

T.C.
BOLU ABANT İZZET BAYSAL ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI



DERİN ÖĞRENME TABANLI AKILLI GENİŞ ALAN
SULAMA SİSTEMİ

YÜKSEK LİSANS TEZİ

MESUT BUDAK

TEZ DANIŞMANI
Dr. Öğr. Üyesi Mehmet Milli

BOLU, TEMMUZ - 2024

KABUL VE ONAY SAYFASI

Mesut BUDAK tarafından hazırlanan “**DERİN ÖĞRENME TABANLI AKILLI GENİŞ ALAN SULAMA SİSTEMİ**” adlı tez çalışması jürimiz tarafından Bilgisayar Mühendisliği Anabilim Dalı’nda Yüksek Lisans Tezi olarak oy birliği ile kabul edilmiştir. 1/07/2024

Jüri Üyeleri

İmza

Danışman
Dr. Öğr. Üyesi Mehmet MİLLİ
Bolu Abant İzzet Baysal Üniversitesi

.....

Üye
Doç. Dr. Ferhat DEMİRAY
Bolu Abant İzzet Baysal Üniversitesi

.....

Üye
Dr. Öğr. Üyesi Emre ÜNSAL
Sivas Cumhuriyet Üniversitesi

.....

Lisansüstü Eğitim Enstitüsü Onayı

Prof. Dr. İbrahim KÜRTÜL
Lisansüstü Eğitim Enstitüsü Müdürü

ETİK BEYAN

Bolu Abant İzzet Baysal Üniversitesi, Lisansüstü Eğitim Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu bildirir,

aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Teze ilişkin Turnitin adlı programında enstitü müdürlüğünce belirlenen filtrelemeler uygulanarak alınmış olan benzerlik raporuna göre, tezin benzerlik oranı % 30'u geçmemektedir.

.....
MESUT BUDAK

ÖZET

DERİN ÖĞRENME TABANLI AKILLI GENİŞ ALAN SULAMA SİSTEMİ YÜKSEK LİSANS TEZİ

MESUT BUDAK

BOLU ABANT İZZET BAYSAL ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

(TEZ DANIŞMANI: DR. ÖĞR. ÜYESİ MEHMET MİLLİ)

BOLU, TEMMUZ - 2024

XIII + 66

Geniş alan sulama faaliyetleri, aşırı su tüketimi, heterojen güneş ışığı maruziyeti ve tüm yüzeyde sulama adaletinin sağlanamaması gibi sorunlarla karşılaşmaktadır. Bu sorunlara çözüm olarak, sulama stratejilerinin bölgesel özelliklere uygun şekilde özelleştirilmesi ve su kaynaklarının verimli kullanılabilmesi amacıyla derin öğrenme ve görüntü segmentasyonu gibi modern teknolojilerin kullanımı bu tez çalışması kapsamında düşünülmüştür. Araştırmanın golf sahaları, stadyum çimleri ve şehir peyzajı gibi su tüketimi yüksek alanlarda hem su tasarrufu sağlayacağı hemde sulama kaynaklı heterojen bitki gelişimi sorunlarında çözüm olacağı öngörülmektedir. Bu tez kapsamında kullanılan derin öğrenme yaklaşımı ve algoritmalar, bitkinin su ihtiyacını belirleme ve sulama gereksinimlerini öngörme konusunda etkili olabilir. Aynı zamanda canlı kamera görüntüleri veya alanın fotoğraflanması gibi görüntü tabanlı veri toplama yöntemlerinden elde edilen verilerle eğitilen bir yapay zeka modelinin, sulama konusunda uzmanlaşmış bir profesyonelin becerilerini herkesin kullanımına sunması hedeflenmektedir. Görüntü tabanlı veri toplama yönteminin sensör tabanlı sistemlere nazaran daha kolay uygulanabilir olması, daha az bir maliyet gerektirmesi, işletme ve bakım maliyetinin olmaması avantajları olarak karşımıza çıkmaktadır.

ANAHTAR KELİMELER: Derin Öğrenme, Yapay Zeka, Yüzeysel Sulama, Geniş Alan, Verimlilik, Optimizasyon

ABSTRACT

**DEEP LEARNING BASED SMART WIDE AREA IRRIGATION SYSTEM
MSC THESIS
MESUT BUDAK
BOLU ABANT IZZET BAYSAL UNIVERSITY
INSTITUTE OF GRADUATE STUDIES
DEPARTMENT OF COMPUTER ENGINEERING
(SUPERVISOR: DR. LECTURER MEMBER MEHMET MİLLİ)**

BOLU, JULY 2024

XIII + 66

Wide-area irrigations face challenges such as excessive water consumption, heterogeneous exposure to sunlight, and the inability to ensure fair irrigation across the entire surface. As a solution to these problems, it is suggested to customize irrigation strategies according to regional characteristics and utilize modern technologies such as deep learning and image segmentation for efficient use of water resources. The research anticipates that it will provide both water savings and solutions to problems related to heterogeneous plant development due to irrigation in high-water-consuming areas such as golf courses, stadium turfs, and urban landscapes. Therefore, deep learning is expected to be effective in determining plant water needs and predicting irrigation requirements. Simultaneously, the goal is to make the skills of an irrigation-specialized professional accessible to everyone through an artificial intelligence model trained with data obtained from image-based data collection methods such as live camera images or field photography. The advantages of image-based data collection methods over sensor-based systems include easier applicability, lower cost, and no operational and maintenance costs.

KEYWORDS: Deep Learning, Artificial Intelligence, Surface Irrigation, wide Area, Efficiency, Optimization

İÇİNDEKİLER

Sayfa

KABUL VE ONAY SAYFASI.....	iii
ETİK BEYAN	iv
ÖZET.....	v
ABSTRACT	vi
İÇİNDEKİLER.....	vii
ŞEKİL LİSTESİ.....	ix
TABLO LİSTESİ.....	x
FOTOĞRAF LİSTESİ.....	xi
KISALTMA VE SEMBOLLER LİSTESİ.....	xii
TEŞEKKÜR.....	xiii
1. GİRİŞ.....	1
1.1 Tez Çalışmasının Amacı	2
2. GENEL BİLGİLER	3
2.1 Yapay Zeka	3
2.2 Makine Öğrenmesi.....	5
2.2.1 Kümeleme Algoritmaları.....	7
2.2.2 Sınıflandırma algoritmaları	15
2.2.3 Regresyon Algoritmaları.....	18
2.3 Derin Öğrenme	21
2.3.1 Yapay sinir ağları(ANN).....	23
2.3.2 Tekrarlayan sinir ağları (RNN).....	25
2.3.3 Uzun Kısa Süreli Bellek ağları(LSTM)	25
2.3.4 Evrişimli Sinir Ağları(CNN)	26
2.3.5 Üretken Çekişmeli Sinir Ağları(GAN)	27
2.3.6 Derin İnanç Ağları(DBN).....	27
2.3.7 Doğal Dil İşleme(NLP).....	28
3. LİTERATÜR TARAMASI.....	29
4. MATERYAL VE YÖNTEM.....	31
4.1 MFECB (Çoklu Fonksiyonel Elektronik Kontrol Kartı)	32
4.2 NodeMCU	35
4.3 Edimax Ic3030Wn IP Kamera	37
4.4 Wi-Fi Erişim Noktası(Access Point)	39
4.5 AC Motor Devir Kontrol Ünitesi.....	40
4.6 Sensör ve Aktüatörler	41
4.6.1 DS1820 Sıcaklık Sensörü.....	41
4.6.2 Basınç Şalteri.....	41
4.6.3 Selenoid Valf ve Sprinkler(Sulama Başlığı)	42
4.7 AC Jet Su Pompası	43

4.8 Yazılım ve Yazılım Geliştirme Araçları	43
4.8.1 MPLAB Yazılım Geliştirme Ortamı.....	44
4.8.2 IC-Prog Programlayıcı	44
4.8.3 Arduino Yazılım Geliştirme Ortamı	45
4.8.4 VS Code ve PlatformIO	46
4.8.5 Python Programlama Dili.....	46
4.9 Yöntem.....	52
5. BULGULAR.....	55
6. SONUÇ VE ÖNERİLER	62
6.1 Öneriler ve Gelecek Çalışmalar.....	63
7. KAYNAKLAR	64



ŞEKİL LİSTESİ

Sayfa

Şekil 2.1. Yapay zekanın alt alanları, organizasyon ve yapılanması.	5
Şekil 2.2. Makine öğrenimi kavramı ve venn diyagramı (Goodfellow, 2016). ..	7
Şekil 2.3. Makine öğrenmesi ve derin öğrenme (Goodfellow, 2016)	21
Şekil 2.4. Perceptron temel yapısı (Staudemeyer, 2019).	24
Şekil 2.5. Çok katmanlı ileri beslemeli sinir ağı (Staudemeyer, 2019).	24
Şekil 2.6. Tekrarlayan sinir ağı(RNN) grafik gösterimi.	25
Şekil 2.7. Uzun kısa süreli bellek ağları(LSTM) grafik gösterimi.	25
Şekil 2.8. Evrişimli sinir ağları (CNN) (Albawi, 2017).	26
Şekil 2.9. Üretken çekişmeli ağ (GAN) mimarisi (Bau, 2018).	27
Şekil 2.10. Derin inaç ağları ve kısıtlı Boltzmann makinesi.	28
Şekil 2.11. Doğal dil işleme(NLP) kelime vektör gösterimi (Goyal, 2018).	29
Şekil 4.1. MFECB şematik gösterimi.	33
Şekil 4.2. Çoklu fonksiyonel elektronik control kartına(MFECB) ait Pcb.	34
Şekil 4.3. Node MCU	36
Şekil 4.4. DS1820 sıcaklık sensörü.	41
Şekil 4.5. Sistem katmanları ve hiyerarşik yapılanması.	53
Şekil 4.6. Sulama sisteminin tam yapılanması ve organizasyon şeması.	54
Şekil 5.1. 2023 Nisan ayı sıcaklık ortalaması haritası.	59
Şekil 5.2. 2024 Nisan ayı sıcaklık ortalaması haritası.	59
Şekil 5.3. 2023 Mayıs ayı sıcaklık ortalaması haritası.	60
Şekil 5.4. 2024 Mayıs ayı sıcaklık ortalaması haritası.	60

TABLO LİSTESİ

Sayfa

Tablo 4.1. Dahili GPIO pinlerine karşılık gelen NodeMCU pin haritası.....	37
Tablo 4.2. NodeMCU pin numaraları ve fonksiyonları (Dutta, 2021).....	37
Tablo 5.1. Aylık bazda su tüketimi (m^3/Ay).....	61

FOTOĞRAF LİSTESİ

Sayfa

Fotoğraf 4.1. Çoklu fonksiyonel elektronik control kartı(MFECB).	32
Fotoğraf 4.2. MFECB 3D görünüm.	35
Fotoğraf 4.3. Edimax IC-3030Wn IP kamera.	38
Fotoğraf 4.4. Wi-Fi Erişim Noktası(Access Point).	39
Fotoğraf 4.5. AC asenkron motor devir hız control ünitesi.	40
Fotoğraf 4.6. Sıvı akışkan basınç şalteri.	42
Fotoğraf 4.7. Selenoid valf ve Sprinkler(Sulama Başlığı)	42
Fotoğraf 4.8. Asenkron 1 Hp 220V AC jet su pompası.	43
Fotoğraf 4.9. Microchip MPLAB yazılım geliştirme ortamı.	44
Fotoğraf 4.10. JDM uyumlu programlayıcılar için IC-Prog yazılımı.	45
Fotoğraf 4.11. NodeMCU için Arduino IDE yazılım geliştirme ortamı.	45
Fotoğraf 4.12. VS Code ve PlatformIO eklentisi.	46
Fotoğraf 4.13. PixelLib Ade20k modeli örnek girdi görsel.	48
Fotoğraf 4.14. PixelLib Ade20k modelin örnek segmentasyon çıktısı.	48
Fotoğraf 4.15. PixelLib Ade20k modelin semantic segmentasyon çıktısı.	49
Fotoğraf 4.16. PixelLib anlık segmentasyon ve sınıflandırma girdisi.	49
Fotoğraf 4.17. PixelLib anlık segmentasyon görsel çıktısı.	50
Fotoğraf 4.18. PixelLib anlık segmentasyon ve sınıflandırma çıktısı.	50
Fotoğraf 5.1. Uygulama sahasına ait 02 Nisan 2024 tarihli görsel.	55
Fotoğraf 5.2. Uygulama sahası 02 Nisan 2024 segmentasyon haritası.	56
Fotoğraf 5.3. Uygulama sahasına ait 28 Nisan 2024 tarihli görsel.	56
Fotoğraf 5.4. Uygulama sahası 28 Nisan 2024 segmentasyon haritası.	57
Fotoğraf 5.5. Uygulama sahasına ait 19 Mayıs 2024 tarihli görsel.	57
Fotoğraf 5.6. Uygulama sahası 19 Mayıs 2024 segmentasyon haritası.	58
Fotoğraf 5.7. Sitemin çalışma süresi içindeki bir zaman dilimine ait görsel. ..	58

KISALTMA VE SEMBOLLER LİSTESİ

YZ	: Yapay Zeka
ML	: Makine Öğrenimi
DL	: Derin Öğrenme
HOG	: Gradyan Histogramları
SIFT	: Ölçekten Bağımsız Özellik Dönüşümü
NLP	: Doğal Dil İşleme
TF-IDF	: Terim Frekansı-Ters Belge Frekansı
POS	: Konuşma Etiketleme
SVM	: Destek Vektör Makinesi
CNN	: Evrişimli Sinir Ağları
RNN	: Tekrarlayan Sinir Ağları
GAN	: Üretici Karşıt Sinir Ağları
RMSE	: Kök Ortalama Kare Hatası
MAPE	: Ortalama Mutlak Yüzde Hatası
RF	: Random Forest
KNN	: K-Nearest Neighbors
MST	: Minimum Spanning Tree
EM	: Expectations Maximization
FCM	: Fuzzy C- Means
GMM	: Gaussian Mixture Model
SVM	: Support Vector Machines
GBRT	: Gradient Boosted Regression Trees
YSA	: Yapay Sinir Ağları
MFECB	: Çoklu Fonksiyonel Elektronik Kontrol Kartı
GPIO	: Genel Amaçlı Giriş/Çıkış Bağlantı Noktası

TEŞEKKÜR

Lisansüstü eğitim sürecim boyunca, aldığım derslerle bilgi ve birikimlerinden büyük ölçüde istifade ettiğim danışmanım Dr. Öğr. Üyesi Mehmet Milli'nin yanı sıra değerli hocalarım Doç. Dr. Ferhat Demiray, Doç. Dr. Mehmet Dinçer Erbaş, Doç. Dr. Önder Eyecioğlu, Dr. Öğr. Üyesi Ümit Şentürk, Dr. Öğr. Üyesi Ali Çalım'a en içten dileklerimle teşekkür ederim. Mehmet Milli, tez hazırlama ve yayın yapma süreçlerinde sabrı, hoşgörüsü ve destekleriyle zaman zaman gösterdiğim direncimin üstesinden gelmiş, öğrenci-danışman ilişkisini inceliklerle korurken, aynı zamanda güçlü dostluk ve arkadaşlık bağlarını da inşa etmiştir.

Anadolu'nun bir köyünde okuma yazma bilmediği halde 'Ben onu okutup Ankara'ya reis-i cumhur yapacağım' diyen ve dünyasından büyük hayallere sahip olan sevgili annem Makbule Budak'a, yine okuma yazmayı askerlikte öğrenen, fakat eğitim ve öğretimimi devam ettirmem için imkanlarını öğrenim hayatım boyunca fedakarca esirgemeyen sevgili babam Ali Budak'a, Türkiye'de 4 adet üniversitenin bulunduğu bir dönemde yüksek öğrenimini başarıyla tamamlayan, çevresindeki herkesin itibar ettiği ve saygı duyduğu biri olarak, akademik kariyer arzusunun ailesinde yaşadığını görmek isteyen sevgili ağabeyim Adem Budak'a, öğrenim hayatımın bir döneminde evlerinin kapılarını açan, kendi çocuklarından ayırt etmeden ilgi, alaka ve sevgisini eksiltmeden veren sevgili yengem Sevim Budak ve ablam Emsal Budak'a teşekkür ediyor içlerinden ebedi aleme intikal edenleri saygı ve rahmetle anıyorum. Katıldığım veli toplantılarında, öğretmenlerinden 'Haklarında söyleyecek tek bir sözümüz yok, vatana millete faydalı olabilecek ancak böyle çocuklar yetiştirilebilirdi' sözünü işittiren, sahip oldukları erdem, ahlak değerleriyle dolu varlıkları ve başarılarıyla her zaman göğsümü kabartan, babaları olmaktan onur duyduğum sevgili çocuklarım Alperen ve Tuana Budak'a, 'İyiki benim çocuklarımsınız' demekten büyük mutluluk duyuyor ve şükrediyorum. Bilgi, birikim ve deneyim bakımından bana geniş imkanlar sunan ve çalışanı olmaktan gurur duyduğum Türkiye Radyo Televizyon Kurumu'na ve bana kazandırdığı değerler için bu kurumun sahip olduğu çok sayıdaki nadide insanlarına teşekkür borçluyum.

1. GİRİŞ

Su kaynaklarının giderek azaldığı günümüz dünyasında, geniş alan sulamaları sadece tarım alanlarını değil, aynı zamanda spor sahalarını, şehir peyzajındaki yeşil alanları ve ticari/kurumsal binaların çevresindeki manzarayı şekillendiren çim alanlarını da içermektedir. Özellikle golf sahaları ve stadyum çimleri gibi geniş yüzeylere sahip olan ve doğası gereği su tüketiminin oldukça fazla olduğu bu alanlarda, suyun etkin ve adil bir şekilde dağıtılması, çevresel sürdürülebilirlik ve su kaynaklarının etkin yönetimi açısından hayati bir önem taşımaktadır. Ancak, bu sulama pratiğinin karmaşıklığını artıran önemli faktörler ise, bitkilerin güneşe maruz kalma süreleri, mevsimsel faktörler ve havada ki nemin büyük ölçüde değişken olmasıdır. Bir diğer faktör ise bazı bölgeler yoğun güneş ışığına maruz kalırken, diğer bölgelerde gölge alanların oluşabilmesidir. Su kıtlığı yaşayan bölgeler tam olarak tespit edilebiliyor olsa bile, sabit sulama menziline sahip fiskiye başlıkları, bu bölgelerin sınırlarıyla tam olarak örtüşecek bir sulama pratiği sağlayamamaktadır. Bu durum da yine heterojen bir sulamaya yol açmakta ve su kullanımındaki etkinliği azaltmaktadır. Sulama pratiğinin yapılacağı tüm yüzey alanında eşit buharlaşma ve emilim olamayacağı için bitkiye eşit miktarda su vermek ise bitki açısından adil bir sulama olmadığını ortaya koymaktadır.

Bu noktada, sulama stratejilerinin bölgesel özelliklere uygun bir şekilde özelleştirilmesi ve su kaynaklarının verimli kullanılması amacıyla, görüntü segmentasyonu kullanılarak elde edilen veriler ve derin öğrenme yöntemleriyle eğitilen bir yapay zeka modelinin sulama sistemine entegre edilmesi önemli bir çözüm olarak öne çıkmaktadır. Derin öğrenme, büyük veri setlerinden karmaşık desenleri tanıma yeteneği sayesinde, bitkinin su ihtiyacını belirlemede ve sulama gereksinimlerini öngörmede etkili bir rol oynayabilir. Yine derin öğrenme heterojen bölge sınırları ile uyumlu su basınç bilgisini ve haritasını da üretebilir. Pratiğe dökülecek bu yaklaşımlar, hem heterojenliği engelleyebilir, hem de su tüketiminde azalmaya katkıda bulunabilir. Aynı zamanda, görüntü segmentasyonu teknikleri, sulama alanlarındaki gölge ve güneşli bölgeleri hassas bir şekilde ayırt ederek üretilen basınç haritası ile heterojen bölgeler ve fiskiye başlıklarının sulama alanını tam olarak örtüştürebilir. Böylece suyun adil ve optimal bir şekilde dağıtılmasını sağlayarak özelleştirilmiş sulama stratejilerinin geliştirilmesine katkıda bulunabilir.

İleri teknoloji tabanlı sulama sistemlerinin benimsenmesi, geniş alan yüzeysel sulamalarının gelecekteki etkinliği ve sürdürülebilirliği açısından önemli bir adımdır.

1.1 Tez Çalışmasının Amacı

Bu çalışmanın temel amacı, mevcut sulama sistemlerinin sınırlamalarını aşma veya gözardı ettiği problemlerine çözüm sunma potansiyelini araştırmaktır. Bu amaç doğrultusunda aşağıdaki alt hedefler belirlenmiştir:

- Araştırma, canlı kamera görüntüleri veya alanın fotoğraflanması gibi görüntü tabanlı veri toplama yöntemlerinin kullanımını değerlendirmeyi amaçlar. Bu sayede, sulama sistemlerinde bitki durumu, nem seviyeleri ve diğer önemli parametreler hakkında daha detaylı verilere ulaşmak mümkün olacaktır.
- Sulama konusunda uzmanlaşmış bir profesyonelin birikimini, tamamen otonom bir sulama sistemi aracılığıyla herkes tarafından ulaşılabilir bir imkana dönüşmesi amaçlanmaktadır.

Sonuç olarak görüntü tabanlı veri toplama yöntemleri, tarım alanlarında hassas ve çeşitlilik gösteren verilere erişmek için daha kolay ve etkili bir çözüm sunarak, kapsamlı bir etki analizi sağlayacağı düşünülmektedir. Bu aynı zamanda sensör tabanlı sistemlere nazaran uygulanabilirlik ve maliyet açısından daha kolay ve ekonomik bir çözüm sunma potansiyeline de sahip olduğu düşünülmektedir.

