

YALOVA ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**BİTKİLERDE GÖRÜLEN HASTALIKLARIN DERİN ÖĞRENME
YÖNTEMLERİYLE
TESPİTİ VE SINIFLANDIRILMASI**

YÜKSEK LİSANS TEZİ

Melike (SARDOĞAN) DOĞAN

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

EYLÜL 2019

YALOVA ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**BİTKİLERDE GÖRÜLEN HASTALIKLARIN DERİN ÖĞRENME
YÖNTEMLERİYLE
TESPİTİ VE SINIFLANDIRILMASI**

YÜKSEK LİSANS TEZİ

**Melike (SARDOĞAN) DOĞAN
165105002**

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

Tez Danışmanı: Dr. Öğr. Üyesi Adem TUNCER

EYLÜL 2019

YALOVA Üniversitesi Fen Bilimleri Enstitüsü'nün 165105002 numaralı Yüksek Lisans Öğrencisi **Melike SARDOĞAN**, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı "**BİTKİLERDE GÖRÜLEN HASTALIKLARIN DERİN ÖĞRENME YÖNTEMLERİYLE TESPİTİ VE SINIFLANDIRILMASI**" başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı : **Dr. Öğr. Üyesi Adem TUNCER**
Yalova Üniversitesi




Jüri Üyeleri : **Dr. Öğr. Üyesi Adem TUNCER**
Yalova Üniversitesi



Dr. Öğr. Üyesi Osman Hilmi KOÇAL
Yalova Üniversitesi



Doç. Dr. Halil YİĞİT
Kocaeli Üniversitesi



Teslim Tarihi : **25 Temmuz 2019**
Savunma Tarihi : **12 Eylül 2019**



Eşime, aileme, arkadaşlarıma ve saygıdeğer danışman hocama,



ÖNSÖZ

Lisans ve yüksek lisans eğitimim boyunca benden yardımlarını esirgemeyen, akademik anlamda bana her zaman destek olan ve doğru yolu gösteren sayın danışman hocam Dr. Öğr. Üyesi Adem TUNCER'e teşekkür ederim.

Eşime, aileme ve arkadaşlarıma her zaman yanımda oldukları ve benden desteklerini hiçbir zaman esirgemedikleri için teşekkür ederim.

Temmuz 2019

Melike (SARDOĞAN) DOĞAN
(Bilgisayar Mühendisi)



İÇİNDEKİLER

Sayfa

ÖNSÖZ.....	vii
İÇİNDEKİLER	ix
KISALTMALAR	xi
ÇİZELGE LİSTESİ.....	xiii
ŞEKİL LİSTESİ.....	xv
ÖZET.....	xvii
SUMMARY	xix
1. GİRİŞ	1
1.1 Tezin Amacı ve Katkısı.....	2
1.2 Literatür Araştırması	3
2. GÖRÜNTÜ İŞLEME.....	9
2.1 Uzaysal Alanda Görüntü İyileştirme	10
2.2 Görüntü Restorasyonu.....	10
2.3 Görüntü Sıkıştırma	14
2.4 Morfolojik İşlemler	15
3. DERİN ÖĞRENME.....	19
3.1 Yapay Sinir Ağları	19
3.1.1 Yapay sinir ağları tarihçe	20
3.1.2 Yapay sinir ağlarının genel yapısı.....	21
3.1.3 Yapay sinir ağlarının yapıları.....	22
3.1.3.1 Yapılarına göre yapay sinir ağları	22
3.1.3.2 Öğrenme algoritmalarına göre yapay sinir ağları.....	24
3.1.3.3 Öğrenme zamanına göre yapay sinir ağları.....	24
3.1.4 Yapay sinir ağlarının özellikleri.....	24
3.1.5 Yapay sinir ağlarının eğitilmesi	27
3.1.6 LVQ ağı.....	28
3.2 Konvolüsyonel Sinir Ağları	30
3.3.1 Havuzlama katmanı.....	34
3.3.2 Aktivasyon katmanı	35
3.3.3 Tam bağlı katman.....	35
4. MATERYAL VE YÖNTEM.....	37
4.1 Materyal	37
4.1.1 Domates yaprakları veri seti	37
4.1.2 Elma yaprakları veri seti	38
4.2 Yöntem	39
4.2.1 Domates yapraklarında hastalık tespiti	39
4.2.2 Elma ağaçlarında hastalık tespiti.....	43
5. SONUÇ VE ÖNERİLER.....	45

KAYNAKLAR.....	47
ÖZGEÇMİŞ.....	51



KISALTMALAR

KSA	: Konvolüsyonel Sinir Ađı
LVQ	: Learning Vector Quantization
İHA	: İnsansız Hava Aracı
RGB	: Red (Kırmızı), Green (Yeşil), Blue (Mavi)
CNN	: Convolutional Neural Networks
DVM	: Destek Vektör Makineleri
YSA	: Yapay Sinir Ağları
ADD	: Ayrık Dalgacık Dönüşümü
AKD	: Ayrık Kosinüs Dönüşümü
ReLU	: Rectified Linear Unit (Doğrultulmuş Doğrusal Birim)
HSI	: Hue (Ton), Saturation (Doygunluk), Intensity (Yoğunluk)



ÇİZELGE LİSTESİ

Sayfa

Çizelge 3.1 : Geleneksel algoritmalar ve yapay sinir ağları karşılaştırması.....	26
Çizelge 4.1 : Karmaşıklık matrisi şeklinde sınıflandırma sonuçları.....	42
Çizelge 4.2 : Karmaşıklık matrisi şeklinde sınıflandırma sonuçları.....	43





ŞEKİL LİSTESİ

Sayfa

Şekil 2.1 : Ortalama filtresi: (a) uygulanmamış, (b) uygulanmış.	11
Şekil 2.2 : Gaussian filtresi: (a) uygulanmamış, (b) uygulanmış.	12
Şekil 2.3 : Medyan filtresi: (a) uygulanmamış, (b) uygulanmış.	12
Şekil 2.4 : Unsharp filtresi: (a) uygulanmamış, (b) uygulanmış.	13
Şekil 2.5 : Motion filtresi: (a) uygulanmamış, (b) uygulanmış.	13
Şekil 2.6 : Blurring filtresi: (a) uygulanmamış, (b) uygulanmış.	14
Şekil 2.7 : Genişletme işlemi: (a) uygulanmamış, (b) uygulanmış.	15
Şekil 2.8 : Aşındırma işlemi: (a) uygulanmamış, (b) uygulanmış.	16
Şekil 2.9 : Açma işlemi: (a) uygulanmamış, (b) uygulanmış.	16
Şekil 2.10 : Kapama işlemi: (a) uygulanmamış, (b) uygulanmış.	17
Şekil 3.1 : Örnek bir yapay sinir ağı.	22
Şekil 3.2 : İleri beslemeli ağ örneği.	23
Şekil 3.3 : Geri beslemeli ağ örneği.	23
Şekil 3.4 : LVQ'nun çalışma yapısı.	29
Şekil 3.5 : KSA'nın genel yapısı.	31
Şekil 3.6 : Konvolüsyon katmanı çalışma yapısı.	33
Şekil 3.7 : Havuzlama işleminin çalışma şekli.	34
Şekil 4.1 : Domates yaprak hastalığı veri seti örnek görüntüleri: (a) Sağlıklı. (b) Bakteriyel leke. (c) Geç yanıklık. (d) Septoria yaprak lekesi. (e) Sarı yaprak kıvrıcılığı.	38
Şekil 4.2 : Elma ağaçlarından alınan gerçek verilere örnek görüntüler: (a) Sağlıklı. (b) Hastalıklı.	39
Şekil 4.3 : Domates yapraklarında hastalık tespitinde kullanılan metodun yapısı	41
Şekil 4.4 : Elma ağaçlarında hastalık tespitinde kullanılan metodun yapısı.	44



BİTKİLERDE GÖRÜLEN HASTALIKLARIN DERİN ÖĞRENME YÖNTEMLERİYLE TESPİTİ VE SINIFLANDIRILMASI

ÖZET

Günümüzde, teknolojide meydana gelen hızlı gelişmelere bağlı olarak tarımsal çalışmalar da bu gelişmelerden olumlu olarak etkilenmiş ve hassas tarım kavramı ortaya çıkmıştır. Hassas tarım tarlada çalışan insan iş gücünü en aza indirmeyi ve bununla birlikte teknolojiyi kullanarak verimi en üst düzeye çıkarmayı hedeflemektedir. Hassas tarım uygulamaları, çiftçinin ihtiyaçları doğrultusunda gelişim göstermektedir. Çiftçinin en büyük ihtiyaçlarından birisi de elde ettiği ürünün kaliteli ve verimli olmasıdır. Bu yüzden bitkinin yaşam evrelerinde hastalıklardan uzak tutulması gerekmektedir. Bir bitki yaşam evresinin her adımında gözlemlenmeli ve hastalıklardan olabildiğince uzak tutulmalıdır. Bitkinin herhangi bir hastalığa yakalanması durumunda bu hastalık ilk evrelerde tespit edilebilmelidir ki bitkiye ve ürününe çok fazla bir zarar vermeden engellenebilsin. Bitkideki hastalığın başlangıç durumunda tespit edilip iyileştirilmesi, bitki ürünlerinin kalitesinin düşmesinin önüne geçebilmektedir. Hassas tarım uygulamaları ile bitkilerde görülen hastalıkların otomatik olarak tespit edilmesi ve hastalık türünün belirlenmesi mümkün olmaktadır.

Bu tez çalışmasında, domates ve elma yapraklarındaki hastalıkların tespiti ve sınıflandırılması derin öğrenme yöntemleri ile gerçekleştirilmiştir. Derin öğrenme yöntemlerinden biri olan Konvolüsyonel Sinir Ağı (KSA) modeli ve Learning Vector Quantization (LVQ) algoritması tabanlı bir yaklaşım sunulmuştur. Deneysel çalışmalar domates ve elma yaprakları için ayrı ayrı yapılmıştır. Domates yapraklarından oluşan veri setinde 4 farklı hastalıklı yaprak ve 1 sağlıklı yaprak olmak üzere toplam 5 farklı sınıf için deneysel çalışmalar gerçekleştirilmiştir. Elma yaprakları üzerinde gerçekleştirilen çalışmalarda ise Yalova ilinde bulunan elma bahçelerindeki yaprakların kamera ile görüntüleri alınmış ve deneysel çalışmalar ağaç üzerindeki yaprak görüntüleri üzerinde gerçekleştirilmiştir. Deneysel sonuçlar, çalışmada önerilen konvolüsyonel sinir ağı tabanlı yöntemin yapraklardaki hastalıkların tespitinde başarılı olduğunu ve tarımsal alanlarda hastalık tespiti ve sınıflandırma çalışmaları için etkili bir şekilde kullanılabileceğini göstermektedir.



DETERMINATION AND CLASSIFICATION OF DISEASES IN PLANTS WITH DEEP LEARNING METHODS

SUMMARY

Due to the rapid developments in the latest technology, agricultural studies have been positively affected by these developments and the concept of sensitive agriculture has emerged. Precision agriculture aims to minimize the human labor in the field and to maximize productivity by using technology beside this. Precision agriculture practices are developing in line with the needs of the farmer. One of the biggest needs of the farmer is that the product farmer's obtains is of high quality and efficient. Therefore, the plant should be kept away from diseases during its life stages. A plant should be observed at every step of the life stage and kept as far away from diseases as possible. Diseases should be detected at an early stage if the plant catches any disease, so that it can be prevented without causing too much damage to the plant and its crop. Detection and rehabilitation of plant disease in the first case may prevent deterioration of the quality of plant products. It is possible to detect the diseases seen in plants automatically and to determine the disease type with precision agriculture applications.

In this thesis, detection and classification of diseases in tomato and apple leaves were carried out by deep learning methods. An approach based on Convolutional Neural Network (CNN) model which is one of the deep learning methods and Learning Vector Quantization (LVQ) algorithm is presented. Experimental studies were done separately for tomato and apple leaves. In the data set consisting of tomato leaves, experimental studies were conducted for 5 different classes, 4 different diseased leaves and 1 healthy leaf. In the studies carried out on apple leaves, the images of the leaves in the apple orchards in Yalova province were taken with camera and the experimental studies were carried out on the leaf images on the tree. Experimental results show that the convolutional neural network based method proposed in the study is successful in detecting diseases in leaves and can be used effectively for disease detection and classification studies in agricultural areas.



1. GİRİŞ

Dünya nüfusunun her geçen gün artması ile birlikte tarıma duyulan ihtiyaç da gün geçtikçe katlanarak artmaktadır. Tarımdan elde edilen ürünler ile insanların yeme, giyinme, ısınma gibi birçok ihtiyacı karşılanabilmektedir. Elde edilen ürünler bir ülkenin ithalat ve ihracatında oldukça büyük rol oynayabilmektedir. Tarım sayesinde elde edilen gelir, ülkenin ekonomisinin gelişmesinde ve kalkınmasında önemli rol oynamaktadır. Bu bakımdan, bitkilerden elde edilen ürünlerin kaliteli olması gerekmekte ve ürünlerin kaliteli olabilmesi için de bitkilerin hastalıklardan korunması gerekmektedir. Bitkide bir hastalık oluşması durumunda ise hastalığın erkenden tespit edilebilmesi ve bitkinin tedavi edilebilmesi çok önemli bir gerekliliktir.

Bitkinin hastalığa yakalanmasında olumsuz çevre koşulları, mantarlar, bakteriler ve virüsler büyük rol oynamaktadır. Hastalıklar bitkinin fotosentez, tozlaşma, dölleme, çimlenme gibi yaşamsal işlevlerine zarar verebilmektedir. Bu nedenle hastalığın erken aşamalarda tespiti oldukça önemlidir.

Bitkilerde hastalık oluşması durumunda elde edilen ürünlerin miktarı ve kalitesinde azalma olmaktadır. Bu durumun oluşmaması için hastalığın erkenden tespit edilmesi ve tedavi uygulanması büyük önem arz etmektedir. Bitkilerdeki hastalıkları gözlemlemek ve tedavi etmek çiftçi için oldukça zaman alan bir iş olabilmektedir. Hastalığın tespit edilemediği durumlarda ise dışarıdan konuda uzman bir insanın yardımına ihtiyaç duyulabilmektedir. Bu durum mali açıdan çiftçiye ek bir yük getirmektedir. Çiftçiyi bu iş yükünden ve zarardan kurtarmak için hızlı ve daha doğru yöntemler kullanmak gerekmektedir. Günümüzde dışarıdan bir uzman yerine teknolojik cihazların kullanılması ve bu cihazlar vasıtasıyla bitkide hastalık olup olmadığı, hastalık var ise hastalık türünün tespiti mümkün hale gelmektedir. Teknolojik cihazların görüntü elde etme kalitesinin oldukça artması ile görüntü üzerinden nesne tespiti yapma, sınıflandırma gibi işlemler, görüntü işleme ve yapay

zeka algoritmaları kullanılarak çok başarılı sonuçlar ortaya koymaktadır. Bazı durumlarda, insan gözünün ayırt edemediği bir ayrıntı bu tür teknolojik cihazlarda çalışan görüntü işleme ve yapay zeka algoritmaları ile kolay bir şekilde ortaya çıkartılabilmektedir. Teknolojik gelişmelerin ve tarımın bu şekilde bir araya getirilmesi ile birlikte hassas tarım kavramı ortaya çıkmıştır.

