

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



**KAYISI YAPRAKLARININ EVRİŞİMSEL SINIR  
AĞLARI KULLANILARAK SINIFLANDIRILMASI**

**Berna ARI**

**Yüksek Lisans Tezi**

**Anabilim Dalı: Elektrik ve Elektronik Teknolojileri**

**Tez Danışmanı: Prof. Dr. Abdulkadir ŞENGÜR**

**TEMMUZ-2017**

T.C.  
FIRAT ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ

KAYISI YAPRAKLARININ EVRİŞİMSEL SINIR AĞLARI KULLANILARAK  
SINIFLANDIRILMASI

YÜKSEK LİSANS TEZİ  
Berna ARI  
132138103

Tezin Enstitüye Verildiği Tarih:06.06.2017

Tezin Savunulduğu Tarih: 14.07.2017

Tez Danışmanı : Prof. Dr. Abdulkadir ŞENGÜR (F.Ü.)

Diğer Jüri Üyeleri : Doç. Dr. Erkan TANYILDIZI (F.Ü.)

Yrd. Doç. Dr. Ömer Faruk ALÇİN (B.Ü.)



TEMMUZ-2017

## ÖNSÖZ

Bu tez çalışmasında, eğitim hayatımda bana sürekli olarak destek olan, yardımlarını hiçbir zaman benden esirgemeyen ve yapacağım diğer çalışmalarda da desteğini esirgemeyeceğini düşündüğüm değerli hocam Prof. Dr. Abdulkadir ŞENGÜR'e ve hayatımın her alanında elinden gelenin fazlasını yapmaya çalışan değerli eşim Ali Arı'ya teşekkürü bir borç bilirim.



## İÇİNDEKİLER

	<u>Sayfa No</u>
ÖNSÖZ .....	II
İÇİNDEKİLER.....	III
ÖZET .....	V
SUMMARY .....	VI
ŞEKİLLER LİSTESİ .....	VII
TABLOLAR LİSTESİ .....	VIII
KISALTMALAR.....	IX
SİMGELER LİSTESİ .....	X
<b>1. GİRİŞ .....</b>	<b>1</b>
1.1. Literatür .....	1
1.2. Problemin Tanımı .....	3
1.3. Tezin Amacı.....	4
1.4. Tezin Organizasyonu .....	4
<b>2. DERİN ÖĞRENME .....</b>	<b>5</b>
2.1. Yapay Sinir Ağları .....	6
<b>3. EVRİŞİMSEL SİNİR AĞLARI .....</b>	<b>8</b>
3.1. Evrişim Katmanları.....	8
3.2. Aktivasyon İşlemleri.....	9
3.3. Havuzlama Katmanı .....	10
3.4. Tamamen Bağlanmış Ağ .....	10
3.5. Çıktı Katmanı.....	11
3.6. ESA'ların Eğitimi .....	11
3.6.1. Geri Yayılım Algoritması .....	11
3.7. İlgili Yapılar.....	13
<b>4. YEREL ALICI ALANLAR AŞIRI ÖĞRENME MAKİNESİ .....</b>	<b>15</b>
4.1. Aşırı Öğrenme Makinesi.....	15
4.2. Yerel Alıcı Alanlar Aşırı Öğrenme Makinesi.....	16
<b>5. DENEYSEL ÇALIŞMALAR .....</b>	<b>18</b>
5.1. Veri Tabanı .....	18
5.2. ESA'ya ait Deneysel Parametreler ve Sonuçlar .....	19

5.3.	YAA-AÖM'e ait Deneysel Parametreler ve Sonuçlar.....	20
5.4.	ESA ve YAA-AÖM'ye Ait Performans Sonuçları.....	21
<b>6.</b>	<b>SONUÇLAR</b> .....	<b>24</b>
	<b>KAYNAKLAR</b> .....	<b>25</b>
	<b>ÖZGEÇMİŞ</b> .....	<b>29</b>



## ÖZET

Bitkilerin insan hayatındaki önemi büyüktür. Bitkiler, besin sektörü, endüstri ve medikal gibi birçok alanda kullanılmaktadır. Dünya üzerinde 310.000 – 450.000 arasında bitki çeşidinin varlığı bilinmektedir. Her geçen gün bilinmeyen yeni türler de ortaya çıkmaktadır. Günümüzde bitki taksonomisi daha çok geleneksel yöntemler ile yapılmaktadır. Uygun, pratik ve otomatik bir bitki tanıma sisteminin oluşturulması bitkilerin sınıflandırılması, anlaşılması ve yönetilebilmesi açısından oldukça kullanışlı bir çalışmadır. Yaprak özelliklerinden bitki tanıma, bitki sınıflandırmada kullanılan yöntemlerden bir tanesidir. Bitkilerin muhafazası ve tanınabilirliğini sağlamak için her türe ait bir bitki veri tabanı kurmak bitki tanımada uzmanlar için önemli bir adımdır. Dijital bitki kataloglama sistemi hazırlamak için bilgisayar görmesi uygulamaları ve görüntü işleme tekniklerini kullanan otomatik yaprak tanıma sistemlerinin tasarlanması, bitki türlerinin sınıflandırılmasına da hız ve verimlilik katacaktır.

Bu tezde, yedi farklı kayısı türüne ait yaprak imgeleri kullanılmıştır. Bu türler sırası ile Apikoz Şalak, Çataloğlu, Çekirge İz, Hacıkızı, Hırmanlı, Paviot ve Tokatoğlu Erzincan'dır. Yedi farklı kayısı türüne ait yaprak imgeleri kullanılarak kayısı türleri derin öğrenme yöntemleri ile sınıflandırılmıştır. Derin öğrenme yöntemlerinden biri olan Evrişimsel Sinir Ağları kullanılarak %91,34 ±0,77 doğru sınıflandırma başarımı ile kayısı türleri sınıflandırılmıştır. Diğer bir derin öğrenme yöntemi olan Yerel Alıcı Alanlar-Aşırı Öğrenme Makinesi kullanılarak %97,26±0,95 doğru sınıflandırma başarımı ile kayısı türleri sınıflandırılmıştır.

**Anahtar Kelimeler:** Kayısı Bitkilerinin Sınıflandırılması, Yaprak Tanıma, Derin Öğrenme, Evrişimsel Sinir Ağları, Yerel Alıcı Alanlar Aşırı Öğrenme Makinesi

## SUMMARY

### CLASSIFICATION OF APRICOT LEAFS VIA CONVOLUTIONAL NEURAL NETWORK

Plants are very significant for human life. They are used in various fields such as food sector, industry and medical. It is known that there are 310000 – 450.000 kinds of plants in the world. As the days pass, unknown species are coming to light. Nowadays traditional Methods are generally used while classifying the plants. Creating appropriate, practical and automatic system to introduce the plant is a practical study in terms of classifying, understanding and managing the plants. Introducing the plant on account of its leaf is one of the methods which are used for classifying plants. To protect and identify the plants, creating a database for each species is an important development for specialists. To create a digital plant classification system, designing automatic leaf recognition system which uses computer vision applications and image processing techniques will provide velocity and productivity the process.

In this paper, 7 different apricot images were used. These species are apikoz şalak, Çatalođlu, Çekirge İz, Hacıkızı, Hırmanlı, Paviot ve Tokatođlu Erzincan. These apricot species are classified by using deep learning methods. Apricot species are classified with %91,34 ±0,77 success by using convolutional neural network which is among deep learning methods. By using local receptive field extreme learning machine which is another deep learning method, apricot species are classified with %97,26±0,95 success.

**Key Words:** Apricot Plant Classification, Leaf Recognition, Deep Learning, Convolutional Neural Networks, Local Receptive Field Extreme Learning Machine

## ŞEKİLLER LİSTESİ

	<u>Sayfa No</u>
Şekil 2.1 YSA'nın yapısı.....	6
Şekil 3.1 Evrişim operatörünün uygulanması .....	9
Şekil 3.2 2x2 filtre ve 2 birim adıma sahip maksimum havuzlama .....	10
Şekil 5.1 Kullanılan kayısı yaprağı imgeleri, a) Apikoz Şalak, b) Çataloğlu, c) Çekirge iz, d) Hacıkızı, e) Hırmanlı, f) Paviot ve g) Tokatoğlu Erzincan .....	18
Şekil 5.2 Kullanılan ESA'nın yapısı .....	19
Şekil 5.3 Hata ve iterasyon.....	20





## TABLULAR LİSTESİ

	<u>Sayfa No</u>
<b>Tablo 5.1.</b> Apikoz şalak ve diğer türlerin sınıflandırılmasına ait performans sonuçları.....	22
<b>Tablo 5.2.</b> Çataloğlu ve diğer türlerin sınıflandırılmasına ait performans sonuçları .....	22
<b>Tablo 5.3.</b> Çekirge iz ve diğer türlerin sınıflandırılmasına ait performans sonuçları .....	22
<b>Tablo 5.4.</b> Hacıkızı ve diğer türlerin sınıflandırılmasına ait performans sonuçları .....	22
<b>Tablo 5.5.</b> Hırmanlı ve diğer türlerin sınıflandırılmasına ait performans sonuçları .....	23
<b>Tablo 5.6.</b> Hırmanlı ve diğer türlerin sınıflandırılmasına ait performans sonuçları .....	23
<b>Tablo 5.7.</b> Tokatoğlu Erzincan ve diğer türlerin sınıflandırılmasına ait performans sonuçları.....	23

## KISALTMALAR

<b>AÖM</b>	: Aşırı Öğrenme Makinesi
<b>ÇBMM</b>	: Çok boyutlu mesafe matrisi
<b>ÇKA</b>	: Çok Katmanlı Algılayıcı
<b>DVM</b>	: Destek Vektör Makinaları
<b>EKDVM</b>	: En Küçük Kareler Destek Vektör makinaları
<b>ESA</b>	: Evrişimsel Sinir Ağları
<b>GSEM</b>	: Gri Seviye Eş-oluşum Matrisi
<b>K-NN</b>	: En Yakın Komşu
<b>KVA</b>	: Kalite Veri Analizleri
<b>NBD</b>	: Nöro Bulanık Denetleyici
<b>OMH</b>	: Ortanca Merkezlerin Hareketi
<b>OSA</b>	: Olasılıksal Sinir Ağları
<b>TGKIB</b>	: Tek Gizli Katmanlı ve İleri Beslemeli
<b>YAA</b>	: Yerel alıcı alanlar
<b>YİÖ</b>	: Yerel İkili Örüntüler
<b>YSA</b>	: Yapay Sinir Ağları
<b>DP</b>	: Doğru Pozitif
<b>DN</b>	: Doğru Negatif
<b>YP</b>	: Yanlış Pozitif
<b>YN</b>	: Yanlış Negatif