2. GENEL BİLGİLER

Bu tez çalışması, birçok bilimsel disiplinin çalışma alanındaki konuların bir arada entegrasyonunu gerektirdiğinden, bu bölümde ilgili konu başlıklarına kapsamlı bir şekilde değinilmiştir. Farklı disiplinler arasındaki bu etkileşim ve ilerlemenin aynı hedef doğrultusunda yönlenmesi, bilginin daha geniş bir perspektiften ele alınmasını sağlamakta ve ilerlemede ivmelenmeyi arttırmaktadır. Bu bölüm farklı disiplinlerdeki araştırmacıların ileride karşılaşacakları konu ve kavramlar hakkında daha derin ve sağlam bir anlayış geliştirmelerine yardımcı olmak ve ortak paydada buluşmayı amaçlamaktadır.

2.1 Yapay Zeka

İnsanın yeryüzünde varlığını göstermesiyle birlikte yaşamındaki zorluklarla baş etme ve karşılaştığı problemlere çözüm arayışı ve onları kolaylaştırma çabası, o zamandan günümüze kadar cazibesini koruyan bir arzusu olmuştur. Günlük yaşamını kolaylaştırmak adına ilk olarak alet ve araçlar geliştirmesiyle başlayan süreç, zaman içerisinde insanın fiziksel yükünü üstünden alacak ekipmanlara dönüşmüştür. Ancak bununla yetinilmemiş ve sonraki adımı kendi fiziksel işlevlerini taklit eden makinelerin geliştirilmesine bırakmıştır. Günümüzde ise, bu arzuyu bir adım daha ileri taşıyarak kendini zihnen ve bilişsel olarak taklit eden makineler olmuştur. Yapay zeka bu serüvenin geldiği son nokta olarak düşünülebilir. Büyük olasılıkla sonraki adım, duygusal zeka ve öz farkındalığı taklit eden makinelerin geliştirilmesi olacaktır. Bu alandaki gelişmeler, insan ile makine arasındaki sınırların bulanıklaşmasıyla sonuçlanmış ve insan ile makine arasındaki etkileşimin değişmesine yol açmıştır.

Yapay zekanın temelleri, sibernetiğin ilk adımlarını atan ve ilk robotu tasarlayıp çalıştırdığı kabul edilen Cezerî'ye (1136-1206) dayandırılabilir. (Aleksandr D. Knyž, 1999) Cezerî'nin robot çizimlerinin, ünlü sanatçı Leonardo da Vinci'ye ilham kaynağı olduğu düşünülmektedir. (Al-Jazari, 2015) Ancak, modern anlamda yapay zeka, ikinci dünya savaşında Alan Mathison Turing'in çalışmalarıyla önem kazanmıştır. Alan Turing, İkinci Dünya Savaşı sırasında Bletchley Park'ta geliştirdiği cihaza "Bombe" adını verdi. Bombe, İngiliz kriptologlar tarafından II. Dünya Savaşı sırasında Alman Enigma makinesiyle

şifrelenmiş gizli mesajların şifresini çözmeye yarayan elektro-mekanik bir cihazdı. (Welchman, 2018,2015,1997)

Alan Turing, 1950 yılında yayınladığı "Computing Machinery and Intelligence" makalesinde "Makineler düşünebilir mi?" sorusunu tartışarak ve bu sorudan yola çıkarak yapay zekanın düşünsel temellerini oluşturmuştur. Ancak tek başına Yapay Zekanın kurucu babasıdır diyemeyiz. Alan Turing'in yanı sıra John McCarthy, Marvin Minsky, Allen Newell ve Herbert A. Simon gibi isimleri de eklememiz gerekir. McCarthy, Minsky, Nathaniel Rochester ve Claude E. Shannon, 1956 yazında ünlü Dartmouth konferansı için yazdıkları bir öneride "yapay zeka" terimini icat ettiler. Bu konferans, yapay zekayı bir alan olarak başlattı. (Roberts, 2016). Daha sonrasında yapay zeka hızla yükselişe geçmiş ve çok sayıda program ve model geliştirilerek isimlendirilmişlerdir. 1974 ile 1980 yılları arasında eleştirel birçok olumsuz yayın yapılmıştır. Marvin Minsky ve Roger Schank, 1984'teki Amerikan Yapay Zeka Birliği toplantısında, iş dünyasını yapay zekaya olan ilginin 1980'lerde kontrolden çıktığı ve hayal kırıklığının kesinlikle takip edeceği konusunda uyardılar. Yapay zeka topluluğunda kötümserlikle başlayacak, bunu basında kötümserlikle takip edecek, ardından finansmanda ciddi bir kesinti ve ardından ciddi araştırmaların sona ermesiyle başlayacak, "nükleer kışa" benzer bir zincirleme reaksiyon tanımladılar. (Crevier, 1993) Üç yıl sonra milyar dolarlık yapay zeka endüstrisinin çökmeye başladığı bu dönem "Yapay Zeka Kışı" (Artificial Intelligence Winter) olarak adlandırılmaktadır.

İngiltere'nin Japonya'ya karşı avantaj sağlamak amacıyla yapay zekaya fon ayırmasıyla başlayan süreç, IBM firmasının ürettiği Deep Blue bilgisayar yazılımının dünya satranç şampiyonu Gary Kasparov'u yenmesinin büyük ses getirmesiyle yeniden ivmelendi. *“İşte buydu, Yapay zekanın şafağı! Alan Turing'in, Claude Shannon'ın, Norbert Weiner'in, tüm kurucu babaların hayalini kurduğu an! Ve... Bu yapay zeka değildi. Aptaldı. Olsa olsa çalar saatiniz kadar zekiydi. Sadece daha büyük ve daha pahalıydı, ama zeki değildi. Zeki olması da gerekmiyordu. Tek yapması gereken, insandan daha az hata yapmaktı.”* Kasparov, Deep Blue ile arasında geçen maç ile ilgili olarak böyle diyordu. Deep Blue, oyundaki durumları ve yapılabilecek muhtemel hamleleri bir ağaç olarak ifade edip, alfa-beta budaması adlı bir arama algoritması ile yapılabilecek en optimum hamleleri yapıyordu.

Günümüzde ise yapay zeka, hem tasarlanmaları, hem işleyiş mekanizmaları, hem de tanımlanmaları bakımından Kasparov'un Deep Blue hakkındaki tanımlamasından çok çok fazla yol alarak büyük bir ilerleme kaydetti. Öyle ki, insan yaşamının ve bilimsel disiplinlerin en ücra, en fethedilmez bilinen köşelerine kadar sızmış durumdadır. Bu etkileşim, her iki tarafın, yani makine ve insanın gelişimi için yeni kapılar aralamakta ve yeni ufuklar açmaktadır.



Şekil 2.1. Yapay zekanın alt alanları, organizasyon ve yapılanması.

Birçok kaynak, makine öğreniminin yapay zekanın bir alt alanı olmaya devam ettiği iddiasını sürdürüyor(Garbade, 2018). **Şekil 2.1.** yapay zeka kavramlarını açıklamaya yönelik görseli içermektedir.

2.2 Makine Öğrenmesi

Makine öğrenimi (ML), örneklerden ve gözlemlerden anlamlı ilişkileri ve desenleri otomatik olarak öğrenmeyi amaçlar (Bishop, 2006). Makine öğrenimi, belirli görevleri yerine getirebilmek için sıkı sıkıya programlanmaya gerek duymadan, daha esnek algoritmaların veriden öğrenebilmesi ve öğrendiklerini görünmeyen verilere de uygulayabilmesidir. Makine öğrenimi araştırmalarının odaklandığı konu ise bilgisayarlara karmaşık örüntüleri algılama ve veriye dayalı en iyi akılcı kararlar çıkarabilme becerisini kazandırmaktır. Çok sayıda olası yanıtın olduğu durumlarda, doğru yanıtlardan bazılarını geçerli olarak etiketlemek bir yaklaşımdır. Bu, daha sonra bilgisayarın doğru yanıtları bulmasında eğitim verisi olarak kullanılabilir.

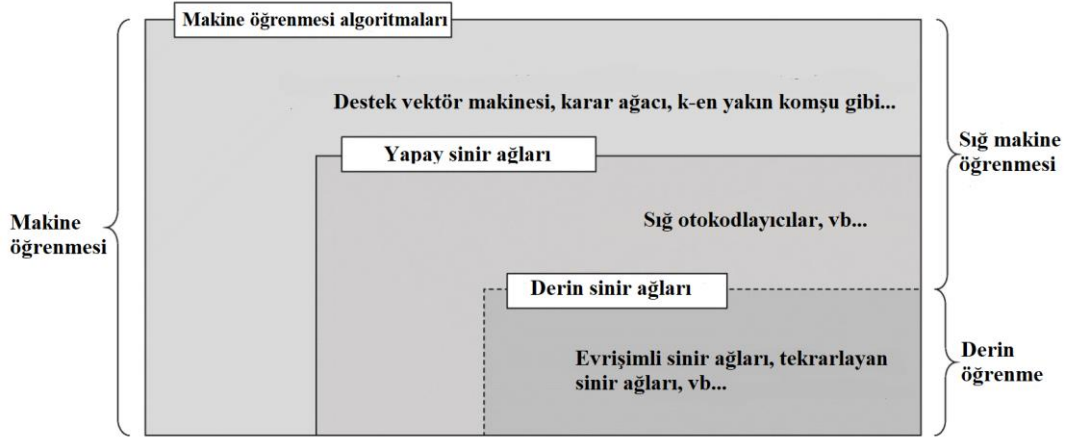
Makine öğrenimi terimi 1959'da bilgisayar oyunları ve yapay zeka alanında öncü ve IBM çalışanı olan Amerikalı Arthur Samuel tarafından icat edildi (Samuel, 1959). 1960'larda makine öğrenimi araştırmasının temsili bir kitabı, Nilsson'un Öğrenme Makineleri hakkındaki kitabıydı ve çoğunlukla örüntü sınıflandırması

iin makine ğrenimi ile ilgiliydi (Nilsson, 1965). Őekil 2.2. Makine ğrenimi kavramlarını ve venn diyagramını iermektedir.

Makine ğrenimi yaklařımları, doęal dil iřleme, bilgisayar grüşü, konuřma tanıma, e-posta filtreleme, tarım ve tıp dahil olmak üzere birok alana uygulanmıřtır (Hu, Niu, Carrasco, Lennox, & Arvin, 2020). Makine ğrenimindeki ilerlemeler, son zamanlarda iř ve kiřisel yařamımıza nřfuz eden ve elektronik pazarlardaki aę tabanlı etkileřimleri her řekilde řekillendiren insan benzeri biliřsel kapasiteye sahip akıllı sistemlerin yřkselmesini saęladı. Bu, řirketlerin verimlilik, katılım ve alıřanları elinde tutma konularındaki karar alma sřrelerini arttırmasıyla birlikte elektronik pazarlarda aę tabanlı etkileřimleri řekillendirdi (Shrestha, 2021).

Verilen probleme ve mevcut verilere dayanarak, üç třr makine ğreniminden sz edebiliriz: denetimli ğrenme, denetimsiz ğrenme ve destekli ğrenme. Elektronik pazarlardaki birok uygulama ise denetimli ğrenmeyi kullanır (Brynjolfsson, 2017).

- **Denetimli ğrenme:** Bu yaklařımda, bir modelin eęitimi iin girdi verileriyle birlikte hedef ıktılar saęlanır. Model, girdi-veri iftleri üzerinden eęitilir ve belirli bir girdi verildięinde doęru ıktıyı tahmin etmeyi ğrenir. Denetimli ğrenme, sınıflandırma ve regresyon gibi eřitli grevler iin kullanılabilir. Sınıflandırma problemlerinde, model, girdi verilerini belirli bir sınıfa atamak iin eęitilirken, regresyon problemlerinde model, sřrekli bir ıktı deęeri tahmin etmek iin eęitilir. Bu yntem, geniř bir uygulama yelpazesine sahiptir ve rneęin, tıbbi tanı, grüntü tanıma, finansal tahminler ve doęal dil iřleme gibi birok alanda kullanılır.
- **Denetimsiz ğrenme:** Bu yntemde, model, veri setindeki kalıpları ve yapıları tespit etmek iin eęitilir. Denetimsiz ğrenme algoritmaları, veri setindeki benzerlikleri, grupları, yoęunlukları veya dięer desenleri belirlemek iin kullanılır. Křmelenme, boyut indirgeme, ve zellik ıkarma gibi teknikler bu bařlık altında yer alır. Denetimsiz ğrenme genellikle veri setindeki gizli yapıları ve iliřkileri keřfetmek, veri keřfi yapmak ve veri setlerinden anlamlı bilgiler elde etmek amacıyla kullanılır.
- **Destekli ğrenme:** Bu yaklařımda, bir modelin eęitimi iin hem girdi verileri hem de doęru ıktılar saęlanır, ancak modelin doęru ıktıları nasıl tahmin edeceęi konusunda kesin talimatlar yerine, bir geribildirim mekanizması aracılıęıyla ğrenmesi saęlanır. Bu geribildirim mekanizması, modelin tahminlerinin gerek sonularla karřılařtırıldıęı ve bu karřılařtırmanın sonularına dayanarak modelin performansını iyileřtirmek iin ayarlamaların yapıldıęı bir sřretir. Destekli ğrenme genellikle bir karar sřrecini optimize etmek iin kullanılır ve bu sřrete belirli bir amaca ulařmak iin ardıřık kararlar alınır.



Şekil 2.2. Makine öğrenimi kavramı ve venn diyagramı (Goodfellow, 2016).

Derin sinir ağları genellikle derinlemesine iç içe geçmiş ağ mimarilerinde bulunan birden fazla gizli katmana sahiptir. Ayrıca, genellikle basit yapay sinir ağlarının aksine gelişmiş sinir hücreleri içerirler. Yani, bunlar basit bir aktivasyon fonksiyonu kullanmak yerine gelişmiş işlemler veya bir sinir hücresinde birden fazla aktivasyon fonksiyonunu kullanabilirler. Bu özellikler, derin sinir ağlarının ham giriş verileri ile beslenmesine ve karşılık gelen öğrenme görevi için gereken temsilin otomatik olarak keşfedilmesine olanak tanır. Literatürde henüz iki kavram arasında kesin bir ayrım olmadığı için Basit yapay sinir ağları ve diğer makine öğrenimi algoritmaları, bu tür işlevsellikleri sağlamadıkları için sığ makine öğrenimi terimi altında toplanabilirler (Schmidhuber, 2015).

Derin öğrenme (DL), geniş ve yüksek boyutlu veriye sahip alanlarda özellikle faydalıdır, bu yüzden metin, görüntü, video, konuşma ve ses verisinin işlenmesi gereken çoğu uygulamada, derin sinir ağları sığ makine öğrenimi algoritmalarını geride bırakır (LeCun, 2015). Ancak, düşük boyutlu veri girişi durumlarında, özellikle sınırlı eğitim verisi bulunabilirliğinde, sığ makine öğrenimi hala üstün sonuçlar üretebilir ve hatta derin sinir ağları tarafından üretilenden daha iyi yorumlanabilir sonuçlar ortaya çıkarabilir (Zhang, 2018).

2.2.1 Kümeleme Algoritmaları

Kümelenmeyi, "birbirine benzer üyelere sahip grupların kümeler halinde düzenlenme süreci" olarak tanımlayabiliriz. Bu nedenle, bir küme, aralarında benzerlik bulunan ve diğer kümelerdeki nesnelerle benzerlik taşımayan bir nesne topluluğudur. Kümelenmede benzerlik ölçüsünü iki küme arasındaki mesafe olarak değerlendirebiliriz. Bu özelliği bakımından kümelenme, denetimsiz makine öğrenme teknikleri arasında kullanılmaktadır. Kümeleme uygulanacak verilerde belirgin bir tanımlama bulunmasına ihtiyaç duyulmaz. Verilerin benzerlik ve farklılıklarına göre ayrılımları sağlanır. Sınıflandırmada ise etiketli verilerle işlem yapılır. Kümeleme ve sınıflandırma teknikleri arasındaki temel farklılık buradan kaynaklanmaktadır.

- **Affinity Propagation:** Affinity Propagation, veriler arasında "mesaj" alışverişi yaparak veri kümesinde bulunan örnekler arasındaki benzerlikleri bulup bu benzerlikleri kümelemede kullanmaya çalışır. Diğer kümeleme algoritmaları kümeleme sayısının önceden belirlenmesini gerektirirken, Affinity Propagation küme merkezlerini keşfeder ve verileri bu küme merkezlerine özerk olarak atar. Bu algoritma, küme sayısının önceden bilinmediği veya verilerin küresel veya dışbükey küme şekillerine uymadığı durumlarda özellikle kullanışlıdır. Algoritmanın amacı, genel verileri en iyi temsil eden örnekleri belirlemek ve bunları, verileri uyumlu gruplar halinde kümelemek için kullanmaktır.

Algoritma çalışırken, verileri temsil etmek için benzerlik matrisi, sorumluluk matrisi ve kullanılabilirlik matrisi olmak üzere üç adet matris hesaplar. Algoritmadaki mesaj aktarım işlemi, sorumluluk mesajları ve kullanılabilirlik mesajlarının aktarılmasıdır. Her veri noktası potansiyel bir örnek olarak değerlendirilir. Gerçek değerli mesajlar, yüksek kaliteli bir örnek kümesi ve karşılık gelen kümeler yavaş yavaş ortaya çıkana kadar veri noktaları arasında değiş tokuş edilir.

- **Birch:** (Balanced Iterative Reducing and Clustering using Hierarchies), büyük ölçekli veri kümelerinin kümeleneğinde işlevselliği yüksek etkili bir algoritmadır. Küme merkezlerini çevreleyen veri noktalarının temsil edilmesinde ağaç yapısını kullanır. Bu sayede veri noktalarını alt kümelere ayırmış olur ve her alt küme için bir dizi özet istatistik çıkartır. Böylece, veri kümesinin genel yapısını özetleyen bir hiyerarşi oluşturulmuş olur. Algoritmanın ana adımları ağaç oluşturma, kümeleme ve küme atama şeklindedir.
- **Dbscan:** Yoğunluk tabanlı mekânsal bir kümeleme algoritmasıdır. Veri noktalarının yoğunluklarını temel alarak veri setlerinden küme oluşturur. (Ester, 1996, August) veri setinin yapısı hakkında önceden bilgi sahibi olunmadığı durumlarda kullanışlıdır.

DBSCAN, Epsilon (ϵ) ve MinPts olmak üzere iki önemli parametreye dayanır. Epsilon, komşuluk yarıçapını belirler. Bu değer, bir veri noktasının etrafındaki diğer noktaların ne kadar yakınında olması gerektiğini ifade eder. Algoritma, veri setinden rastgele bir nokta seçerek başlar. Seçilen noktanın Epsilon içindeki komşuları bulunur ve seçilen noktanın komşuluk içindeki nokta sayısı MinPts'ten büyük veya eşitse, bu nokta "yoğun" olarak işaretlenir. Yoğun olarak işaretlenmiş noktalar ve bunların komşuları ile bir küme oluşturulur. Diğer noktalar işlenene kadar işlemler tekrarlanır. İşlenmemiş noktalar, gürültü olarak kabul edilir veya var olan kümelerin bir parçası olabilir sonucuna varılır.

- **K-Means:** Bir veri kümesini belirli sayıda K kümesine (veya kümeye) ayırmak için kullanılan bir kümeleme algoritmasıdır. Her bir küme, içindeki noktaların merkezini temsil eden bir "centroid" ile karakterize edilir (MacQueen, 1967, June). K-Means algoritması, küme sayısının önceden belirlenmesi gerektiği bir (K) parametresine sahiptir. Ancak, bu algoritma hızlı ve kolayca uygulanabilir olması nedeniyle sıkça kullanılır.

İlk olarak, K tane başlangıç merkez noktası belirlenir. Bu merkez noktaları genellikle rastgele seçilir. Her veri noktası, mevcut merkezlere olan mesafesine göre en yakın merkeze atanır. Bu, her noktanın bir kümeye atandığı ve bu kümeyi temsil eden bir merkez bulunduğu anlamına gelir. Sonrasında her kümenin veri noktalarının ortalamasının alınmasıyla her küme için yeni bir merkez hesaplanır. Bu, Yeni hesaplanan merkezler, eski merkezlerin yerini alır. Küme merkezlerinin ve kümelerin değişmediği duruma kadar bu işlem tekrarlanır. Merkezlerin ve kümelerin değişmezliği durumunda, algoritma sona erer ve kümeleme sonuçlanmış olur.

- **Mini-Batch K-Means:** Geleneksel K-Means algoritmasının bir çeşitidir ve büyük ölçekli veri kümeleri üzerinde daha hızlı çalışmak için tasarlanmıştır. Bu algoritma, veri kümesini kümelemek için yalnızca küçük bir örnek bölüm (mini-batch) kullanır, bu da hesaplama süresini azaltırken benzer kümeleme sonuçlarını elde etmeyi mümkün kılmış olur. K-Means algoritmasına göre daha hızlı çalışır, çünkü her iterasyonda tüm veri seti yerine sadece bir mini-batch kullanılır. Bu, özellikle yüksek boyutlu veri kümeleri için hesaplama maliyetinin azalmasına neden olur.
- **Mean Shift:** Bir diğer yoğunluk tabanlı kümeleme algoritmasıdır. Veri noktalarını yoğunluğu en yüksek olan bölgelere iteratif olarak kaydırarak kümeleme yapar (Comaniciu, 2002). Mean Shift algoritması, özellikle görüntü işleme, nesne takibi, desen tanıma gibi alanlarda ve video izleme gibi uygulamalarda hareket izleme ve analizinde kullanılır. Ayrıca, görüntü segmentasyonunda benzer renk veya özelliklere sahip pikselleri gruplamak için kullanılan bir algoritmadır.

İlk olarak, her bir veri noktası başlangıçta bir küme merkezi olarak kabul edilir. Her bir veri noktası, çevresindeki diğer noktaların ağırlıklı ortalamasını alarak yeni bir konuma kaydırılır. Bu kaydırma işlemi, yoğunluğun en yüksek olduğu yöne doğru gerçekleşir. Küme merkezlerinin kaydırma hareketi durduğunda veya bir yakınsama kriteri ile karşılaşıldığında algoritma sonlanır. Her bir veri noktası, son iterasyonda bulunduğu küme merkezi ile ilişkilendirilir. Aynı küme merkezi ile ilişkilendirilen noktalar aynı küme olarak kabul edilir.