Hassas tarım uygulamalarını, teknolojik cihazlar kullanarak uzaktan veri toplama, toprak haritalama, toprağın nem ve sıcaklığını ölçme, tarımsal ürünleri kontrol etme (sağlam, çürük, olgunlaşmış vb.), ürün sayma, sulama, ilaçlama gibi farklı kategorilere ayırmak mümkündür. Hassas tarım sayesinde düşük maliyet ile üründen daha fazla verim ve gelir alınması, aynı zamanda çevreye verilen zararın da daha aza indirilmesi amaçlanmaktadır.

Yapay zekanın alt alanı olan derin öğrenme yöntemleri günümüzde bir çok alanda yaygın olarak kullanılmaktadır. Tarımsal alanlarda kullanılan yapay zeka çalışmalarından biri de bitkilerde hastalık ve türünün tespitinin yapılmasıdır. Bu tez çalışmasında, bitkide hastalık tespitinin yapay zeka ve görüntü işleme yöntemleri kullanılarak otomatik bir şekilde gerçekleştirilmesi sunulmuştur.

1.1 Tezin Amacı ve Katkısı

Son yıllarda derin öğrenme yöntemlerinin pek çok alanda olduğu gibi görüntü işleme alanında da başarılı performans sergilemesi ile birlikte geleneksel görüntü işleme tekniklerine alternatif olarak literatürde sıkça kullanılmaktadır.

Tez kapsamında bitkilerde hastalık tespiti ve sınıflandırma işlemlerinin otomatik olarak yapılması hedeflenmiştir. Deneysel çalışmalarda, hazır olarak alınan domates yaprakları görüntüleri ile birlikte Yalova ilinde bulunan elma ağaçları yapraklarının kamera ile görüntüleri elde edilerek toplanan veriler kullanılmıştır. Hastalık tespiti için verilerden özellik çıkarma işlemlerini de otomatik olarak yapabilen derin öğrenme yöntemlerinden KSA tabanlı bir yaklaşım sunulmuştur. Literatürdeki KSA tabanlı çalışmalar genellikle hazır derin öğrenme araç ve kütüphaneleri ile gerçekleştirilmiştir. Tez çalışmasında, literatürde KSA'da kullanılan öğrenme algoritmalarından geri yayımlı algoritma veya vb. algoritmalar yerine LVQ

algoritmasının kullanılması ve performansının ölçülmesi amaçlanmıştır. Gerçekleştirilen literatür araştırmaları sırasında KSA ve LVQ'nun birlikte kullanıldığı başka bir çalışmaya rastlanmamıştır. Bu bakımdan tez çalışması KSA ve LVQ ağının birlikte kullanıldığı ilk çalışmadır. Çalışmanın gerçek ağaçlar üzerinde de gerçekleştirilerek, gerçek dünya problemleri karşısındaki başarımının da ortaya konması hedeflenmiştir.

Standart KSA'da uygulanan işlemlerin ilkinden sonra verilere ait farklı katmanlar tek bir katman olarak birleştirilmektedir. Bu tez çalışmasında standart KSA'dan farklı olarak her bir kanala ayrı ayrı konvolüsyon işlemleri uygulanmış, en son aşamada tek bir sütun olarak birleştirilmiş ve tam bağlı katmana girdi olarak verilmiştir. Böylece farklı kanalların, özellikle RGB kanalları için, özellikleri korunarak derin öğrenme modelinin eğitilmesi gerçekleştirilmiştir.

1.2 Literatür Araştırması

Bitkide hastalık tespiti alanında yapılmış birçok çalışma bulunmaktadır. Yapılan çalışmalar genellikle hazır veri setleri ile hazır model ve kütüphaneler kullanılarak gerçekleştirilmiştir. İncelenen literatür çalışmaları aşağıda verilmiştir.

Arivazhagan ve arkadaşları [1], bitki görüntüsünün özelliklerini görüntü işleme teknikleri ile elde etmişlerdir. Elde edilen özelliklere öncelikle minimum uzaklık kriteri uygulamışlar ve elde ettikleri sonuçları destek vektör makineleri (DVM) kullanarak sınıflandırıp, bitkinin hastalığını tespit etmişlerdir.

Liu ve arkadaşları [2], elma bitkisinde mozaik, pas, alternarya ve kahverengi nokta hastalıklarının tespiti ve sınıflandırılması üzerine bir çalışma ortaya koymuşlardır. Hastalık sınıflandırma işlemini standart AlexNet modeli üzerine bir yapı geliştirerek gerçekleştirmişlerdir.

Islam ve arkadaşları [3], pirinç bitkisinde hastalık tespiti üzerine bir çalışma ortaya koymuşlardır. Ön işleme aşamasında elde ettikleri görüntüye Gaussian filtreleme uygulayarak görüntünün gürültüsünü azaltmışlardır. Gürültüsünü azalttıkları görüntülere eşik değeri uygulayarak sağlıklı ve hasta olan bitkilerin piksel değerlerini karşılaştırıp sonuç elde etmişlerdir.

Kaur ve arkadaşları [4] gerçekleştirdikleri çalışmada çeşitli hastalıklı ağaç yapraklarından oluşan bir veri seti üzerinde DVM ve karınca kolonisi optimizasyonu algoritması kullanmışlardır.

Pujari ve arkadaşları [5], DVM ve YSA kullanarak, buğday, üzüm, ayçiçeği gibi bitkilerin üzerinde hastalık tespiti çalışması yapmışlardır.

Barbedo [6], yaprakların görünen kısımları üzerinde çeşitli resim işleme teknikleri uygulayarak farklı bitki türlerinde görülebilen benzer hastalıkların tespitini yapmıştır. Çalışmada bitkilerde hastalık tanıma işlemi için renk dönüşümü ve renk histogramları kullanmıştır. Çalışma, geniş bir veri tabanında 12 farklı yaprak türünü etkileyen 82 farklı hastalığa ait belirtilerin görüntülerinin üzerinde test gerçekleştirilmiştir.

Padol ve Yadav [6], üzüm yapraklarında görülen hastalıkların tespiti için yaptıkları çalışmada hastalıklı yaprakları k-ortalama kümeleme algoritması ile bölütleyip, bu bölgeler üzerinde lineer DVM ile sınıflandırma yapmışlardır. Kullandıkları 137 farklı üzüm yaprağı üzerindeki hastalıkların sınıflandırılmasını %89 başarı oranı ile gerçekleştirmişlerdir.

Singh ve Misra [8] tarafından yapılan çalışma bitki yapraklarında meydana gelen hastalıkların otomatik tespiti ve sınıflandırılmasında kullanılan görüntü segmentasyon tekniği üzerine bir algoritma sunmaktadır. Aynı zamanda farklı hastalık sınıflandırma teknikleri ile ilgili anketi de içermektedir. Araştırmanın en önemli kısmı hastalık tespiti aşamasında genetik algoritma kullanmalarıdır. Beş farklı hastalık üzerinden yapılan çalışmanın ortalama başarı oranı %97.6'dır.

Akram ve arkadaşları [9], yaprak hastalıklarını sınıflandırmak için gerçek zamanlı çalışacak bir gömülü sistem tasarımı gerçekleştirmiştir. RGB görüntüler üzerinde çalışan bir algoritmayı bu sistem üzerinde çalıştırıp paralel işleme yaparak hastalık tespiti işlemini hızlandırmayı amaçlamışlardır. Bu çalışma sonucunda gerçek zamanlı görüntü üzerinde hastalık tespiti yapan otomatik bir sistem geliştirmenin mümkün olabileceği görülmektedir.

Zhang ve arkadaşları [10], salatalık yaprakları veri tabanı üzerinde hastalık tespiti yapmışlardır. Bu çalışmada hastalıklı yaprakları k-ortalama kümeleme algoritması ile

bölütlemişler, hastalıkları tespit etmek için şekil ve renk özelliklerini çıkarıp seyrek temsil yöntemi ile bu hastalıklı yaprakları sınıflandırmışlardır. %86 doğruluk oranı ile sınıflandırmayı başarmışlardır.

Elangovan ve Nalini [11] tarafından yapılan çalışmada, görüntü bölütlemesi ve DVM kullanılarak bitkisel hastalık sınıflandırma işlemi gerçekleştirilmiştir. RGB formatında alınan yaprak görüntüsüne renk dönüşümü uygulamışlardır.

Khirade ve Pati [12], tarafından yapılan çalışmada bitki hastalıklarının saptanmasında yaprak görüntülerini kullanan yöntemler ele alınmıştır. Çalışma beş temel adımdan oluşmaktadır; ilk adımda görüntünün RGB formatında elde edilmesi ve renk dönüşüm yapısının uygulanması yer almaktadır. İkinci adımda gürültüden arındırma ve uygun boyuta indirgeme işlemleri gerçekleştirilmektedir. Üçüncü adımda otsu metodu, k-ortalama kümeleme algoritması kullanılmakta ve RGB'den HSI modeline dönüşüm yapılmaktadır. Dördüncü adımda Color co-occurrence metodu kullanılarak öznitelikler çıkarılmaktadır. Son adımda YSA geri yayılım algoritması kullanılarak sınıflandırma işlemi tamamlanmaktadır.

Sladojevic ve arkadaşları [13] tarafından yapılan çalışmada derin konvolüsyonel ağlar kullanılarak bitki yaprakları üzerinden hastalıklar tespit edilmiştir. Geliştirilen model bitki yapraklarını çevrelerinden ayırt edebilme yeteneği sayesinde on üç farklı bitki hastalığını yapraklar üzerinden tanıyabilmektedir. Derin KSA eğitimini gerçekleştirmek için Berkley Vision and Learning Center tarafından geliştirilen Caffe derin öğrenme kütüphanesi kullanılmaktadır. Deneysel sonuçlar ayrı sınıf testleri için ortalama %96.3 başarı göstermektedir.

Ferentinos [14] tarafından yapılan çalışmada, derin öğrenme yöntemleri ile bitki hastalıklarının tespit ve teşhisini gerçekleştirmek için KSA modelleri geliştirilmiştir. Modellerin eğitimi 87,848 görüntü içeren veri seti üzerinde yapılmıştır. Ayrıca bu veri seti 25 farklı bitki türünün 58 [bitki, hastalık] kombinasyonunu içermektedir. Deneysel sonuçlar çalışmanın %99.53'lük bir başarı oranına sahip olduğunu göstermektedir.

Kamilaris ve Prenafeta-Boldú [15] tarafından yapılan çalışma, tarımsal alanlarda derin öğrenme teknikleri, görüntü işleme ve veri analizi kullanılarak gerçekleştirilen 40 farklı araştırmanın incelenmesini kapsamaktadır.

Lu ve arkadaşları [16] tarafından yapılan çalışma, derin konvolüsyonel sinir ağları tekniklerini temel alarak pirinçlerde hastalık tespiti yapmaktadır. Sağlıklı ve hastalıklı pirinç yapraklarından oluşan 500 görüntülük bir veri seti ile eğitim gerçekleştirilmiştir. Eğitilen model sık rastlanan 10 farklı hastalığı tespit etmektedir. Deneysel sonuçlara göre hastalığı tespitindeki başarı oranı %95.48'dir.

Amara ve arkadaşları [17] tarafından yapılan çalışma, muz yapraklarında hastalık sınıflandırması yapmaktadır. Derin öğrenme temelli yaklaşımda veri setinin sınıflandırılmasında konvolüsyonel sinir ağları modellerinden LeNet mimarisi kullanılmıştır. Deneysel sonuçlar farklı çözünürlük, arka plan ve boyuttaki veri seti üzerinde kullanılan modelin başarısını ortaya koymaktadır.

Fuentes ve arkadaşları [18] ortaya koydukları çalışmada, domates bitkisi üzerinde hastalık tespiti yapmışlardır. Çalışmada üç farklı model kullanılmış ve modellerin performanslarının karşılaştırılması yapılmıştır. Kullandıkları modeller Faster R-CNN, R-FCN ve SSD modelleridir. Bu üç modeli de VGGNet ve ResNet gibi öznitelik çıkarım mekanizmaları ile birlikte kullanmışlardır. Sonuç olarak kullanılan modeller dokuz farklı hastalığı başarılı olarak tespit etmiştir. Karşılaştırma yapıldığında ise ResNet-50 ile birlikte çalıştırılan R-FCN tespit yüzdesi olarak diğerlerini geride bırakarak en başarılı model olmuştur.

Mohanty ve arkadaşları [19] tarafından yapılan çalışmada, on dört farklı bitki türüne ait 26 hastalığın tespiti yapılmıştır. Eğitim için kullanılan veri seti 54,306 görüntü içermektedir. Eğitilen model % 99.35 başarı oranına ulaşmıştır.

Durmuş ve arkadaşları [20] tarafından gerçekleştirilen çalışmada, domates bitkisinin yaprakları üzerinde hastalık tespiti yapılmıştır. Sahada hareket edebilen bir robot aracılığıyla gerçek zamanlı tespit yapılması amaçlanmıştır. On dört farklı ürüne ait 54,309 etiketlenmiş görüntüden oluşan veri seti ile eğitim yapılmıştır. KSA, Caffè kütüphanesi kullanılarak eğitilmiştir. İki farklı mimari üzerinden sınamaya tabi tutulan modelin başarı oranı AlexNet'te %95.65 iken SqueezeNet'te %94.3'tür.

Ashqar ve Abu-Naser [21] tarafından ortaya koyulan çalışmada, domates yapraklarında hastalık tespiti yapılmıştır. Kontrollü koşullar altında toplanmış, hastalıklı ve sağlıklı yapraklardan oluşan 9000 görüntülük bir veri seti kullanılarak eğitim yapılmıştır. Beş farklı hastalığa teşhis koyabilen bir KSA eğitilmiştir. Sonuç olarak bu model, uygulanabilirliğini kanıtlayan test setinde %99.84'lük bir başarı göstermiştir.

Cruz ve arkadaşları [22], tarafından yapılan çalışmada, belirli bir bakteri türü ile hastalıklı zeytin yapraklarının tespiti yapılmıştır. Başka hastalıkları tespit etmek amacı ile daha evvel eğitilmiş derin öğrenme modelinin transfer öğrenme ile tekrar eğitilerek zeytin yapraklarındaki bu hastalığı tespit etmesi amaçlanmıştır. Sonuç olarak başarı oranının 98.60 ± 1.47 olduğu görülmüştür.



2. GÖRÜNTÜ İŞLEME

Sayısal görüntü işleme herhangi bir görüntü cihazı ile elde edilmiş dijital görüntülerin bilgisayar ortamına aktarılıp incelenmesi, üzerinde işlem yapılmasıdır [23]. Görüntü işleme kaydedilmiş olan görüntü verilerini iyileştirmek, değiştirmek ve analiz etmek gibi işlemler için kullanılmaktadır. Görüntü işlemede üç boyutlu bir nesne iki boyutlu bir nesneye dönüştürülmektedir [24]. Görüntü, elektronik cihazlar ile elde edildikten sonra iki boyutlu bir fonksiyon olan $f(x, y)$ ile ifade edilmekte, x ve y değerleri uzaysal koordinatları temsil etmektedir [25]. Sayısal görüntü işlemede çeşitli bilgisayar algoritmaları kullanılarak iki boyutlu $f(x, y)$ fonksiyonu üzerinde işlem yapılmaktadır.

Görüntü işleme sanayi, tıp, tarım, güvenlik, uzaktan algılama uygulamaları gibi birçok farklı alanda kullanılmaktadır [26]. Örneğin; tarım alanında meyve ve sebzelerde renk analizi, sınıflandırma, sayım, yabancı otların belirlenmesi, hastalık analizi gibi işlemler için kullanılmaktadır [27].

