorithms for exploiting negative correlation. Sussex University, Department of Informatics, Brighton, 2008.

ÖZGEÇMİŞ

Burcu KIR, 24.05.1987 de Samsun' da doğdu. İlk ve orta eğitimini Gülsüm Sami Kefeli İ.Ö.O' da tamamladı. Lise eğitimini Samsun Anadolu Lisesi'nden mezun oldu. 2005 yılında başladığı Dumlupınar Üniversitesi Bilgisayar Mühendisliği bölümünü 2009 yılında bitirdi. 2009 yılında Sakarya Üniversitesi'nde Bilgisayar ve Bilişim Mühendisliği Bölümünde yüksek lisans yapmaya başladı. 2010 yılında Kocaeli Üniversitesi Bilgisayar Mühendisliği Bölümü'nde Araştırma Görevlisi olarak göreve başladı ve şu anda hala Kocaeli Üniversitesi'nde görev yapmaktadır.