## SİMGELER LİSTESİ

$x_i$	: Yapay Sinir Ağlarına ait giriş parametreleri
$w_i$	: Yapay Sinir Ağlarının giriş ağırlıkları
$t_i$	: Yapay Sinir Ağının çıkış parametreleri
$\theta_i$	: Bias değeri
$a(.)$	: Aktivasyon fonksiyonu
$m$	: Örneklem sayısı
$J(W, b)$	: Maliyet fonksiyonunun birinci terimine ait toplam kare hatası
$\tilde{N}$	: Gizli sinir hücresi
$g(x)$	: Aktivasyon fonksiyonu
$H$	: Gizli katmanın çıkış matrisi
$\beta$	: Çıkış ağırlıkları
$K$	: Evrişim filtresinin katsayıları
$dxd$	: Öznitelikleri çıkarılacak giriş imgesinin boyutu
$rxr$	: Evrişim filtresinin boyutu,
$X$	: Evrişim işleminde önceki katmanın değeri
$Y$	: Sonraki evrişim katmanının değeri,
$T$	: Veri seti için sınıf etiketleri
$I$	: Birim matrisi
$C$	: Regülasyon katsayısı
$\delta_i^{n_l}$	: Hata terimi

## 1. GİRİŞ

Dünya üzerinde 310.000 – 450.000 arasında bitki çeşidinin varlığı bilinmektedir. Her geçen gün bilinmeyen yeni türler de ortaya çıkmaktadır [1]. Bitkilerin, gıda sektöründe, tıp alanında ve endüstride de yoğun olarak kullanılmaları, bitki türleri üzerine yapılacak çalışmaların önemini bir kat daha artırmaktadır. Günümüzde bitki taksonomisi daha çok geleneksel yöntemler ile yapılmaktadır. Uygun, pratik ve otomatik bir bitki tanıma sistemi oluşturulması, bitkilerin sınıflandırılması, anlaşılması ve yönetilebilmesi açısından oldukça kullanışlı bir çalışma olacaktır. Bitki tanımda, bitkilerin özellikleri, benzer ve ayırt edici karakteristikleri önemlidir, çünkü bu özellikler ve farklılıkların belirlenmesi botanikçilere bitkileri tanıma ve sınıflandırmada yardımcı olmaktadır. Yaprak tanıma, otomatik bitki taksonomisi için temel oluşturacaktır. Bunun yanında, bitki ürünlerinin kullanıldığı endüstriyel faaliyetlerde üretim kalitesi ve hızının artırılması için destek sağlayacaktır. Bitkilerin tanıma bilirliliğini sağlamak için her türe ait bir bitki veri tabanı kurmak bitki tanımda uzmanlar için önemli bir adımdır. Dijital bitki kataloglama sistemi hazırlamak için bilgisayar görmesi uygulamaları ve görüntü işleme tekniklerini kullanan otomatik yaprak tanıma sistemlerinin tasarlanması, bitki türlerinin sınıflandırılmasına hız ve verimlilik sağlayacaktır. Yaprakların şekil ve doku özelliklerini kullanarak bitki tanıma yaklaşımı bu çalışmanın temelini oluşturmaktadır. Bu alanda son yıllarda birçok çalışma yapılmıştır.

### 1.1. Literatür

[2] nolu çalışmada üzüm yaprağı imgelerinden elde edilmiş dokuz farklı morfolojik özellik, çoklu-sınıf Destek Vektör Makinaları (DVM) ile dokuz farklı üzüm türü sınıflandırılmıştır. [3] nolu çalışmada 32 farklı bitki türüne ait 1800 imgeden beş temel geometrik özniteliklere dayalı 12 öznitelik parametresi elde edilmiş ve Olasılıksal Sinir Ağları (OSA) kullanılarak %90 doğruluk ile sınıflandırılmıştır. [4] nolu çalışmada altı farklı bitkiye ait 267 adet yaprak görüntülerinden, en boy oranının tespiti (yaprağın kendi eninin, kendi boyuna oranı), yaprak görüntüsünün resme olan oranı, yaprağın genişliği, baş kısmı ve uç kısmı gibi her yaprağa göre değişebilecek özellikler tespit edilmiş ve sınıflandırma işlemi k-NN (En Yakın Komşu) algoritması ile gerçekleştirilmiştir. [5] nolu çalışmada, Flavia ve SLID veri tabanlarına ait yaprak görüntüleri kullanılmıştır. Nesnenin başlangıç ve

bitim noktalarını belirledikten sonra, bitişik noktalar arasına konik veya doğru uydurulmuştur. Uydurulan koniklerin tiplerine göre imgedeki nesneyi tanımlayan öznitelik vektörü elde edilmiştir. Sınıflandırma işleminde DVM kullanılmıştır. [6] nolu çalışmada yeşil çay yapraklarını sınıflandırmak amacıyla örtüşmeyen pencereyi Yerel İkili Örüntüler (YİÖ) ve Gri Seviye Eş-oluşum Matrisi (GSEM) dokusal özelliklerini birleştiren yeni bir metod önermişlerdir. [7] nolu çalışmada veri tabanı olarak, Flavia ve SLID veri tabanı kullanılmıştır. Yaprakların genel biçim öznitelikleri, Fourier tanımlayıcı öznitelikleri ve çok boyutlu mesafe matrisi (ÇBMM) öznitelikleri ile belirlenmeye çalışılmış ve tüm bu özniteliklerin birleşiminin performansa etkisi incelemiştir. Sınıflandırma işlemi için Kalite Veri Analizleri (KVA), DVM ve K-NN kullanılmıştır. [8] nolu çalışmada bitki yapraklarından öznitelikler elde edebilmek için yaprağa ait şekil, yaprak ağzı ve doku, özellikleri ile birlikte Zernike Momentler yöntemi kullanılmıştır. Sınıflandırma işlemi Radyal Temel Olasılıksal Sinir Ağı(OSA) ile yapılmıştır. [9] nolu çalışmada 20 çeşit üzüm türüne ait 354 yaprak imgesi kullanılarak, Hu'nun değişmez moment metodu ile öznitelikler belirlenerek sınıflandırma işlemi Öklid uzaklığı ile yapılmıştır. [10] nolu çalışmada fotoğraf ile ya da video içerisinden belirlenen 6 türe ait yaprak görüntülerinden çıkarılan 12 adet özellik K-NN sınıflandırma algoritması kullanılarak sınıflandırılmıştır. [11] nolu çalışmada Acer (Japon akçaağacı) bitkisine ait görüntülerin bulunduğu veri tabanındaki dokuz cinse ait imgeler kullanılmıştır. Bu imgelerden merkez kenar uzunlukları, geometrik ve istatistiksel öznitelikler çıkarımı yapılmıştır. Sınıflandırma işlemi Aşırı Öğrenme Makineleri (AÖM) ile gerçekleştirilmiştir. [12] nolu çalışmada morfolojik özelliklere dayalı özellik çıkarımı önermişlerdir. Sınıflandırma aşamasında Rastgele Orman (Random forest) algoritması, C4,5 algoritması, k-NN, Naive Bayes ve DVM yöntemleri uygulanarak sınıflandırıcıların sınıflandırma performansları karşılaştırılmıştır. [13] nolu çalışmada bitki yapraklarının doku ve şekil özniteliklerinin birleşimleri kullanılarak yaprakların karakterize edilmesi ve tanınması için yeni bir yöntem önerilmiştir. Yaprak dokusu Gabor filtresi ve GSEM kullanılarak modellenmiştir. Şekil öznitelikleri ise değişmez momentum ve Curvelet transform ile çıkarılmıştır. 31 sınıf yaprağın sınıflandırılması İleri Beslemeli Geri Yayılım Çok Katmanlı Algılayıcı ile gerçekleştirilmiştir. [14] nolu çalışmada, bitki yapraklarından en-boy oranı, Zernike Momentleri, Hu'nun Değişmezleri, form faktörü, dikdörtgenlik, dairesellik, alan, entropi dizilimi gibi özellikler çıkartılmış ve sınıflandırma işlemi Yapay Sinir Ağları (YSA) ile gerçekleştirilmiştir. [15] nolu çalışmada yaprak şekli ve yaprağın damar düzeni kullanılarak öznitelik çıkarımı yapılmış ve çıkarılan bu değerler YSA ile

sınıflandırılarak LeaVeS adında yaprak tanıma uygulaması yapılmıştır. LeaVeS uygulaması ile bir bitki türünü tek başına bir yaprak görüntüsü ile sınıflandırmayı hedeflemiştirlerdir. Bitki türlerinin sınıflandırılmasında %95 doğruluk gözlemlenmiştir. [16] nolu çalışmada geometrik özniteliklere dayalı 8 temel morfolojik özniteliklere ek olarak değişmez moment uygulanarak toplam da 15 öznitelik çıkarılmıştır. Her sınıf için 20 örnek alınarak 20 farklı bitki türü incelenmiştir. Sınıflandırıcı olarak Ortanca Merkezlerin Hareketi (OMH) yöntemi kullanılarak başarımlar ve sınıflandırma zamanları diğer çalışmalar ile kıyaslanmıştır. [17] nolu çalışmada Flavia veri setinde olan yaprak görüntüleri 2 ve 4 parçaya ayrılarak her bir parçadan renksel özellikler çıkarılmıştır. Daha sonra yaprağın morfolojik özelliklerinden boyut, dış merkezlik ve çapın ana uzunluğuna oranı gibi öznitelikler çıkarılmıştır. Sınıflandırma işlemi En Küçük Kareler Destek Vektör makinaları (EKDVM) ile gerçekleştirilmiş ve %95,80 doğruluk oranı elde etmişlerdir. [18] nolu çalışmada Değiştirilmiş YİÖ metodu ile yaprak görüntülerinden öznitelikler çıkarılmış ve sınıflandırma işlemi k-NN ile yapılmıştır. [19] nolu çalışmada yaprağa ait şekil, yaprak ağzı ve doku özellikleri ile birlikte Fourier tanımlayıcılarını kullanarak öznitelikler elde edilmiştir. Temel Boyut Analizi yöntemi ile elde edilen özniteliklerin boyutu azaltılmış ve DVM kullanılarak sınıflama işlemi gerçekleştirilmiştir. [20] nolu çalışmada iki boyutlu video görüntülerinden bitki çeşitlerini sınıflandırabilmek için renksel ve dokusal öznitelikler birleştirilerek DVM sınıflandırıcısı ile sınıflandırma işlemi gerçekleştirilmiştir.