Görüntü işleme ihtiyaç duyulan konulara göre farklı amaçlarla kullanılmaktadır. Bunlar;

- Görmekte zorlanılan nesnelerin görünmesini kolaylaştırma (görselleştirme)
- İçerisinde gürültü bulunan görüntüleri iyileştirme (görüntünün keskinleştirilmesi ve yenileme)
- Yüksek çözünürlüklü görüntü arama (görüntü alımı)
- Görüntüde bulunan farklı nesnelerin tanımlanması (desen tanıma)
- Görüntüde bulunan farklı nesnelerin ayırt edilmesi (görüntü tanıma)

Görüntü işlenirken öncelikle görüntünün elde edilmesi gerekmektedir. Elde edilen görüntü sayısallaştırılmakta ve sayısallaştırılan görüntü griye dönüştürülmektedir.

Görüntüdeki gürültü temizlendikten ve çeşitli görüntü işleme teknikleri uygulandıktan sonra yeni görüntü elde edilmektedir.

Görüntü işleme sırasında kullanılan bazı teknikler aşağıda verilmektedir:

- Uzaysal alanda görüntü iyileştirme
- Görüntü restorasyonu
- Görüntü sıkıştırma
- Morfolojik görüntü işleme

Kullanılan bu teknikler ile görüntünün anlaşılması kolaylaşmaktadır. Görüntü üzerinde gürültüler giderilebilmekte, bozulmalar düzeltilebilmekte, görüntü içerisindeki istenilen nesnelere kolay bir şekilde ayırt edilebilmektedir.

2.1 Uzaysal Alanda Görüntü İyileştirme

Görüntü iyileştirme, görüntünün kalitesini arttırarak daha iyi anlaşılabilmesini sağlamaktır. Bunun için gri seviyeli dönüşümler, histogram işlemleri, aritmetik ve mantıksal işlemler, filtre işlemleri gibi teknikler kullanılmaktadır. Histogram, verilerin sütun grafiğiyle gösterilmesidir. Histogram işlemleri sırasında bu grafik değerleri üzerinde işlem yapılmaktadır. Histogram germe ve eşitleme işlemleri ile görüntünün daha iyi anlaşılabilmesi sağlanmakta, görüntünün parlaklık dağılımı eşitlenmektedir. Görüntü üzerinde toplama, çıkarma gibi işlemler yapılabilmektedir. Bu sayede iki farklı resim birleştirilebilmektedir.

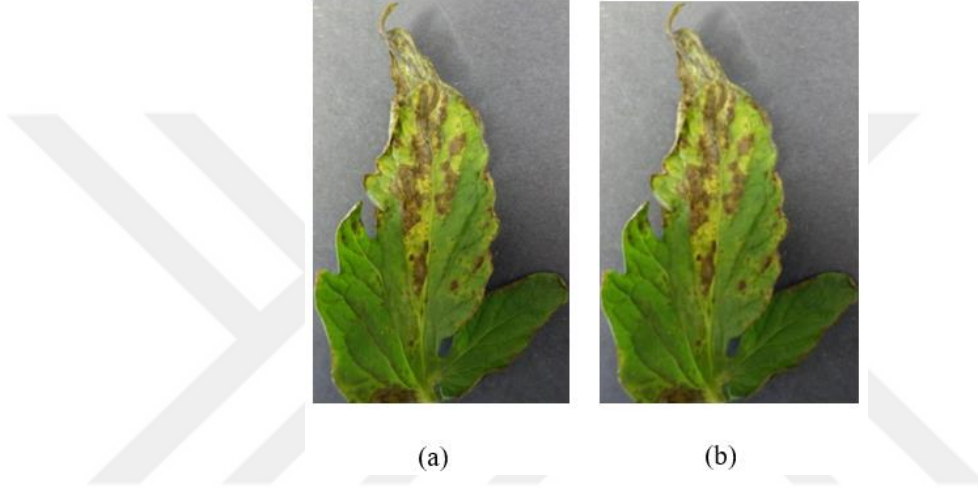
2.2 Görüntü Restorasyonu

Görüntü elde edilirken bir takım bozulmalar da meydana gelebilmektedir. Bu bozulmalar veri kayıplarına sebep olabilmektedir. Bu kayıplar görüntüde bulunan gürültü sebebi ile oluşmaktadır. Görüntünün elde edilmesi sırasında hareket edilmesi, ışığın etkisi, objektif bozulmalar gibi çeşitli sebeplerle gürültü oluşabilmektedir. Tuz-biber gürültüsü, gaussian gürültü ve uniform gürültü bunlardan en önemlileridir. Görüntüde oluşan gürültüleri giderebilmek için çeşitli filtreleme işlemleri uygulanmaktadır.

Aşağıda meyve görüntüleri üzerinde Matlab programı ile gerçekleştirilen çeşitli filtre uygulamaları gösterilmektedir;

Ortalama Filtresi: Fotoğraftaki her piksel komşuları ile birlikte ortalaması alınarak hesaplanmakta ve pikselin yeni değeri bu ortalama değer olmaktadır. Bu filtre ile görüntüdeki keskin geçişler daha yumuşak bir hale gelmektedir.

Şekil 2.1’de ortalama filtresi için örnek bir uygulama gösterilmektedir. Şekil 2.1 (a) ortalama filtresi uygulanmamış görüntüyü, Şekil 2.1 (b) ortalama filtresi uygulanmış görüntüyü göstermektedir.



Şekil 2.1 : Ortalama filtresi: (a) uygulanmamış, (b) uygulanmış.

Gaussian Filtresi: Ortalama filtresinin Gaussian dağılımı kullanılarak gerçekleştirilmiş halidir. Bir görüntü üzerindeki gürültüyü, detayları kaldırmak ve görüntüyü yumuşatmak için kullanılmaktadır. Görüntünün bulanıklaştırma derecesinin ayarlanmasına olanak vermektedir. Görüntüyü hafifçe bulanıklaştırmaktan üzerinde kalın bir sis tabakası varmış izlenimine kadar değişen efektler verilmesini sağlamaktadır.

Şekil 2.2’de Gaussian filtre için bir örnek gösterilmektedir. Şekil 2.2 (a) gaussian filtresi uygulanmamış görüntüyü, Şekil 2.2 (b) gaussian filtresi uygulanmış görüntüyü göstermektedir.

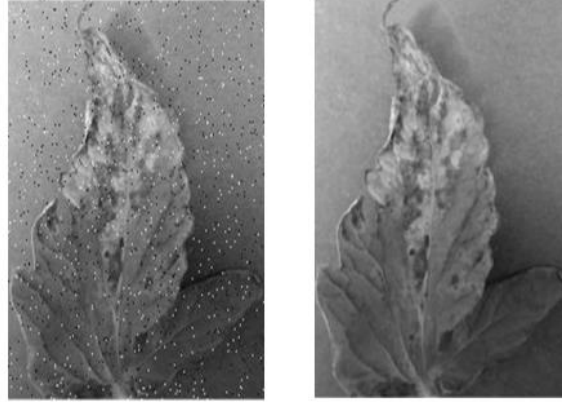


(a)

(b)

Şekil 2.2 : Gaussian filtresi: (a) uygulanmamış, (b) uygulanmış.

Medyan Filtresi: Üzerinde işlem yapılan piksel, komşu pikseller ile birlikte küçükten büyüğe sıralanmakta ve ortada bulunan değer pikselin yeni değeri olmaktadır. Bağımsız nokta ve çizgi gürültülerini temizlemekte kullanılmaktadır. Şekil 2.3'te medyan filtre için örnek bir uygulama gösterilmektedir. Şekil 2.3 (a) medyan filtresi uygulanmamış görüntüyü, Şekil 2.3 (b) medyan filtresi uygulanmış görüntüyü göstermektedir.



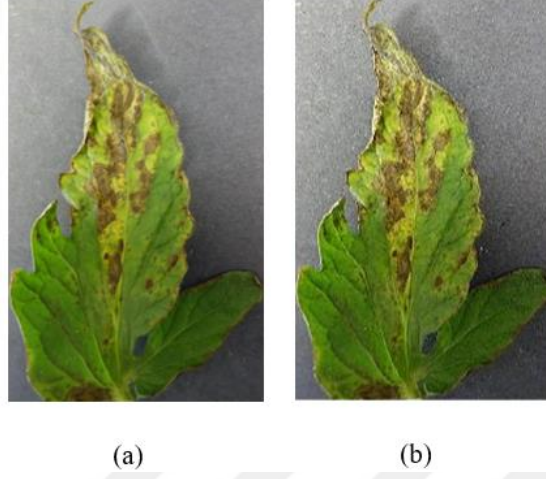
(a)

(b)

Şekil 2.3 : Medyan filtresi: (a) uygulanmamış, (b) uygulanmış.

Unsharp Filtresi: Bulanık görüntülerde ayrıntıları tekrar elde edebilmek, görüntüyü keskinleştirebilmek için kullanılmaktadır.

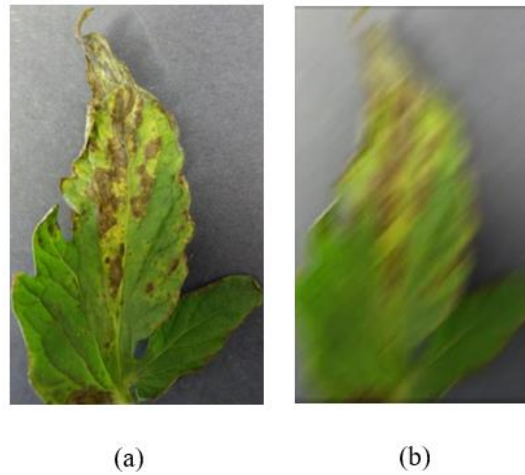
Şekil 2.4'te unsharp filtresi için örnek bir uygulama gösterilmektedir. Şekil 2.4 (a) unsharp filtresi uygulanmamış görüntüyü, Şekil 2.4 (b) unsharp filtresi uygulanmış görüntüyü göstermektedir.



Şekil 2.4 : Unsharp filtresi: (a) uygulanmamış, (b) uygulanmış.

Motion Filtresi: Görüntü herhangi bir hareket sırasında çekilmiş izlenimi oluşmasını sağlamaktadır.

Şekil 2.5'te motion filtresi için örnek bir uygulama gösterilmektedir. Şekil 2.5 (a) motion filtresi uygulanmamış görüntüyü, Şekil 2.5 (b) motion filtresi uygulanmış görüntüyü göstermektedir.



Şekil 2.5 : Motion filtresi: (a) Uygulanmamış, (b) Uygulanmış.

Blurring Filtresi: Görüntünün bulanıklaşmasını sağlamaktadır.

Şekil 2.6'da blurring filtresi için örnek bir uygulama gösterilmektedir. Şekil 2.6 (a) blurring filtresi uygulanmamış görüntüyü, Şekil 2.6 (b) blurring filtresi uygulanmış görüntüyü göstermektedir.



(a)

(b)

Şekil 2.6 : Blurring filtresi: (a) Uygulanmamış, (b) Uygulanmış.

2.3 Görüntü Sıkıştırma

Görüntü sıkıştırma, veri miktarını azaltmak için gerçekleştirilen bir işlemdir. Kayıplı ve kayıpsız olmak üzere ikiye ayrılmaktadır.

Kayıplı Görüntü Sıkıştırma: İlk olarak renk dönüşümü yapılmakta ve daha sonra fazlalıklar atılmaktadır. Ayrık dalgacık dönüşümü (ADD) ve ayrık kosinüs dönüşümü (AKD) gibi dönüşümler uygulanarak verinin daha az bit ile ifade edilmesi sağlanmaktadır. Veri, niceleme işlemi kullanılarak uygun bit oranına indirgenmekte ve kayıpsız bir yöntem kullanılarak sıkıştırılmaktadır [28].

Kayıpsız Görüntü Sıkıştırma: Veri bütünlüğünün bozulmasının istenmediği durumlarda kullanılmaktadır. Olasılık tabanlı veya sözlük tabanlı yöntemler kullanılabildiği gibi ikisinin bir arada kullanıldığı bir yöntem de uygulanabilmektedir [28].

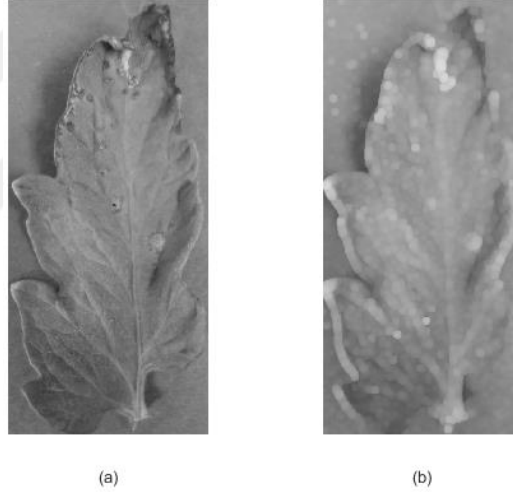
2.4 Morfolojik İşlemler

Morfoloji, biyolojinin canlıların şekil ve yapıları ile ilgilenen dalıdır. Matematiksel morfoloji ise temel küme işlemlerine dayanan, görüntüdeki sınırların tanımlanması ve çıkartılması, gürültüden kurtulma gibi uygulamalar için kullanılmaktadır [29].

İkili morfolojik işlemler aşağıda açıklanmaktadır:

Genişletme İşleme: Görüntü üzerinde gerçekleştirilen büyütme veya kalınlaştırma gibi morfolojik işlemlerdir [30].

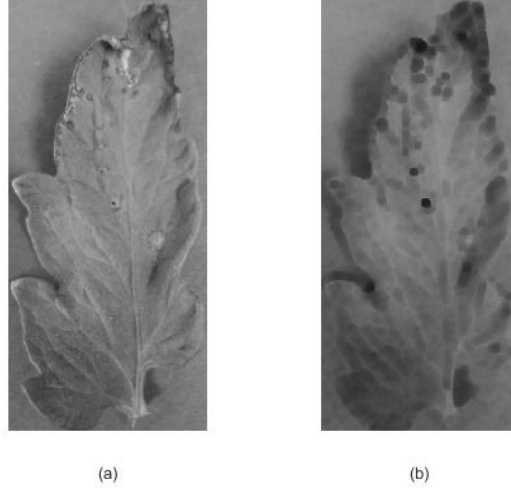
Şekil 2.7’de görüldüğü üzere genişletme işleminden sonra görüntüdeki beyaz kısımlar genişlemiştir. Şekil 2.7 (a) morfolojik işlem uygulanmamış gri görüntüyü, Şekil 2.7 (b) genişletme işlemi uygulanmış görüntüyü göstermektedir.



Şekil 2.7 : Genişletme işlemi: (a) uygulanmamış, (b) uygulanmış.

Aşındırma İşlemi: Görüntü üzerinde gerçekleştirilen küçültme veya inceltme gibi morfolojik işlemlerdir [30].

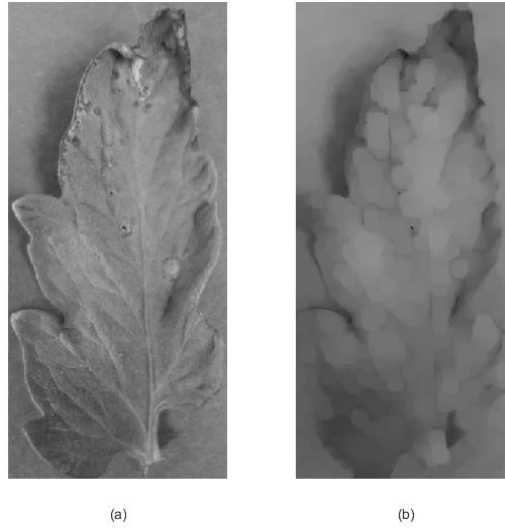
Şekil 2.8’de görüldüğü üzere aşındırma işleminden sonra görüntüdeki beyaz kısımlar aşınmıştır. Şekil 2.8 (a) morfolojik işlem uygulanmamış gri görüntüyü, Şekil 2.8 (b) aşındırma işlemi uygulanmış görüntüyü göstermektedir.