- **Optics:** OPTICS (Ordering Points To Identify the Clustering Structure), DBSCAN kümeleme algoritmasından ilham alınmış bir diğer kümeleme algoritmasıdır. Farklı yoğunluk ve şekillerde kümeler oluşturabilir. OPTICS, DBSCAN kümeleme kavramına Çekirdek Mesafesi ve Ulaşılabilirlik Mesafesi olmak üzere iki terim daha ekler. Algoritma, ulaşılabilirlik grafiği adı verilen sıralı bir nokta listesi oluşturarak verilerin yoğunluk temelli bir temsilini oluşturur. Listedeki her nokta, veri kümesindeki diğer noktalardan o noktaya ulaşmanın ne kadar yakın olduğunun ölçüsü olan ulaşılabilirlik mesafesiyle ilişkilidir. Benzer erişilebilirlik mesafelerine sahip noktaların aynı kümede olması muhtemeldir. Bu kümeleme tekniği, verileri açıkça kümelere ayırmaması açısından diğer kümeleme tekniklerinden farklıdır. Bunun yerine, erişilebilirlik mesafelerinin bir görselleştirmesini üretir ve bu görselleştirmeyi verileri kümelemek için kullanır.

Kümelerin minimum yoğunluğunu kontrol eden bir yoğunluk eşik parametresi olan Eps tanımlanır. Veri kümesindeki her nokta için k-en yakın komşularına olan mesafe hesaplanır. Rastgele bir noktadan başlayarak, komşularının yoğunluğuna bağlı olarak veri kümesindeki her noktanın ulaşılabilirlik mesafesi hesaplanır. Noktalar erişilebilirlik mesafelerine göre sıralanır ve ulaşılabilirlik grafiği oluşturulur. Birbirine yakın ve benzer erişilebilirlik mesafelerine sahip noktalar gruplandırılarak erişilebilirlik grafiğinden kümeler elde edilir.

- **Spectral Clustering:** Spectral Clustering algoritması, veri noktalarını doğrudan konumlarına göre değil, veri noktaları arasındaki ilişkileri temsil eden bir kenar grafik yapısına dönüştürüldüğü verilerin grafik temsili fikrine dayanmaktadır.

Verilerin Benzerlik Grafiğinin Oluşturulması: Bu adım, Benzerlik Grafiğini A ile temsil edilen bir bitişiklik matrisi biçiminde oluşturur.

Epsilon-Komşu Grafiği: Epsilon parametresi önceden sabitlenmiştir. Daha sonra her nokta kendi epsilon yarıçapındaki tüm noktalara bağlanır. Herhangi iki nokta arasındaki tüm mesafeler ölçek açısından benzerse, tipik olarak kenarların ağırlıkları, yani iki nokta arasındaki mesafe, herhangi bir ek bilgi sağlamadıkları için saklanmaz. Dolayısıyla bu durumda oluşturulan grafik yönsüz ve ağırlıklandırılmamış bir grafikdir.

K-En Yakın Komşular: Bir k parametresi önceden sabitlenmiştir. Daha sonra, u ve v gibi iki köşe için, yalnızca v, u'nun k-en yakın komşuları arasındaysa, bir kenar u'dan v'ye yönlendirilir. Bunun ağırlıklı ve yönlendirilmiş bir grafiğin oluşmasına yol açar çünkü k-en yakın komşularından biri olarak v'ye sahip olan her u için her zaman v'nin k-en

yakın komşuları arasında u'ya sahip olması durumu aynı olacaktır. Bu grafiği yönsüz hale getirmek için aşağıdaki yaklaşımlardan biri izlenir:

Eğer v , u 'nın k -en yakın komşuları arasındaysa VEYA u , v 'nin k -en yakın komşuları arasındaysa, u 'dan v 'ye ve v 'den u 'ya bir kenar yönlendirilir.

v , u 'nın k -en yakın komşuları arasındaysa VE u , v 'nin k -en yakın komşuları arasındaysa, u 'dan v 'ye ve v 'den u 'ya bir kenarı yönlendirilir.

Tamamen Bağlantılı Grafik: Bu grafiği oluşturmak için her nokta, iki nokta arasındaki mesafenin diğer noktalara olan uzaklığı kadar yönlendirilmemiş bir kenarla bağlanır. Bu yaklaşım yerel komşuluk ilişkilerini modellemek için kullanıldığından, mesafeyi hesaplamak için tipik olarak Gauss benzerlik metriği kullanılır.

Verilerin daha düşük Boyutlu Uzaya yansıtılması: Bu adım, aynı kümenin üyelerinin belirli boyutlu uzayda çok uzakta olabileceği olasılığını hesaba katmak için yapılır. Daha sonra boyutsal alan azaltılır, böylece bu noktalar azaltılmış boyutsal alanda daha yakın olur ve geleneksel bir kümeleme algoritması ile birlikte kümelenebilir. Grafik Laplace Matrisinin hesaplanmasıyla yapılır.

- **Gaussian Mixture Model:** Farklı bileşenlerin Gauss dağılımı altında bir araya geldiğini varsayan bir parametrik olasılık yoğunluk fonksiyonudur. Bu model, her bir bileşenin ağırlığı ile ayrı ayrı temsil edilir. Özellikle model tabanlı kümeleme için bir araç olarak popülerlik kazanmıştır. Çeşitli alanlarda, örneğin patern tanıma, görüntü analizi, biyometrik verilerin analizi gibi birçok alanda yaygın bir şekilde kullanılır. Gaussian karışım modelleri, karmaşık veri yapılarını modelleme yetenekleri nedeniyle tercih edilir ve farklı bileşenlerin esnek bir kombinasyonunu sağlar.

Başlangıç Merkezlerinin Belirlenmesi: İlk olarak, her bir Gauss dağılımının merkezi (ortalama) ve yayılımı (varyans) başlangıç değerleri belirlenir. Bu başlangıç değerleri genellikle rastgele seçilir veya başka bir başlangıç tahmini yöntemi kullanılır.

E-step (Expectation Step): Bu adımda, her veri noktasının her bir Gauss dağılımı ile ilişkili olasılığını hesaplamak için beklenen değerler hesaplanır. Yani, her bir veri noktasının her bir Gauss dağılımı altında olma olasılığı belirlenir.

M-step (Maximization Step): E-step adımı hesaplanan olasılıkları kullanarak, Gauss dağılımlarının parametreleri (ortalama ve varyans)

güncellenir. Bu güncelleme, olasılık yoğunluğunun maksimize edilmesini sağlar.

İterasyonlar ve Yakınsama: E-step ve M-step adımları tekrarlanır ve GMM'nin parametreleri iteratif olarak güncellenir. Bu adımlar, genellikle bir yakınsama kriteri sağlanana kadar tekrarlanır. Örneğin, parametrelerin değişimi belirli bir eşiğin altına düştüğünde veya belirli bir iterasyon sayısına ulaşıldığında durabilir.

Sonuçların Elde Edilmesi: İterasyonlar tamamlandıktan sonra, Gauss dağılımlarının parametreleri ve her bir veri noktasının hangi bileşene ait olduğu sonuçları elde edilir.

- **K-Medians:** K-medians, k-mean'e alternatif olarak kullanılan bir kümeleme algoritmasıdır. Bu yöntem, küme merkezlerini belirlemek için k-means'den farklı olarak medyanı kullanır. K-medians'in tercih edilme nedeni, medyanın uç değerlere karşı sağlam olmasıdır. Ortalama, bir veya daha fazla radikal uç değer varlığında veri setinin çoğunluğundan uzaklaşabilir, ancak medyanın bu etkiye daha az duyarlı olduğu bilinmektedir. K-medians, aykırı değerlerin etkilerini azaltmak için çeşitli varyasyonlarla birlikte kullanılabilir.

Küme Merkezlerinin Başlangıç Değerlerinin Belirlenmesi: İlk adımda, kümeleme işlemi için başlangıç merkezlerinin belirlenmesi gerekir. Genellikle bu merkezler rastgele seçilir veya veri noktaları arasından rastgele örnekler alınır.

Noktaların Kümeye Atanması: Her veri noktası, bu noktaya en yakın olan küme merkezine atanır. Bu işlem genellikle Öklid mesafesi gibi bir uzaklık ölçüsü kullanılarak gerçekleştirilir.

Küme Merkezlerinin Güncellenmesi: Her bir kümenin medyanı, o kümenin yeni merkezi olarak seçilir.

İterasyon: Noktaların kümeye atanması ve küme merkezlerinin güncellenmesi adımları iteratif olarak tekrarlanır.

Sonuçların Elde Edilmesi: İterasyonlar tamamlandıktan sonra, her bir veri noktasının ait olduğu kümeyi belirlemek için son küme merkezleri kullanılır.

- **Fuzzy C- Means:** (FCM) algoritması, veri kümesindeki her bir noktanın bir veya birden fazla kümeye olan ilişkisinin, o kümelerin merkezlerine olan uzaklığa dayalı olarak belirlendiği bir bulanık kümeleme yöntemidir. Bu

algoritma, veri noktalarının kesin bir şekilde bir kümeye atanması yerine, her bir noktanın her küme ile olan ilişkisinin belirsizliğini hesaba katar. Ruspini (Ruspini, 1970) tarafından tanıtilan bulanık C-bölüm kavramına dayanmaktadır ve sonrasında geliştirilmiştir. Algoritma, bir özellik vektörleri kümesini küme merkezlerine yaklaştırmak için veri kümesinin küçük bir alt kümesinin kullanılabileceği varsayımına dayanmaktadır. Bu varsayım altında, FCM, orijinal veri kümesine ait uygun boyutlu bir alt kümesinden küme merkezlerini hesaplamak için kullanılır.

İlk adımda, algoritma orijinal veri kümesinden rastgele C başlangıç küme merkezi seçer. Sonraki adımlarda, algoritma birkaç iterasyon sonrasında sonuçları gerçek küme merkezlerine yakınsar. Bu nedenle, iyi bir başlangıç küme merkezi seti seçmek, algoritmanın başarısı için önemlidir.

- **Fuzzy K- Modes:** K-Modes algoritmasının genişletilmiş bir versiyonu olarak Ruspini (Ruspini E. , 1969) ve Bezdek (Bezdek, 1980) tarafından duyuruldu. Fuzzy K-Modes, veri noktalarının bir veya birden fazla küme ile ilişkisinin belirsizliğini hesaba katar. Bu yöntem, her bir veri noktasının her bir küme ile olan ilişkisini bir olasılık dağılımı olarak ifade eder, böylelikle veri noktalarının direk olarak bir kümeye atanması yerine her bir kümeye olan ilişkinin belirsizliğini yansıtır. Kategorik verilerin net bir şekilde kümeleme yapılamadığı durumlarda, belirsizlikle başa çıkmak önemlidir.

Başlangıç: İlk adımda, algoritma k küme merkezi oluşturmak için veri setinden rastgele k nokta seçer.

Atama Adımı (Assignment Step): Her veri noktası, k küme merkezi arasındaki mesafeleri hesaplar ve her bir kümeye atanma olasılığını belirler. Bu atama, bulanık bir şekilde yapılır, yani her veri noktası her bir kümeye olan aidiyeti için bir olasılık dağılımı elde eder.

Küme Merkezi Güncelleme (Centroid Update Step): Her küme merkezi, o kümeye atanan veri noktalarının ağırlıklı ortalaması olarak güncellenir. Bu, belirli bir kümenin merkezinin, o kümeye ait veri noktalarının ortalama özellik değerlerine en iyi şekilde uymasını sağlar.

Yeniden Atama ve Güncelleme (Re-assignment and Update): Atama adımı ve küme merkezi güncelleme adımı tekrarlanır. Bu adımlar, belirlenmiş bir durma kriteri sağlanana veya belirlenen bir maksimum iterasyon sayısına ulaşılanaya kadar devam eder.

Sonuç: Algoritma sona erdiğinde, her veri noktası için her bir küme ile ilişkili olasılıklar belirlenir. Bu olasılıklar, veri noktalarının her bir küme ile ilişkisinin belirsizliğini ifade eder.

- **Fuzzy Clustering:** Ana fikir, veri noktalarını kümelere atarken her bir noktanın her bir kümeye olan aidiyetini belirsizlikle ifade etmek olan ve bazılarına önceki sayfalarda geniş yer verilen bulanık mantık temelli kümeleme algoritmalarını içerir.
- **Expectations Maximization:** özellikle veri setlerinde gizli (görünmeyen veya eksik) değişkenler olduğunda ve bu değişkenlerin değerinin belirsiz olduğu durumlarla başa çıkmak için tercih edilen bir istatistiksel algoritmadır. Bu algoritma, parametrik bir olasılık dağılımının maksimum olabilirlik tahminlerini hesaplamak için kullanılır, ancak verilerde eksiklik olduğunda da uygulanabilir. EM algoritması, Dempster'a (A.P. Dempster, 1977) kadar bağımsız birçok araştırmacı tarafından keşfedilip kullanılmıştır. Özellikle gözlemlenmemiş veriler ve boşlukların olduğu genetik araştırmalarda kullanıldığı gibi ayrıca, sonuçları bilinmeyen faktörlerin etkilediği ekonomik, klinik ve sosyolojik verilerin olduğu alanlarda da tercih edilmiştir. Sinyal işleme alanında ise büyük bir ilgi görmüştür. Konuşma tanıma alanında ise gizli Markov modellerinin eğitiminde kullanılmıştır (Rabiner, 1989). Yine görüntü işlemede yeniden yapılandırma modellerinde de kullanılır.

EM algoritması, iki temel adımdan oluşur:

Beklenti (Expectation): Eksik veri değerlerini doldurarak, eksik verilerin dağılımının parametrelerine dair tahminler yapılır. Ancak, bu tahminler kesin değil, sadece olasılıklı tahminlerdir.

Maksimizasyon (Maximization): Parametrelerin mevcut tahminlerine dayanarak yeni değerleri tahmin edilir. Bu adım, verinin tümünde olabilirlik fonksiyonunu maksimize eden parametreleri bulmaya çalışır.

- **Hierarhcial Clustering:** Amaç, benzer nokta gruplarını ardışık olarak birleştiren verilerin ikili bir ağacını oluşturmaktır. Bu işlem, tüm X öğelerini içeren ağacın köküne ulaşana kadar devam eder. X 'in herhangi iki alt kümesi arasındaki mesafe bağlantı mesafesi olarak adlandırılan $\Delta(X_i, X_j)$ ile gösterilir. Birleştirici (Agglomerative) Hiyerarşik Kümelenme ve Bölücü (Divisive) Hiyerarşik Kümelenme olmak üzere iki ana türü vardır. Birleştirici kümelenmede her veri noktası bir grup olarak başlar ve benzerlik ölçüsüne dayalı olarak en yakın iki grup birleştirilir. Bu birleştirme işlemi adım adım devam eder ve sonunda tüm veri noktaları tek bir grup

oluşturacak şekilde birleştirilir. Bölücü kümelenmede ise tüm veri noktaları tek bir grupta başlar ve en farklı gruplar belirlenene kadar gruplar bölünür.

Hiyerarşik kümelenme yöntemi, genellikle veri setinde doğal bir hiyerarşi olduğu durumlarda ve grupların birbirine göre ilişkilerinin önemli olduğunda kullanılır. Özellikle görselleştirme amacıyla dendrogramlar şeklinde gösterilen hiyerarşik yapılar, veri setinin yapısının daha iyi anlaşılmasını sağlar. Ayrıca, grup sayısının önceden belirtilmek zorunda olunmaması esnek bir yöntem olmasını sağlar.

- **Minimum Spanning Tree:** Bir ağırlıklı bağlı grafiğin tüm düğümlerini içeren en az toplam ağırlığa sahip alt ağacını bulmayı amaçlar. Bu ağaç, grafiğin tüm düğümlerini içerir ve tüm düğümleri bağlar, ancak döngü içermez. Pek çok MST çeşidi olmasına rağmen, en yaygın kullanılanları Prim algoritması ve Kruskal algoritmasıdır. İlk algoritma, bir düğüm kümesi ile başlar ve bu küme her adımda yeni bir düğüm eklenene kadar büyür. Her adımda, mevcut düğüm kümesinden başlayarak en az maliyetli bir kenar seçilir ve ağaca eklenir. Bu işlem, ağaç tüm düğümleri içerecek şekilde tamamlanana kadar devam eder. İkinci algoritma ise kenarları ağırlıklarına göre sıralar ve her adımda ağaçta döngü oluşturmayacak şekilde en küçük ağırlıklı kenarı seçer. Eğer seçilen kenar ağaçta döngü oluşturacaksa, o kenar ihmal edilir.

MST kümeleme algoritması pratikte geniş bir şekilde kullanılır. Ancak, çok renkli karmaşık arka planlara sahip olan ve aynı zamanda düşük mekansal çözünürlüğü ve görüntülerdeki renklerin büyük sayısı nedeniyle web görüntülerinin renk kümelemesinin oluşturulmasında tercih edilen uygulamalardan birisidir. Bu uygulamadaki amaç, binlerce rengi bir görüntüdeki ilgi çekici nesneleri net bir şekilde ayırt edebilecek birkaç temsilciye indirgemektir.

2.2.2 Sınıflandırma algoritmaları

Sınıflandırma, genel olarak belirli özelliklere sahip verileri gruplara ayırmak veya kategorize etmek için kullanılan bir tanımlamadır. Bu gruplama, veri setindeki benzerliklere dayanabilir. Sınıflama, her bir veri için sistemli ve düzenli bir atama işlemi gerektirir. Atama işleminin en doğru ve hızlı şekilde yapılmasını amaçlayan birçok algoritma zaman içinde geliştirilmiş ve birbirlerine karşı eksiklikleri giderilmeye çalışılmıştır.

Temel olarak, sınıflandırma, belirli bir girişin önceden tanımlanmış aranan özellikleri ile ilişkili etiketlenmiş belirli bir kategoriye veya sınıfa atanmasıdır. Öncelikle oluşturulacak sınıfın özellikleri belirlenir, sonrasında ise içine o grubun niteliklerini taşıyan veriler atanır. Atama işlemi grubun niteliklerine ve grubun oluşturulma amacıyla yakından ilişkilidir. Bu nedenle sınıflandırma yapılırken

amaçın ne olduğu büyük önem taşır. Sınıflandırmadaki amaç, aranan ve ihtiyaç duyulan verilerin yerlerini kolayca saptamak, verilere kısa sürede zaman kaybetmeden ulaşmak, veriler arasında düzen ve ilişki kurmak şeklinde özetlenebilir.

- **K-Nearest Neighbors:** En temel ve basit sınıflandırma yöntemlerinden biridir ve verilerin dağılımı hakkında önceden bilginin çok az olduğu veya hiç olmadığı durumlarda ilk tercihlerden biri olmuştur. Bu algoritma, yeni bir veri noktasını sınıflandırmak için etrafındaki en yakın k komşularının etiketlerini kullanır. Veri noktası, bu komşuların çoğunluğunun sınıf etiketiyle etiketlenir. Örneğin, bir veri noktası etrafındaki en yakın üç komşuya bakıldığında, iki tanesi "A" sınıfına ve bir tanesi "B" sınıfına aitse, bu veri noktası "A" sınıfına ait olarak sınıflandırılır. K-en yakın komşu sınıflandırması, olasılık yoğunluklarının güvenilir tahminlerinin bilinmediği veya belirlenmesinin zor olduğu durumlarda diskriminant analizi gerçekleştirme ihtiyacından geliştirilmiştir. 1951 de ABD Hava Kuvvetleri Havacılık Tıbbi Okulu'nun yayınlanmamış bir raporunda, Fix ve Hodges, o zamandan beri k-en yakın komşu kuralı olarak bilinen, model sınıflandırması için parametrik olmayan bir yöntem tanıttı (Fix, 1951). Daha sonra 1967'de k-en yakın komşu kuralının bazı biçimsel özellikleri üzerinde çalışıldı.

KNN algoritması çalışırken, sınıflandırma için kullanılan k değeri önceden belirlenir. Bu k değeri, yakın komşuların sayısını karşılık gelmektedir. Küçük k değerleri, modelin daha esnek olmasını sağlar ancak aşırı uyum riskini artırabilir. Büyük k değerleri ise daha net sınıflandırma kararları üretir ancak detayları göz ardı edebilir.

- **Random Forest:** Çoklu karar ağaçlarının çıktılarını bir araya getirerek sınıflandırma işlemini icra eden bir algoritmadır. Her bir karar ağacı, veri setindeki rastgele örneklerle eğitilir. Sonrasında her bir ağaç bir tahminde bulunur ve bu tahminlerin çoğunluğu alınarak ağaçlar bir araya getirilmiş ve böylelikle nihai sınıflandırma sonucu belirlenmiş olur. Bir RF'yi eğitmek için, ormanda bulunan ağaç sayısı ve her ağaç düğümünde değerlendirilmek üzere rastgele seçilen özellik/değişken sayısı gibi iki parametre sağlanmalıdır, ayrıca bir eğitim veritabanı kullanılarak gerçek sınıf etiketlerine sahip olunmalıdır. RF, popüler ve güçlü bir topluluk denetimli sınıflandırma yöntemidir (Breiman, 2001). Bu nedenle biyoinformatik ve tıbbi görüntüleme de dahil olmak üzere çeşitli makine öğrenimi uygulamalarına etkili bir şekilde uygulanmıştır.

Temel olarak, RF rastgele örnekleme yoluyla eğitim verilerinin birden fazla alt kümesini değiştirmek suretiyle oluşturma ve bu alt kümeleri bir karar ağaçlarını eğitmede kullanma süreci(Bootstrap Aggregating) ve topluluk

içindeki her bir karar ağacı için özellik kümesinden bir alt kümesinin rastgele seçilmesini ifade eden(Random Subspace) iki kavramdan oluşur. RF algoritması hem sınıflandırma da hemde regresyon da kullanılır kullanım farkları ise nihai tahmin alma şekline göre değişir. Sınıflamada her karar ağacının oylaması sonucu belirlerken regresyonda her karar ağacı bir olasılık tahmininde bulunur ve olasılıkların ortalaması alınarak regresyon sonucu belirlenir.