Bu tezde, yedi farklı kayısı türüne ait yaprak imgeleri kullanılarak kayısı türleri derin öğrenme ile sınıflandırılmıştır. Bu türler Apikoz Şalak, Çataloğlu, Çekirge İz, Hacıkızı, Hırmanlı, Paviot ve Tokatoğlu Erzincan'dır. Tezde kullanılan derin öğrenme yöntemlerinden biri olan Evrimsel Sinir Ağları ile %91,34 ±0,77 doğru sınıflandırma başarımları elde edilmiştir. Diğer bir derin öğrenme yöntemi olan Yerel Alıcı Alanlar-Aşırı Öğrenme Makinesi ile de %97,26±0,95 doğru sınıflandırma başarımları elde edilmiştir.

## **1.2. Problemin Tanımı**

Malatya Kayısı Araştırma Merkezinden alınan bilgiye göre Malatya ve Elazığ bölgesinde 285 farklı çeşit kayısı türü bulunmaktadır. Botanikçiler ve bitki uzmanları kayısı türlerinin sınıflandırılmasında geleneksel yöntemleri kullanmaktadırlar ve bu durum kayısı sınıflarının karıştırılması gibi bazı ciddi sıkıntılar yaratmaktadır. Herhangi bir yerde yeni yetişen kayısı ağacının hangi türe ait olduğuna karar vermekte uzmanlar zorlanmaktadırlar.

Aynı zamanda farklı disiplinlerden kayısı üzerinde çalışma yapan arařtırmacılar, kendilerine verilen tür adlarının yaptıkları analiz sonuçlarında farklı bir türe ait olduğunu gördüklerinde, yapılan tüm çalışmalarının bořa zaman kaybı olduğunu ifade etmişlerdir.

### **1.3. Tezin Amacı**

Yaprak tanıma, yaprakların türlerine göre sınıflandırılması amacıyla kullanılır. Böyle bir sınıflandırma işlemi özellikle otomatik bitki taksonomisi için temel oluşturacaktır. Bu sayede, uygun ve pratik bir bitki tanıma sistemi oluşturularak bitkilerin sınıflandırılması, anlaşılması ve yönetilebilmesi sağlanacaktır. Bu durum, özellikle botanik bilimiyle ilgilenenler için kolaylık sağlayacaktır. Bu kapsamda yapılan tezin temel amacı, bölgemizde yetişen ve çiftçilerimiz için önemli bir gelir kaynağı olan kayısı bitki türlerinin sınıflandırılmasıdır. Bu sayede bölgede yetişen kayısı türlerinin tanınması daha hızlı yapılacaktır.

### **1.4. Tezin Organizasyonu**

Tezin organizasyonu şöyledir: İkinci bölümde derin öğrenme hakkında bilgiler verilmiş ve yapay sinir ağlarına değinilmiştir. Üçüncü bölümde Evriřimsel Sinir Ağları hakkında bilgi verilmiştir. Dördüncü bölümde, Yerel Alıcı Alanlar-Ařırı Öğrenme Makinası hakkında bilgi verilmiştir. Beşinci bölümde geliştirilen yaprak tanıma sistemleri ile ilgili yapılan deneysel çalışmalar ve sonuçları anlatılmıştır. Altıncı bölümde ise, deneysel çalışmalar sonucunda elde edilen veriler ışığında, çalışma ile ilgili sonuçlar ortaya konularak gelecekte yapılması planlanan çalışmalara değinilmiştir.

## 2. DERİN ÖĞRENME

Yapay zekâyı kısaca tanımlarsak, makinelere insanlar gibi düşünme, karar verme, analiz etme gibi birtakım fonksiyonların kazandırılmasıdır [21]. Yapay zekânın fikir babası Alan Turing olarak bilinir. Alan Turing, “Makineler düşünebilir mi?” sorusuyla makine zekâsını tartışmaya açmıştır. Turing’den sonra Yapay Zekâ konusundaki ilk çalışmalar ise McCulloch ve Pitts tarafından yapılmıştır [22]. 2000’lerin başından itibaren Derin Öğrenme, yapay zekâ alanında yaygınlaşmaya başlamıştır [23,24]. Google, Microsoft ve Apple gibi önemli bilişim şirketleri ses ve görüntü tanımda derin öğrenme teknolojisini kullanmaya başlamışlardır. Yapay zekâ alanında yeni bir teknoloji olan derin öğrenme, görüntü ve ses tanımda farklı bir boyut kazandırmaktadır. İnsanlar için nesnelere tanımak ve onları belli bir kategoriye dâhil etmek oldukça basittir. Mesela bir insan elma ve armudun farkını ayırt edebilir ve bu iki nesneyi sınıflandırabilir. Ancak makineler için durum daha farklıdır. Derin öğrenme mimarisi insan beyni gibi bir karar mekanizmasına sahiptir.

Derin öğrenme algoritması öncelikle imgenin en alt katmanından başlayarak (piksellerden) bir sınıflandırma yapmaya çalışır. Eğer sınıflandırma yapılamaz ise bir üst katmana soyutlama yapar. Resimde en alt katman olan pikselleri, bir üst katmana yani sınır çizgilerine soyutlar. Derin öğrenme algoritmasının bir önceki katmandan soyutladığı her başarılı yeni katman, derin öğrenme algoritmasının tanıma ve sınıflandırma gücünü daha çok arttırmaktadır. Oluşturulan yeni sınıf ise en üst katmana yani sınıflandırmanın tamamlandığı katmana geçmiş olur [23,24]. Bu katmanlar gizlenmiş katmanlar olarak bilinir. Bu özellik derin öğrenme algoritmasını diğer algoritmalara göre üstün kılar.

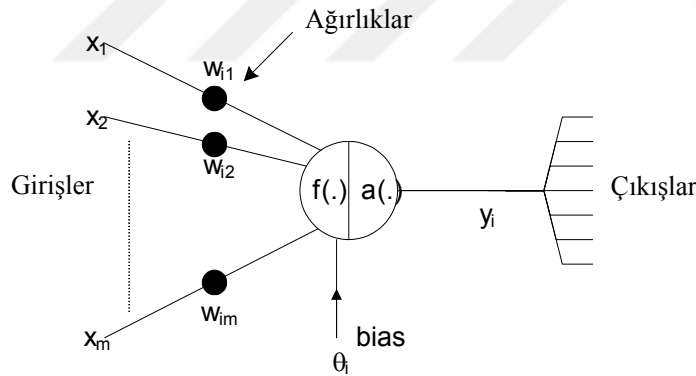
Kullanım şekline göre derin öğrenme mimarilerini üç sınıfa ayırmak mümkündür: Olasılıksal üretim modelleri, ayırım modelleri ve her ikisinin birlikte kullanıldığı melez modeller. Üretimsel derin mimariler, örüntü analizi ve sentez amaçları için, gözlenen verinin yüksek dereceli ilinti özniteliklerinin anlaşılmasını sağlar ve ilişkili olduğu sınıflarla ortak olasılıksal dağılımlarını karakterize eder. Bu sınıf mimariler Bayes kuralının kullanılması ile ayırıcı bir özellik kazanır. Eğitimi gözetimsiz öğrenme ile gerçekleştirilir. Bu yaklaşımda verinin modellenmesi amaçlanmaktadır. Ayırıcı özelliği olan derin mimariler, örüntü sınıflandırması için doğrudan ayırıcı özniteliklerin belirlenmesine yoğunlaşırlar. Doğal olarak sınıflandırma maksatlı özniteliklerin çıkarılması hedeflendiğinden, eğitim etiketli veri kullanılarak gözetimli şekilde gerçekleştirilir. Melez yapılar, en uygun ve düzenli sonuçlar



elde ederek ayırım yapmak amaçlandığında kullanılır. Bu yapılarda gözetimsiz olarak katmanların eğitilmesi sağlanır, tek istisna son katmandaki eğitimin veriye ait etiket bilgisi kullanılarak gözetimli şekilde yapılmasıdır.

## 2.1. Yapay Sinir Ağları

Birçok nörondan meydana gelen Yapay Sinir Ağları (YSA)'lar insan beyninin biyolojik yapısından ilham alınarak oluşturulmuş yapılardır. YSA'nın modeli Şekil 2.1'de gösterilmiştir. Bu nöronlar ağırlıklarla birleştirilerek kullanılırlar. Her bir ağırlık değeri, ağdan aktarılan sinyal ile çarpılmaktadır. Her nöron için çıkışlar aktivasyon fonksiyonu ile belirlenir. Literatüre çok fazla aktivasyon fonksiyonu sunulmuştur. Bunlar arasında tanjant sigmoid fonksiyonu en çok tercih edilen aktivasyon fonksiyonu olarak belirlenmiştir [25]. YSA'ların genelleme özellikleri vardır bu sayede ağa daha önce verilmemiş bir girdi ilk defa verildiğinde çıktı üretebilirler. YSA'nın çıkış değeri Denklem 2.1'de verildiği gibi hesaplanmaktadır.



Şekil 2.1 YSA'nın yapısı

$$y(t+1) = a\left(\sum_{j=1}^m w_{ij} x_j(t) - \theta_i\right), \quad f_i \triangleq net_i = \sum_{j=1}^m w_{ij} x_j - \theta_i \quad (2.1)$$

Denklem 2.1'de  $x=(xX1, xX2 \dots xXm)$  nöronun  $m$  giriş parametrelerini gösterir.  $w_i$ ,  $x_i$  girişi için ağırlıkları ve  $\theta_i$  bias değerini temsil eder.  $a(.)$  ise aktivasyon fonksiyonudur. YSA'nın paralel işlem yetenekleri olduğu için veri madenciliği, tıbbi uygulamalar ve örüntü tanıma gibi işlemlerde kullanılmaktadır. YSA modeli tasarlanırken ağın tipi, katman sayısı

ve her katmandaki nöron sayısı belirlendikten sonra işleme başlanmalıdır. Günümüzdeki yüksek kapasiteye sahip bilgisayarlar ile bu işlemler kolayca modellenmektedir [26,27].