Şekil 2.8 : Aşındırma işlemi: (a) uygulanmamış, (b) uygulanmış.

Açma İşlemi: Aşındırma işleminin uygulanmasının ardından genişletme işlemi uygulanması ile elde edilmektedir. Görüntüdeki nesnelere ve boşluklara temizlenmekte, görüntüde kalan nesnelere ilk durumlarına göre daha küçük bir duruma gelmektedir.

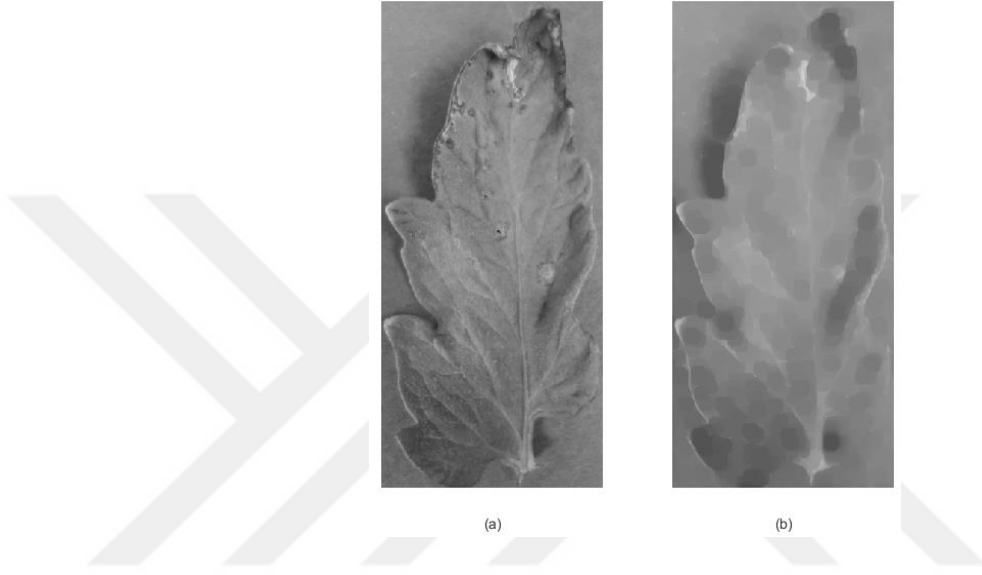
Şekil 2.9'da görüldüğü üzere açma işleminden sonra görüntüdeki siyah kısımlar genişlemiş ve beyaz kısımlar küçülmüştür. Şekil 2.9 (a) morfolojik işlem uygulanmamış gri görüntüyü, Şekil 2.9 (b) açma işlemi uygulanmış görüntüyü göstermektedir.



Şekil 2.9 : Açma işlemi: (a) uygulanmamış, (b) uygulanmış.

Kapama İşlemi: Genişletme işleminin uygulanmasının ardından aşındırma işlemi uygulanması ile elde edilmektedir. Görüntüde bulunan birbirine yakın nesnelere arasında boşlukların kapanmasını ve noktaların birleşmesini sağlamaktadır.

Şekil 2.10'da görüldüğü üzere kapama işleminden sonra görüntüdeki beyaz kısımlar genişlemiştir. Şekil 2.10 (a) morfolojik işlem uygulanmamış gri görüntüyü, Şekil 2.10 (b) kapama işlemi uygulanmış görüntüyü göstermektedir.



Şekil 2.10 : Kapama işlemi: (a) uygulanmamış, (b) uygulanmış.



3. DERİN ÖĞRENME

Derin öğrenme, büyük boyuttaki verileri daha düşük boyutlu verilere dönüştürmek amacı ile derin mimarileri kullanmakta olan çok sayıda algorithmadan oluşmaktadır [31]. Derin öğrenmenin temelini genellikle YSA oluşturmaktadır ve YSA'dan farklı olarak YSA'nın derinleştirilmiş halidir. Derin öğrenme olarak adlandırılmasının sebebi ağda birçok gizli katman olmasından kaynaklıdır [32]. YSA'da olduğu gibi beynin çalışma sistemindeki sinyal aktarımının simüle edilmesi ile gerçekleştirilmektedir [33].

Derin öğrenmenin kullanımının artmasındaki en büyük etken teknolojinin gelişmesi ile birlikte işlemci yeteneklerinin artmasıdır. İşlemci yetenekleri ile birlikte sistemin eğitiminde kullanılan veri sayısı ve ağlarda kullanılan katman sayısı artmıştır. Bu sayede elde edilen sonuçlarda büyük oranda iyileşmeler gerçekleşmiş bulunmaktadır. Katman sayısının ve veri setinin artması ile daha karmaşık problemler üzerine odaklanmak ve çözüm üretmek kolaylaşmıştır.

3.1 Yapay Sinir Ağları

YSA insan beyninin çalışma yapısından esinlenerek geliştirilmiş algoritmalarıdır. Beyindeki sinir hücrelerinin (nöron) çalışma yapısına benzer bir şekilde çalışmaktadırlar.

İnsan beyninde bulunan sinir hücreleri, sinir sisteminin temel yapı ve görev birimini oluşturmaktadır. İnsan vücudunda bulunan sinir hücrelerinde, dentritler kendilerine ulaşan sinyalleri çekirdeğe iletmektedirler. Çekirdek dentritten aldığı sinyalleri bir araya toplamakta ve aksonlara iletmektedir. Elde edilen sinyaller akson tarafında işlemlerden geçerek sinapslara iletilmektedir. Sinapslar yeni üretilmiş olan bu sinyalleri diğer sinir hücrelerine veya dış dünyaya iletmektedir.

YSA'da bulunan yapay sinir hücreleri biyolojik sistemde bulunan sinir hücrelerine benzer şekilde çalışmaktadır. Yapay sinir hücreleri, elde ettikleri verilerin üzerine bir toplama fonksiyonu uyguladıktan sonra bir aktivasyon fonksiyonu işleminden geçirmektedir. Aktivasyon işleminden geçen veriler ise diğer hücelere iletilmektedir.

Yapay sinir hücresi genel olarak beş bölümden oluşmaktadır:

Girdiler: YSA'ya eğitilmesi için verilen verilerdir.

Ağırlıklar: Hücrelerin birbirleri arasında bulunan bağlantıların sayısal olarak değerini göstermektedir. Hücreye dışarıdan gelen bilginin değerini ve hücreye olan etkisini belirtmektedir.

Toplama Fonksiyonu: Hücreye dışarıdan verilen girdilerin ağırlıklarıyla çarpılması ve toplanması ile hücrenin net girdisinin belirlenmesini sağlamaktadır.

Aktivasyon Fonksiyonu: Elde edilen net girdinin işlenmesini ve hücrenin bu girdi için oluşturacağı çıktı değerinin belirlenmesini sağlamaktadır.

Çıktılar: Aktivasyon fonksiyonu işleminden sonra elde edilen değerlerdir. Elde edilen çıktı, hücrenin kendisine veya başka bir hücreye girdi olarak verilebilmekte ya da sonuç olarak sunulabilmektedir.

YSA, görüntü işleme, işaret işleme, finansal sistemler, örüntü tanıma, sağlık, askeri sistemler ve yapay zeka gibi bir çok alanda kullanılmaktadır. Yüz tanıma, ses tanıma, nesne tespiti, sınıflandırma gibi bir çok işlem YSA kullanılarak yapılabilmektedir.

3.1.1 Yapay sinir ağları tarihçe

Warren McCulloch ve Walter Pitts tarafından YSA'ya ait ilk model 1943 yılında geliştirilmiş ve YSA'nın temelleri oluşturmuştur [34]. McCulloch ve Pitts, insan beyninin sahip olduğu hesaplama yeteneğinden örnek alarak, basit bir sinir ağı modelini elektrik devreleri kullanarak modellemişlerdir. Donald Hebb, Hebb kuralı ile sinir ağlarında bulunan bağlantı sayısının öğrenme ile ilişkili olduğunu ortaya koymuştur [35].

İlk nörobilgisayarlar 1950'lerde geliştirilmiştir. 1958 yılında tek katmanlı sinir ağı modeli olan perceptron Frank Rosenbatt tarafından ortaya atılmıştır. Perceptron, girdi ve çıktı katmanlarından oluşmaktadır.

Adaline modeli, 1959 yılında Bernard Widrow ve Ted Hoff tarafından geliştirilmiştir. Adaline yapı olarak Perceptron modeline benzese de öğrenme kuralı ondan farklıdır. Adaline modelinde, verilen giriş elemanlarına ağırlık değerleri ile çarpma işlemi uygulandıktan sonra bias değeri ile toplanmaktadır. Çıkan sonucun 0 değerinden büyük olması durumunda çıkış değeri 1 olarak, küçük olması durumunda ise -1 olarak atanmaktadır.

1969 yılında Minsky ve Papert tek katmanlı yapay sinir ağlarının özel-veya (XOR) işlemi için yetersiz olduğunu ortaya koymuştur. Bu durum YSA ile ilgili çalışmaların belli bir süre duraklama dönemine girmesine sebep olmuştur.

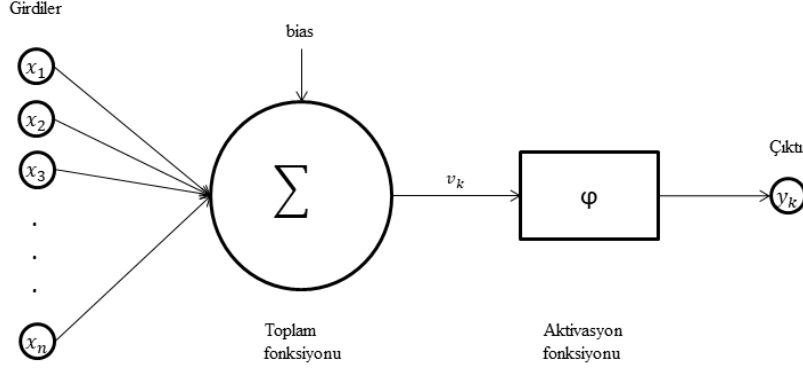
1974 yılında Kohonen algoritması geliştirilmiştir. Kohonen algoritması eğiticişiz bir öğrenme algoritmasıdır.

Hopfield ağı, 1974'te Little tarafından açıklanan ancak 1982'de John Hopfield tarafından ön plana çıkarılan tekrarlayan bir YSA'dır. Hopfield ağları geri beslemeli sinir ağları içerisinde yer almaktadır. Bu yaklaşımdaki temel amaç, enerjiyi en düşük seviyeye indirebilmek için gerekli olan sinaptik ağırlık değerlerine yakınsamadır. Hopfield ağları insan hafızasını anlayabilmek için bir model sağlamaktadır.

1986 yılında, David Rumelhart ve James McClelland geliştirdikleri geri yayılım algoritması ile çok katmanlı YSA'nın eğitilebileceğini ortaya koymuşlardır. Bu sayede YSA'nın popülerliği tekrar artmış ve duraklama döneminden çıkmıştır.

3.1.2 Yapay sinir ağlarının genel yapısı

YSA, sinir hücrelerinin birbirlerine bağlanması sonucuyla oluşmaktadır. YSA, girdi katmanı, bir veya daha fazla ara (gizli) katman ve çıktı katmanı olarak üç katmandan oluşmaktadır. Şekil 3.1'de YSA için örnek bir yapı verilmiştir.



Şekil 3.11 : Örnek bir yapay sinir ağı.

Girdi katmanı: Eğitilmek istenen verilerin girdi olarak YSA'ya verildiği katmandır.

Ara katman: Girdi katmanından elde edilen veriler bu katmana verilmektedir. Ara katman sayısı problemin boyutuna göre değişebilmektedir. Ara katman YSA'nın ayrılmaz bir parçası değildir. Bazı durumlarda kullanılmamaktadır, ağda hiç ara katman bulunmamaktadır. Bazı durumlarda ise YSA'da birden fazla ara katman kullanılmaktadır.

Çıktı katmanı: Ara katmanda elde edilen veri işlenmekte ve girdi katmanında verilen veriye uygun bir çıktı üretilmektedir. Bir çıktı hücrelerinin bir tane çıktısı olabilmektedir. Her bir çıktı hücresi bir önceki katmanda bulunan bütün hücreler ile bağlantı içerisindedir.

3.1.3 Yapay sinir ağlarının yapıları

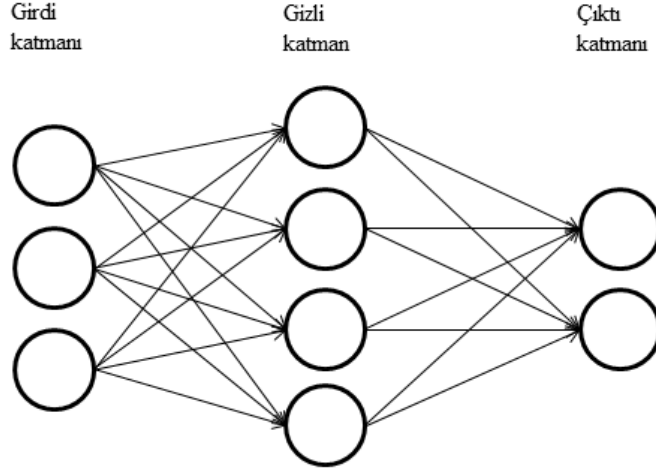
YSA yapılarına göre, öğrenme algoritmalarına göre ve öğrenme zamanına göre olmak üzere 3'e ayrılmaktadır.

3.1.3.1 Yapılarına göre yapay sinir ağları

YSA, yapılarına göre ileri beslemeli ağlar ve geri beslemeli ağlar olmak üzere ikiye ayrılmaktadır.

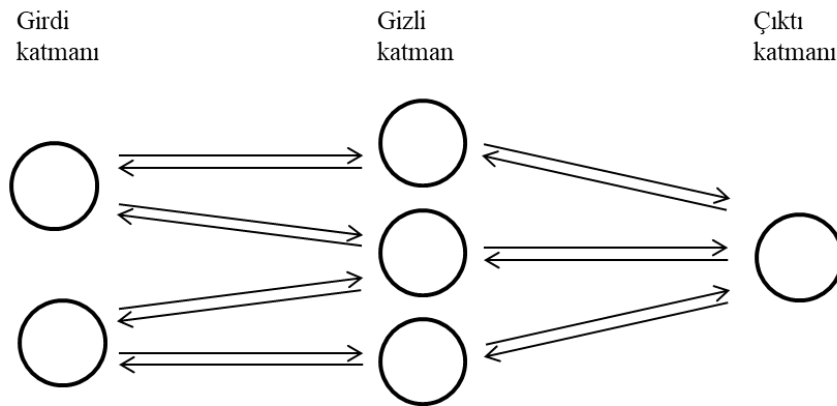
İleri beslemeli YSA'da sinir hücreleri girdi katmanından çıktı katmanına doğru düzenli bir şekilde ilerlemektedir. Ağda önceki sinir hücrelerine dönülecek şekilde bir etkileşim bulunmamaktadır. Kendinden sonra gelen katmana doğru ilerleyen bir

bağ bulunmaktadır. İleri beslemeli ağda gelen veriler girdi katmanına verilmekte, buradan elde edilen sonuçlar da ara katmanlara iletilmektedir. Ara katmanlardan elde edilen verilerde çıktı katmanına iletilerek bir sonuç elde edilmektedir. Ağda herhangi bir şekilde geriye dönüş işlemi bulunmamaktadır. Şekil 3.2’de ileri beslemeli ağa örnek bir yapı gösterilmektedir.