- **Logistic Regression:** Lojistik regresyon bir ikili sınıflandırma yöntemidir. Bir veya birden fazla bağımsız değişkenin değerlerini kullanarak bir bağımlı değişkenin olasılığını tahmin ederek bu değişkenin ait olduğu sınıfı tahmin etmek için kullanılır. Bağımlı değişkenin dağılımı logit dönüşümü kullanılarak tanımlanır. Logit dönüşümünde bağımlı değişkenin logaritması alınır. Daha sonra, bu dönüştürülmüş değer, bir sigmoid fonksiyonu ile işlenir ve sonuç olasılık değerine dönüştürülür. Sınıflandırma işlemi elde edilen değerlerin 0.5 eşik değeri ile karşılaştırılması ile yapılır. Eşik değerinden büyük olan olasılıklar bir kategoriye, eşik değerinden küçük olan olasılıklar ise diğer bir kategoriye atanır. Logistic regresyon modelinin kullanımı 1845'e kadar uzanmaktadır. İlk olarak o dönemde nüfus artışı üzerine yapılan matematiksel çalışmalarda ortaya çıktı (Gürçan, 1998). İkili sınıflandırma problemlerinde yaygın olarak kullanılmaktadır.
- **Naive Bayes:** Sınıflandırması, "naive" yani "saf" olarak adlandırılır, çünkü özellikler arasında bağımsızlık olduğu varsayımını yapar. Yani, her özellik atanacak sınıfın belirlenmesinde birbirinden bağımsızdır ve birbirleri üzerinde etkileşimleri yoktur. Bu varsayım sayesinde, model basitleştirilmiş ve hesaplama süreci hızlanmış olur. Algoritma, Bayes Teoremi'ni kullanarak her sınıf için tanımlanan özellikler başlığı altındaki bir sınıfa ait olma olasılığını hesaplar. Bu olasılık sonucu kullanılarak en yüksek olasılığa sahip sınıfa örneğin atama işlemi gerçekleşir. Uygulamada, özellik değerlerinin çoğu eğitim verilerinde bulunmayabilir veya yeterli sayıda olmayabilir. Bu durumda çok değişkenli olasılığın doğrudan tahmini güvenilir olmayabilir. Naïve Bayes, bu durumu koşullu bağımsızlık varsayımıyla telafi eder. Bu sıkı bağımsızlık varsayımına rağmen, Naïve Bayes birçok gerçek dünya uygulamasında oldukça yetenekli bir sınıflandırıcıdır. Naive Bayes sınıflandırması, genellikle metin sınıflandırması gibi doğal dil işleme uygulamalarında aynı zamanda spam filtreleme gibi alanlarda kullanılır. Algoritma, sınıflandırma işlemi için önceden etiketlenmiş eğitim veri setinden faydalanır. Yeni örnekleri eğitim verileri üzerinde test ederek sınıflandırma işlemini gerçekleştirir.
- **Decision Tree:** Belirli bir problem için tüm olası durumları haritalayan ve karar ağaçları oluşturarak her bir durumun karşılık gelen kararlarını analiz eden bir akış şemasıdır. Amacı, en uygun sınıfa atama işlemini

gerçekleştirmektedir. Karar ağacında, iç düğüm bir özelliğe, her dal bir karar kuralına ve her yaprak düğüm ise bir sonuca karşılık gelir. En üst düğüm kök düğüm olarak adlandırılır ve hiç alt düğümü olmayan düğümler yaprak düğüm olarak bilinir. Algoritma kök düğümünden başlar ve yaprak düğüme ulaşana kadar özellik değerlerine göre kararlar alarak ağaç boyunca ilerler. Yaprak düğümdeki değer, çıktı değerini veya atanan sınıfı temsil eder. Bir karar ağacı, böl ve yönet stratejisi kullanılarak bir örnek kümesinden oluşturulabilir. Eğer tüm örnekler aynı sınıfa aitse, ağaç o sınıfı etiket olarak içeren bir yaprak olur. Aksi takdirde, farklı sonuçları olan en az iki örneğe sahip bir test seçilir ve bu sonuca göre bölünen örnekler seçilir.

- **Support Vector Machines:** SVM'ler istatistiksel öğrenme teorisine dayanır ve sınıfların optimal bir şekilde ayrılmasını sağlayan karar sınırlarının konumunu belirlemeyi amaçlarlar. Başlangıçta ikili (iki sınıf) problemler için tasarlanmıştır. Birden fazla sınıfı ele alırken uygun bir çoklu sınıf yöntemi gereklidir (VAPNIK, 1995). Destek vektör makinesinin iyi tanıtılmış olması ve daha yüksek bir doğruluk sağlaması nedeniyle, makine öğrenimi topluluğunun araştırma odağı haline gelmiştir. Bu nedenle geniş bir uygulama alanına sahiptir ve özellikle biyomedikal mühendislik, biyoinformatik, görüntü işleme, finans ve diğer birçok alanda başarılı sonuçlar elde edilmiştir.

SVM'nin temel prensibi, veri noktalarını bir hiperdüzlemle sınıflandırmaktır. İkili sınıflandırma problemleri için kullanıldığında, SVM iki sınıf arasında bir ayırıcı hiperdüzlem oluşturur. Bu hiperdüzlem, veri noktalarını iki sınıfa ayıran en geniş marjı elde etmeyi hedefler. Marj, hiperdüzlemle en yakın veri noktaları arasındaki mesafe olarak tanımlanır. Amaç bu marjı maksimize etmektir. Marjın maksimize edilmesi, destek vektörleri olarak adlandırılan veri noktalarının belirlenmesine dayanır. Destek noktaları, eğitim veri setinde bulunan ve sınıflandırma işlemi için en uygun konuma sahip olan veri noktalarıdır. Bu noktalar, sınıflar arasındaki ayrımı en iyi şekilde temsil ederler ve sınıflandırma sürecindeki karar sınırlarının belirlenmesinde kullanılırlar.

2.2.3 Regresyon Algoritmaları

Regresyon istatistiksel alanda ve veri analizinde kullanılan bir analiz tekniğidir. Bağımsız değişkenlerin bir veya daha fazla bağımlı değişken üzerindeki etkisini tahmin etmek için hedefler. Bağımlı değişken, tahmin etmek veya açıklamak istenilen değişkendir, bağımsız değişkenler ise bu değişkeni etkileyebilecek potansiyel faktörlerdir. Regresyon analizi, değişkenler arasındaki ilişkiyi belirlemek ve bu ilişkiyi ifade etmek için bir matematiksel model oluşturmaya çalışır. Makine öğrenmesinde regresyon analizinin amacı ise makinenin veri setini ezberlemek yerine öğrenmesi için oluşturulan algoritmadır. Amaçlanan makinenin tutarlı tahmin yeteneğine sahip olmasıdır. Regresyon analizi, bağımlı değişkenin

sürekli olduğu durumlar için kullanılır. Örneğin, bir kişinin gelirini etkileyen faktörleri modellenmek istendiği durumda gelir bağımlı değişkeni kullanılabilirken, bu kişinin eğitim seviyesi, yaş, meslek gibi faktörler ise bağımsız değişkenler olabilir.

- **Linear Regresyon:** Sir Francis Galton, 1894 yılında doğrusal regresyon kavramını ilk kez önermiştir. Bir bağımlı değişken ile bir ya da daha fazla bağımsız değişken arasındaki ilişki, doğrusal bir fonksiyon ile temsil ediliyorsa, bu model Linear Regresyon Modeli olarak adlandırılır. Başka bir deyişle, bağımlı ve bağımsız değişkenler arasında doğrusal ilişkiler kuran bir veri değerlendirme ve modelleme yöntemidir; aynı zamanda en basit ve en yaygın makine öğrenimi algoritmalarından biridir. Model, bir bağımlı ve bir bağımsız değişkenden oluşur. Linear regresyonun amacı, regresyon katsayılarını verilen veri setine en iyi şekilde uyan bir doğruyu bulmaktır. Gerçek ve tahmini değerler arasındaki karelerin toplamını en aza indirecek şekilde katsayılar tahmin edilir.
- **Multiple Linear Regresyon:** Lineer regresyondaki tanımlama aynen geçerlidir, fakat farkı bir bağımlı ve birden fazla bağımsız değişken içermesidir. Bu durumda, Çoklu Lineer Regresyon Modeli olarak tanımlanır. Ancak, gerçek hayat problemleri çok daha karmaşık olduğundan, çoğu zaman bir bağımlı değişken birden çok faktöre bağlıdır. Örneğin, ikinci el bir otomobilin fiyatı, bulunduğu il, üretim yılı, kaza durumu, km olarak yapılan yol, modeli vb. gibi birçok faktöre bağlı olabilir. Doğrusal regresyonda girdi değişkeni ile çıktı değişkeni arasında birbirine karşılık gelen bir ilişki vardır, ancak çoklu doğrusal regresyonda, birkaç bağımsız girdi değişkeni ile bir bağımlı çıktı değişkeni arasında çoklu ilişki bulunur. Çoklu girdi değişkeninin olması, regresyonun daha iyi sonuçlar vereceği anlamına gelmez. Çoklu ve doğrusal regresyon kullanım tercihi farklı durumlar göz önünde bulundurularak yapılır. Birbirlerine üstünlükleri yoktur.
- **Polinomial Linear Regresyon:** Doğrusal regresyon modelinde olduğu gibi, polinomial regresyon da bir bağımlı değişkenin bir veya daha fazla bağımsız değişkenle olan ilişkisini modellenmeyi amaçlar. Ancak, polinomial regresyon, bağımsız değişkenlerin birinci dereceden olmayan terimlerini içerir. Bu terimler, bağımsız değişkenlerin kuvvetleri veya çapraz terimler olabilir. Lineer veya Çoklu Doğrusal Regresyon'da olduğu gibi veriler arasında doğrusal bir ilişki kurulamadığı durumlarda bu karmaşık ilişkileri modellemek için kullanılır. Ancak polinom derecesi mutlaka belirlenmelidir. Polinom derecesinin artmasıyla birlikte modelin esnekliği de artar fakat bu durumda aşırı uyumlanma ihtimali de artar. Bu nedenle, polinomial regresyon için optimal derecenin belirlenmesi oldukça

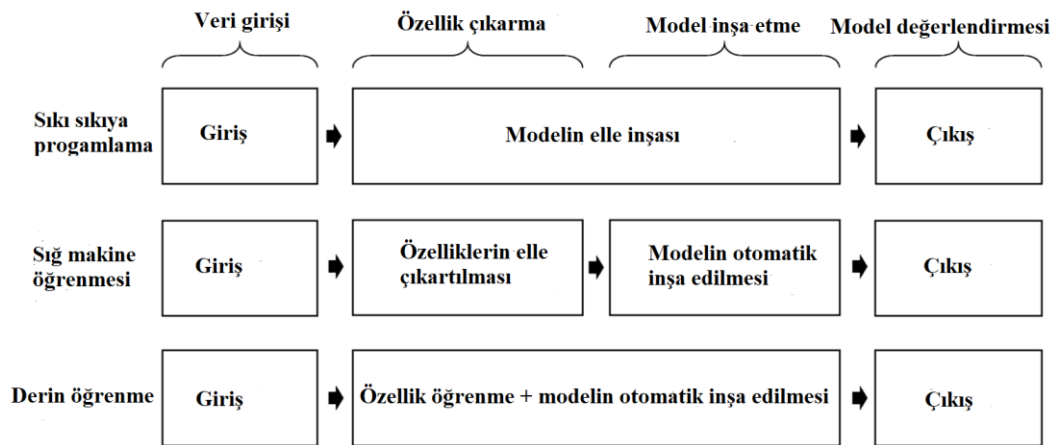
önemlidir. Optimal bir değer ile modelin başarılı ve uygun hale getirilmesi sağlanmış olur.

Birinci dereceden bir polinomun basit bir örneği, $y=\beta_0+\beta_1x$ formülü ile gösterilirken ikinci dereceden bir polinomda, formül $y=\beta_0+\beta_1x+\beta_2x^2$ olabilir.

- **Karar Ağacı(Decision Tree) Regresyonu:** Karar ağacı algoritmasının regresyon problemlerine uyarlanmasıdır. Amacı, yine bağımlı değişkenin tahmin edilmesi için daha önce sınıflandırmada değinildiği gibi bir karar ağacı oluşturmaktır. Bir karar ağacı regresyonu, veri seti üzerinde yine bir dizi bölünme işlemi gerçekleştirir. Her bölünme, veri kümesini belirli bir özellik ve eşik değerine göre iki alt küme ayırır. Bu bölünmeler, veri setindeki değişkenlerle hedef değişken arasındaki ilişkinin belirlenmesi için kullanılır. Regresyon problemlerine uygulanan karar ağacının, sınıflandırma problemlerine uygulanan karar ağacından farklılık gösterdiği unutulmamalıdır. Sınıflandırma karar ağaçlarında, yaprak düğümleri belirli bir sınıfı temsil ederken, regresyon karar ağaçlarında yaprak düğümleri belirli bir sayısal değeri temsil ederler
- **Random Forest Regresyonu:** Sınıflandırma algoritmaları bölümünde değinildiği gibi, Random Forest algoritması hem sınıflandırma hem de regresyon problemlerinin çözümünün modellenmesinde kullanılır. Sınıflandırma veya regresyon modeli olmasındaki farklılığı, rastgele oluşturulan karar ağaçlarının çıktıların çıkışa yansıtılma şeklinden ileri gelir. Her bir karar ağacının bağımsız tahminlerinin ortalaması alınarak hedef değişken tahmin ediliyorsa, bu karar ağacı regresyonu olarak tanımlanır. Her bir karar ağacının tahminlerinin oylaması hedef değişkeni belirlemede kullanılıyorsa, bu durumda sınıflandırma modeli inşa edilmiş olur.
- **Gradient Boosted Regresyon Ağaçları:** GBRT, veri setindeki doğrusal olmayan ilişkileri ve etkileşimleri modelleyebilme yeteneğine sahiptir. Algoritma öncelikle bir dizi ardışık karar ağacı oluşturur ve bu ağaçları bir araya getirerek güçlü bir tahmin modeli oluşturur. Ağaçtaki zayıf karar bölümleri tahminci olarak adlandırılır. Her bir karar ağacı, veri setinin farklı alt kümelerine göre belirli bir özellik ve eşik değerine göre bölünmüş veriler kullanılarak oluşturulur. Ve her bir ağaç, bir önceki ağacın tahmin hatalarını düzeltmeye odaklanır. Önceki ağaçların hatalarını telafi etmek için yeni bir ağaç oluşturulur ve bu süreç birkaç kez tekrarlanır. İşlemler sonunda veri setindeki ilişkiler daha iyi ve ayırt edici olarak öğrenilir ve bu da hedef değişkenin tahmin edilmesindeki doğruluğu artırır.

- **Pace Regresyon:** Projection Adjustment by Contribution Estimation regresyonu Wang ve Witten (Wang Y, 2002) tarafından geliştirilmiş ve yüksek boyutlu veriler için diğer regresyon modellerine kıyasla en iyi performansı gösterdiğini kanıtlamıştır. Algoritma her bir değişkenin etkisini değerlendirerek ve katkılarını tahmin etmek ve benzer modellerin dikkate alınmasına dayanan doğrusal modellere uymaya yönelik yeni bir yaklaşım sunmaktadır. Öncelikle, her bir değişkenin etkisinin hesaplanması için bir kümeleme analizi yapılır. Bu analiz sonucunda değişkenler benzer özelliklere sahip olan gruplara bölünmüş olur. Daha sonra, her bir grup için bir regresyon modeli oluşturulur ve her değişkenin katkısı ayrı ayrı değerlendirilir. Son olarak, bu katkılar, modelin genel etkisi dikkate alınarak yeniden düzenlenir ve modelin doğruluğu artırılır. Zaman içerisinde PACE1'den PACE6'ya kadar adlandırılan altı farklı regresyon prosedürü geliştirildi.

2.3 Derin Öğrenme



Şekil 2.3. Makine öğrenmesi ve derin öğrenme (Goodfellow, 2016)

Giriş verileri web siteleri, uygulamalar, sosyal medya platformları ve mobil uygulamalar tarafından geniş bir yelpazede ve çeşitlilikte üretilebilirler. Ortak sayısal verilerin yanı sıra zaman serileri, görüntüler ve metin gibi yapılandırılmamış ve çapraz ilişkili veriler de dahil olmak üzere çeşitlikte içerebilirler. Bu veriler, analitik model oluşturmak için kullanılabilirler. Ancak, insanın bilişsel kapasitesini aşan desenler ve ilişkileri elle çıkarmak mümkün olmayabilir, bu nedenle büyük ve yüksek boyutlu verilerle uğraşırken algoritmik destekten faydalanmak kaçınılmaz olmaktadır.

Zaman serisi verileri, genellikle regresyon problemleri veya trend sınıflandırma görevleri için zaman içindeki sıralı bağımlılıkları ve desenleri içeren verilerdir. Finansal piyasaların tahmin edilmesi veya süreç davranışlarının

öngörülmesi gibi görevleri içerir (Heinrich, 2021). Görüntü verileri genellikle nesne tanıma veya nesne sayma bağlamında karşımıza çıkar ve uygulama alanları tarım verim tahminleri için mahsul tespitinden otonom sürüşe kadar çeşitlenebilir (Grigorescu, 2020). Metin verileri, kurumsal e-postalar veya sosyal medya gönderileri gibi büyük hacimli belgelerin analiz edildiği durumlarda karşımıza çıkar. Örnek uygulamalar arasında duygu analizi veya belgelerin makine tabanlı çevirisi ve özetlenmesi bulunur (Young, 2018).

Özelliklerin çıkarılması, büyük veri setlerinden desenlerin ve ilişkilerin otomatik olarak tanımlanması için önemli bir aşamadır. Genel olarak, bir özellik, uygun bir temsil sağlamak amacıyla ham veri girişinden türetilen bir özelliktir. Dolayısıyla, özellik çıkarma, bilgiyi korumayı ve öğrenme görevi için önemli olan faktörleri ayırmayı amaçlar.

Sığ makine öğrenimi, bu tür iyi tanımlanmış özelliklere ağırlık verir ve bu nedenle performansı başarılı bir çıkarma sürecine bağlıdır. Zamanla, farklı veri türlerine uygulanabilen çeşitli özellik çıkarma teknikleri ortaya çıkmıştır. Görüntü analizi için uygun yaklaşımlar arasında yönlendirilmiş gradyan histogramları (HOG), ölçekten bağımsız özellik dönüşümü (SIFT) ve Viola-Jones yöntemi bulunur. Doğal dil işleme (NLP) alanında terim frekansı-ters belge frekansı (TF-IDF) vektörleri, konuşma etiketleme (POS) ve kelime şekli özellikleri kullanılır. Manuel özellik tasarımı ise çok fazla alan uzmanlığı gerektirdiğinden zaman alıcı, iş gücü yoğun ve esnek olmayan bir süreç olarak kabul edilir. Derin sinir ağları, bu sınırlamaları aşar. Gelişmiş mimarileri, özellik temsillerini otomatik olarak çıkarmak için minimal insan çabasıyla öğrenme yeteneği sağlar. Özellik öğrenme süreci genellikle hiyerarşik bir şekilde ilerler; yüksek seviyeli soyut özellikler, daha basit olanlar tarafından bir araya getirilir.

Otomatik model oluşturma sürecinde, girişe beslenen veri, öğrenme görevi için gerekli desenleri ve ilişkileri tanımlamada bir öğrenme algoritması tarafından kullanılır. Her öğrenme algoritması, analitik model oluşturmak için farklı mekanizmaları kullanır. Örneğin, bir sınıflandırma modeli oluştururken, karar ağacı algoritmaları, özellik uzayını kullanarak veri setlerini artan derecede homojen bölümlere ayıran ve hiyerarşik, ağaç benzeri bir yapıyı takip eden bir yol izlerken bir destek vektör makinesi (SVM), farklı sınıfların veri noktaları arasına ayırt edici bir hiperdüzlem oluşturmayı hedefler ve giriş verisi genellikle daha iyi ayırt edilebilirlik için yüksek boyutlu bir özellik uzayına yansıtılır. Her birinin giriş verisine ve elde edilen özelliklere bağlı olarak bireysel avantajları ve dezavantajları bulunmaktadır (Kotsiantis, 2006). Zaman içinde çeşitli derin öğrenme mimarileri ortaya çıkmıştır. Temelde mevcut geliştirilmiş her mimari, her görev için kullanılabilirken, bazı mimariler belirli veri tipleri için daha uygun olabilmektedir. Bu mimariler arasında konvolüsyonel sinir ağları (CNN'ler), tekrarlayan sinir ağları (RNN'ler), ve üretici karşıt sinir ağları (GAN'lar) sayılabilir.