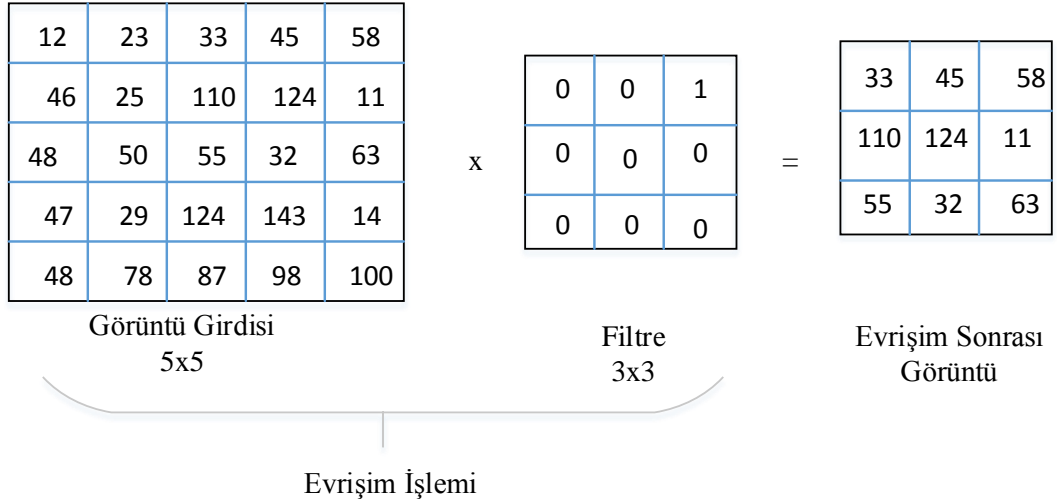


### 3. EVRİŞİMSEL SINIR AĞLARI

Evrişimsel Sinir Ağları (ESA), ileri beslemeli yapay sinir ağlarının en gelişmiş halidir ve nesne tanımada sıklıkla kullanılan bir yöntemdir. Evrişimsel Sinir Ağları (ESA), önceki bölümde bahsedilen sıradan yapay sinir ağlarına benzemekle beraber öğrenilebilir tercihlere ve yanlılıklara sahip olan nöronlardan oluşmaktadır [28-31]. ESA'da veri parçalara ayrılır ve her parçaya filtre uygulanır, uygulanan filtrenin boyutuna göre resimde küçülmeler olur. Örneğin;  $8 \times 8$  boyutundaki bir filtre resmin  $8 \times 8$  alanındaki bir bölümüne uygulandıktan sonra çıkan sonuç bir pikselde tutulur ve bunun sonucu olarak çıktı  $1 \times 1$  boyutunda olur. Ortaya çıkan bu pikseller birbiriyle ilişkilendirilerek tanımlama yapılmaya çalışılır. ESA'da ham görüntü piksellerinden çıkışa kadar tüm ağ tek bir bütün şekilde türevlenebildiği için parametrelerin miktarı büyük oranda azalmaktadır. ESA'lar eğitilebilen birçok seviyeden oluşmaktadır ve her basamak da kendi içerisinde filtre banka katmanı, doğrusal olmayan katman ve havuzlama katmanı olmak üzere üç katmandan oluşmaktadır[33]. ESA'nın katmanları genişlik, yükseklik ve derinlik olmak üzere 3 boyutludur. ESA'daki tüm nöronların birbirleriyle tamamen bağlantılı olmayıp, sonraki katman önceki katmanın küçük bir bölümüne bağlantılıdır[33].

#### 3.1. Evrişim Katmanları

Evrişim katmanları, ESA'nın ana katmanıdır. Bu katmanlar bir dizi öğrenilebilir filtrelerden oluşmaktadır. Her filtre ileriye geçişte evrişim işleminden sorumludur. Bu üretim işleminden sonra evrişim katmanının büyüklüğüne ve boyutuna bağlı olarak birkaç özellik haritası oluşturur. Bu haritalar bir katmanda depolanır ve hepsi kendi filtrelerine aittir [34]. Girdideki bazı üç boyutlu pozisyonlarda spesifik özellikler gördüklerinde ağlar filtreyi otomatik olarak öğrenir. *Şekil 3.1'de* görüldüğü üzere,  $5 \times 5$  boyutundaki girdi verisi üzerinde belirlenen özellikteki  $3 \times 3$  boyutundaki filtre gezdirilerek evrişim operatörünün uygulanışı gösterilmiştir.



**Şekil 3.1** Evrşim operatörünün uygulanması

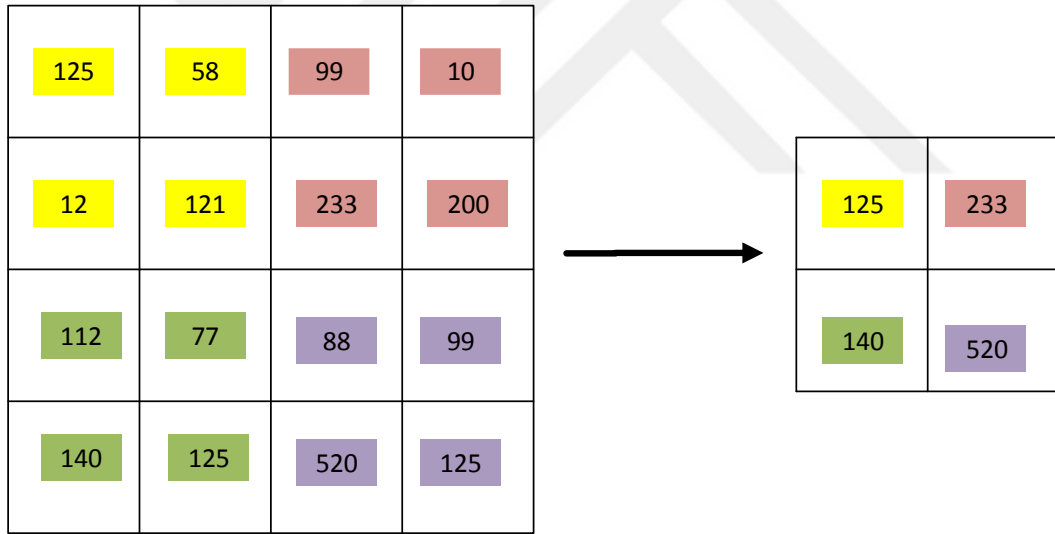
### 3.2. Aktivasyon İşlemleri

YSA’larda aktivasyon işlevinin amacı nöronun girdi değerini çıktı değerine dönüştürmektir [35,36]. Genellikle, YSA’larda aktivasyon işlevi sigmoidal fonksiyonu kullanılarak yapılmaktadır. “S” şeklinde bir kıvrım olan Sigmoid fonksiyon işlevi sigmoid ve tanjant hiperbolik işlevini içerir. Sigmoid fonksiyonu minimum ve maksimum değerlere sahiptir. Bu değerler nöral ağın yüksek katmanlarında doymuş nöronlara sebep olurlar. Bu doymuş nöronlar nöral ağın derinliğini belirlemektedir[36]. Bir diğer önemli aktivasyon işlevi düzeltici doğrusal birim işlevidir. Bu işlev sigmoid operatöründen farklıdır. Bunun Denklem 3.1’de verilen maksimum işlev olduğunu söyleyebiliriz. Bu yalnızca minimum değer olan sıfır ile sınırlandırılır ve herhangi bir negatif olmayan gerçek değeri sembolize eder. Bu işlev aynı zamanda gerçek bir sıfır aktivasyon değerine sahip olmak için iyi seyreklik özelliklerine sahiptir.

$$Rect(x) = maks(0; x) \quad (3.1)$$

### 3.3. Havuzlama Katmanı

ESA yapısında ardışık evrişim katmanlarının arasına periyodik olarak havuzlama katmanı eklemek oldukça yaygındır. Havuzlama katmanının işlevi ağdaki hesap ve parametrelerin miktarını azaltmak için verinin hacimsel büyüklüğünü sürekli olarak azaltmaktır. Havuzlama katmanı girdinin her bir derinlik diliminde bağımsız olarak çalışır ve maksimum uygulamayı kullanarak hacimsel olarak yeniden şekillendirir. Bir havuzlama operatörü bireysel özellik kanallarında uygun bir operatör uygulamasıyla yakın özellik değerlerini birleştirerek çalışır. Maksimum ve toplam operatörleri ile havuzlama işlemi sıklıkla yapılmaktadır [38]. Örnek olarak havuzlama katmanına gelen girdinin boyutu 224x224x64 ise 2x2 filtre ve 2 birim adıma sahip maksimum havuzlama sonunda 112x112x64 boyutuna dönüşecektir. Havuzlama katmanında derinlik boyutu değişmemektedir.



Şekil 3.2 2x2 filtre ve 2 birim adıma sahip maksimum havuzlama

### 3.4. Tamamen Bağlanmış Ağ

Tamamen bağlanmış bir katmandaki nöronlar, YSA'larda görüldüğü gibi, önceki katmanlardaki tüm aktivasyonlarla tam bağlantı halindedir. Bu yüzden onların aktivasyonu sapma ofseti tarafından takip edilen dikey çarpım ile yönetilebilmektedir.

### 3.5. Çıktı Katmanı

Çıktı katmanı, çıktı grupları üzerinden olasılık dağılımını verir. Bu katmanda kullanılacak birkaç kayıp işlev vardır. Softmax, Sigmoid ve Öklid bunlardan bazılarıdır. Softmax en yaygın kullanılan kayıp fonksiyondur. Softmax aynı zamanda  $\log$  olasılık dağılımını kullandığı için doğrusal bir sınıflayıcıdır [36].

### 3.6. ESA'ların Eğitimi

ESA'lar, neredeyse her makine öğrenimi probleminde olduğu gibi danışmanlı öğrenme ile eğitilir. ESA'ları eğitmek için pek çok farklı algoritma vardır. ESA'ların eğitiminde genellikle geri yayılım algoritması kullanılmaktadır. Evrişim katmanının tek özellik haritasında tüm nöronlar aynı boyuta sahiptir. Bu yüzden; evrişim katmanı tam birleşik çok katmanlı perseptron katmanından daha az parametreye sahiptir ve bu boşluk test ile eğitim arasındaki farkın azalmasını sağlamaktadır. Bu çalışmada ağırlarımızı eğitmek için geri yayılım algoritması kullanılmıştır.

#### 3.6.1. Geri Yayılım Algoritması

ESA eğitiminde, genellikle toplu gradyant azaltım (batch gradiend descent) algoritması kullanılmaktadır.  $m$  adet örneklemin bulunduğu bir eğitim kümesinin  $\{x^{(1)}, y^{(1)}, x^{(2)}, y^{(2)}, \dots, x^{(m)}, y^{(m)}\}$  tek bir örneğe ait  $(x, y)$  maliyet fonksiyonu denklem 3.2'deki formül ile hesaplanmaktadır.

$$J(W, b; x, y) = \frac{1}{2} \|h_{w,b}(x) - y\|^2 \quad (3.2)$$

Tüm örneklemlerin toplam maliyet fonksiyonu ise;

$$J(W, b) = \left[ \frac{1}{m} \sum_i^m j(W, b; x^{(i)}, y^{(i)}) \right] + \lambda/2 \sum_l^{n_l-1} \sum_i^{s_l} \sum_j^{s_{l+1}} (W_{ji}^l)^2 \quad (3.3)$$

fonksiyonu ile hesaplanır.