Şekil 3.12 : İleri beslemeli ağ örneği.

Geri beslemeli ağlarda ise hücreden elde edilen çıktı hem kendinden sonra gelen hücreye hem de kendinden önce gelen hücreye girdi olarak verilmektedir. Ağ bu şekilde en uygun sonuç elde edene kadar eğitilmektedir. Şekil 3.3’te geri beslemeli ağa örnek bir yapı gösterilmektedir.



Şekil 3.13 : Geri beslemeli ağ örneği.

3.1.3.2 Öğrenme algoritmalarına göre yapay sinir ağları

YSA öğrenme algoritmalarına göre, danışmanlı öğrenme, danışmansız öğrenme ve destekleyici öğrenme olarak üç grupta incelenmektedir.

Danışmalı öğrenmede, ağa verilen girdi değerleri için uygun çıktı değerleri de verilmektedir. Ağ, ağırlıklarını verilen girdi değerlerine uygun çıktı değerlerine göre güncellemektedir. Ağdan elde edilen çıktı değerleri ve başlangıçta ağa verilen çıktı değerleri arasında bir karşılaştırma yapılmakta ve ağın hatası hesaplanmaktadır. Ağ için oluşturulan yeni ağırlık değerleri bu hataya göre yeniden hesaplanmaktadır.

Danışmansız öğrenmede ağa sadece eğitilmesi istenen girdi değerleri verilmektedir. Örnek bir çıktı verisi verilmemektedir. Girdi katmanında verilen verilere göre ağ her bir veriyi kendi arasında sınıflandıracak şekilde eğitmektedir.

Destekleyici öğrenmede ise ağa her iterasyon sonucu elde ettiği bilginin iyi veya kötü bir sonuç olduğuna göre bilgi verilmektedir. Ağ verilen bu bilgilere göre kendi üzerinde yeni düzenlemeler yapmaktadır. Bu durum ağın herhangi bir girdi verisi ile hem öğrenip hem sonuç çıkarmasını sağlamaktadır.

3.1.3.3 Öğrenme zamanına göre yapay sinir ağları

YSA öğrenme sürelerine göre statik öğrenme ve dinamik öğrenme olmak üzere ikiye ayrılmaktadır.

Statik öğrenmede, YSA çalıştırılmadan önce eğitimden geçirilmektedir. Ağ eğitim tamamlandıktan sonra amacına göre kullanılmaktadır. Statik öğrenmede, kullanım aşamasında ağ üzerinde bulunan ağırlıklarda herhangi bir değişim olmamaktadır.

Dinamik öğrenmede ise YSA eğitim işlemi sırasında öğrenecek şekilde çalıştırılmaktadır. Eğitim aşaması bittikten sonra elde ettikleri çıkışın onay durumuna göre ağ üzerinde bulunan ağırlıklar güncellenmektedir. Eğitim çalışma aşamasında da devam etmektedir.

3.1.4 Yapay sinir ağlarının özellikleri

YSA'nın öğrenebilme ve kendi kendini düzenleyebilme yetenekleri bulunmaktadır. Verilen bilgilerde eksiklikler olması durumunda da çalışabilmektedir. Oluşan bir

hataya karşı hata toleransı bulunmaktadır. Verilen bilgilerin belirsiz ve tam olmaması durumunda da işlem yapabilmektedir.

YSA'nın diğer algoritmalara göre birçok avantajı bulunmaktadır. Birden fazla sayıda hücreden oluştukları ve ağda bulunan hücreler eş zamanlı olarak çalışabildikleri için geleneksel algoritmaların çözmekte zorlandıkları karmaşık problemleri çözebilmekte oldukça başarılıdır. Eğitim esnasında verilen verilerden genel olarak bir sonuç elde etmesi ile eğitim aşamasında kullanılmayan veriler için de anlamlı sonuçlar ortaya koyabilmektedir.

Bilgiyi geleneksel algoritmalarından farklı şekilde işledikleri için geleneksel algoritmalarda bulunan olumsuzlukları bulundurmamaktadır. Elde edilen bilgiler bütün ağda saklandığı için hücrelerden bazılarının işlevini kaybetmesi durumunda da ağdaki anlamlı bilgi kaybolmamaktadır.

YSA, kendisine verilen örneklerden genellemeler çıkardığı için ağa daha önce hiç verilmemiş veriler hakkında da bilgi üretebilmektedir. YSA, örüntü ilişkilendirme ve sınıflandırma işlerini yapabilmektedir.

Kendisine verilen örüntü örnek verilerini örüntünün kendisi ile ya da diğer örüntüler ile ilişkilendirebilmektedir. Verilen örüntünün eksik olması durumunda, eksik gelen bilgileri tamamlayarak örüntü tamamlama yapabilmektedir. Hata toleransına sahip oldukları için, ağlar ani bozulma göstermemektedir. Bir hata olması durumunda, ağ yavaş ve göreceli olarak dereceli bir şekilde bozulma göstermektedir.

YSA'nın avantajlarının yanında bazı dezavantajları da bulunmaktadır. Eğitim sırasında genellikle iyi bir işlemciye ihtiyaç duymaktadır. Bu yüzden donanıma bağımlıdır. Donanımın kötü olması eğitimi etkileyebilmekte, yavaşlatabilmektedir. Problem için uygun olan ağ yapısını seçebilmek için bir kurala sahip olmadığından, uygun olan ağ yapısını seçmek için deneme yanılma yolu kullanılmaktadır.

Ağda kullanılacak parametreler de deneme yanılma yolu ile belirlenmektedir. YSA, sayısal bilgilerle çalıştığı için veriler ağa verilmeden önce sayısal değerlere çevrilmek zorundadır. YSA, verilen probleme ürettiği çözüm hakkında kesin bir bilgi vermemektedir. Bu sebeple ağın davranışları kesin olarak açıklanamamaktadır.

Geleneksel algoritmalar ve YSA'nın karşılaştırılması Çizelge 3.1'de verilmiştir.

Çizelge 3.1 : Geleneksel algoritmalar ve yapay sinir ağıları karşılaştırması.

Geleneksel Algoritmalar	Yapay Sinir Ağları
Algoritmanın oluşturduğu çıkışlar, girişlerin belirlenen kurallara uygulanması sonucu meydana gelir.	Giriş çıkış bilgileri öğrenme sırasında verilir ayrıca kurallar da bu anda belirlenir.
Algoritmalar ve bilgiler mutlak hesaplamalara dayanır.	Tecrübelerden faydalanır.
Merkezi, ardışık ve eş zamanlı hesaplama yapılıır.	Toplu, öğrenmeden sonra paralel devam eden ve eş zamansız hesaplama yapılıır.
Hazır bilgi depolanmış ve hafıza paketlenmiştir.	Hafıza ayrılmış ve ağa yayılmıştır.
Yanılma toleransı mevcuttur.	Yanılma toleransı mevcut değildir.
Oldukça hızlıdır.	Donanıma tabiidir ayrıca yavaştır.

3.1.5 Yapay sinir ağlarının eğitilmesi

YSA, insan beyninden esinlenerek tasarlandığı için insan beynine benzer şekilde çalışmaktadır. Beyin zaman içerisinde kendisine gelen bilgilerle gelişmekte, öğrenmekte ve tecrübe kazanmaktadır. Karşılaşılan yeni olaylara tepkisini daha önce edindiği ve yorumladığı verilere göre vermektedir. YSA da karşılaştığı yeni bir problemi daha önce eğitildiği ve yorumladığı duruma göre çözmekte, cevaplamaktadır. Elde ettiği cevabı daha önce eğittiği sistemdeki çıkış ile karşılaştırarak bir hata oranı belirlemektedir.

Hata oranını azaltmak ve istenilen çıkış değerine yaklaşabilmek amacı ile çeşitli öğrenme algoritmaları geliştirilmiştir. Bu algoritmalarda, ağ çalıştığı sürece ağırlıklar güncellenmektedir. Verilen girdiler ve istenilen çıkışların birbirlerine uyumlu olması durumunda, bu sonucun elde edilmesini sağlayan ağırlıklar saklanmaktadır. Verilerin girdi olarak verilmesinden, beklenen sonuca ulaşana kadar geçen zaman öğrenme olarak adlandırılmaktadır.

Ağ, öğrenme işlemini tamamladıktan sonra eğitim sırasında kullanılmayan veriler ile test edilmektedir. Test edilmesinin sonunda verdiği sonuçların doğru ve yanlış oranı istenilen oranın altındaysa ağ iyi eğitilememiş, istenilen değer üzerinde ise ağ ezberleme yapmış demektir. Test verilerinden elde edilen sonuç istenilen orana yaklaşıyor ise ağ iyi bir şekilde eğitilmiş demektir.

YSA'nın eğitim aşaması birkaç adımdan oluşmaktadır. Ağ eğitime başlanmadan önce, probleme uygun veriler toplanmaktadır. Eldeki verilerin bir kısmı ağa başlangıçta eğitim verisi olarak verilmekte, kalan kısmı ise test verisi olarak kullanılmaktadır. Ağın eğitim verileri ile öğrenmesi tamamlandığında, test verileri ile test edilmektedir. Sonucun iyi olması durumunda ağ başarılı bir şekilde öğrenmiş demektir.

Çözülmesi gereken probleme göre ağın yapısı başlangıçta belirlenmektedir. Girdi sayısı, ara katman sayısı, çıktı sonucunda oluşacak sınıf sayısı burada belirlenmektedir. Ağın öğrenme katsayısı, toplama ve aktivasyon fonksiyonları gibi parametreler ağ eğitime başlanmadan önce belirlenmektedir.

Eđitim bařlangıç ařamasında ađırlık deđerleri manuel belirlenebileceđi gibi rastgele olarak oluřacak řekilde de belirlenebilmektedir. Ađ đrenme sırasında iteratif olarak ađırlıkları gncelleyerek en uygun ađırlık deđerlerini bulmaya alıřmaktadır. Bu iřlemi de kendisine verilen yeni rneklere uyum sađlayacak řekilde yapmakta ve en uygun deđerlere ulařmaya alıřmaktadır. Ađ eđitim sırasında elde ettiđi ıktıyı, bařlangıta verilen ıktı ile karřılařtırmakta ve hata deđerini belirlemektedir. Yeni ađırlık deđerleri, ıkan hata deđerinin azalmasını sađlayacak řekilde gncellenmektedir. Uygun ađırlık deđerlerinin ne olduđu bilinmediđi iin YSA'nın davranıřlarına bir yorum ve aıklama getirmek mmkn olmamaktadır.

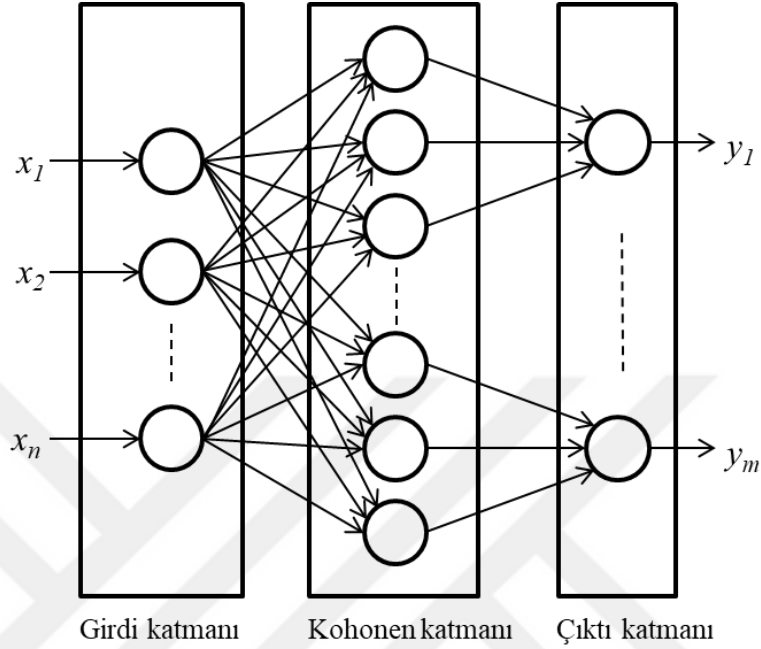
Ađ bazen hata dzeyinin stnde takılabilmektedir. Bu durumlar iin ađın parametrelerinde deđiřiklikler yapılmakta ve ađ yeniden eđitilmektedir. Ya da ađın modeli deđiřtirilebilmektedir.

3.1.6 LVQ ađı

LVQ ađı, 80'li yıllarda Kohonen'in z rgtlemeli ađ alıřmasından hemen sonra sunulmuřtur. Hem bu ađ hem de z rgtlemeli ađlar Kohonen katmanı zerine inřa edilmiřtir. Katman, maddeleri benzer nesnelerin bulunduđu uygun kategorilere ayırma zelliđine sahiptir. Kohonen [36] tarafından sunulan LVQ modeli, rekabeti đrenmeyi denetimli đrenme ile birleřtiren bir sinir ađıdır. LVQ, sınıflandırma problemlerini ozmek iin kullanılan gl ve sezgisel bir algoritmadır. Basit topolojisi ve uyarlanabilir modeli nedeniyle, LVQ birok uygulamada yaygın olarak kullanılmaktadır. Verilen verileri sabit sayıda sınıf ile sınıflandırmaktadır [37]. LVQ, hem grnt iřleme hem de sınıflandırma problemleri iin kullanılan bir sinir ađı modelidir.

Yapı olarak ađ, girdi katmanı, Kohonen katmanı ve ıktı katmanından oluřmaktadır. Ađın ıktı katmanında ayrı sınıflar veya kategoriler olduđu gibi ok sayıda iřleme elemanı bulunmaktadır. Her bir sınıftaki iřleme elemanlarının sayısı, girdi ve ıktı arasındaki iliřkinin karmařıklıđına bađlıdır. Genellikle, her bir sınıf katman boyunca eř sayıda elemana sahip olmaktadır. Bir eđitim kmesi zerinden bađlantılı sınıflandırmayı đrenen ve gerekleřtiren katman Kohonen katmanıdır. Sınıflandırılan sonular, Kohonen katmanından dođrusal ıktı katmanına

aktarılmaktadır [38]. Şekil 3.4'te LVQ'nun çalışma yapısına bir örnek verilmiştir. Ağlar eğitim kümesinin kalıtsal topolojisini koruyup girdileri sınıflandırmak üzere eğitilebilmektedirler. Örneğin, önceden öğrenilmemiş olan girdi kalıpları eğitim verileri içinde en yakın komşuları tarafından kategorize edilmektedir.



Şekil 3.14 : LVQ'nun çalışma yapısı

Öğrenme biçiminde, Kohonen katmanını kullanmaktadır. Örneğin bir eğitim vektörünün her bir elemanına uzaklığı hesaplanmakta ve en yakın eleman kazanan ilan edilmektedir. Tüm katmanda tek bir kazanan bulunmaktadır. Kazanan, girdi vektörünün ait olduğu kategori veya sınıfı açıklayarak sadece bir tane çıktı işleme elemanının harekete geçmesini sağlamaktadır. Eğer kazanan eleman eğitim vektörünün beklenen sınıfına ait ise eğitim vektörü doğrultusunda güçlendirilmektedir.

LVQ'da, ağırlıklardan oluşan referans vektörleri öğrenme sınıflarını temsil etmek için kullanılmaktadır. Öğrenme, girdi vektörü ve referans vektörleri arasındaki benzerliğe dayanmaktadır. Çıktının yalnızca biri 1 değerini almakta, diğerleri 0 değerini almaktadır. 1 değerini alan referans vektörü girdi vektörünün sınıfını vermektedir.