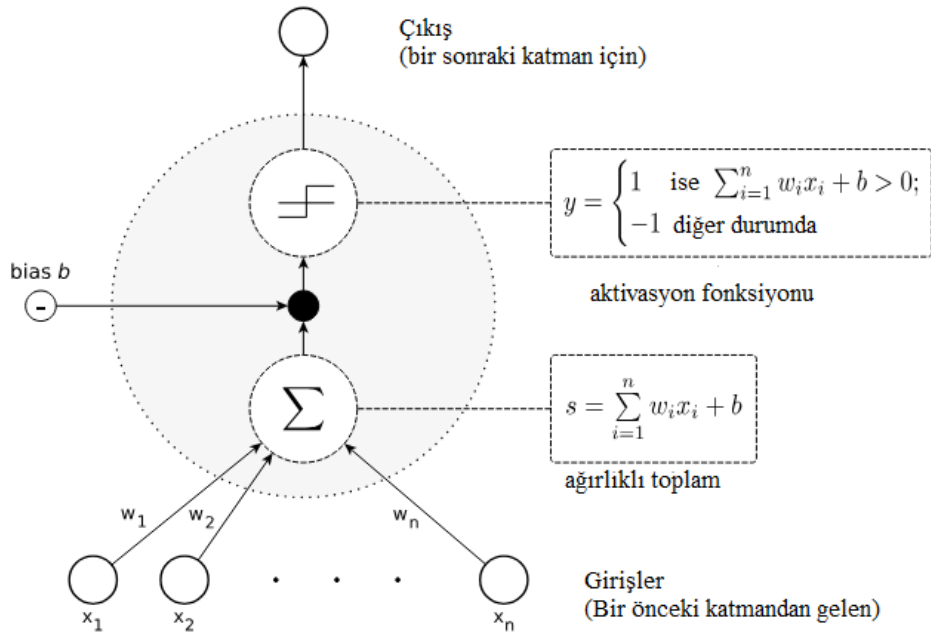
Bir modelin ürettiği çıktılarının değerlendirilmesi için performans ve yorumlanabilirlik gibi çeşitli yönlerinin dikkate alınması gerekmektedir. Performans temelli metrikler, bir modelin öğrenme görevini belirlenen amaca ne kadar uygun gerçekleştirdiği konusunda değerlendirir. Bir modelin aşırı öğrenmesini(Overfitting) önlemek için k-katlı çapraz doğrulama yönteminin kullanılması yaygın bir uygulamadır. Çapraz doğrulama, karşılaştırmalı istatistiksel testleri sağlayan birden fazla veri örneği sağlayarak ML modellerinin güvenilirliğini karşılaştırma olanağı sunar (García, 2008). Regresyon modelleri, kök ortalama kare hatası (RMSE) veya ortalama mutlak yüzde hatası (MAPE) gibi tahmin hatalarını ölçülerek değerlendirilirken, sınıflandırma modelleri doğru veya yanlış tahmin edilen örneklerin farklı oranları hesaplanarak değerlendirilir; bu oranlar arasında doğruluk, hatırlama, hassasiyet ve F1 skoru da bulunur.

Uygun bir tahmin modelini belirlemek için, alternatif modelleri karşılaştırarak karar vermek mantıklı olmalıdır. Bu sayede rekabet eden model sınıfları ve aynı model sınıfının alternatif çeşitleri de dikkate alınmış olur. Bir modelin karmaşıklığı, öğrenme mekanizmalarının türü, manuel veya otomatik olarak çıkarılan özelliklerin sayısı, ve öğrenilebilir parametrelerin sayısı gibi özelliklerle karakterize edilebilir. Diğer yandan, aşırı karmaşık modeller, aşırı uyum riskini de beraberinde getirir. Aynı zamanda seçilen modelin hesaplama maliyeti de dikkate alınmalıdır. Hesaplama maliyeti, bir modelin veriler üzerinde yürütülmesi için gerekli bellek gereksinimlerini ve çıkarsama zamanını ifade eder. Bu kriterler, özellikle milyonlarca parametrenin işlenip depolanabileceği derin sinir ağları değerlendirilirken önemlidir. Böyle bir durum modelin donanım kaynaklarına olan talebinin artması anlamına gelir.

2.3.1 Yapay sinir ağları(ANN)

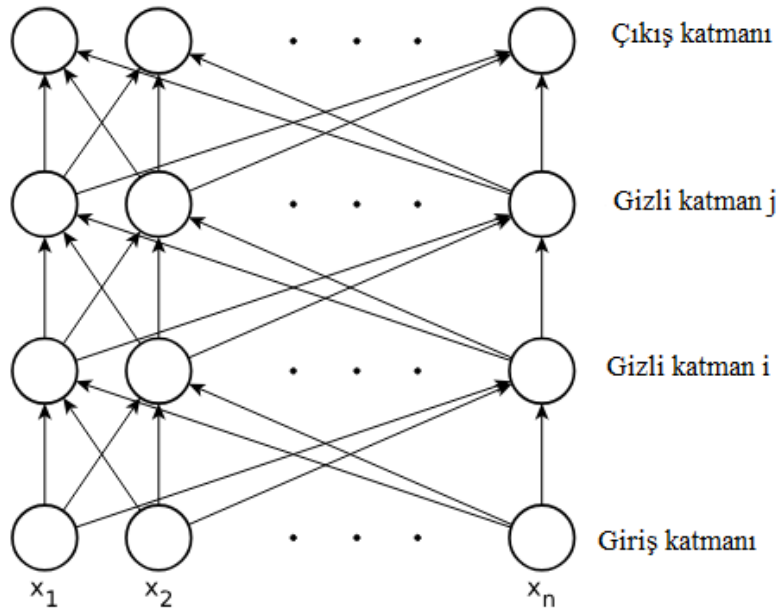
Yapay Sinir Ağları (YSA), biyolojik öğrenme sistemlerinin derinlemesine araştırılması sonucunda öğrenme işlevlerinin genel bir modelini çıkarmaya çalışır. Biyolojik öğrenme, nöron adı verilen sinir yapılarının oluşturduğu karmaşık ağlarla gerçekleşir. Nöronlar, girdileri kabul eden ve tek bir çıktı üreten temel yapılardır. Klasik sınıflandırma görevlerini yerine getiren sinir ağlarına ileri beslemeli sinir ağları denir. Bu ağlar, bir giriş katmanı, bir çıkış katmanı ve bir de gizli katmandan oluşur. Yapay sinir ağındaki en temel nöron yapısına perceptron adı verilmiştir. Perceptron, bir dizi giriş bağlantısına ve bir çıkış bağlantısına sahiptir. Ayrıca, bir de bias olarak adlandırılan iç bir giriş bağlantısına da sahiptir. Bir perceptronun girişine verilen bilgiyi ne şekilde değerlendireceğini ise tercih edilen aktivasyon fonksiyonu belirler. Perceptron, girişe verilen bilgiyi bir ağırlıkla çarpar ve eğitim aşamasında bu ağırlıkları öğrenir. Tüm ağırlıklı giriş değerlerini toplar; sonuç, önceden belirlenmiş bir eşik değerinin üzerindeyse ateş alır. Perceptronun çıkışı boolean 1'dir; eşik seviyesi 0, devre dışı bırakıldığında ise -1'dir.

Şekil 2.4. Perceptron temel yapısı



Şekil 2.4. Perceptron temel yapısı (Staudemeyer, 2019).

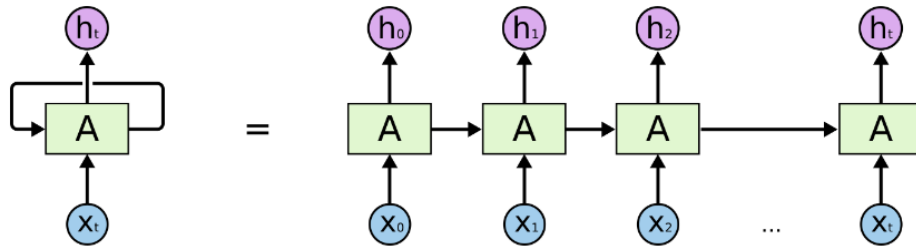
Eğitim aşamasında, perceptron bir çıktı üretir ve bu çıktıyı eğitim verileri tarafından sağlanan değerle karşılaştırır. Eğer sınıflandırma hatalı yapılmışsa ağırlıkları buna göre değiştirir. Ağırlıklar başlangıçta rastgele değerlerle başlar. Perceptron eğitimi bir denetimli öğrenme eğitimidir. Bir perceptronun temel yapısı Şekil 2.4. de verilmiştir.



Şekil 2.5. Çok katmanlı ileri beslemeli sinir ağı (Staudemeyer, 2019).

2.3.2 Tekrarlayan sinir ağıları (RNN)

Tekrarlayan Sinir Ağı (RNN), sıralı veri türlerini işlemek için tasarlanmış bir yapay sinir ağı türüdür. RNN'ler, (Şekil2.6.) geçmiş zaman dilimindeki bilgiyi ve öğrenmeyi, şimdiki zaman dilimindeki çıktıyı üretirken kullanabilen yapıyla önem kazanırlar. Bu, zaman serileri, doğal dil işleme, ses tanıma gibi sıralı verilerin işlendiği birçok alanda RNN'lerin kullanılmasına imkan tanır. RNN'ler, ileri beslemeli sinir ağlarına (feedforward neural networks) kıyasla daha esnek bir yapıya sahiptir, çünkü birçok durumda bir girdiye ek olarak önceki bilgi ve birikimleri de değerlendirip çıkışa yansıtılmış olurlar.



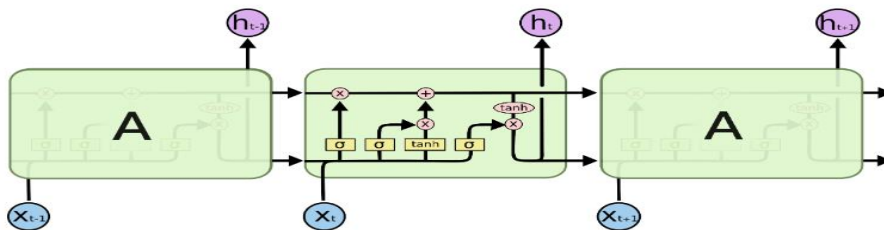
Tekrarlayan sinir ağı (RNN)

Şekil 2.6. Tekrarlayan sinir ağı(RNN) grafik gösterimi.

Basit verilerin kullanıldığı bir hisse senedine ait borsa geçmiş verileri 15, 42, 30... gibi değerlere sahip olsun. Bu durumda $x_0=15$, $x_1=42$, $x_2=30$ içeriğine sahip olacaktır ve bu değerler bir sonraki değer tahmin edilmesinde kullanılacaktır. Geleneksel sinir ağında girdiler ve çıktılar birbirinden bağımsız olmasına rağmen, RNN'nin geçmiş girdileri hatırlaması sayesinde çıktı dizisi içindeki önceki öğelere bağlıdır.

2.3.3 Uzun Kısa Süreli Bellek ağları(LSTM)

Veriler arasındaki bağımlılıkları uzun süreli olarak öğrenebilen özel bir RNN türüdür. Hochreiter ve Schmidhuber (Hochreiter, 1997) tarafından tanıtılmış ve daha sonraki çalışmalarla birçok kişi tarafından geliştirilip popüler hale getirilmiştir. Geniş bir problem yelpazesinde son derece başarılı sonuçlar elde etmişlerdir. Hala yaygın bir şekilde kullanılmaktadırlar. LSTM'ler uzun süreli bağımlılık sorununa çözüm olmak için tasarlanmıştır.

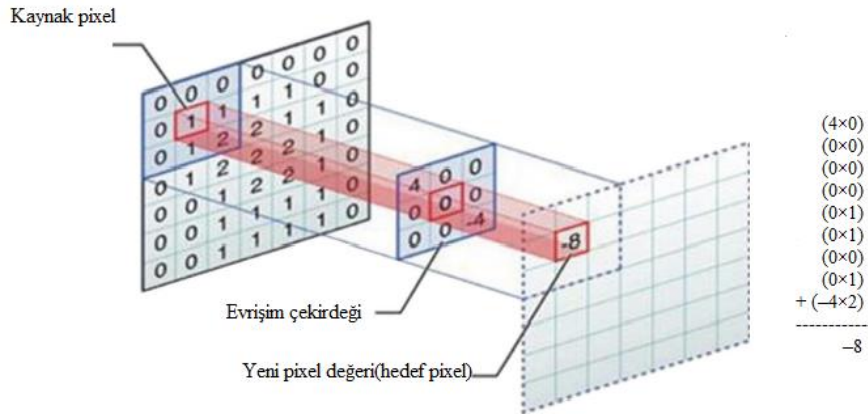


Şekil 2.7. Uzun kısa süreli bellek ağları(LSTM) grafik gösterimi.

Bir LSTM hücresi, giriş kapısı, unutma kapısı ve çıkış kapısı bölümlerinden oluşur. Unutma kapısı, geçmiş bilgilerin hangisinin unutulması gerektiğini belirlerken, giriş kapısı giriş ve geçmiş durum bilgilerine göre hangi bilgilerin hafızada tutulması gerektiğini belirler; bu kısım bilgi güncelleme şeklinde de ifade edilebilir. Çıkış kapısı ise güncellenmiş hafıza durumunu çıkışa yansıtır. **Şekil 2.7.**'de pembe daireler vektör toplama işlemini temsil ederken, sarı kutular öğrenilmiş sinir ağı katmanlarını temsil eder. Birleşen oklar birleştirme işlemini, çatallanan oklar ise içeriğin kopyalandığını ve kopyaların farklı konumlara gittiğini göstermek için kullanılmıştır.

2.3.4 Evrişimli Sinir Ağları(CNN)

Evrişimli sinir ağları Le Cun (B Boser Le Cun, 1990) tarafından tanıtılmıştır. CNN, bir evrişimli katman ile diğer türde katman ve filitreleri de içerebilen yapay sinir ağı türüdür(**Şekil 2.8.**). Doğrusal olmayan havuzlama ve tam bağlantılı katmanlar eklenerek evrişimli sinir ağı daha da derinleştirilebilir (S. Albawi, 2017). CNN'de evrişimli filtreler ve geri yayılım yöntemleri kullanılır, filtre yapılarının şekilleri hedeflenen görevlere bağlıdır. Yüz tanıma gibi bir uygulamada bir filtre kenar özelliklerini çıkartabilirken, başka bir filter göz tespiti ve özelliklerini çıkartmada kullanılabilir.

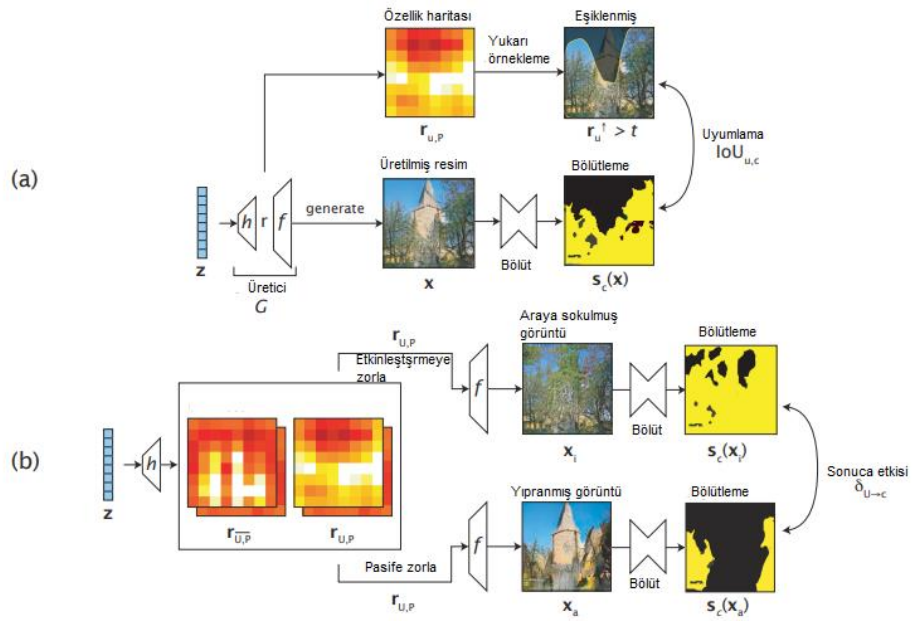


Şekil 2.8. Evrişimli sinir ağları (CNN) (Albawi, 2017).

Evrişim katmanında, birçok filtre giriş verileri üzerinde kayar. Bu giriş verileri genellikle bir görüntüye aittir. Belirlenen filtre boyutu ile örtüşen alandaki giriş verileri eleman eleman çarpılarak toplanır ve sonuç bu katmanın çıkış verisi olur. Aynı zamanda, bu veri bir sonraki katmanın giriş verisini oluşturur. Daha sonra filtre kaydırılır ve işlemler tekrarlanır. Filtrenin kaydırma miktarı adım büyüklüğü olarak adlandırılır. 1 adım her seferinde filtreyi giriş verisi üzerinde bir birim sağa kaydırmak demektir. Sıfır dolgu, temel olarak giriş matrisinin kenarındaki verileri dahil etmeyi amaçlar. Sıfır dolgu olmadan, evrişim çıkışı girişten daha küçük boyutta olur. Bu nedenle, çok katmanlı evrişimlerin olduğu ağlarda ağ boyutu küçülür, bu da ağların çok katmanlı olmasını sınırlar. Bunun önüne geçebilmek için sıfır dolgu kullanılır. Bu şekilde ağların küçülmesi önlenmiş olur ve sınırsız derin katmanlar kullanmanın önü açılmış olur.

2.3.5 Üretken Çekişmeli Sinir Ağları(GAN)

Birbirleriyle rekabet halinde olan iki sinir ağından oluşmuş ve birlikte çalıştığı bir modeldir. Sinir ağlarından birisi üretici (generator) ve diğeri ayırt edici (discriminator) olarak adlandırılır. Üretici, rastgele gürültüden başlamak üzere gerçekçi görüntüler üretmeye çalışır. Amacı, verilen bir gürültü girdisiyle gerçek görüntülere benzer görüntüler üretmektir. Ayırt edici, üretilen görüntülerin gerçek görüntülerden ayırt etmeye çalışır. Eğitim sırasında gerçek görüntülerle üretilen görüntüler arasındaki farkı öğrenir. Üretici, ayırt ediciyi yanıltmaya çalışırken, ayırt edici gerçek görüntüler ile sahte görüntüler arasında ayırım yapmada uzmanlaşmaya çalışır. Her eğitim döngüsünde, üretici daha gerçekçi görüntüler üretmeyi öğrenirken, ayırt edici de daha hassas hale gelmiş olur.



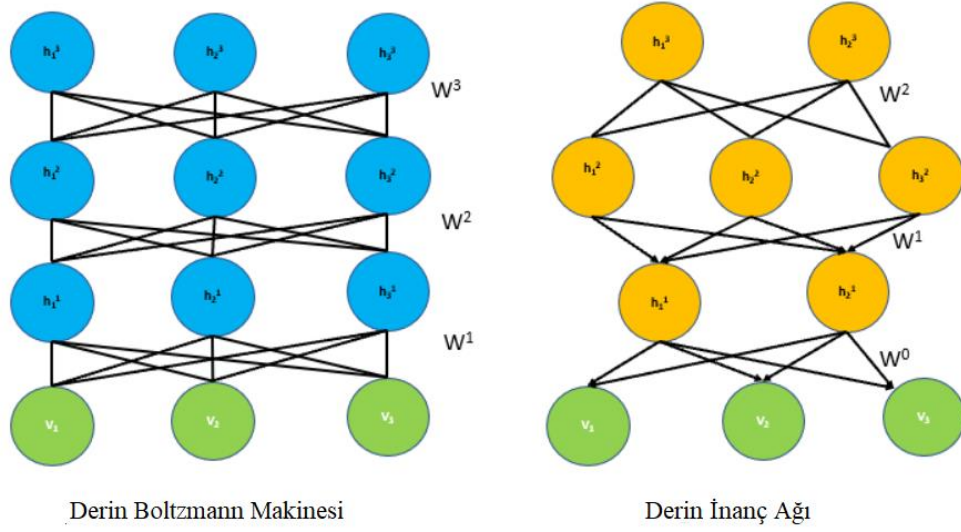
Şekil 2.9. Üretken çekişmeli ağ (GAN) mimarisi (Bau, 2018).

Şekil 2.9. da iki yöntem olan a) dissection, b) Intervention yöntemlerini göstermektedir. Dissection yönteminde, birimin eşiklenmiş ve örneklendirilmiş ısı haritası, üretilen görüntünün anlamsal segmentasyonu ile karşılaştırılır. Intervention ise segmentasyon ile birim ekleme sonrasında ağaçların arttığını ve birim yıpratması sonrasında azaldığını ortaya koyar. Ağaç piksellerindeki ortalama fark, ortalama neden-sonuç etkisini ölçer.

2.3.6 Derin İnanç Ağları(DBN)

Derin inanç ağları(DBN) kısıtlı Boltzmann makinelerinin(RBM) çok katmanlı olarak bir araya getirilmesiyle oluşturulmuş derin öğrenme de önemli bir model mimarisidir(Şekil 2.10.). DBN'ler, genellikle özellik çıkarma, sınıflandırma, regresyon ve olasılıksal modelleme gibi birçok görevde kullanılır. DBN ler klasik sinir ağları ile ilgili yavaş öğrenme, çok sayıda eğitim veri seti gerektirme, ve yerel minumumlara takılma gibi olumsuzluklara çözüm üretme potansiyeline sahiptir.

DBN lerde her bir alt ağı gizli katmanı bir sonraki katman için görünür katman görevi görür.



Şekil 2.10. Derin inaç ağları ve kısıtlı Boltzmann makinesi.