$J(W, b)$  maliyet fonksiyonunun birinci terimi toplam kare hatasıdır. İkinci terimi ise ağırlık çürüme (weight decay) terimidir. Çürüme terimi, overfitting probleminin oluşması ve toplam maliyet fiyatının azaltılması arasında dengeyi sağlamaktadır.

Tanımlanan maliyet fonksiyonu sınıflandırma ve regresyon problemlerinde kullanılmaktadır. Sınıflandırma problemlerinde  $y = 0$  veya  $1$  değerini almakta (hiperbolik tanjant fonksiyonunda  $[-1,1]$ ), regresyon problemlerinde  $[0,1]$  aralığında değer almaktadır.

$(W$  ve  $b)$  parametrelerine bağlı  $J(W, b)$  fonksiyonu ile hesaplanan toplam maliyetin azaltılması kullanılacak parametreler rassal olarak başlatılmakta sonrasında optimizasyon algoritmaları kullanılarak güncellenmektedir. Toplu gradyant azaltımı ile yapılan optimizasyonda, parametrelerin güncellenmesi maksadıyla kısmi türevin hesaplanmasında geri yayılım algoritması kullanılır. Toplam maliyet fonksiyonunun  $(W, b)$  parametrelerine bağlı kısmi türevleri denklem 3.4 ve 3.5'deki gibi hesaplanır.

$$\frac{\delta J(W,b)}{\delta W_{ij}^l} = \left[ \frac{1}{m} \sum_{i=1}^m \frac{\delta J(W,b;x^i,y^i)}{\delta W_{ij}^l} \right] + \lambda W_{ij}^l \quad (3.4)$$

$$\frac{\delta J(W,b)}{\delta b_i^l} = \left[ \frac{1}{m} \sum_{i=1}^m \frac{\delta J(W,b;x^i,y^i)}{\delta b_i^l} \right] \quad (3.5)$$

Bu iki terim arasındaki fark, ağırlık çürüme teriminin, sonuç üzerinde büyük bir etkisi olmaması nedeniyle  $b$  parametresine uygulanmamasıdır.

Geri yayılım algoritmasında;

1. Gizli katmanlardaki hatanın hesaplanmasında öncelikle her bir katmanın kendine ait aktivasyonu ele alınır.

2. Çıktı katmanında bulunan herbir birim için hata terimi denklem 3.6'daki gibi hesaplanır.

$$\delta_i^{n_l} = \frac{\delta \left( \frac{1}{2} \|y - h_{W,b}(x)\| \right)}{\delta Z_i^{n_l}} = -(y - a_i^{n_l}) \cdot f'(Z_i^{n_l}) \quad (3.6)$$

3.  $l = n_{l-1}, n_{l-2}, n_{l-3}, \dots, 2$  gizli katmanlardaki tüm düğümlerde hata terimleri denklem 3.7'deki gibi hesaplanır.

$$\delta_i^{n_l} = \left( \sum_j^{s_{l+1}} W_{ji}^l \delta_j^{l+1} \right) \cdot f'(Z_i^{n_l}) \quad (3.7)$$

4. Kısmi türevler hesaplanır.

$$\frac{\delta J(W,b;x,y)}{\delta W_{i,j}^l} = a_j^l \cdot \delta_i^{l+1} \quad (3.8)$$

$$\frac{\delta J(W,b;x,y)}{\delta b_i^l} = \delta_i^{l+1} \quad (3.9)$$

5. Yukarıdaki denklemlerde belirtilen hesaplamaların yapılmasına müteakip parametrelerin güncellemesi gradyant inişi yöntemi ile gerçekleştirilir.

### 3.7. İlgili Yapılar

ESA ağ yapısını kullanan birkaç yapı vardır ve bunların en yaygın olanları şunlardır:

**LeNet;** Evrimsel Ağların ilk başarı uygulamalarından olan LeNet 1990'larda Yann LeCun tarafından geliştirilmiştir [28].

**AlexNet:** Bilgisayar görmesinde Evrimsel Ağlarda popüler ilk çalışma Alex Krizhevsky, Ilya Sutskever ve Geoff Hinton tarafından geliştirilen AlexNet'dir[30]. Ağın yapısı LeNet ile benzerlikler taşımaktadır. Fakat LeNet'e göre daha derin, daha büyük ve birbirinin üzerine yığılmış evrim katmanları içermektedir [29].

**ZF Net:** AlexNet'in geliştirilmiş halidir. AlexNet'in yapısal hiper parametreleri üzerinde küçük düzenlemeler yapılmış, orta evrim katmanlarının boyutları genişletilmiş ve ilk katmandaki filtre boyutları küçültülmüştür [30].

**GoogLeNet:** Bu ağın asıl katkısı ağdaki parametrelerin sayısını ciddi ölçüde azaltan Inception(başlangıç) modülünün gelişmesidir. GoogLeNet'in devamı niteliğinde birkaç versiyonu daha bulunmaktadır ve en yakın zamanlı olanı Inception-v4'dür [28-30].

**VGGNet:** VGGNet, Karen Simonyan ve Andrew Zisserman tarafından geliştirilen ağıdır. VGGNet'in başlıca katkısı ağın derinliğinin iyi bir performans için önemli bir parça olduğunu göstermesidir. Ön eğitilmiş modelleri tak-çalıştır kullanım için ulaşılabilir konumdadır. VGGNet'in dezavantajı çok pahalı olması, çok fazla hafıza ve parametre (140M) kullanmasıdır. Bu parametrelerin çoğu ilk tam bağlantılı katmanda bulunmaktadır [28-30].



**ResNet:** Kaiming He ve arkadaşları tarafından geliştirilen ađ yapısıdır. ResNet'in özelliđi yığın normalizasyonunu yoğun şekilde kullanması ve bağlantıları atlamasıdır. Bu yapı, aynı zamanda tam bağlantılı katmanları ađın sonunda kaybetmektedir. ResNet'ler, şunda ESA modellerinde en gelişmiş olan ve sıklıkla çalışmalarda kullanılan ađ yapısıdır [28-30].



## 4. YEREL ALICI ALANLAR AŞIRI ÖĞRENME MAKİNESİ

### 4.1. Aşırı Öğrenme Makinesi

Huang ve arkadaşları tarafından önerilen Aşırı Öğrenme Makinesi (AÖM), Tek Gizli Katmanlı ve İleri Beslemeli (TGKIB) bir sinir ağı modeli için kullanılan bir öğrenme algoritmasıdır [39]. AÖM, geriye yayılım algoritmasının bazı dezavantajlarını gidermek için önerilmiş popüler bir sınıflandırma yöntemidir. AÖM, giriş ağırlıkları rastgele çıkış ağırlıkları ise analitik olarak hesaplanan TGKIB bir YSA modelidir [40]. AÖM’de gizli katmanda sigmodial, Gaussian ve hard-limitin gibi aktivasyon fonksiyonları kullanılırken çıkış katmanında doğrusal fonksiyon kullanılmaktadır. Geleneksel TGKIB bir YSA’da ağı eğitilmesi iteratif olurken, bu süreç AÖM’de analitik bir denkleme dönüştürülmüştür [41]. AÖM iyi genelleme performansı, aşırı hızlı öğrenme yeteneği ve düşük işlem karmaşıklığı gibi önemli avantajlara sahiptir. Belirli bir N girdi-çıkış ilişkisi  $x_i$  ve  $t_i$  parametreleri kabul edilsin. Burada  $x_i$ ,  $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in R^n$  ve  $t_i$  ise  $t_i = [t_{i1}, t_{i2}, \dots, t_{in}]^T \in R^m$  şeklinde ifade edilir.  $\tilde{N}$  tane gizli sinir hücresi ve  $g(x)$  aktivasyon fonksiyonuna sahip standart TGKIB ağ modeli matematiksel olarak denklem 4.1’deki gibi modellenir:

$$\sum_{i=1}^{\tilde{N}} \beta_i g(w_i \cdot x_j + b_i) = o_j, \quad j = 1, \dots, N \quad (4.1)$$

Burada  $w_i = [\omega_{i1}, \omega_{i2}, \dots, \omega_{in}]^T$ ,  $i$ ’ninci gizli katman sinir hücresine ve giriş sinir hücresine bağlı olan ağırlık vektörüdür,  $\beta_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{im}]^T$  ise  $i$ ’ninci gizli sinir hücresine ve çıkış sinir hücrelerine bağlı olan ağırlık vektörüdür.  $b_i$  ise  $i$ ’ninci gizli sinir hücresinin eşik değeridir.  $w_i \cdot x_j$  ise  $w_i$  ve  $x_j$ ’nin içsel çarpımını ifade eder.  $\tilde{N}$  tane gizli sinir hücresine ve  $g(x)$  aktivasyon fonksiyonuna sahip standart TGKIB ortalama sıfır hataya yaklaşabilir,  $\sum_{j=1}^{\tilde{N}} \|o_j - t_j\| = 0$ ,  $\beta_i$ ,  $w_i$  ve  $b_i$  arasındaki ilişki denklem 4.2’de verilmiştir;

$$\sum_{i=1}^{\tilde{N}} \beta_i g(w_i \cdot x_j + b_i) = t_j, \quad j = 1, \dots, N. \quad (4.2)$$

Denklem 4.2’de verilen  $N$  tane eşitlik, Denklem 4.3’deki gibi kısaltılabilir:

$$H\beta = T \quad (4.3)$$

Burada;

$$H = \begin{bmatrix} g(w_1 \cdot x_1 + b_1) & \cdots & g(w_{\tilde{N}} \cdot x_1 + b_{\tilde{N}}) \\ \vdots & \cdots & \vdots \\ g(w_1 \cdot x_N + b_1) & \cdots & g(w_{\tilde{N}} \cdot x_N + b_{\tilde{N}}) \end{bmatrix}_{N \times \tilde{N}} \quad i\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_{\tilde{N}}^T \end{bmatrix}_{\tilde{N} \times m} \quad \text{ve } T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N \times m} \quad (4.4)$$

Burada T ağırlık çıkışı, H gizli katman çıkış matrisini ve  $\beta$  ise çıkış ağırlıklarını temsil etmektedir.