LVQ modeli, "kazanan hepsini alır" yaklaşımına göre çalışmakta ve yalnızca her girişte girdi vektörüne en yakın olan kazanan referans vektörünün ağırlıkları güncellenmektedir. Kazanan referans vektörü, girdi vektöründen referans vektörlerin her birine Öklid mesafesinin Denklem 3.1 ile hesaplanmasıyla bulunmaktadır.

$$d = \arg \min_i \{|x - w_i|\} \quad (3.1)$$

Denklem 3.3'te , x giriş vektörünü, w_i , i. referans vektörünü, $\eta \in (0, 1)$ ve η öğrenme oranını temsil etmektedir. $\eta(t)$ kademeli olarak azaltılmaktadır. Sınıflandırma doğrusa referans vektörler Denklem 3.2 ile güncellenmekte, aksi takdirde 3.3 ile güncellenmektedir.

$$w_i(t + 1) = w_i(t) + \eta(t)(x - w_i(t)) \quad (3.2)$$

$$w_i(t + 1) = w_i(t) - \eta(t)(x - w_i(t)) \quad (3.3)$$

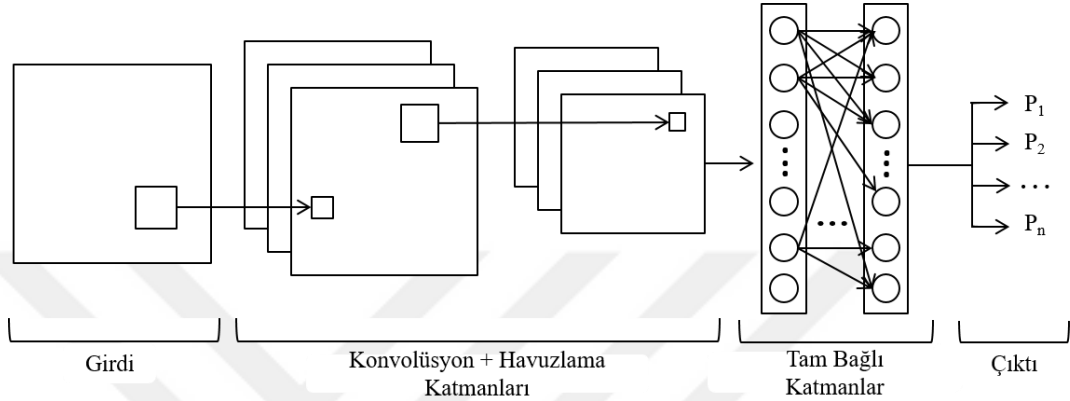
Referans vektörü ve girdi vektörü sınıfları eşleşirse, referans vektörü girdi vektörüne yaklaşmaktadır. Aksi takdirde girdi vektöründen [14] uzaklaşmaktadır. Güncelleme referans vektörlerinin bir işlemi, sınıflandırma oranına ulaşılan veya maksimum çevrim sayısına ulaşılan kadar tekrar etmektedir.

3.2 Konvolüsyonel Sinir Ağları

KSA, genellikle görüntüleri sınıflandırmakta, benzerliklerine göre kümelemekte ve görüntüde nesne tanıma yapmakta kullanılan derin YSA'lerden oluşmaktadır. KSA ile yüz tanıma, insan tanıma, hastalık tespit etme gibi birçok işlem yapılabilmektedir. Ses, spektrum olarak görsel bir şekilde sunulduğu zaman KSA ile üzerinde işlem yapılabilmektedir. KSA'nın görüntü işlemedeki başarısının yüksek olması, KSA'ya olan ilgiyi arttırmaktadır.

KSA, konvolüsyon katmanı, havuzlama katmanı, aktivasyon katmanı ve tam bağlı katman olmak üzere dört katmandan oluşmaktadır. Girdi olarak verilen görüntüler ilk olarak konvolüsyon katmanında işlenmektedir. Girdi görüntülerine çeşitli filtreleme işlemleri uygulanarak hem görüntünün özellik haritası çıkarılmakta hem de görüntünün boyutu düşürülmektedir.

Konvolüsyon işleminden sonra elde edilen yeni veriler havuzlama katmanından geçirilmekte ve boyut tekrar azaltılmaktadır. Çıkan sonuçlara normalizasyon işlemi uygulanmakta ve yeni değerler elde edilmektedir. Bu işlem, başlangıçta tanımlanan konvolüsyon ve havuzlama sayısına ulaşıncaya kadar devam etmektedir. İstenilen sonuca ulaşıldığında tam bağlı katmanda yeni veriler ile eğitim işlemi gerçekleştirilmektedir. KSA'nın genel yapısı Şekil 3.5'te gösterilmektedir.



Şekil 3.15 : KSA'nın genel yapısı.

3.3 Konvolüsyon katmanı

Konvolüsyon katmanı, ağır hesaplama işlerinin gerçekleştiği, KSA'nın çekirdeği olan katmandır. Bu katmanda, giriş görüntüsünün özellik haritasını çıkarmak için bir dizi matematiksel işlem gerçekleştirilmektedir [39]. Bu katmanda filtreleme işlemleri gerçekleştirilmektedir.

Konvolüsyon katmanının parametreleri bir dizi öğrenilebilir filtre içermektedir. Katmanda bulunan her filtre uzamsal olarak küçüktür ancak girişte verilen verilerin hacminin tam derinliği boyunca uzanmaktadır. Örneğin, $250 \times 250 \times 3$ boyutlu girdi verileri için konvolüsyonda kullanılan filtreler $G \times Y \times 3$ şeklinde olmaktadır. Burada filtrenin genişliğini G , yüksekliğini ise Y göstermektedir.

Filtreler, görüntünün genişliği, yüksekliği ve hacmi boyunca görüntü üzerinde kaydırılmakta ve filtre ve görüntüde aynı pozisyonda bulunan pikseller üzerinde hesaplama yapılmaktadır. Filtrenin, görüntü üzerinde bu şekilde kaydırılması ve

noktasal bazlı hesaplama işlemleri bittikten sonra çıkan sonuçlardan iki boyutlu bir etkinleştirme haritası elde edilmektedir.

Ağ, bir renk lekesi, petek veya tekerlek deseni gibi her hangi bir desene denk geldiğinde etkinleştirme haritası sayesinde sezgisel olarak öğrenmektedir. Çıkan etkinleştirme haritaları, derinlik boyutu boyunca istiflenmekte ve çıktı hacmi bu şekilde elde edilmektedir. Çıktı hacminin boyutu derinlik, adım ve sıfır dolgusu olmak üzere üç parametreye bağlıdır.

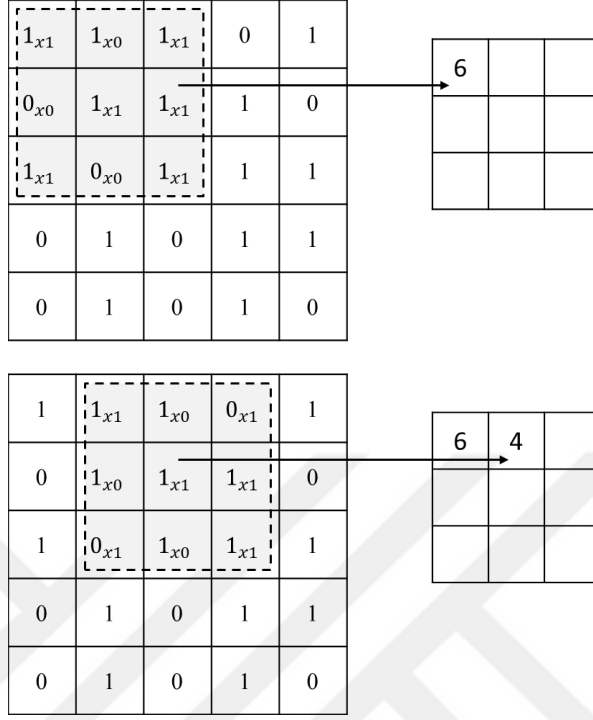
Derinlik filtre sayısına karşılık gelmekte ve her bir girdide farklı bir özellik aramayı öğrenmektedir. Örneğin, ilk konvolüsyon katmanında ham görüntü girdi olarak alınır, farklı sinir hücreleri derinlik boyutu boyunca çeşitli kenarların, renklerin veya lekelerin olması durumunda aktive olabilmektedir.

Adım sayısı, filtrenin kaç piksel kaydırılacağına karar vermektedir. Adım sayısının bir olması durumunda, filtre görüntü boyunca her seferinde bir piksel kaydırılmaktadır. İki olması durumunda ise her seferinde ikişer piksel kaydırılmaktadır. İstenirse adım sayısı ikiden büyük alınabilmektedir ama bu nadir görülen bir durumdur. Adım sayısı konvolüsyon işleminin sonunda çıkan verinin boyutunu etkilemektedir. Adım sayısının büyük olması durumunda elde edilen veri sayısı az olmakta ve ağ daha kolay eğitilmektedir. Ancak bu da verilerden elde edilen bilgilerde kayıplar olmasına sebebiyet verebilmektedir.

Filtre görüntü boyunca istenilen adım sayısına göre kaydırıldıktan sonra yeni bir matris (etkinleştirme haritası) elde edilmektedir. Elde edilen matrisi diğer adımlarda daha kolay kullanabilmek için boyutunda ayarlamalar yapmak gerekebilmektedir. Bu ayarlamaların gerekli olduğu durumlarda elde edilen matrise sıfır dolgusu uygulanmaktadır.

Konvolüsyon katmanında, filtre verilen görüntü girdi matrisinin üzerinde yukarı sol baştan başlayarak istenilen adım sayısı kadar kaydırılmaktadır. Her adımda filtredeki pikseller, görüntüde buldukları noktaya karşılık gelen pikseller ile çarpılmakta ve sonuçlar toplanmaktadır. Elde edilen toplama sonucu filtre genişliğinin filtre yüksekliğine çarpımına bölünmektedir. Çıkan değer, yeni matristeki uygun olan

noktaya yazılmaktadır. Şekil 3.6'da konvolüsyon katmanı çalışma yapısı gösterilmektedir.



Şekil 3.16 : Konvolüsyon katmanı çalışma yapısı.

Konvolüsyon işleminde kullanılan formül Denklem 3.4'te verilmiştir. Burada ζ , işlem sonucunda elde edilen çıktıyı, F filtre matrisinin boyutunu (genişlik, yükseklik), B filtre matrisini, G ise girdi matrisini temsil etmektedir.

$$\zeta_{i,j} = \sum_{k=1}^F \sum_{l=1}^F B_{k,l} G_{i+k-1,j+l-1} \quad (3.4)$$

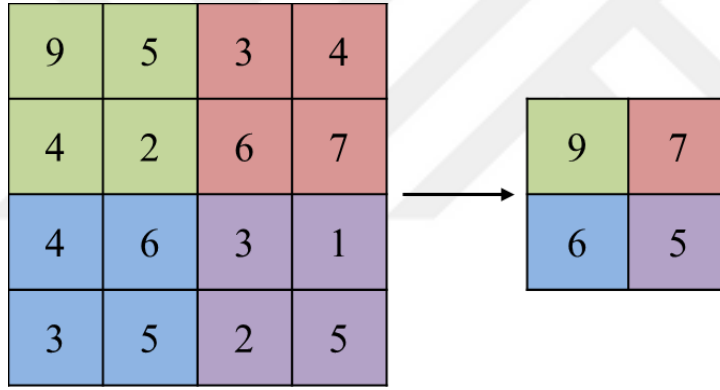
Konvolüsyon işlemi sonunda elde edilecek yeni matrisin boyutu Denklem 3.5'e göre hesaplanmaktadır. G , girdi matrisinin boyutunu ve F filtre matrisinin boyutunu temsil etmektedir. A , adım sayısını ve S ise sıfır doldurma sayısını temsil etmektedir.

$$((G - F + (2 \times S))/A) + 1 \quad (3.5)$$

3.3.1 Havuzlama katmanı

Konvolüsyon katmanının ardından genellikle havuzlama katmanı eklenmektedir. Bu katmanda amaç, ağda bulunan parametre ve hesaplama miktarını azaltmaktır. Konvolüsyon katmanında olduğu gibi burada da filtre görüntü üzerinde sol baştan sağa doğru kaydırılıp yeni matris oluşturulmaktadır. Bu katmanda kullanılan filtre genellikle 2×2 boyutlu olmaktadır. Filtreleme işlemi için genellikle maksimum havuzlama tercih edilse de, ortalama havuzlama ve L2-norm havuzlama gibi farklı havuzlama yöntemleri de kullanılmaktadır.

Maksimum havuzlama fonksiyonu ile yapılan havuzlama işleminde, filtre görüntünün üzerinde kaydırılmakta ve görüntüde filtrenin bulunduğu alana karşılık gelen değerlerden en büyüğü alınmaktadır. Filtre kaydırıldıkça alınan değerlerden yeni bir matris oluşturulmaktadır. Şekil 3.7'de örnek bir maksimum havuzlama işlemi gösterilmektedir.



Şekil 3.17 : Havuzlama işleminin çalışma şekli.

Bu katmanda derinlik korunmakta ancak yükseklik ve genişlik azaltılmaktadır. Elde edilen yeni matrisin boyutu Denklem 3.6 ile hesaplanmaktadır. Denklemden G , girdi matrisinin boyutunu ve F filtre matrisinin boyutunu temsil etmektedir. A ise adım sayısını göstermektedir.

$$((G - F) / A) + 1 \quad (3.6)$$

3.3.2 Aktivasyon katmanı

Aktivasyon katmanı ile elde edilen yeni matris verilerinde düzenlemeler yapılmakta ve veriler YSA'da eğitmeye uygun hale getirilmektedir. Normalizasyon işlemi sırasında genellikle ReLU aktivasyon fonksiyonu kullanılmaktadır. ReLU, rectified linear unit (doğrultulmuş doğrusal birim)'in kısaltılmış halidir. ReLU, verilere doygun olmayan aktivasyon fonksiyonunu uygulamaktadır. Verilerde bulunan negatif değerleri sıfır değerine ayarlayarak etkili bir şekilde kaldırmaktadır. Denklem 3.7'de ReLU fonksiyonu gösterilmiştir.

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases} \quad (3.7)$$

ReLU dışında hiperbolik tanjant veya sigmoid gibi farklı aktivasyon fonksiyonları da kullanılabilir ancak çok büyük bir hata oranı olmadan ağı daha hızlı eğitilmesini sağladığı için yaygın olarak ReLU aktivasyon fonksiyonu tercih edilmektedir [40].

3.3.3 Tam bağlı katman

Tam bağlı katman, bir önceki katmanda bulunan verilerin tümüne bağlıdır. Bu katmandaki verileri almakta ve YSA ile eğitilmesini sağlamaktadır. Katman sonucunda başlangıçta istenilen sınıf sayısı kadar sınıf oluşturulmaktadır. Nesne tanıma ve sınıflandırma işlemleri bu katmanda gerçekleştirilmektedir. Eğitim işlemi için uygun YSA bu katmanda uygulanmaktadır.

Ağ olarak, genellikle geri yayımlı algoritma tercih edilse de herhangi bir yapay sinir ağı modeli kullanılabilir. Seçilen ağa bazı durumlarda seyreltme (dropout) işlemi uygulanmaktadır. Bu sayede aşırı uyumsuzluk engellenmiş olmaktadır [31]. Seyreltme, derin öğrenmede sıkça kullanılan regülerizasyon yöntemlerinden biridir. Tam bağlı katmanda belli bir eşik değerinin altında kalan bağlantıların koparılması esasına dayanan bir tekniktir. Bağlantıların koparılması, zayıf bilgilerin unutulması anlamına gelmektedir.