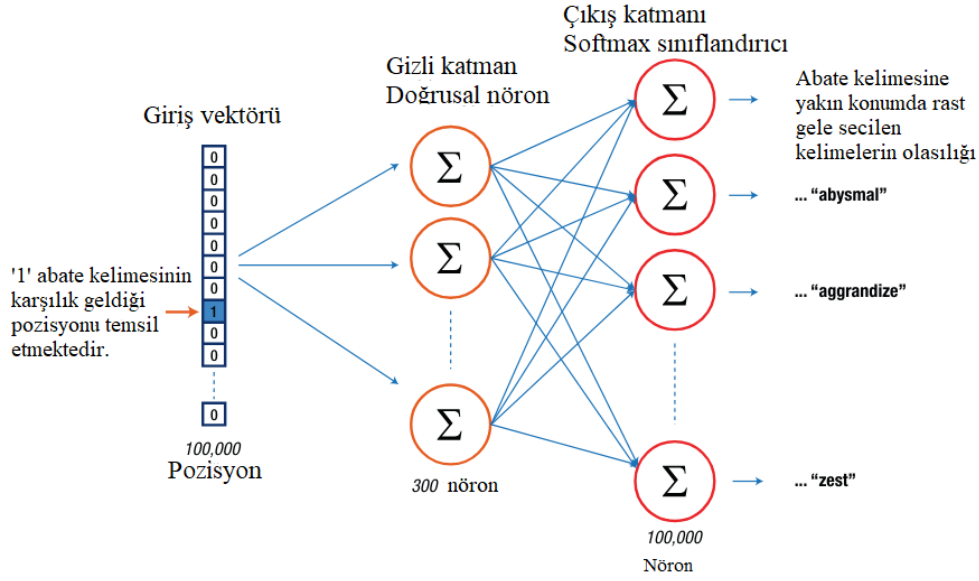
Boltzmann makinesinin “çıkış” katmanının sonucu ardı ardına bir sonraki Boltzmann makinesine girdi olarak eklenir. Daha sonra yakınsayana kadar eğitilir ve ağ tamamlanana kadar aynı işlem uygulanır. DBN'nin tüm katmanları aynı anda eğitilmez. Bunun yerine, her bir RBM katmanı tek tek eğitilir. Bu, "greedy layer-wise pretraining" olarak adlandırılan bir süreçtir. Bu süreçte, veri setindeki örüntüler, giriş katmanından başlayarak her bir gizli katman tarafından öğrenilir. Bu sayede ağın daha iyi bir şekilde özellikleri öğrenmesi sağlanır (Hinton, 2006).

2.3.7 Doğal Dil İşleme(NLP)

Doğal Dil İşleme (NLP), İngilizce, Japonca, İtalyanca veya Rusça gibi bir veya daha fazla insan dilini analiz etmeye, anlamaya veya üretmeye çalışan bilgisayar sistemlerini ifade eder (Allen, 2003). Doğal dilin bir sistem tarafından gerçekten "anlaşıldığının" nasıl belirleneceğini tanımlamak son derece zordur. Aslında test edilebilecek şey, sistemin dilin anlaşıldığına benziyormuş gibi davranıp davranmadığıdır. Bu konuda Turing testi klasik model olmuştur (Turing, 1950). Bu testte, sistem, bir terminal üzerinden bir insan tarafından rastgele sorgulamalara yanıt verirken bir insanla ayırt edilemez olmalıdır.

Şekil 2.11. de skip-gram modeline ait grafik görülmektedir. Model mevcut kelimeyi sıradaki kelimeleri tahmin etmek için kullanır. Bunu çevredeki kelimelerin sınıflandırma puanı, merkez kelime ile olan sözdizimsel ilişki ve ortaklıklara dayandırarak yapar. Sıralamada bulunan her kelime, log-lineer sınıflandırıcıya giriş olarak alınır, bu da önceden belirlenmiş kelime aralığı altına düşen kelimelerin tahminini yapar. Kelime aralığının seçimi ile sonuçta elde edilen kelime vektörlerinin hesaplama karmaşıklığı ve kalitesi arasında bir denge vardır.

İlgili kelimeye olan uzaklık arttıkça, daha düşük seviyede merkez kelime ile ilişkilidir (Goyal, 2018).



Şekil 2.11. Doğal dil işleme(NLP) kelime vektör gösterimi (Goyal, 2018).

3. LİTERATÜR TARAMASI

Teknolojik ilerleme, insanlığın tarih boyunca karşılaştığı problemlere çözüm bulma veya yaşamını kolaylaştırma arayışının, hatta hayal ettiğini gerçekleştirme çabasının bir sonucu ve ortak birikimi olarak karşımıza çıkmaktadır. Bu süreçte, bireysel yeteneklerin yanı sıra takım çalışmaları da önemli bir rol oynamıştır. Bireyler veya gruplar, farklı uzmanlık alanlarından gelen bilgileri bir araya getirerek, teknolojinin evrimine katkıda bulunmuşlardır. Karşılaşılan ortak problemler, genellikle kolektif çabalarla aşılmış, bu da toplumsal düzeyde inovasyonu teşvik etmiştir. Hal böyle olunca, elbette ki yakın bir tehdit olarak karşımıza çıkan temiz suya ulaşım imkanının azalması ve bölgesel kuraklıkların hem sayıca hem de derece olarak artması, bu alanda yapılan çalışmaların çoğalmasına neden olmuştur. Konu hakkında yapılan çalışmaların incelenmesi sonucunda yapılan çalışmayla örtüşen paralel bir araştırmaya rastlanmamış olması, konuya daha fazla odaklanılması için bir motivasyon kaynağı olmuştur. Bu doğrultuda yapılan bazı çalışmalar ise şu şekildedir.

Guevara ve arkadaşları(Guevara, 2020), Peru Yaylaları için nesnelerin interneti tabanlı bir sulama sistemi önermişlerdir. Elektrik enerjisine ulaşım imkanının olmadığı çalışma sahasında ihtiyaç duyulan enerji güneş panelleri ile şarj olunan bataryalardan sağlanmıştır. Önerilen çalışmada sensörler aracılığıyla toplanan veriler sıcaklık ve toprak nem değerleri açısından ölçümlendirilmiş ve Arduino Uno kullanılarak Digi Xbee kablosuz modülüne aktarılmıştır. Bu modül gömülü

sistemlere internet erişim imkanı sunmaktadır. Kullanıcı bilgisayarında çalışan bir uygulama ise planlama ve verileri monitörize etmede kullanılmıştır. Sulama işlemi sensörden alınan ve belirlenen bir nem eşik değerine göre Arduino Uno üzerinden kontrol edilen valflerin açılıp kapatılması ile gerçekleştirilmiştir. Önerilen sulama sistemi düşük maliyetli otonom bir sistemdir.

Katherine Milla ve Stephen Kish (Kish, 2006), toprak infiltrasyon hızını analiz etmek ve elde ettikleri verileri bitkinin su ihtiyacını belirlemede kullanmak amacıyla tek ve çift halkalı infiltrasyonmetrelere entegre ettikleri lazer seviye ölçüm sensörü ve mikro işlemci tabanlı bir otonom veri toplama sistemi inşa ettiler. Sistem tarafından sulama miktarı ve zamanlaması kaydedildi ve bu verilerin, detaylı bir analizi ve değerlendirilmesinin yapılması için bir bilgisayara aktarımı sağlandı.

Viviana Gavilan ve arkadaşları (Gavilan, 2019), Evapotranspirasyon (su döngüsünün bir parçası olup atmosferdeki su buharının oluşumunu sağlar. Bir ekosistem içindeki bitkilerin ve toprağın yüzeyinden atmosfere su buharı kaybını ifade eden bir terimdir. Evapotranspirasyon miktarı, bitki örtüsü, iklim, toprak özellikleri ve diğer çevresel faktörlere bağlı olarak değişebilir.) değerini tarım alanlarının uzaktan algılama ile izlenmesinde kullanılan Landsat-8 ve Sentinel-2 uydu görüntülerinden faydalanan tahmin etmeye çalışmışlardır. Gerçekleştirilen çalışma da elde ettikleri verilere Normalize Fark Bitki Örtüsü 'İndeksi (NDVI)' ni entegre ederek ETo değerini NDVI zaman serileri kullanılarak tahmin etmede başarılı sonuçlar elde etmişlerdir.

Henner Gimpel ve arkadaşları (Gimpel, 2021), Frankfurt'taki 200.000 ağaçtan 4000'inden fazlasının kuraklıktan ölmesi üzerine bu duruma önlem almak için Nesnelerin İnterneti (IoT) ve sensörlere dayalı DSR (bir soruna etkili bir çözüm bulmayı amaçlayan bir arama süreci Hevner et al., 2004) kurmuşlardır. Yapılan çalışmada 8 adet ağaç pilot olarak seçilmiş ve bu ağaçlara 18 adet sensör üç aylık bir için dönem kurulmuştur. Toprak nemi ve kalitesi zaman serileri olarak toplanıp değerlendirilmiş ve Frankfurt'taki 5000 genç ağaca uygulandığı takdirde ilkbaharda bir sulama dönemi içinde su dağıtımının 1 milyon litreye kadar azaltılabileceği sonucuna varmışlardır.

John R. dela Cruz ve arkadaşları (Dela Cruz, 2017), YSA tabanlı bir sistemi bir çiftlikte sulama suyunun optimize edilmesinde kullanmışlardır. Sistem, su seviyesi sensörü, toprak nem ölçüm sensörü, sıcaklık sensörü, hava nem sensörleri kullanılarak inşa edilen Su Tankı İzleme ve Kontrol Sistemi, Açık Sulama Kontrol Sistemi ve Bölge Sulama Kontrol Sistemi olmak üzere üç ana bölümden oluşmaktadır. 2 giriş ve 1 çıkış verisine sahip 20 adet nöron içeren YSA çalışma sonunda başarılı olarak görmüşlerdir.

Ehsan Raei ve arkadaşları (Raei., 2022), görüntü segmentasyonu ile ABD'nin Idaho eyaletindeki tarım alanlarının sulama türlerini tahmin etmek ve belirlemek amacıyla U-Net derin öğrenme mimarisıyla eğittikleri bir yapay zeka modelini kullandılar. Modelin eğitiminde ABD Tarım Bakanlığı'nın Ulusal Tarım

Görüntüleme Programı(NAIP) tarafından sağlanan uzaktan algılama yöntemleri ve ticari uydulardan elde ettiği yüksek çözünürlüklü görüntülerden faydalandılar. Eğitilen model verilen görüntüden

- 1: Merkez-Pivot (CP) sulama sistemiyle sulanan alanlar
- 2: Doğrusal yağmurlama Sulama (LI) sistemiyle sulanan alanlar
- 3: Sallanır Kollu Merkezi Pivot (CPSA) sulama sistemiyle sulanan alanlar
- 4: Karıklı Yüzey Sulama (SI) sistemiyle sulanan alanlar
- 5: Kentsel Alan (UA)
- 6: Yabani Alan/Arka Plan Alanı (BG)

olmak üzere 6 kategorideki sulama sistemlerinden (SI, LI, CP ve CPSA) sınıflandırma tahminini en az %72 ve %92'si için doğru şekilde yaptı.

Literatür taraması sonucunda, bu tez çalışmasına paralel nitelikte bir araştırmaya rastlanılmamıştır. En yakın çalışma, Ehsan Raei ve arkadaşlarının (Raei., 2022) ABD'de gerçekleştirdikleri araştırma olmakla birlikte, bu tez çalışmasıyla ortak bir amacı taşımaktan çok uzaktır. Çalışmaları herhangi bir sulama aksiyonu veya stratejisi barındırmamakta olup, sadece uydu görüntülerinden yola çıkarak daha önce tercih edilmiş veya sürdürülmekte olan sulama türünün altı türden hangisi olduğunu tahmin etmeye yöneliktir.

4. MATERYAL VE YÖNTEM

Birçok akıllı sulama sistemi ve algoritma, bitkinin su ihtiyacını belirlemede sensör verilerini kullanır. Elde edilen veriler ışığında ya belirlenen bir eşik seviyesine göre bir sulama kararı alırlar ya da sulama zaman çizelgesi kullanarak önceden belirlenmiş programı uygulurlar. Sensörler uygulama sahasına konumlandırılırken, karar destek birimi ise Nesnelerin İnterneti (IoT) kullanılarak ulaştırılan sensör verilerini işleyen bir bilgisayar olabilmektedir.

Bitkinin su ihtiyacını belirlemede kullanılan diğer bir yaklaşım ise sensör verilerinden evapotranspirasyon (ET_o) değerini hesaplamaktır. ET_o değeri, bir bitkinin su tüketim miktarını bulmak için kullanılmaktadır. Her bitkinin yaşam evreleri, bitkinin bulunduğu coğrafi konuma ve mevsimsel koşullara göre farklılıklar göstermektedir (Allen R. G., 1998).Bitkilerin hassas sulama programının oluşturulması amacıyla Penman-Monteith yöntemi, FAO tarafından 1990 yılında önerilmiştir.

Yapılan tez çalışması kapsamında planlanan sistem, geniş bir yelpazede ayakları olan düşük akım otomasyon kontrol donanımları, yüksek akım ekipmanları, yazılım, donanım, gömülü sistem yazılımı, sensörler, kamera, mekanik ve elektronik ekipmanların bir arada uyum içinde çalışmasını gerektiren karmaşık bir sistemden oluşmaktadır. Bu kapsamda ilk olarak uygulama sahasında ekipmanların merkezi sinir ağı ve otomasyonunu gerçekleştirecek elektronik kontrol ve otomasyon sistemi tasarlanıp geliştirilmiştir. Bu elektronik kart, çoklu

fonksiyonel elektronik kontrol kartı olarak adlandırılan MFECB adını almıştır. **Şekil 4.1.** ve **Şekil 4.2.** MFECB'nin şematik ve pcb gösterimidir. MFECB, uluslararası 11. ICEEE 2024 Kongresi'nde sunulmuş ve ilgili yayın organında yayınlanmıştır.

4.1 MFECB (Çoklu Fonksiyonel Elektronik Kontrol Kartı)

MFECB tasarlanırken, uygulama sahasında gerçekleştirilmesi beklenen görevler ve ana başlıklar belirlendi. Bu başlıklar şunlardır:

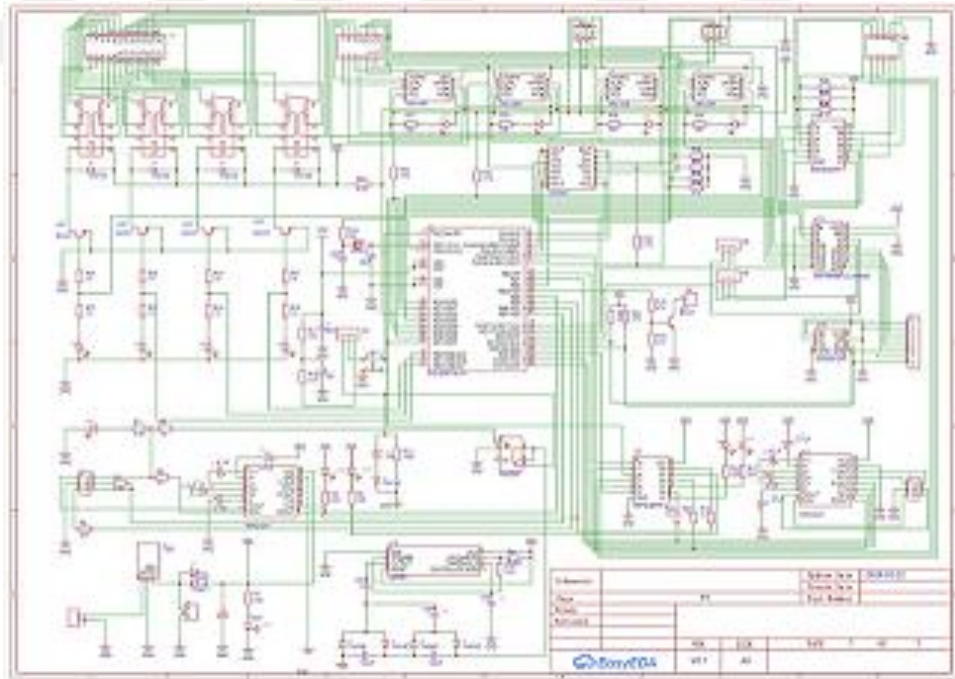
- Sulama başlıklarından su deşarjının akış hızını ve çapını ayarlamak için, su temininde kullanılan AC pompada ayarlanabilir hız kontrolü olması gerekiyordu. Bu amaçla, AC motor hız kontrol ünitesinin gereksinime göre yazılım tarafından kontrol edilebilmesi için sayısal potansiyometrelere kontrol kartında yer verildi.
- Su pompasını çalıştırmak ve sulama başlıklarını etkinleştiren veya devre dışı bırakan aktuatörleri kontrol etmek için dört adet çift kutuplu röle dahil edildi.
- Gerekli olduğunda kartın işletim durumunu izlemek ve bir PC kullanarak yazılım güncellemeleri gerçekleştirmek için bir RS-232 protokollü bağlantı portuna yer verildi
- Kart üzerine, iş istasyonunda çalışan derin öğrenme algoritmaları ile veri transferi için IoT cihazlarının entegrasyonunu kolaylaştırmak amacıyla donanım akış kontrolü etkinleştirilmiş ikinci bir RS-232 protokole sahip iletişim portu dahil edildi.
- Tüm bu başlıklara ek olarak, kart üzerinde yerleşik mikrodenetleyici programlayıcı, dahili genişletilebilir EEPROM bellek birimi ve çevre donanım birimleri ile iletişimi sağlayacak çok sayıda GPIO bağlantı noktası dahil edildi.



Fotoğraf 4.1. Çoklu fonksiyonel elektronik kontrol kartı(MFECB).

- 2 adet sıcaklık sensör bağlantı terminali.
- Analog sistemleri doğrusal olarak kontrol edebilmek maksatlı tarafından kontrol edilen fiziksel 0-100K potansiyometre.
- Sisteme eklenecek aktüatörler için 4 adet 250V AC 1A çift bağlantı portu
- Gerektiğinde çevre donanım birimlerini donanımsal resetlem için resetleme çıkışı.

- Mikroişlemci 16F877.
- Yerleşik PIC serisi mikroişlemci programlayıcı.
- 512KB yerleşik EEPROM hafıza birimi(genişletilebilir).
- İki adet RS232 iletişim portu bunlardan biri, PC bağlantısı için yazılım akış kontrolüne sahipken, diğeri (ESP8266, NodeMCU, ESP32, Arduino, Wi-Fi, Bluetooth, GSM modem, GPS modülü vb. gibi) eklenebilecek çevresel donanım cihazları ile iletişim kurabilmek için donanım akış kontrolüne sahiptir.
- 2 adet sıcaklık sensor bağlantı terminali.
- Analog sistemleri doğrusal olarak kontrol edebilmek maksatlı 4 adet yazılım tarafından kontrol edilen fiziksel 0-100K potansiyometre.
- Sisteme eklenecek aktüatörler için 4 adet 250V AC 1A çift kontak röle bağlantı portu
- Gerektiğinde çevre donanım birimlerini donanımsal resetleme yapabilmek için resetleme çıkışı.

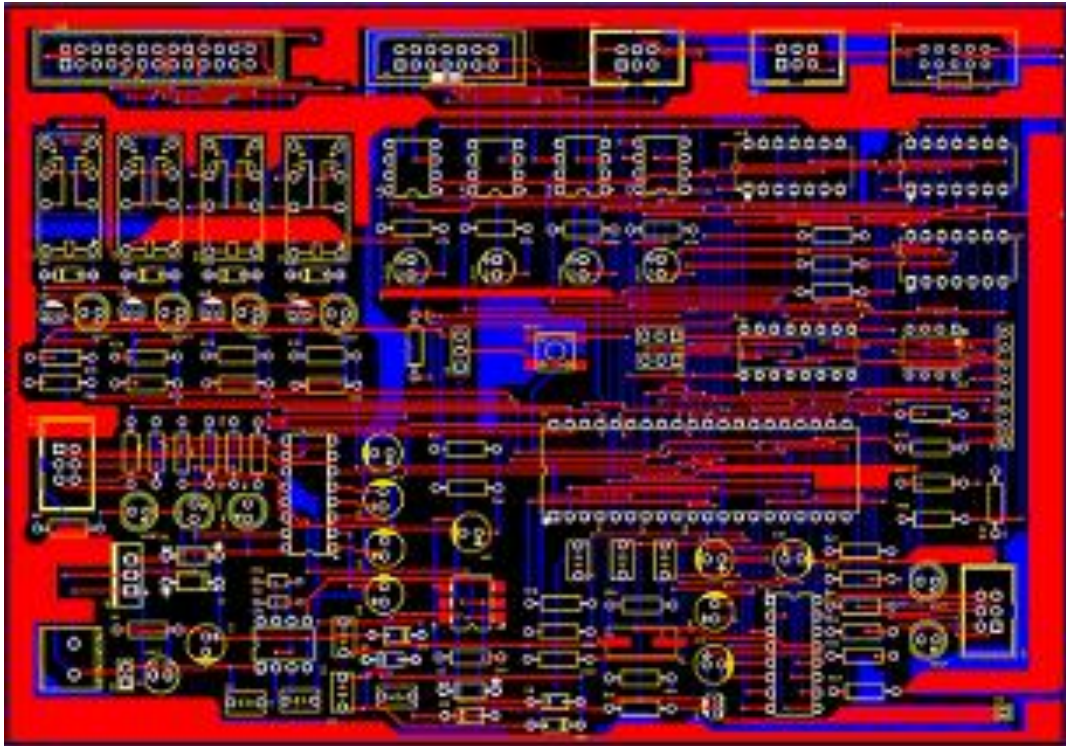


Kontrol kartındaki merkezi işlem birimi bir PIC MCU'dur ve donanım yeteneklerini genişletmek için yardımcı elektronik bileşenlerle desteklenmektedir. Çevre donanım cihazlarından kaynaklanabilecek donanım arızalarının kontrol kartını etkileyerek işlevsiz hale getirmesini önlemek için, tüm giriş ve çıkışlar

merkezi sistemden izole edilmiştir. Ayrıca, yazılım çökmesini ve donmalarının önüne geçebilmek için de çip üzerinde çalışan gömülü sistem yazılımı watchdog timer denetimi altında çalışacak şekilde tasarlanmıştır.