## 4.2. Yerel Alıcı Alanlar Aşırı Öğrenme Makinesi

Geriye yayılım öğrenme algoritması kullanan Evrişimsel sinir ağlarının (ESA) başarılı nesne tanıma uygulamaları mevcuttur. ESA'nın eğitim sürecinin uzun olması ve eğitim sürecinde yerel bir çözüme takılma ihtimali önemli dezavantajları olarak bilinir [42]. Yerel alıcı alanlar (YAA) bilgisinin AÖM içine entegre edildiği YAA-AÖM yapısı, ESA'ya alternatif bir model olarak önerilmiştir[43]. Yerel Alıcı Alanlar Aşırı Öğrenme Makinesi (YAA-AÖM) yeni bir derin öğrenme kavramı olup, temel olarak birbirinden bağımsız iki farklı yapıyı bünyesinde barındırmaktadır [42-44]. Bu yapılardan ilkin, öğrenmenin olmadığı, evrişim ve toparlama süreçlerinin olduğu yapı oluşturmaktadır. İkinci yapı ise en küçük kareler çözümü ile  $\beta$ 'nin analitik hesaplanması aşamasıdır. Birinci yapıda kullanılan toparlama için kare/karekök işlevi kullanılmaktadır.

**Birinci yapı:** Daha öncede belirtildiği gibi bu bölümde herhangi bir öğrenme olmamaktadır. Diğer bir ifade ile bu bölümde ağırlık yenilenmesine ihtiyaç duyulmamaktadır. Kullanılan K adet evrişim filtresinin katsayıları başlangıçta rasgele seçilir. Eğer öznitelikleri çıkarılacak giriş imgesinin boyutu  $d \times d$  ve evrişim filtresi boyutu  $r \times r$  ise, evrişim katmanı sonunda  $(d - r + 1) \times (d - r + 1) \times K$  boyutlu öznitelik haritası elde edilir [42-44]. Toparlama katmanında, verilen bir pencere boyutunda öznitelikler toparlanır. Evrişim katmanında  $Y$  sonraki evrişim katmanının değerini,  $X$  de önceki katmanın değeri olarak kabul edilirse evrişim denklemi denklem 4.5'deki gibi ifade edilir.

$$\gamma_{i,j} = g \left( \sum_{m=1}^r \sum_{n=1}^r x_{i+m-1,j+n-1} \cdot w_{m,n} + b \right), \quad i, j = 1, \dots, (d - r + 1) \quad (4.5)$$

Evrişim işleminden sonra yapılan toparlama işlemi verinin boyutunu indirgemek için kullanılmaktadır. Bu yolla gereksiz bilgiler kullanılmayarak sadece önemli bilgilerin toparlanması amaçlanmaktadır.

**İkinci yapı:** Bir önceki yapıda her bir giriş imgesi için elde edilen öznitelikler bir matriste birleştirilerek eğitim veri seti elde edilmişti. Bu yapıda ise sadece AÖM'nin gizli katman ile çıkış arasındaki ağırlık vektörü  $\beta$ 'nin analitik olarak hesaplanmasını gerektirir [42-44]. Diğer bir ifade ile önceki aşamada elde edilen öznitelik matrisi,  $H \in R^{N \times (d-r+1)^2}$  matrisi olarak kabul edilir ve  $\beta$  analitik olarak denklem 4.6'daki gibi hesaplanır;

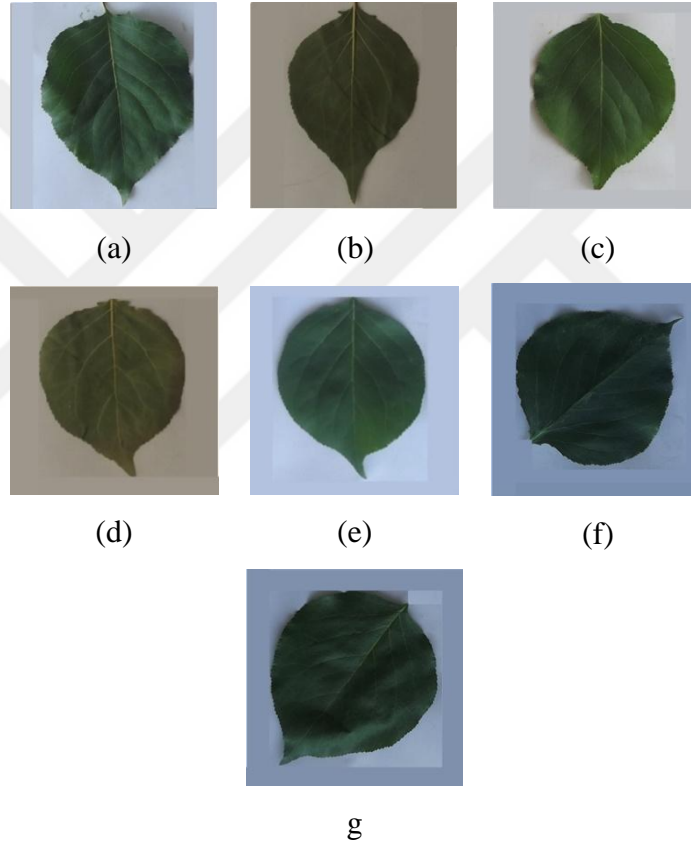
$$\beta = \begin{cases} H^T \left( \frac{I}{C} + HH^T \right)^{-1} T & \text{eğer } N \leq K \cdot (d-r+1)^2 \\ \left( \frac{I}{C} + HH^T \right)^{-1} H^T T & \text{eğer } N > K \cdot (d-r+1)^2 \end{cases} \quad (4.6)$$

Denklem 4.6'da eğitim veri seti için sınıf etiketleri T, birim matrisi I ve regülasyon katsayısı C olarak adlandırılır.

## 5. DENEYSEL ÇALIŞMALAR

### 5.1. Veri Tabanı

Bu tezde, Malatya ilinde bulunan Gıda Tarım ve Hayvancılık Bakanlığına bağlı Kayısı Araştırma Enstitüsü Müdürlüğü bahçesinde yetişen kayısı türüne ait 7 farklı genotip için veri tabanı oluşturulmuştur [44,45]. Bu türler sırası ile Apikoz Şalak, Çataloğlu, Çekirge iz, Hacıkızı, Hırmanlı, Paviot ve Tokatoğlu Erzincan'dır. Şekil 5.1'de her bir türe ait bir resim verilmiştir.



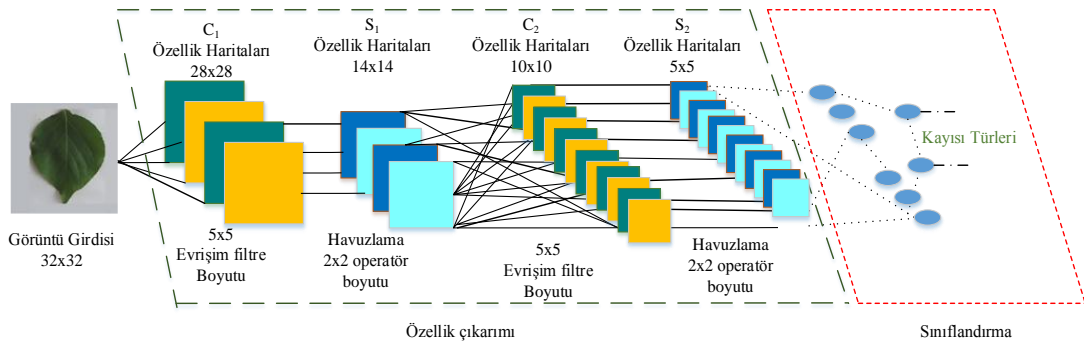
**Şekil 5.1** Kullanılan kayısı yaprağı imgeleri, a) Apikoz Şalak, b) Çataloğlu, c) Çekirge iz, d) Hacıkızı, e) Hırmanlı, f) Paviot ve g) Tokatoğlu Erzincan

Bu genotiplere ait ağaçlardan olgunlaşmış yapraklar ziraat mühendisi ile birlikte toplanarak görüntüleri veri tabanına kaydedilmiştir. Her yaprak, beyaz bir fon üzerine farklı açılarla yerleştirilmiş ve sabit 30 cm yükseklikten görüntüleri alınarak veri tabanına kaydedilmiştir. Bu yükseklikten, ölçüleme (kalibrasyon) nesnesi olarak kullanılan uzunluğu bilinen bir nesnenin (madeni para) görüntüsü alınmış, paranın resimdeki boyutları

ile gerçek boyutları karşılaştırılmış ve değerlerin aynı olduğu görüldüğünden kullanılan yükseklik geçerli kılınmıştır. Elde edilen resimlerin boyutu 256x256 dır.

## 5.2. ESA'ya ait Deneysel Parametreler ve Sonuçlar

Çalışmada 7 türe ait toplam 356 adet imge kullanılmıştır. Kullanılan bütün imgeler ilk önce gri tona çevrilmiş ve 32x32 boyutuna yeniden boyutlandırılmıştır. Bu imgelerden rasgele seçilen 210 adedi ESA'nın eğitiminde, geri kalan 146 adet imge ise ESA'nın başarımlı testinde kullanılmıştır. Kullanılan bütün imgeler ilk önce gri tona çevrilmiştir. Kullanılan ESA modeli 6 katmandan oluşmaktadır. İlk katman giriş katmanıdır. İkinci katman 6 adet evrişim filtresi bulunduğu evrişim katmanı olarak adlandırılır. Daha sonraki katman alt örnekleme katmanı olup, bu katmanda yarı örnekleme stratejisi benimsenmiş ve maksimum değerleri alan (max) operatörü kullanılmıştır. Dördüncü katman yine bir evrişim katmanı olup, 12 adet evrişim filtresini barındırmaktadır. Beşinci katman yine bir alt örnekleme katmanı olup, yarı örnekleme kullanılmıştır. Son katman ise yine bir evrişim katmanı olup, 12 adet evrişim filtresini barındırır. Evrişim katmanlarında kullanılan filtrelerin boyutu 5x5 pencere boyutu seçilmiştir. Aktivasyon fonksiyonu olarak signum fonksiyonu kullanılmıştır. Kullanılan ESA modeli Şekil 5.2'de gösterilmiştir [45].