4. MATERYAL VE YÖNTEM

4.1 Materyal

4.1.1 Kullanılan program

Tez çalışmasında program olarak MathWorks tarafından geliştirilen MATLAB (MATrix LABoratory) programı kullanılmıştır.

4.1.2 Domates yaprakları veri seti

Tez çalışmasında, bitkide hastalık tespitini gerçekleştirebilmek için 500 tane sağlıklı ve hastalıklı domates yapraklarından oluşan görüntü verisi kullanılmıştır. Görüntülerin 400 tanesi eğitim verisi, 100 tanesi de test verisi olarak kullanılmıştır. Kullanılan görüntü verileri PlantVillage [41] veri setinden elde edilmiştir. Veriler dört adet hastalıklı ve bir adet sağlıklı olmak üzere beş farklı sınıftan oluşmaktadır. Hastalık sınıfları; bakteriyel leke (bacterial spot), geç yanıklık (late blight), septoria yaprak lekesi (septoria leaf spot) ve sarı yaprak kıvrıcıklığı (yellow leaf curl)'dır.

Bakteriyel Leke: Hastalığın belirtileri, küçük, sarı-yeşil lezyonlar veya yapraklarda koyu, yağlı görünen lezyonlar olarak ortaya çıkmaktadır. Hastalık çok hızlı yayılmakta ve kontrol etmesi çok zor olmaktadır. Bu sebeple en tehlikeli domates hastalıklarından birisidir. Bu hastalık domatesin bitkisine ve pazarlanabilirliğine zarar verebilmektedir.

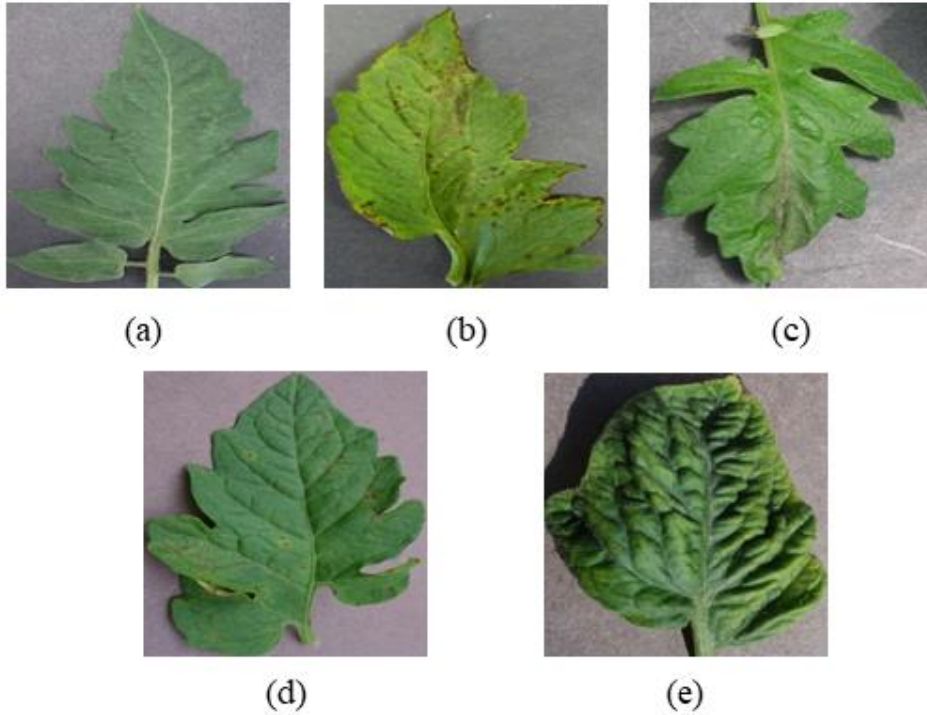
Geç Yanıklık: İlk olarak yaşlı yapraklarda yeşil-gri kenarlı kahverengi lekeler olarak ortaya çıkmaktadır. Hastalık olgunlaştıkça yapraklar üzerinde bulunan lekeler koyulaşmaktadır. Sonunda hastalık bütün bitkiye bulaşmakta ve bitkinin ciddi şekilde zarar görmesine neden olmaktadır.

Septoria Yaprak Lekesi: İlk olarak bitkinin alt yapraklarında ortaya çıkmaktadır. Belirtiler, yaprakların altında oluşan yuvarlak, sarı, suya batırılmış gibi görünen

lekelerdir. Yapraklarda küçük kahverengi veya siyah lekeler neden olmaktadır. Noktaların boyutu 1.5 mm ile 6.5 mm arasında deęişiklik göstermektedir.

Sarı Yaprak Kıvrıkcılığı: Bitkinin cüce ve bodur kalmasına neden olmaktadır. Yapraklar içe ve yukarıya doğru yuvarlanmaktadır. Genellikle yaprakların aşağı doğru bükülmesine neden olmaktadır. Yaprakların sert, normalden daha kalın hale gelmesine ve kösele cilt dokusuna sahip olmasına sebep olmaktadır. Genç ve hastalıklı yapraklar hafif sarımsı hale gelmektedir.

Şekil 4.1’de 5 farklı domates yaprağı için örnek görüntüler verilmiştir. (a) yaprağın sağlıklı durumunu, (b) bakteriyel leke hastalığını, (c) geç yanıklık hastalığını, (d) septoria yaprak lekesi hastalığını, (e) sarı yaprak kıvrıkcılığı hastalığını göstermektedir.



Şekil 4.1 : Domates yaprak hastalığı veri seti örnek görüntüleri: (a) Sağlıklı. (b) Bakteriyel leke. (c) Geç yanıklık. (d) Septoria yaprak lekesi. (e) Sarı yaprak kıvrıkcılığı.

4.1.3 Elma yaprakları veri seti

Yalova’da bulunan elma ağaçlarından alınan görüntüler ile bir veri seti oluşturulmuştur. Elma ağaçlarından telefon kamerası ile elde edilen 180 adet 2900×2900 boyutundaki yaprak görüntüleri kullanılmıştır. Bu görüntülerden 156

tanisi hastalıklı, geri kalanlar ise sağlıklı olmak üzere ayrılmıştır. Görüntülerin elde edildiği ağaçların bulunduğu Yalova'daki bahçede genellikle elma kara leke hastalığı olmasından dolayı, yapraklar üzerinde kara leke hastalığının olup olmadığı tespiti yapılmıştır.

Elma Kara Leke Hastalığı: Belirtileri yaprak, sürgün ve meyvede ortaya çıkmaktadır. Başlangıçta yaprağın alt ve üst kısmında ortaya çıkan yağlımsı görünüşteki lekeler zaman içerisinde zeytin rengine dönüşmektedir. Daha sonra da renkleri kahverengini almaktadır. Yapısı kadifemsidir. Zaman içerisinde lekenin olduğu kısımda bulunan doku ölmekte, üzerinde çatlaklar ve delikler oluşmaktadır. Hastalığı ağır olan yapraklar sararmakta ve erkenden dökülmektedirler [42].

Şekil 4.2'de Yalova'daki elma ağaçlarından elde edilen görüntülerden örnekler verilmiştir. Şekilde (a) sağlıklı yaprakları, (b) hastalıklı yaprakları göstermektedir.



Şekil 4.18 : Elma ağaçlarından alınan gerçek verilere örnek görüntüler: (a) Sağlıklı. (b) Hastalıklı.

4.2 Yöntem

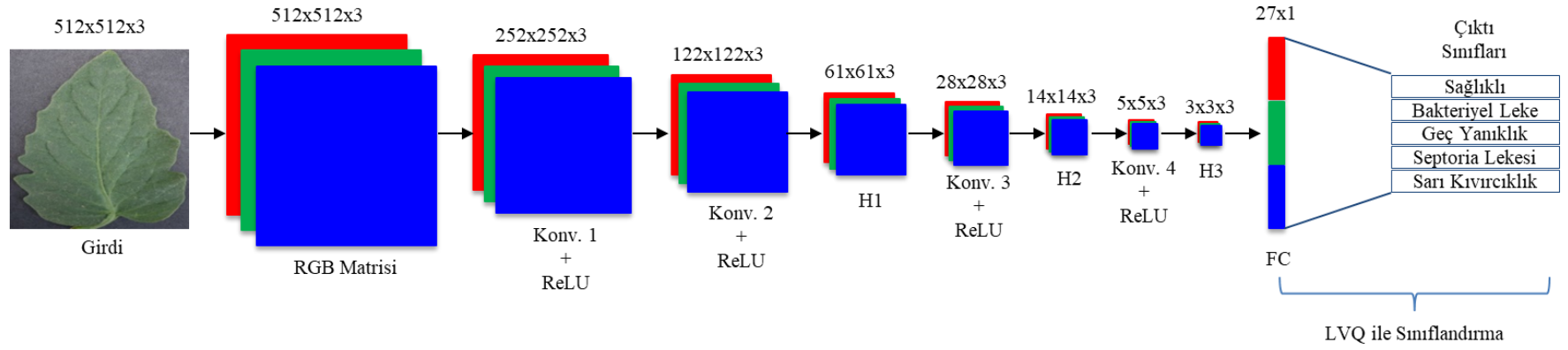
4.2.1 Domates yapraklarında hastalık tespiti

Domates yapraklarında hastalık tespiti için ilk olarak KSA tabanlı bir model oluşturulmuştur. KSA ile eğitilmek istenen görüntüler öncelikle R, G ve B kanallarına ayrılmıştır. Standart KSA modellerinde farklı kanallar ilk konvolüsyon

işleminde sonra kanallardaki aynı koordinatlara denk gelen değerleri toplanarak tek bir kanal haline getirilmektedir. Bu durum da birinci konvolüsyon işleminden sonra kanalların ayrı ayrı önemini yitirmesine neden olmaktadır. Bu bakımdan tez çalışmasında standart KSA'dan farklı olarak her bir kanal ayrı ayrı konvolüsyon aşamasından geçirilmiş ve görüntülerin özellik haritaları çıkarılmıştır. En son tam bağlı katmana her bir kanal verileri tek sütunlu bir vektör olarak birleştirilerek girdi olarak verilmiştir. Bu model ile R, G ve B kanallarının özelliklerinin kaybolmaması amaçlanmıştır. Konvolüsyon işleminden sonra elde edilen matrislere ReLU aktivasyon fonksiyonu uygulanmıştır. Oluşan yeni verilere konvolüsyon ve ReLU işlemi tekrar uygulanmıştır. Daha sonraki aşamada maksimum havuzlama işlemi gerçekleştirilmiştir.

Konvolüsyon, ReLU aktivasyon fonksiyonu ve maksimum havuzlama uygulanması iki kere daha tekrarlanmıştır. Birinci ve ikinci konvolüsyon aşamasında 9×9 'luk, üçüncüsünde ve dördüncüsünde ise 5×5 'lik filtreler uygulanmıştır. Standart KSA çalışmalarından birden fazla filtre kullanılmaktadır. Bu çalışmada daha az filtre ile eğitim gerçekleştirilip tek bir filtre ile modelin başarısı test edilmiştir. Diğer yandan da tek bir filtre uygulayarak modelin hızlı çalışması amaçlanmıştır. Bu işlemlerden sonra her bir kanal için 3×3 boyutunda R, G ve B kanalları için 3 tane matris elde edilmiştir. Elde edilen matrisler tek boyutlu bir vektör haline dönüştürülmüş ve 27×1 boyutunda bir vektör elde edilmiştir. Bu vektörün ilk 9 elemanı R kanalını, ikinci 9 elemanı G kanalını ve üçüncü 9 elemanı B kanalını temsil etmektedir.

Orijinal görüntülerden elde edilen 500 özellik vektörü eğitim ve test işlemleri için kullanılmıştır. 500 özellik vektörünün 400 tanesi eğitim için, geri kalan 100 tanesi ise test için kullanılmıştır. Özellik vektörleri tam bağlı katmanda LVQ algoritması ile eğitilmişlerdir. LVQ algoritmasında, Kohonen katmanındaki ağda her sınıf için 10 tane olmak üzere 50 tane sinir hücresi kullanılmıştır. Çıktı katmanı her sınıfı temsil eden bir nöron olmak üzere beş nörondan oluşmaktadır. Maksimum epoch sayısı bütün deneylerde 300 olarak ve öğrenme oranı da 0.1 olarak seçilmiştir. Sunulan metodun yapısı Şekil 4.3'te verilmiştir. Eğitimin sonuçları Çizelge 4.1'de verilen karmaşıklık matrisinde gösterilmiştir. Eğitim sonuçları sadece bir filtre uygulandığında ortaya çıkan sonuçlardır.



Şekil 4.19 : Domates yapraklarında hastalık tespitinde kullanılan metodun yapısı



Çizelge 4.1 : Karmaşıklık matrisi şeklinde sınıflandırma sonuçları.

Yaprak Hastalığı	Sağlıklı	Bakteriyel Leke	Geç Yanıklık	Septoria Yaprak Lekesi	Sarı Kıvrıcılık	Doğruluk
Sağlıklı	18	0	0	0	2	%90
Bakteriyel Leke	0	18	0	0	2	%90
Geç Yanıklık	0	0	17	0	3	%85
Septoria Lekesi	1	0	0	16	3	%80
Sarı Kıvrıcılık	0	0	0	3	17	%85
Ortalama						%86

4.2.2 Elma ağaçlarında hastalık tespiti

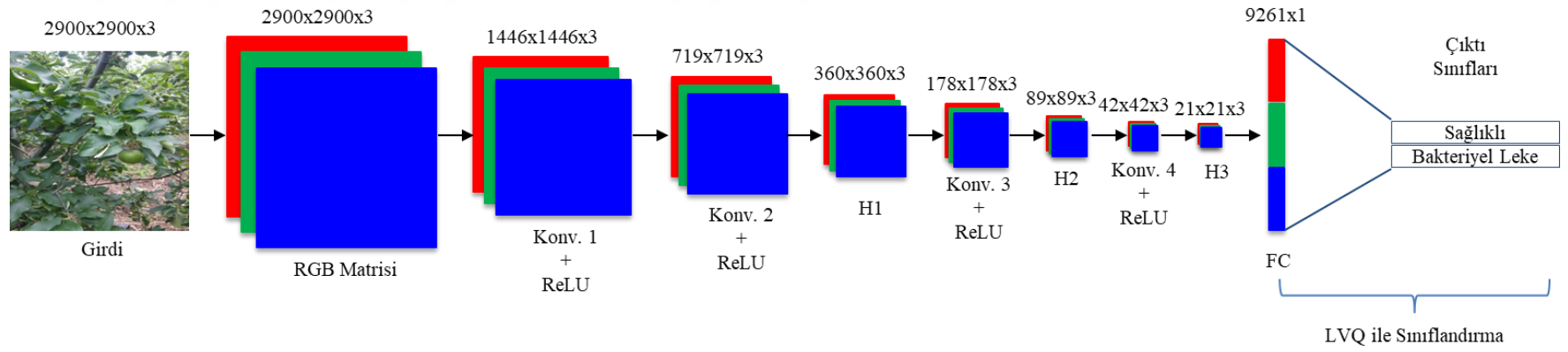
Domates yaprakları için veri setini oluşturan görüntülerde sadece bir tane yaprak görüntüsü bulunmaktadır fakat elma yaprakları için kullanılan görüntülerde elma ağaçları üzerindeki yaprakların görüntüleri alındığından dolayı çok fazla sayıda yaprak görüntüsü bulunmaktadır. Elma yapraklarında hastalık tespiti bu bakımdan daha karmaşık olmaktadır. Hastalıkların tespiti için bir görüntü verisinde bulunan tüm yaprakların tek tek etiketlenmesi gerekmektedir.