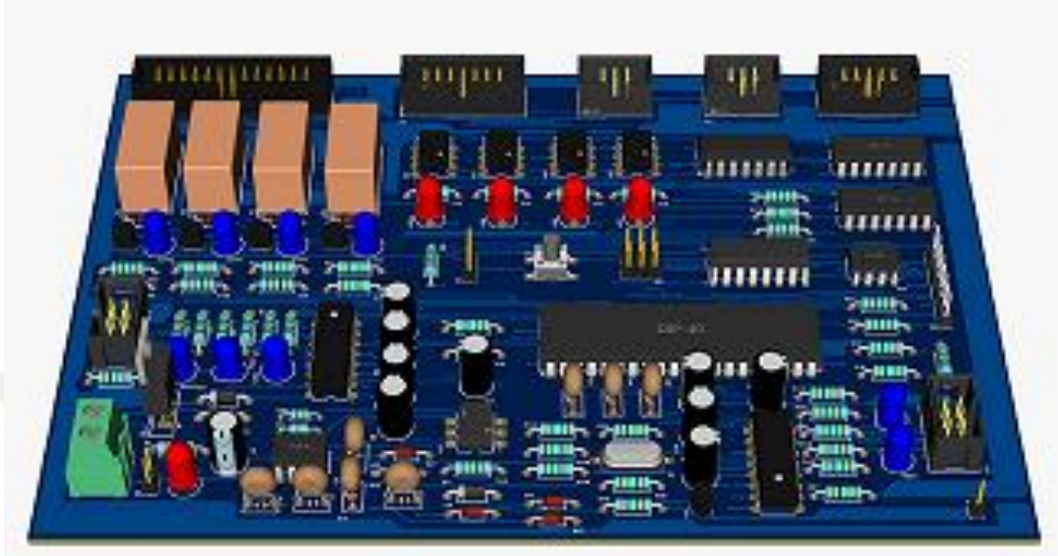
- 4 adet tampon korumalı analog giriş.
- Aktüatörler için kullanılabilecek 4 adet tampon korumalı dijital çıkış.
- Sensörlere bağlanabilecek 3 adet tampon korumalı genel amaçlı dijital giriş.
- Tüm analog ve dijital giriş/çıkış portlarına ek olarak, genel amaçlı kullanım için bulunan 4 adet giriş/çıkış portu kameraların bağlanması veya düşük güç aktüatörlerin kontrolü için kullanılabilir.
- 2 adet PWM (Darbe Genişliği Modülasyonu) portu.
- 20 MHz çalışma frekansı.
- Watchdog Timer denetimi altında çalışan gömülü system yazılımı.

Elektronik kontrol üniteleri endüstriyel uygulama ve otomasyon sistemlerinde önemli bir rol oynamaktadır. Bu sistemler çeşitli sensörler ve aktüatörler aracılığıyla çevresel koşulları algılar ve kontrol ederler. Mevcut elektronik kontrol kartları incelendiğinde, bazılarının belirli sorunları ele almak için tasarlandığı, diğerlerinin ise yaygın kullanıma yönelik olduğu görülmektedir. Geliştirilen otomasyon kartı, sadece üzerinde çalışacak yazılım ile yaygın kullanım ihtiyaçlarını karşılamakla kalmaz, aynı zamanda özel çözümler de sunmaktadır. Elektronik kart üzerinde çalışan yazılım, geniş yüzey alanlarında sulama özelleştirilmesi için derin öğrenme algoritmalarıyla uyumlu olarak çalışacak şekilde kodlanmıştır.



Şekil 4.2. Çoklu fonksiyonel elektronik control kartına(MFECB) ait Pcb.

Kartın iki işlevi yenilik olarak öne çıkmaktadır. Bu işlevler arasında, çevresel cihazların yazılım kilidinde kaldığı durumlarda kilitli donanımı donanımsal olarak(HardReset) sıfırlamak için fiziksel bir sıfırlama çıkışı ve entegre analog sistemlerin doğrusal kontrolü için yazılım tarafından kontrol edilen fiziksel potansiyometreler yer almaktadır.

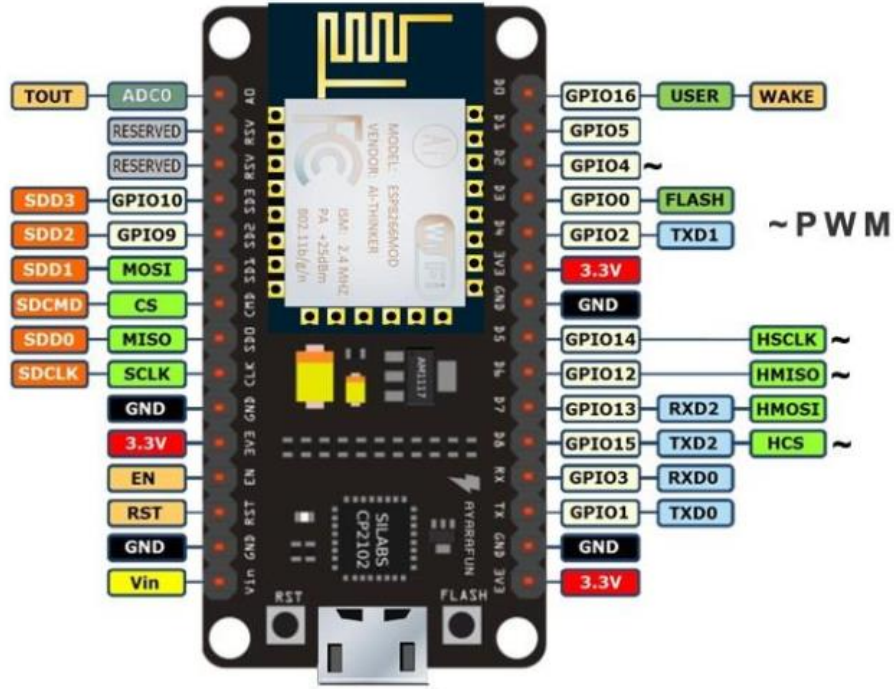


Fotoğraf 4.2. MFECB 3D görünüm.

Kart, işletme sahasında hizmet verirken çevresel sıcaklık ve diğer sensör verilerini toplar ve ayrıca sulanan alanın periyodik olarak görüntülerini yakalar ve bunları internet aracılığıyla bir iş istasyonunda çalışan derin öğrenme (DL) algoritmalarına iletir.

4.2 NodeMCU

NodeMCU, IoT tabanlı uygulamalarda sıklıkla kullanılan açık kaynak bir platformdur. Bu platform, cihazlara WiFi erişim noktasına bağlanma imkanı sunarken bu sayede internet üzerinden veri transferini kolaylaştırır. Bu modül, Espressif Systems tarafından üretilen ESP8266 WiFi yonga tabanlı olarak geliştirilmiştir. ESP8266'nın TCP/IP protokol yığınına kolayca kullanma imkanı sunmanın yanı sıra, mikro işlemcinin yeteneklerinden de faydalanma imkanı tanır. Modülde bulunan birden fazla GPIO pini, modülün diğer çevre birimlerine PWM, I2C, SPI ve UART seri iletişimleriyle bağlanmasına izin verir. Lua yazılım geliştirme ortamı veya Arduino IDE kullanılarak modül için uygun yazılım geliştirilebilir. NodeMCU ismi aslında ESP8266 çipi üzerinde çalışan Firmware'e verilen isimdir ve donanım ESP-12E modülüdür. Lua script dili, çip üzerindeki Firmware'i güncellemek için kullanılır.



Şekil 4.3. Node MCU

Projede, uygulamanın merkezi sinir ağı olan MFECB kontrol kartına ağ bağlantısı ve internet erişim imkanı sağlayarak merkezi iş istasyonunda çalışan uygulamalar ile veri aktarımı sağlamak için kullanılmıştır.

Teknik özellikleri:

- Açık kaynak
- Arduino benzeri donanım
- Durum LED'i
- MicroUSB bağlantı noktası
- Reset/Flash düğmeleri
- Etkileşimli ve Programlanabilir
- Düşük maliyet
- Dahili Wi-Fi özelliğine sahip ESP8266
- USB'den UART dönüştürücü
- GPIO pinleri
- Arduino benzeri donanım IO
- Lua script diliyle etkileşimli olarak Arduino gibi kodlama.
- Node.js tarzı ağ API'si.

Tablo 4.1. Dahili GPIO pinlerine karşılık gelen NodeMCU pin haritası.

D0	D1	D2	D3	D4	D5	D6
GPIO16	GPIO5	GPIO4	GPIO0	GPIO2	GPIO14	GPIO12
D7	D8	RX	TX	SD2	SD3	
GPIO13	GPIO15	GPIO3	GPIO1	GPIO9	GPIO10	

Tablo 4.2. NodeMCU pin numaraları ve fonksiyonları (Dutta, 2021).

NodeMCU Modül Pin Numaraları	Kesme Fonksiyonu	I2C veya PWM Fonksiyonu	Açıklama
D0(GPIO16)	✗	✗	Açılışta Lojik 1 e ESP8266 yı derin uykudan uyandırmak içinde RST pinine bağlanmalıdır.
D1(GPIO5)	✓	✓	I2C iletişim için SCL ucu
D2(GPIO4)	✓	✓	I2C iletişim için SDA ucu
D3(GPIO0)	✓	✓	Bu uç program tuşuna bağlıdır. Eğer bu uç lojik 0 a çekilirse boot işlemi olmaz.
D4(GPIO2)	✓	✓	Bu uç modüldeki Led'e bağlıdır. Giriş olarak kullanılacağı zaman pull up yapılmalıdır.
D5(GPIO14)	✓	✓	SPI haberleşme için SCLK
D6(GPIO12)	✓	✓	SPI haberleşme için MISO
D7(GPIO13)	✓	✓	SPI haberleşme için MOSI
D8(GPIO15)	✓		SPI haberleşme için CS. Bu giriş olarak kullanılacağı zaman pull down yapılmalıdır.
RX(GPIO3)	✓	✗	Bu uç boot zamanında HIGH olmalıdır. Çıkış olarak kullanıldığında boot zamanında davranışı öngörülemez.
TX(GPIO1)	✗	✓	Bu uç boot zamanında HIGH olmalıdır. Çıkış olarak kullanıldığında boot zamanında davranışı öngörülemez.
A0	✓	✗	0 ile 3.3V arasında analog girişi kabul eder. ADC çözünürlüğü 10 Bittir.

4.3 Edimax Ic3030Wn IP Kamera

İnternet Protokolü (IP) veya ağ kameraları olarak isimlendirilen bu kameralar geniş alan veya yerel ağ üzerinden ya da internet üzerinden görüntüleri sayısal olarak aktarma kabiliyetine sahiptirler. IP kameralar, WiFi veya Ethernet Üzerinden

Güç (PoE) kablosu aracılığıyla bir ağı bağlanırlar ve bağlandıkları sistem tarafından kendilerine atanmış IP adresine sahip bağımsız bir birim olarak çalışırlar.



Fotoğraf 4.3. Edimax IC-3030Wn IP kamera.

IP kameralar, görüntüleri sayısal kameralarla aynı şekilde yakalar ve ağ üzerinden iletmek için dosyaları sıkıştırarak ethernet kablosuyla bir geniş bant modeme veya yönlendiriciye bağlı kablolu bir ağla ya da bir WiFi yönlendirici aracılığıyla kablosuz olarak aktarırlar. Düşük çözünürlüklü siyah beyaz sadece video çeken basit kameralardan, hareketli servolar ile görüş menzili içindeki insanları ve hayvanları otomatik olarak takip edebilen tam renkli HD cihazlara kadar çeşitlilik gösterir. Hatta yerleşik yapay zeka modülleri bulunan IP kameralar da mevcuttur. Bu kameralardan otomatik nesne tanımlama ve takibi, öğrenilmiş olay ve sahnelerden tehdit veya durum sınıflaması yaparak raporlama ve bildirim gönderme yeteneklerine sahip olanları da vardır. IP kamera ile çekilen canlı görüntü ve sesler, harekete duyarlı tetikleme ile anlık olarak cihaz üzerine kaydedilen görüntüler bilgisayar veya cep telefonu aracılığıyla internet üzerinden dünyanın herhangi bir yerinden izlenebilir. Çoğu durumda kamera, video görüntülerini izleyebilmenin ve ses akışını dinlemenin yanı sıra uzaktan da kontrol imkanına sahiptir.

Uygulamada, iş istasyonunda çalışan derin öğrenme algoritmalarının görüntü segmentasyonu yapabilmesi için uygulama sahasından görüntü yakalama ve aktarma amacıyla kullanılmıştır.

Teknik özellikleri:

- H.264, MPEG4 ve M-JPEG üçlü modlu video yayını.
- Çoklu bölge hareket algılama ve anlık görüntü yakalama ile E-posta ve FTP'ye gönderme.

- Manuel Odaklı lens.
- VGA'da 30 fps Video Kare Hızı.
- İki yönlü ses aktarımı.
- Yüksek hızlı donanım tabanlı sıkıştırma.
- Tetiklenen etkinlik için ön kayıt + son kayıt.
- Anlık görüntüleri ve kaydedilen videoyu yerel olarak bir SDHC/SD hafıza kartına saklamak için dahili SDHC/SD kart yuvası.
- Yazılım aracılığıyla Manuel/Zamanlamalı/Döngü kaydı ve Oynatma/İleriye sarma/ Ara verme/Durdur özelliği.

4.4 Wi-Fi Erişim Noktası(Access Point)

Erişim noktası (Access Point), bir kablosuz ağ oluşturmak ve cep telefonu, IoT cihazları, bilgisayarlar ve gömülü sistem cihazları gibi çok sayıda ve çeşitlilikteki cihazları oluşturulan ağa dahil ederek bu cihazlar arasında veri aktarımını sağlar. Aynı zamanda bu cihazlara internet erişim imkanı tanır.



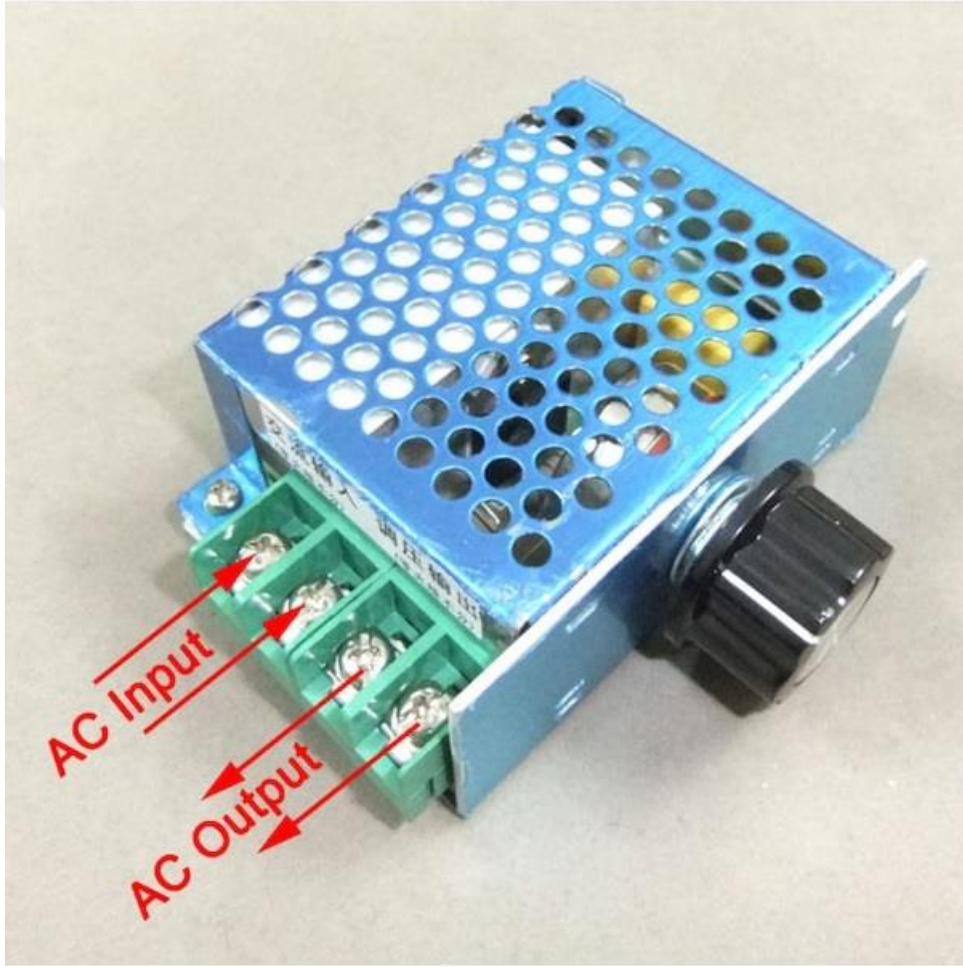
Fotoğraf 4.4. Wi-Fi Erişim Noktası(Access Point).

Projede, NodeMCU, IP kamera ve mobil erişim noktası (Hot Spot) olarak kullanılan cihazlar arasında bir ağ oluşturularak cep telefonu üzerinden internet erişimi sağlanmıştır. Uygulama sahasında sabit internet erişimi bulunmadığı için ve

genellikle uygulama yapılacak çoğu sahada benzer şartlar geçerli olduğundan, internet erişimi mobil olarak sağlanmıştır.

4.5 AC Motor Devir Kontrol Ünitesi

Derin öğrenme algoritmalarının görüntü segmentasyonu sonucunda ürettiği sulama çapı veya su basıncı haritası bilgisini uygulama sahasında pratikte kullanabilmek için sulama başlıklarına giden su basıncının bu bilgiyle ilişkili olarak değiştirilmesi gerekir. Su basıncı ve debisinin kontrolü, AC pompanın devir hızı değiştirilerek sağlanmaktadır. Bu amaçla AC motor devir kontrol ünitesine ihtiyaç duyulmuştur. Su başlıklarının sulama çapı, başlığa uygulanan su basıncı ve debisiyle doğru orantılı olarak değişmektedir.



Fotoğraf 4.5. AC asenkron motor devir hız control ünitesi.

Asenkron AC motorların devir hızı iki parametre ile değişmektedir. Bunlar uygulanan AC akımın frekansı ve motorun imal edilirken statorundaki kutup sayısıdır. Kutup sayısı üretim aşamasından sonra değiştirilemeyeceği için motor hızını değiştirmekte kullanılmak üzere geriye tek parametre olarak AC akım frekansı kalmaktadır. Çok pahalı ve tam profesyonel sistemlerde AC akım önce DC akıma çevrilir ve sonrasında tam dalga sinus inverter aracılığıyla geniş bir frekans aralığı yakalanır. Bu da çok geniş bir alanda devir hızı yakalama imkanı verir fakat

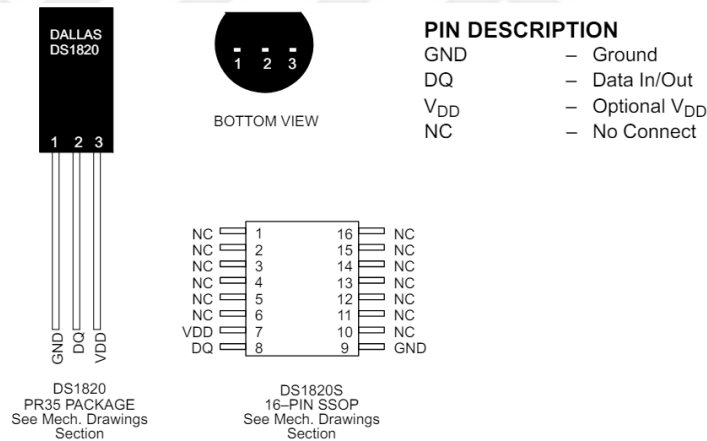
tüm bu aşamalar güç elektroniği kapsamında yüksek gerilim ve akıma dayanıklı elektronik komponentlere ihtiyaç duyar. Bu kapasitede komponent üretmek hem zor hem de oldukça maliyetlidir. Bu duruma alternatif olarak, 0 devir/dakika ile 50 Hz şebeke akımı ve motor kutup sayısının değeri ile ulaşılan maksimum devir hızı arasında değişebilen bir kontrol aralığı triyak tetikleme ile yakalanabilir. Birinci yöntemde maksimum motor devrinin üzerine çıkılabilirken ikinci yöntemde sınır değeri maksimum motor devridir. Uygulamada ikinci yöntem tercih edilmiştir ve görseldeki devir control ünitesi ile ihtiyaç duyulan devir hızı aralığı yakalanmıştır.

4.6 Sensör ve Aktüatörler

Projede kullanılan sensörler, aktüatörler ve diğer ekipmanlar ile kullanım amaçları aşağıdaki alt başlıklar altında verilmiştir:

4.6.1 DS1820 Sıcaklık Sensörü

Uygulama sahasında ortam sıcaklığını ölçebilmek ve bu değerlerden yola çıkarak segmentasyon sonucu sulanması gereken bölgelerin değişken hava koşullarına bağlı su ihtiyacını öngörmeye kullanılmak üzere DS1820 sıcaklık sensörü kullanılmıştır.



Şekil 4.4. DS1820 sıcaklık sensörü.

DS1820 sıcaklık sensörü, 9 bit derinliğinde sıcaklık okuması sağlar. Tek bir kablo üzerinden çift yönlü veri aktarımı yapabilen 1-Wire bağlantısıyla kullanılır. Ayrıca, sensörün güç ihtiyacı da yine bu kablo üzerinden sağlanır. DS1820'nin her biri benzersiz bir adres numarasına sahip olduğu için aynı hat üzerinde birden fazla sensörün kullanımına olanak tanır. Ölçüm aralığı 0.5°C aralıkla -55°C ile +125°C arasındadır.

4.6.2 Basınç Şalteri

Basınç şalteri, bağlı bulunduğu sistemde sıvı veya gaz basıncının belirlenen alt veya üst limitlere ulaştığında içerisinde bulunan elektrik kontaklarını açarak yada kapatarak bir sistemi etkinleştiren veya enerjisini keserek faaliyetini sonlandıran mekanik bir elektrik devresidir.

Proje kapsamında, Derin Öğrenme algoritmalarının yönlendirmesiyle MFECB nin aktif hale getirdiği sulama başlıklarının anlık su basıncı düşmesine sebep sonucu tetiklenerek AC su pompasını çalıştırmak veya tersi durumda, sulamaya ihtiyaç duyulmadığında aktif sulama başlıklarının kapatılmasıyla anlık su basıncı yükselmesi sonucu AC pompanın enerjisini kesmek için kullanılmıştır.



Fotoğraf 4.6. Sıvı akışkan basınç şalteri.