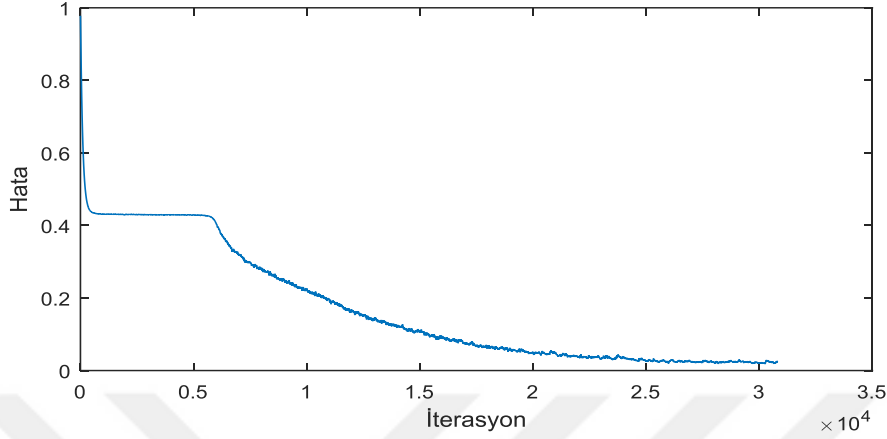


Şekil 5.2 Kullanılan ESA'nın yapısı

Yöntemin uygulanması için kullanılan bilgisayar 3.50GHz işlemci ve 8 GB bellek özelliklerine sahiptir. Benzetim sonuçları MATLAB 2016 ortamında yapılmıştır.

ESA, 50 kez çalıştırılmış ve en yüksek doğruluk oranı %92,26 olarak en düşük doğruluk oranı ise %90,13 olarak bulunmuştur. Ortalama doğruluk oranı %91,34 ±0,77 olarak hesaplanmıştır.

Ayrıca Şekil 5.3’de ESA’nın eğitimi sırasında oluşan hatanın iterasyona bağlı değişimi gösterilmiştir.

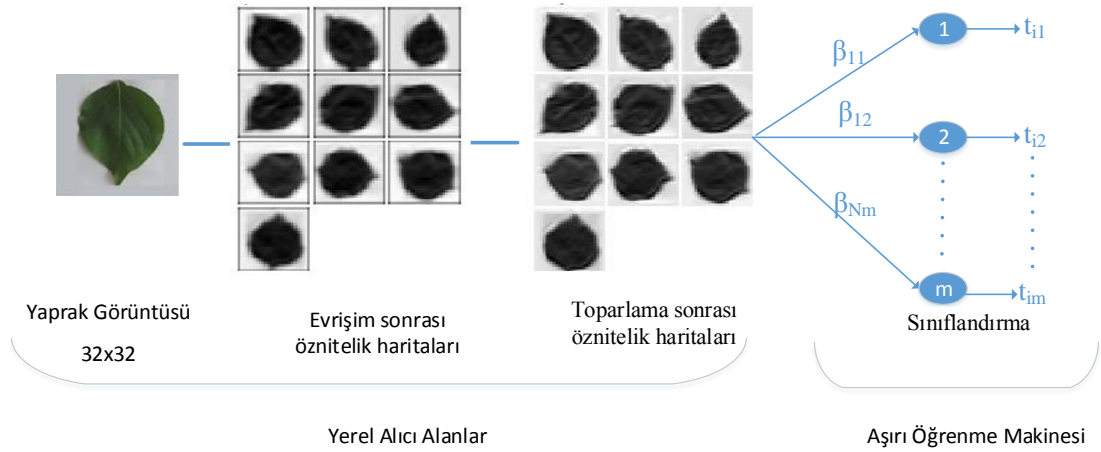


Şekil 5.3 Hata ve iterasyon

Şekil 5.3’de görüldüğü gibi iterasyon sayısı arttıkça hata miktarı azalmaktadır.

### 5.3. YAA-AÖM’e ait Deneysel Parametreler ve Sonuçlar

Bu yöntem için ESA’da kullanılan veri seti kullanılmıştır. Bu imgelerden rasgele seçilen 210 adedi YAA-AÖM’nin eğitiminde, geri kalan 146 adet imge ise YAA-AÖM’nin başarımlarında kullanılmıştır. Kullanılan bütün imgeler ilk önce gri tona çevrilmiş ve 32x32 boyutuna yeniden boyutlandırılmıştır. YAA-AÖM’nin ayarlanabilir 4 parametresi mevcuttur. Bunlar sırası ile evrişim filtresi boyutu  $r$ , evrişim filtresi sayısı  $K$ , toplama boyutu ve  $C$  regülasyon katsayısıdır.  $K$  değeri 10, evrişim filtresi boyutu 3, toplama filtresi boyutu 2 olarak belirlenmiştir[42,44]. Şekil 5.4’de YAA-AÖM’nin blok diyagramı gösterilmiştir.



Şekil 5.4 YAA-AÖM'nin blok diyagramı

Yapılan deneylerde en iyi regülasyon katsayısının 5 olduğu görülmüştür. Ayrıca yığın boyutu 50 olarak seçilmiştir. Yöntemin uygulanması için ESA ile aynı şartlarda gerçekleştirilmiştir.

YAA-AÖM, 50 kez çalıştırılmış ve en yüksek doğruluk oranı %98,51 olarak en düşük doğruluk oranı ise %95,2 bulunmuştur. Ortalama doğruluk oranı %97,26±0,95 olarak bulunmuştur.

#### 5.4. ESA ve YAA-AÖM'ye Ait Performans Sonuçları

Kullanılan yöntemlerin başarımının değerlendirilmesinde daha iyi karar verebilmek amacıyla duyarlılık ve özgüllük gibi yaygın performans kriterleri de kullanılmıştır.

$$Duyarlılık = \frac{DP}{DP+YN} \quad (5.1)$$

$$Özgüllük = \frac{DN}{DN+YP} \quad (5.2)$$

Hesaplamalarda kullanılan, DP Doğru Pozitifleri, DN Doğru Negatifleri, YP Yanlış Pozitifleri ve YN Yanlış Negatifleri göstermektedir.

Apikoz şalak bir tür geriye kalan türler de tek bir tür kabul edilerek hesaplanan performans sonuçları Tablo 5.1'de sunulmuştur.



**Tablo 5.1.** Apikoz şalak ve diğer türlerin sınıflandırılmasına ait performans sonuçları

<b>Yöntem</b>	<b>Duyarlılık</b>	<b>Özgüllük</b>
ESA	%94,12	%95,78
YAA-AÖM	%95,43	%95,89

Çataloğlu bir tür geriye kalan türler de tek bir tür kabul edilerek hesaplanan performans sonuçları Tablo 5.2’de sunulmuştur.

**Tablo 5.2.** Çataloğlu ve diğer türlerin sınıflandırılmasına ait performans sonuçları

<b>Yöntem</b>	<b>Duyarlılık</b>	<b>Özgüllük</b>
ESA	%94,23	%92,67
YAA-AÖM	%96,28	%97,16

Çekirge iz bir tür geriye kalan türler de tek bir tür kabul edilerek hesaplanan performans sonuçları Tablo 5.3’de sunulmuştur.

**Tablo 5.3.** Çekirge iz ve diğer türlerin sınıflandırılmasına ait performans sonuçları

<b>Yöntem</b>	<b>Duyarlılık</b>	<b>Özgüllük</b>
ESA	%91,19	%92,77
YAA-AÖM	%94,34	%95,45

Hacıkızı bir tür geriye kalan türler de tek bir tür kabul edilerek hesaplanan performans kriterleri Tablo 5.4’de sunulmuştur.

**Tablo 5.4.** Hacıkızı ve diğer türlerin sınıflandırılmasına ait performans sonuçları

<b>Yöntem</b>	<b>Duyarlılık</b>	<b>Özgüllük</b>
ESA	%97,19	%98,75
YAA-AÖM	%98,67	%98,89

Hırmanlı bir tür geriye kalan türler de tek bir tür kabul edilerek hesaplanan performans sonuçları Tablo 5.5’de sunulmuştur.

**Tablo 5.5.** Hırmanlı ve diğer türlerin sınıflandırılmasına ait performans sonuçları

<b>Yöntem</b>	<b>Duyarlılık</b>	<b>Özgüllük</b>
ESA	%88,56	%89,12
YAA-AÖM	%90,54	%91,65

Paviot bir tür geriye kalan türler de tek bir tür kabul edilerek hesaplanan performans kriterleri Tablo 5.6’de sunulmuştur.

**Tablo 5.6.** Paviot ve diğer türlerin sınıflandırılmasına ait performans sonuçları

<b>Yöntem</b>	<b>Duyarlılık</b>	<b>Özgüllük</b>
ESA	%88,56	%89,12
YAA-AÖM	%90,54	%91,65

Tokatoğlu Erzincan bir tür geriye kalan türler de tek bir tür kabul edilerek hesaplanan performans kriterleri Tablo 5.7’de sunulmuştur.

**Tablo 5.7.** Tokatoğlu Erzincan ve diğer türlerin sınıflandırılmasına ait performans sonuçları

<b>Yöntem</b>	<b>Duyarlılık</b>	<b>Özgüllük</b>
ESA	%88,56	%89,12
YAA-AÖM	%90,54	%91,65

## 6. SONUÇLAR

Bu çalışmada yedi farklı kayısı türü, bu türlere ait yaprak imgeleri kullanılarak sınıflandırılmıştır. İlk olarak sınıflandırma için 6 katmanlı bir ESA modeli kullanılmıştır. Gerçekleştirilen deneysel çalışmalarda bir doğruluk elde edilmiştir. Kullanılan ESA modeli, 6 katmanlı olup bir giriş katmanı, 3 evrişim katmanı ve iki alt örnekleme katmanı mevcuttur. Önerilen yöntemin başarımlarını değerlendirme ile ilgili deneysel çalışmalar Matlab ortamında gerçekleştirilmiştir. Elde edilen sonuçlar  $91,34 \pm 0,77$  doğruluk göstermektedir.

Daha sonra YAA-AÖM yapısı ile yaprak tanıma gerçekleştirilmiştir. Yöntem, ESA ile karşılaştırıldığında daha basit ve daha kullanışlıdır. Eğitim süreci herhangi bir iterasyon gerektirmediği için zaman almamaktadır. Öğrenilmesi gereken parametre sayısı olarak da düşünüldüğünde yine önemli avantajlar sunmaktadır. YAA-AÖM yapısı ile yaprak tanıma aynı veri setine uygulanmış ve  $97,26 \pm 0,95$  doğruluk elde edilmiştir. Elde edilen sonuçlardan, YAA-AÖM yapısının nesne tanıma uygulamalarında kullanılabilir önemli bir araç olduğu sonucu çıkarılmıştır.