Elma yapraklarından elde edilen görüntülerden oluşan veri setinde hastalık tespiti için domates yapraklarında olduğu gibi KSA ve LVQ kullanılmıştır. Domates yapraklarından farklı olarak burada sobel yatay filtre, sobel dikey filtre, uzaysal filtre ve ortalama filtresi olmak üzere dört farklı filtre kullanılmıştır. KSA aşaması yukarıda 4.2.1’de domates yapraklarında hastalık tespitinde yapıldığı gibi gerçekleştirilmiştir.

KSA sonucunda elde edilen özellik vektörleri LVQ algoritması ile eğitilmiştir. LVQ uygulanırken, Kohonen katmanındaki ağda her sınıf için 25 tane olmak üzere 50 tane sinir hücresi kullanılmıştır. Çıktı katmanı her sınıfı temsil eden bir sinir hücresi olmak üzere iki sinir hücresinden oluşmaktadır. Maksimum epoch sayısı bütün deneylerde 2000 olarak ve öğrenme oranı da 0.1 olarak seçilmiştir. Çalışmanın sonuçları Çizelge 4.2’ de, sunulan metodun yapısı ise Şekil 4.4’te verilmiştir.

Çizelge 4.2 : Karmaşıklık matrisi şeklinde sınıflandırma sonuçları.

Yaprak Hastalığı	Sağlıklı	Bakteriyel Leke	Doğruluk
Sağlıklı	50	0	%100
Bakteriyel Leke	3	100	%97
Ortalama			%98.5



Şekil 4.20 : Elma ağaçlarında hastalık tespitinde kullanılan metodun yapısı.

5. SONUÇ VE ÖNERİLER

Görüntü işleme ve yapay zeka alanlarındaki hızlı gelişmelere bağlı olarak tarımsal alanlardaki çalışmalar da bu teknolojik gelişmelerden payını almış ve hassas tarım kavramı ortaya çıkmıştır. Hassas tarımda toprağın işlenmesi, gübreleme, ilaçlama, sulama, hasat tahmini, ürünlerde hastalık tespiti gibi pek çok uygulama bulunmaktadır. Son yıllarda yapay zeka alanlarından derin öğrenme yaklaşımları bitkilerde hastalık tespitini başarılı bir şekilde gerçekleştirebilmektedir. Ayrıca yapay zeka yöntemleri ile birlikte hastalık tespitinin manuel yaklaşımlara göre çok daha hızlı ve daha başarılı sonuçlar ürettiği görülmektedir.

Tez çalışmasında domates ve elma yapraklarındaki hastalıkların ve hastalık türünün tespitine yönelik derin öğrenme tabanlı bir yaklaşım sunulmuştur. Domates yaprakları verileri PlantVillage veri tabanından alınmıştır. Elma yaprakları ise Yalova'da bulunan elma ağaçlarının kamera ile görüntüleri alınarak elde edilmiştir. Çalışmada KSA ve LVQ ağı tabanlı bir model sunulmuştur. Standart KSA çalışmalarında kullanılan geri yayılım veya benzer öğrenme algoritmalarından farklı olarak LVQ ağı ile hastalıkların sınıflandırılması amaçlanmıştır.

Standart KSA'nın çalışma yapısında ilk konvolüsyon işleminden sonra farklı katmanlar tek bir katman olarak birleştirilmektedir. Örneğin RGB tabanlı bir çalışmada R, G ve B kanalları ilk konvolüsyon işleminden sonra tek bir kanal olarak birleştirilmektedir. Bu durum da kanalların özelliklerinin kaybolmasına sebep olabilmektedir. Bu tez çalışmasında standart KSA'dan farklı olarak her bir kanala konvolüsyon işlemleri ayrı ayrı uygulanmış ve tam bağlı katmana girildiği olarak verilmeden önce tek bir sütun verisi olarak birleştirilerek verilmiştir. Böylece derin öğrenme için R, G ve B kanallarının bilgisi korunarak sistemin eğitilmesi gerçekleştirilmiştir.

Domates yapraklarında tek bir filtre kullanılarak gerçekleştirilen hastalık tespitinde %86'lık bir başarı oranı ve elma ağaçlarında hastalık tespitinde ise dört farklı filtre

kullanılarak %98.5'lik bir başarı oranı elde edilmiştir. Domates yaprakları için veri setini oluşturan görüntülerde sadece bir tane yaprak görüntüsü bulunuyorken, elma yaprakları için kullanılan görüntülerde çok fazla sayıda elma yaprağı görüntüsü bulunmaktadır. Elma yapraklarında hastalık tespiti bu bakımdan daha karmaşık olmaktadır. Domates yapraklarında hastalık tespitinde beş farklı sınıf üzerinden sınıflandırma işlemi gerçekleştirilmiştir. Elma ağaçlarında hastalık tespitinde ise hastalıklı ve sağlıklı olmak üzere iki sınıf üzerinden sınıflandırma işlemi gerçekleştirilmiştir. Elma hastalık tespitinde sadece bir tane hastalıklı sınıf olması Yalova'daki elma ağaçlarından alınan görüntülerin alındığı zamanda elmalarda sadece kara leke hastalığının bulunmasından kaynaklanmaktadır.

Sonraki çalışmalarda tarımsal alandaki görüntülerin otomatik alınması için insansız hava aracı (İHA) kullanılabilir. İHA'lar son yıllarda pek çok alanda olduğu gibi tarımsal alanlarda da kullanılmaya başlamıştır. Özellikle üzerindeki kameraların teknolojik gelişmelere bağlı olarak yüksek görüntü kalitesine sahip olması ile birlikte İHA'lar tarımsal alanlarda otomatik olarak uzaktan yüksek çözünürlüklü görüntü alabilme yeteneğine sahiptir. Ayrıca İHA'lar üzerindeki görüntüler kablosuz bağlantı vasıtasıyla akıllı telefonlar ya da tabletlere de aktarılabilir. Günümüzde akıllı telefonlar ya da tabletler üzerinde derin öğrenme uygulamaları kolay bir şekilde çalışabilmektedir. Bu durumda İHA'lardaki kameradan alınan görüntüler doğrudan akıllı telefon veya tablete gönderilebilir ve hastalık tespiti ayrıca bir bilgisayara gerek kalmadan cihaz üzerinde otomatik olarak gerçekleştirilebilir.

KAYNAKLAR

- [1] **Arivazhagan, S., Shebiah, R. N., Ananthi, S., & Varthini, S. V.** (2013). Detection of unhealthy region of plant leaves and classification of plant leaf diseases using texture features. *Agricultural Engineering International: CIGR Journal*, 15(1), 211-217.
- [2] **Liu, B., Zhang, Y., He, D., & Li, Y.** (2017). Identification of apple leaf diseases based on deep convolutional neural networks. *Symmetry*, 10(1), 11.
- [3] **Islam, T., Rahman, T., Akter, M., & Rana, M. P., and Jenkins, J. M.,** 2018: Noise reduction in rice leaf disease detection using gaussian filtering technique. *European Journal of Scientific Research*, 149, 24-33.
- [4] **Kaur, I., Aggarwal, G., Verma, A.,** 2016: Detection and classification of disease affected region of plant leaves using image processing technique. *Indian Journal of Science and Technology*, 9(48).
- [5] **Pujari, J. D., Yakkundimath, R., & Byadgi, A. S.,** 2015: Image processing based detection of fungal diseases in plants. *Procedia Computer Science*, 46, 1802-1808.
- [6] **Barbedo, J. G. A.,** 2016: A review on the main challenges in automatic plant disease identification based on visible range images. *Biosystems engineering*, 144, 52-60.
- [7] **Padol, P. B., & Yadav, A. A.,** 2016: SVM classifier based grape leaf disease detection. In 2016: Conference on advances in signal processing (CASP) (pp. 175-179). IEEE.
- [8] **Singh, V., & Misra, A. K.** (2017). Detection of plant leaf diseases using image segmentation and soft computing techniques. *Information processing in Agriculture*, 4(1), 41-49.
- [9] **Akram, T., Naqvi, S. R., Haider, S. A., Kamran, M.,** 2017: Towards real-time crops surveillance for disease classification: exploiting parallelism in computer vision. *Computers & Electrical Engineering*, 59, 15-26.
- [10] **Zhang, S., Wu, X., You, Z., Zhang, L.,** 2017: Leaf image based cucumber disease recognition using sparse representation classification. *Computers and electronics in agriculture*, 134, 135-141.
- [11] **Elangovan, K., Nalini, S.,** 2017: Plant disease classification using image segmentation and SVM techniques. *International Journal of Computational Intelligence Research* ISSN 0973-1873, 13(7), 1821-1828.

- [12] **Khirade, S. D., & Patil, A. B.** (2015). Plant disease detection using image processing. In 2015 International conference on computing communication control and automation (pp. 768-771). IEEE.
- [13] **Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., & Stefanovic, D.** (2016). Deep neural networks based recognition of plant diseases by leaf image classification. Computational intelligence and neuroscience, 2016.
- [14] **Ferentinos, K. P.** (2018). Deep learning models for plant disease detection and diagnosis. Computers and Electronics in Agriculture, 145, 311-318.
- [15] **Kamilaris, A., & Prenafeta-Boldú, F. X.** (2018). Deep learning in agriculture: A survey. Computers and Electronics in Agriculture, 147, 70-90.
- [16] **Lu, Y., Yi, S., Zeng, N., Liu, Y., & Zhang, Y.** (2017). Identification of rice diseases using deep convolutional neural networks. Neurocomputing, 267, 378-384.
- [17] **Amara, J., Bouaziz, B., & Algergawy, A.** (2017). A deep Learning-based approach for banana leaf diseases classification. In BTW (Workshops) (pp. 79-88).
- [18] **Fuentes, A., Yoon, S., Kim, S., & Park, D.** (2017). A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. Sensors, 17(9), 2022.
- [19] **Mohanty, S. P., Hughes, D. P., & Salathé, M.** (2016). Using deep learning for image-based plant disease detection. Frontiers in plant science, 7, 1419.
- [20] **Durmuş, H., Güneş, E. O., & Kırcı, M.** (2017, August). Disease detection on the leaves of the tomato plants by using deep learning. In 2017 6th International Conference on Agro-Geoinformatics (pp. 1-5). IEEE.
- [21] **Ashqar, B. A., & Abu-Naser, S. S.** (2018). Image-Based tomato leaves diseases detection using deep learning.
- [22] **Cruz, A. C., Luvisi, A., De Bellis, L., & Ampatzidis, Y.** (2017). Vision-Based plant disease detection system using transfer and deep learning. In 2017 ASABE Annual International Meeting (p. 1). American Society of Agricultural and Biological Engineers.
- [23] **Çomak, B., Beycioğlu, A., Başığit, C., Kılınçarslan, Ş.** (2011). Beton teknolojisinde görüntü işleme tekniklerinin kullanımı. In 6th International Advanced Technologies Symposium (IATS'11).
- [24] **Demirkol, A.,** 2006: Doğrultu yoğunluk fonksiyonu ile radar görüntü işleme. Signal Processing and Communications Applications.
- [25] **Çelik, H. İ., Dülger, L. C., Topalbekiroğlu M.,** 2012: Görüntü işleme teknikleri kullanarak kumaş hatalarının belirlenmesi. Tekstil Teknolojileri Elektronik Dergisi, 6, 22-39.

- [26] **Demirbaş, H. Y., Dursun İ.,** 2007: Buğday tanelerinin bazı fiziksel özelliklerinin görüntü işleme tekniğiyle belirlenmesi. *Tarım Bilimleri Dergisi*, 13, 176-185.
- [27] **Sabancı, K., Aydın, C.,** 2014: Görüntü işleme tabanlı hassas ilaçlama robotu. *Tarım Bilimleri Dergisi*, 20(4), 406-414.
- [28] **Mesut, A.** (2006). *Veri sıkıştırma yeni yöntemler* (Doktora Tezi). <https://tez.yok.gov.tr/UlusalTezMerkezi/> sitesinden alındı.
- [29] **Karhan, M., Oktay, M. O., Karhan, Z., Demir, H.,** 2011: Morfolojik görüntü işleme yöntemleri ile kayıslarda yaprak delen (çil) hastalığı sonucu oluşan lekelerin tespiti. In 6th International Advanced Technologies Symposium (IATS'11), 172-176.
- [30] **Atalı, G., Karayel D., Özkan, S. S.,** 2016: Morfolojik görüntü işleme tekniği ile yapay sinir ağlarında görüntü tahribat analizi. *Academic Platform Journal of Engineering and Science (APJES)*, 4(1).
- [31] **Büyükyılmaz, M.** (2017). *Mikroskopik görüntüler üzerinde derin öğrenme algoritmaları kullanarak hastalıklı hücrelerin otomatik tanımlanması* (Yüksek Lisans Tezi). <https://tez.yok.gov.tr/UlusalTezMerkezi/> sitesinden alındı.
- [32] **Eral, M.** (2016). *Deep learning approach for laboratory mice grimace scaling* (Master Thesis). Retrieved from <https://tez.yok.gov.tr/UlusalTezMerkezi/> sitesinden alındı.
- [33] **Kaya, A.** (2017). *Derin öğrenme ile rakam öğretme* (Yüksek Lisans Tezi). <https://tez.yok.gov.tr/UlusalTezMerkezi/> sitesinden alındı.
- [34] **Keskenler, M. F., & Keskenler, E. F.** (2017). Geçmişten günümüze yapay sinir ağları ve tarihçesi. *Takvim-i Vekayi*, 5(2), 8-18.
- [35] **Hebb, D. O.** (1949). *The organization of behavior; a neuropsychological theory.* A Wiley Book in Clinical Psychology., 62-78.
- [36] **Kohonen, T.** (1989). *Self-organization and associative memory.* Springer-Verlag, third ed.
- [37] **Mokbel, B., Paassen, B., Schleif, F. M., & Hammer, B.** (2015). Metric learning for sequences in relational LVQ. *Neurocomputing*, 169, 306-322.
- [38] **Li, X., Zhang, Y.** (2016). Digital image edge detection based on LVQ neural network. In *2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA)* (pp. 1251-1255). IEEE.
- [39] **Tümen, V., Söylemez, Ö. F., & Ergen, B.** (2017). Facial emotion recognition on a dataset using convolutional neural network. *2017 International Artificial Intelligence and Data Processing Symposium (IDAP)*.
- [40] **Yiğit, A.** (2016). *İş süreçlerinde insan görüşünü derin öğrenme ile destekleme.* (Yüksek lisans tezi). <https://tez.yok.gov.tr/UlusalTezMerkezi/> sitesinden alındı.

[41] **Url-1** <<https://github.com/spMohanty/>>, alındığı tarih: 20.01.2018

[42] **Url-2** <<https://www.tarimorman.gov.tr/GKGM>>, alındığı tarih: 20.01.2018



ÖZGEÇMİŞ

Ad Soyad: Melike (SARDOĞAN) DOĞAN

Doğum Yeri ve Tarihi: İZNİK/BURSA 13.05.1993

Lisans: Yalova Üniversitesi Bilgisayar Mühendisliği

TEZDEN TÜRETİLEN YAYINLAR/SUNUMLAR

- **Sardoğan M.**, Tuncer A., Özen Y., 2018: Plant Leaf Disease Detection and Classification based on CNN with LVQ Algorithm, *3rd International Conference on Computer Science and Engineering*, September 20-23, 2018 Saraybosna, BosnaHersek.