ESA'nın ve YAA-AÖM'nin kullanıldığı bazı uygulamalarda örnek sayısının on binlere ulaşmaktadır. Bu doğrultuda bizde ileri ki zamanlarda veri setimizde tür çeşitliliğini arttırmayı ve türlere ait imgelerin sayısını çoğaltmayı planlamaktayız. Ayrıca bu tez çalışmasını bütün kayısı türlerinin sınıflandırılabilmesini kapsayacak şekilde ilerletmeyi planlamaktayız.

## KAYNAKLAR

- [1] Novotny P. and Suk T. “Leaf recognition of woody species in Central Europe” Biosystems Engineering, Volume 115, Issue 4, August 2013, Pages 444–452.
- [2] Turkoglu, M., Hanbay D., “Destek vektör makinaları kullanılarak yaprak tanıma ile üzüm çeşitlerinin sınıflandırılması”, Signal Processing and Communications Applications Conference (SIU), 2015, 23th, vol., no., pp. 2674 – 2677, 16-19 May 2015, Malatya.
- [3] Stephen Gang Wu, Forrest Sheng Bao, Eric You Xu, Yu-Xuan Wang, Yi-Fan Chang, Qiao-Liang Xiang, “A Leaf Recognition Algorithm for Plant Classification Using Probabilistic Neural Network”, International Symposium on Signal Processing and Information Technology, IEEE, pp 11-16, 2007.
- [4] Stephen Gang Wu, Forrest Sheng Bao, Eric You Xu, Yu-Xuan Wang, Yi-Fan Chang, Qiao-Liang Xiang, “A Leaf Recognition Algorithm for Plant Classification Using Probabilistic Neural Network”, International Symposium on Signal Processing and Information Technology, IEEE, pp 11-16, 2007.
- [5] Kir, B., Oz, C., Gulbag, A., “K-Nn Sınıflandırma Algoritması Kullanılarak Yaprak Tanıma”, Signal Processing and Communications Applications Conference (SIU), 2012, 20th, vol., no., pp. 1 – 4, 18-20 April 2012, Mugla.
- [6] Avcı, T., Kökdemir, G., Kurt, Z., Özkan, K., “Konik Eğrilerine Dayalı Şekil Öznitelikleri ile Yaprak Tanıma”, SIU2014, 22th, vol., no., pp. 1463 – 1466, 23-25 April 2014 Trabzon.
- [7] Tang Z., Su Y., Er M.J., Qi F., Zhang L., Zhou J., “A local binary pattern based texture descriptors for classification of tea leaves”, Neurocomputing 168 (2015) 1011–1023.
- [8] Sari, B., Akgul C.B., Sankur, B., “Şekile Dayalı Yaprak Tanıma”, Signal Processing and Communications Applications Conference (SIU), 2013, 21th, vol., no., pp.1-4, 24-26 April 2013, Girne.
- [9] Kulkarni, A. H., Rai, H. M., Jahagirdar, K. A., & Upparamani, P. S. (2013). A leaf recognition technique for plant classification using RBPNN and Zernike moments. International Journal of Advanced Research in Computer and Communication Engineering, 2(1), 984-988.

- [10] Zhang J., Yanne P. And Li H., “Identification of Grape Varieties via Digital Leaf Image Processing by Computer”, *Bulletin de l'OIV* 84 (959-961), 5-14, 2011.
- [11] Kır B., Öz C., Gülbağ A., “Leaf recognition using KNN classification algorithm”, 2012 20th Signal Processing and Communications Applications Conference (SIU), 1 – 4, 18-20 April 2012, Mugla.
- [12] Türkoğlu M., Alçın Ö.F., Üstündağ M., Hanbay D., “Leaf Recognition and Classification Based On Geometric and Statistical Features”, *International Conference on Natural Science and Engineering (ICNASE'16)*, March 19-20 2016, Kilis.
- [13] Rojas-Hern, R., & Rojas-Hern, C. A. (2016, April). Plant identification using new geometric features with standard data mining methods. In 2016 IEEE 13th International Conference on Networking, Sensing, and Control (ICNSC) (pp. 1-4). IEEE.
- [14] Chaki J., Parekh R., Bhattacharya S., “Plant leaf recognition using texture and shape features with neural classifiers”, *Pattern Recognition Letters* 58 (2015) 61–68.
- [15] Wable, P. B., Chilveri, P. G., “Neural network based leaf recognition”, In *Automatic Control and Dynamic Optimization Techniques (ICACDOT)*, International Conference on (pp. 645-648), September 2016.
- [16] Tang Z., Su Y., Er M.J., Qi F., Zhang L., Zhou J., “A local binary pattern based texture descriptors for classification of tea leaves”, *Neurocomputing* 168 (2015) 1011–1023.
- [17] Du, J. X., Wang, X. F., and Zhang, G. J., "Leaf shape based plant species recognition", *Applied Mathematics and Computation* 185, pp 883-893, 2007.
- [18] Chaki J., Parekh R., Bhattacharya S., “Recognition of whole and deformed plant leaves using statistical shape features and neuro-fuzzy classifier”, *Recent Trends in Information Systems (ReTIS)*, 2015 IEEE 2nd International Conference on, 189 – 194, 9- 11 July 2015, Kolkata.
- [19] Naresh, Y. G., & Nagendraswamy, H. S. (2016). Classification of medicinal plants: An approach using modified LBP with symbolic representation. *Neurocomputing*, 173, 1789-1797.
- [20] Ahmed, N., Khan, U. G., & Asif, S. (2016). An Automatic Leaf Based Plant identification System. *Science International*, 28(1).
- [21] Razavi, S. (2016, May). Plant classification using group of features. In *Signal Processing and Communication Application Conference (SIU)*, 2016 24th (pp. 1957-1960). IEEE.

- [22] Russel Stuart, Norvig Peter, Artificial Intelligence, A Modern Approach, Prentice Hall, 2nd Edition, 2003.
- [23] Kaya, İ., & Engin, O. (2005). Kalite İyileştirme Sürecinde Yapay Zeka Tekniklerinin Kullanımı. Mühendislik Bilimleri Dergisi, 11, 103–112.
- [24] URL-1 : <http://www.quora.com/Machine-Learning/What-are-the-practical-applications-of-deep-learning-What-are-all-the-major-areas-fields>.
- [25] URL-2 : <https://github.com/rasmusbergpalm/DeepLearnToolbox>
- [26] B.I. Colak M.C., Colak C., Kocaturk H., Sagioglu S., Predictin gcoronary artery disease using different artificial neural network models., Anatol. J. Cardiol. (2008) 249–255.
- [27] H.H. Firat F., Arslan A.K., Colak C., Estimation of risk factors associated with colorectal cancer: an application of knowledge discovery in databases., Kuwait J. Sci. 43 (2) (2016) 151–161.
- [28] M.Y. Rafiq, G. Bugmann, D.J. Easterbrook, Neural network design for engineering applications, Comput. Struct. 79 (2001) 1541–1552. doi:10.1016/s0045-7949(01)00039-6.
- [29] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel: Handwritten digit recognition with a backpropagation network, in NIPS’89.
- [30] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner: “Gradient-based learning applied to document recognition,” Proceedings of the IEEE, 1998.
- [31] URL-3 : <http://cs231n.github.io/convolutional-networks/#fc>
- [32] G. Hinton, S. Osindero, Y. Teh : A Fast Learning Algorithm for Deep Belief Nets, In : Neural Computation,2006.
- [33] M. Ranzato, C. Poultney, S. Chopra, Y. LeCun: Efficient Learning of Sparse Representations with an Energy-based Model. In NIPS’ 2006.
- [34] URL-4 : <http://cs.stanford.edu/people/karpathy/convnetjs/demo/cifar10.html>
- [35] A. Krizhevsky. Convolutional deep belief networks on cifar-10. Unpublished manuscript, 2010.
- [36] Van Doorn, J. (2014). Analysis of Deep Convolutional Neural Network Architectures.
- [37] Alper Ulu: “Deep Convolutional Neural Network Based Representations For Person Re-Identification”, Department of Computer Engineering Computer Engineering Programme, Istanbul Technical University, June 2016.

- [38] Dan Claudiu Cireşan, Ueli Meier, Luca Maria Gambardella, and Jurgen Schmidhuber. Convolutional neural network committees for handwritten character classification. In International Conference on Document Analysis and Recognition, pages 1250–1254, 2011.
- [39] URL-5 :[http://web.engr.illinois.edu/~slazebni/spring14/lec24\\_cnn.pdf](http://web.engr.illinois.edu/~slazebni/spring14/lec24_cnn.pdf)
- [40] Huang, G.-B., Zhu, Q.-Y., & Siew, C.-K. (2006). Extreme Learning Machine: Theory and Applications. *Neurocomputing*, 70, 489-501.
- [41] Alçin, Ö.F., ŞENGÜR, A., İnce, “M.C, İleri-Geri Takip Algoritması Tabanlı Seyrek Aşırı Öğrenme Makinesi”, *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, Vol 30, No 1, 111-117, 2015.
- [42] Kaya, Y., Tekin, R., “Epileptik Nöbetlerin Tespiti için Aşırı Öğrenme Makinesi Tabanlı Uzman Bir Sistem”, *Bilişim Teknolojileri Dergisi*, Cilt: 5, Sayı: 2, Mayıs 2012.
- [43] Şengür, A., “Deep Extreme Learning Machine for Face Recognition”, *International Artificial Intelligence and Data Processing Symposium (IDAP'16)*, 43-47, Malatya, 2016.
- [44] G.-B. Huang, Z. Bai, L. L. C. Kasun, and C. M. Vong, “Local receptive fields based extreme learning machine,” *IEEE Computational Intelligence Magazine*, vol. 10, pp. 18–29, 2015.
- [45] Arı, B., Şengür, A., Arı, A.,: “Local Receptive Fields Extreme Learning Machine for Apricot leaf recognition”, *International Artificial Intelligence and Data Processing Symposium (IDAP'16)*, 43-47, Malatya, 2016.
- [46] Ari, B., Sengur, A., Ari, A., Hanbay D., “Apricot Plant Classification Based On Leaf Recognition by Using Convolutional Neural Networks”, *International Conference on Natural Science and Engineering (ICNASE'16)*, March 19-20 2016, Kilis.

## ÖZGEÇMİŞ

1990 yılında Elazığ'da doğdum. 2013 Yılında Fırat Üniversitesi Teknik Eğitim Fakültesi Bilgisayar Öğretmenliği Programından mezun oldum. Aynı yıl Diyarbakır Çermik Çok Programlı Lisesi'nde Bilişim Teknolojileri Öğretmeni olarak mesleki hayatıma başladım. 2016 yılında Mühendislik Tamamlama Programı ile İnönü Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümünden mezun oldum. Evli ve bir çocuk annesiyim.