4.6.3 Selenoid Valf ve Sprinkler(Sulama Başlığı)

Otomasyon ve endüstriyel sistemlerde geniş bir kullanıma sahip selenoid valfler, akışkanların akışının başlatılması veya durdurulması için kullanılır. Tipik olarak, selenoid valfler bir bobin içerir ve bobine bir elektrik akımı uygulandığında, elektromıknatıs devreye girer ve valf içindeki bir armatürü hareket ettirerek akışkanın yolunun açılmasını sağlayarak akışı başlatır. Elektrik akımı kesildiğinde ise akışkanın yolunu kapatarak akışı durdurur. Normalde açık veya normalde kapalı olmak üzere iki türü vardır. Uygulama sahasında kullanılan selenoid valfler, 24V AC akımla çalışan normalde kapalı türden olan valflerdir.



Fotoğraf 4.7. Selenoid valf ve Sprinkler(Sulama Başlığı)

Sprinkler, yağmurlama sulama veya yangın söndürme sistemlerinde bir boru ağı üzerinde yer alan ve suyu homojen olarak atmosfere püskürten ünitelerdir. Döner başlıklı veya sabit başlıklı olarak üretilirler. Sabit başlıklı olanlar da, suyun basıncı ile ters kuvvette olan bir yayın gerilme kuvveti yenilerek başlık uzar ve suyu püskürtmeye başlar. Su kesildiğinde ise yay uzayan kısmı tekrar geri çekerek sprinklerin toprak altında gizli kalmasını sağlar. Uygulama sahasında sabit başlıklı minimum 2 bar su basıncı ile uzayan sprinkler kullanılmıştır.

4.7 AC Jet Su Pompası

AC su pompasına sulama için kullanılacak suyun, su deposundan çekilerek MFECB tarafından kontrol edilen değişken basınç ve debide sisteme akışını tedarik etmek için ihtiyaç duyulmuştur. Uygulama sahasında 1 HP 220V AC asenkron su pompası talep edilen ihtiyacı karşıladığından, bu özelliklere sahip bir pompa tercih edilerek kullanılmıştır.



Fotoğraf 4.8. Asenkron 1 Hp 220V AC jet su pompası.

4.8 Yazılım ve Yazılım Geliştirme Araçları

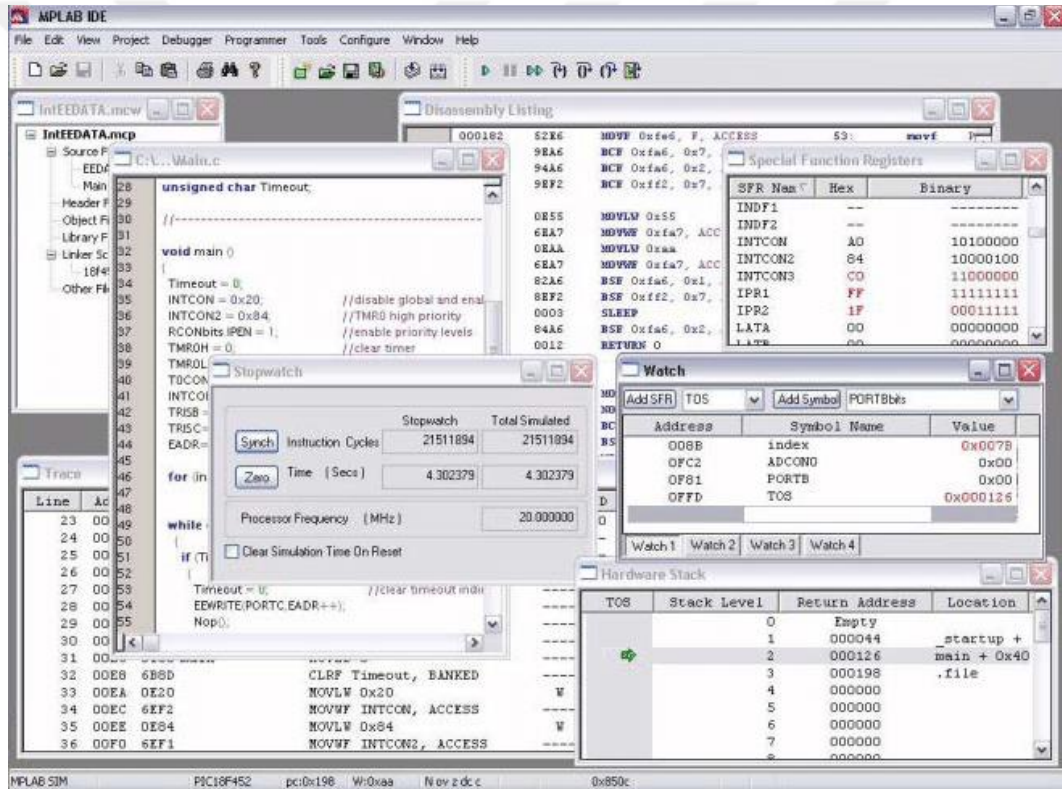
Günümüz teknolojisinde donanımları yazılımdan bağımsız düşünmek artık imkansızdır. En küçük donanım dahi bir yazılım içermekte ve bu yazılım ile işbirliği içinde olmak zorundadır. Yazılımlar, proje hayata geçirilirken karşımıza geliştirme aracı olarak kolaylık sunmak amacıyla çıkmakla birlikte projenin her aşamasında farklı bir yazılım çözümü de yer alabilmektedir. Ya yardımcı unsur olarak ya da ana unsur olarak proje katmanlarında var olmaktadır. Donanımlar üzerinde çalışan çok sayıda yazılım geliştirilmiştir. Donanımların işlevselliğini

artıran ve projeyi daha verimli hale getiren bu yazılımlar, kullanım amaçları ve kullanılan geliştirme ortamları bu başlık altında toplanmıştır.

4.8.1 MPLAB Yazılım Geliştirme Ortamı

MPLAB Entegre Geliştirme Ortamı (IDE), PIC mikrodeneleyiciler için genişletilebilir ve yüksek derecede yapılandırılabilir bir yazılım geliştirme ortamıdır. Bu program, Microchip firmasının üretimini yaptığı çoğu mikrodeneleyici ve dijital sinyal denetleyici için gömülü tasarımlar geliştirmeye, hata ayıklamaya ve nitelendirmeye yardımcı olacak güçlü araçları içermektedir.

MFECB kartının ihtiyaç duyduğu ve üzerinde çalışan gömülü sistem yazılımları, MPLAB yazılım geliştirme ve hata ayıklama ortamı kullanılarak C dilinde yazılmıştır.

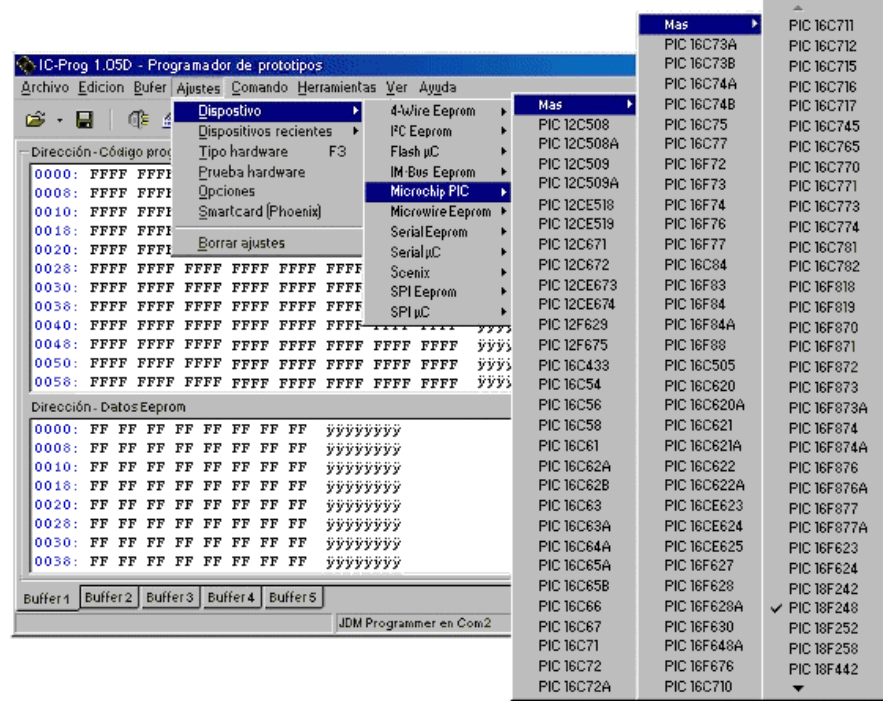


Fotoğraf 4.9. Microchip MPLAB yazılım geliştirme ortamı.

4.8.2 IC-Prog Programlayıcı

JDM, "Johnson Device Microcontroller" (Johnson Cihaz Mikrodeneleyici) uyumlu seri programlayıcılar üzerinden entegre çiplere veya mikrodeneleyicilere yazılım yüklemek ve okumak için kullanılan bir programdır. JDM programlayıcılar çoğunlukla PIC mikrodeneleyicilerin programlanması için tercih edilir. Geliştirilen MFECB kartta yerleşik olarak bulunan programlayıcı JDM tabanlı bir programlayıcıdır. MPLAB yazılım geliştirme ortamında yazılan ve HEX formatında

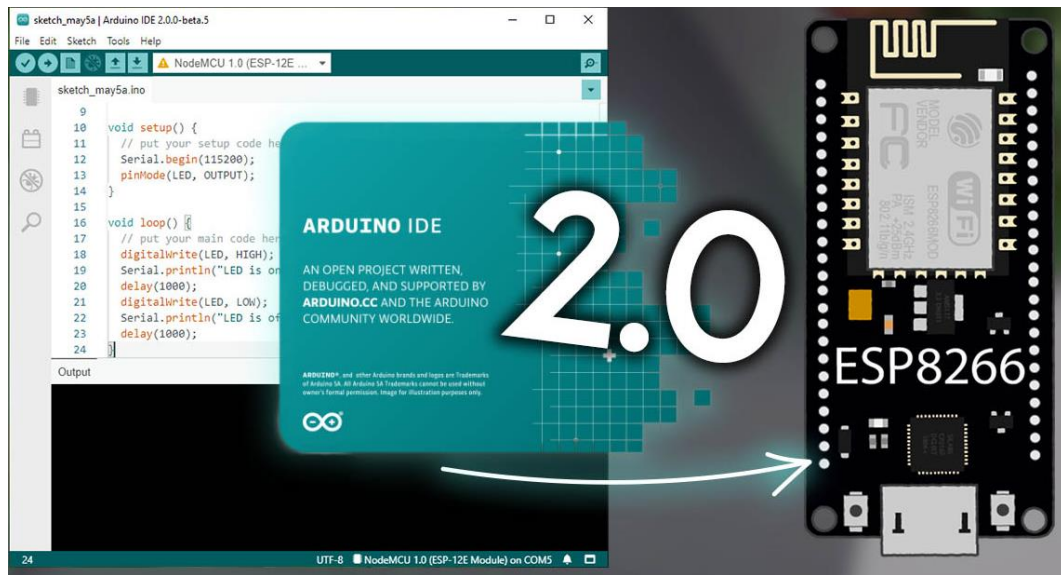
kodu üretilen MFECB gömülü sistem yazılımı IC-Prog yazılımı aracılığı ile mikrodenetleyiciye yüklenmektedir.



Fotoğraf 4.10. JDM uyumlu programlayıcılar için IC-Prog yazılımı.

4.8.3 Arduino Yazılım Geliştirme Ortamı

Projede Arduino modülü kullanılmadı, ancak NodeMCU kullanılmıştır. NodeMCU modülüne yönelik yazılım geliştirme aracı olarak LUA script dili veya ilgili kütüphanelerin kurulmasıyla Arduino IDE, NodeMCU için yazılım geliştirme aracı olarak kullanılabilir. Bu modül için zaman zaman Arduino IDE, zaman zaman da VSCode eklentisi olan PlatformIO tercih edilmiştir.

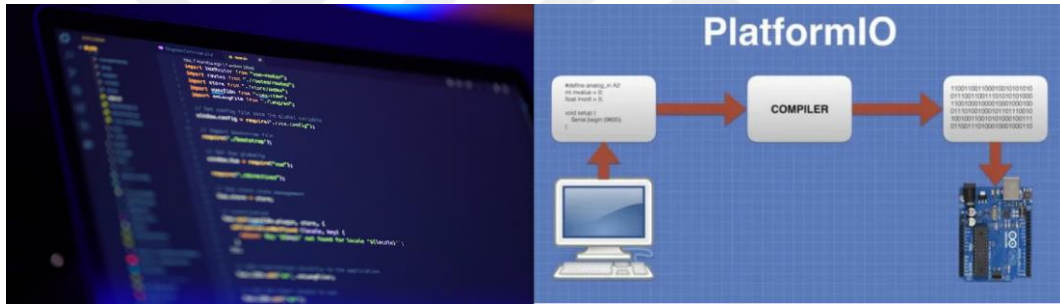


Fotoğraf 4.11. NodeMCU için Arduino IDE yazılım geliştirme ortamı.

4.8.4 VS Code ve PlatformIO

Visual Studio Code, Microsoft tarafından açık kaynak olarak geliştirilen ve yazılım geliştiricilerin kullanımına sunulan ücretsiz bir kod düzenleyicidir. Windows, macOS ve Linux gibi farklı işletim sistemlerinde çalışabilmektedir. Yerleşik olarak JavaScript, TypeScript ve Node.js desteğiyle gelir, fakat eklentiler ile C++, C#, Java, Python, PHP, Go ve .NET gibi çok daha fazla dili aynı ortamda kullanabilme imkanı sunar. Projede Visual Studio Code'dan Python kodlarının yazılması ve debug edilmesinde, ayrıca çok daha önemlisi PlatformIO eklentisi sayesinde mikrodenetleyici ve gömülü sistemlere yönelik kod yazılmasında faydalanılmıştır.

PlatformIO, odak noktası gömülü sistemler ve IoT uygulamaları geliştirmek olan, açık kaynaklı ve çapraz platform desteğine sahip bir yazılım geliştirme ortamıdır. VS Code ve çok sayıda popüler kod düzenleyiciye kolayca entegre olabilmektedir. PlatformIO, C, C++, Python ve Assembly gibi birçok programlama dilini desteklemesiyle birlikte Arduino, ESP8266, ESP32, STM32, AVR, ARM ve daha birçok platform ve kart için geniş donanım desteği sunar. Bu da çok çeşitli donanım projelerinde PlatformIO'nun tercih edilmesine neden olmaktadır.



Fotoğraf 4.12. VS Code ve PlatformIO eklentisi.

4.8.5 Python Programlama Dili

Python, 1990 yılında Guido van Rossum tarafından geliştirilmiş ve ilk kez 1991 yılında piyasaya sürülmüştür. Kullanımı kolay ve geniş bir kullanıcı kitlesine hitap eden, açık kaynaklı, nesne yönelimli, yüksek seviyeli bir programlama dilidir. web geliştirme, veri analizi, makine öğrenimi, bilimsel hesaplama, otomasyon, oyun geliştirme, akademik ve bilimsel çalışmalarda, özellikle yapay zeka çalışmalarında, geniş kütüphane olanakları ve katılımcı desteği ile öne çıkmakta ve bu yüzden de yaygın olarak kullanılmaktadır. Proje kapsamında, merkezi iş istasyonunda çalışan derin öğrenme algoritmalarının kodlarında bu destek ve imkandan faydalanılmıştır. Görüntü segmentasyonunda PixelLib kütüphanesi, görüntü yakalama ve segmentasyon verisi olarak hazırlamak için ise OpenCV kütüphanesi kullanılmıştır. Diğer verileri işleme ve öğrenmede ise Keras kütüphanesinden faydalanarak CNN (Evrişimli Sinir Ağı) modeli tercih edilmiştir. Sıcaklık sensor verilerinde ise LSTM (Uzun Kısa Süreli Bellek Ağları) modeli tercih edilmiştir.

İlk olarak, sıcaklık verisi pandas kullanılarak Excel dosyasından okunur ve değerler numpy dizisine dönüştürülür. Veriler, MinMaxScaler kullanılarak normalize edilir ve bu işlem, verilerin 0 ile 1 arasında bir değere dönüştürülmesini sağlar. 'create_dataset' fonksiyonu ile veriler özellikler (X) ve etiketler (y) olarak ayrılır ve belirli bir sekans uzunluğunda kaydırılarak eğitim ve test verisi oluşturulur. Veriler train_test_split kullanılarak eğitim ve test verisi olarak ikiye ayrılır. Bir Sequential model tanımlanır ve içerisine 50 nöronlu bir LSTM katmanı eklenir. LSTM katmanının ardından, tek bir nöronlu bir Dense katman eklenir. Oluşturulan veri seti literatürde çok sık kullanılan hazır öğrenme yöntemlerinden adam optimizasyon algoritması kullanılarak optimize edilmiştir. Çıkan sonuçların doğruluğunu ölçmek için mean squared error (mse) kayıp fonksiyonu kullanılmıştır. Model, 100 epoch boyunca ve 32 batch size ile eğitilirken, verilerin %20'si doğrulama %80'i eğitim için kullanılır. Eğitilen model, test verisi üzerinde değerlendirilir ve mse kayıp değeri hesaplanır. Model, son 10 veri noktası kullanılarak gelecekteki bir sıcaklık değerini tahmin eder ve bu tahmin edilen değer, normalleştirilmiş halden orijinal değerine dönüştürülür. Kodda öne çıkan iki önemli bileşen vardır: LSTM modeli ve MinMaxScaler. LSTM modeli, zaman serisi verilerindeki ardışık bağımlılıkları öğrenmek için kullanılırken, MinMaxScaler verileri normalleştirerek modelin daha iyi performans göstermesini sağlar. Bu iki bileşen, sıcaklık tahmin modelinin etkinliğinde kilit rol oynamaktadır.

4.8.5.1 Pixellib Python Kütüphanesi

Pixellib, görüntü işleme, segmentasyon ve nesne tespiti için kullanılan bir Python kütüphanesidir. Özellikle, görüntülerde nesneleri tanımlamak, sınıflandırmak, izlemek ve nesnelerin maskelerini çıkarmak gibi işlemlerde kullanılır. Önceden eğitilmiş modelleri kullanarak hızlı ve etkili bir şekilde görüntü işleme işlemlerini gerçekleştirmeye olanak tanır. Ayrıca, özel eğitim verileri kullanarak özel modeller oluşturmaya da olanak sağlar. Görüntü işleme projelerinde Pixellib, güçlü bir araç olarak kullanılmaktadır.



Fotoğraf 4.13. PixelLib Ade20k modeli örnek girdi görsel.



Fotoğraf 4.14. PixelLib Ade20k modelin örnek segmentasyon çıktısı.



Fotoğraf 4.15. PixelLib Ade20k modelin semantic segmentasyon çıktısı.

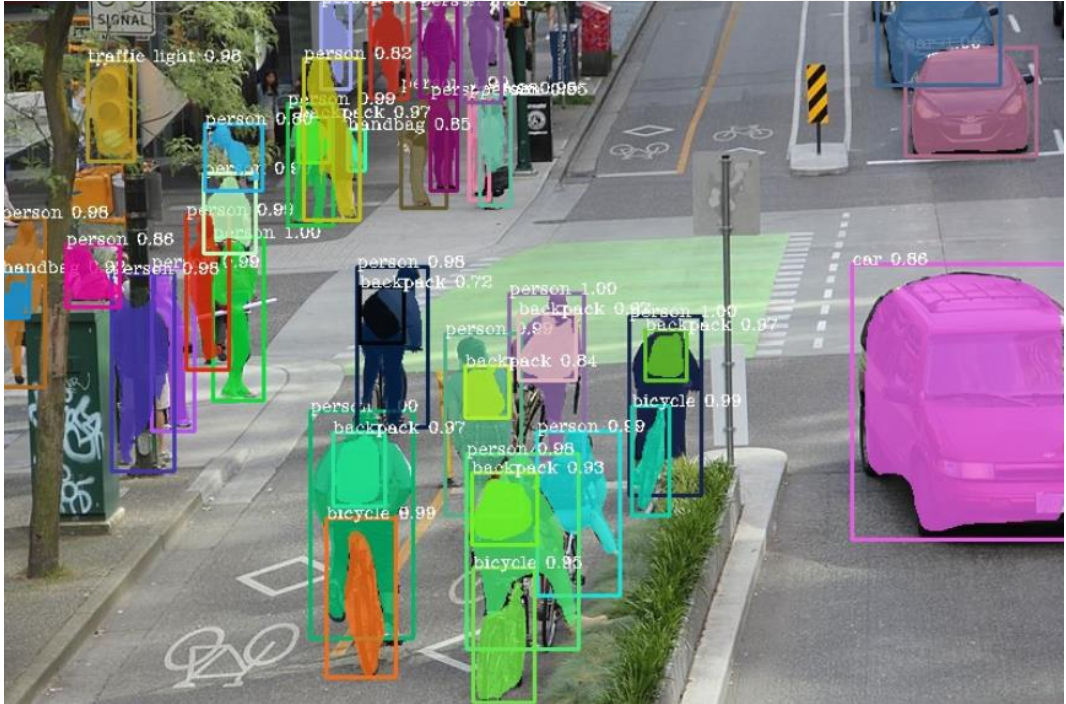
PixelLib kütüphanesi aynı zamanda videolardan veya akıcı görsellerden de anlık görüntü segmentasyon ve sınıflandırma yapabilmektedir.



Fotoğraf 4.16. PixelLib anlık segmentasyon ve sınıflandırma girdisi.



Fotoğraf 4.17. PixelLib anlık segmentasyon görsel çıktısı.



Fotoğraf 4.18. PixelLib anlık segmentasyon ve sınıflandırma çıktısı.

PixelLib önceden eğitilmiş bir çok görüntü ve video işleme modeline sahiptir. Bu yüzden diğer modellere değinilmemiş öne çıkan ve proje unsurunu oluşturan model ve özelliklerine yer verilmiştir.

4.8.5.2 OpenCV Python Kütüphanesi

OpenCV (Open Source Computer Vision Library), endüstride ve akademik çevrelerde yaygın olarak benimsenmiş açık kaynak görüntü işleme ve bilgisayarla görme uygulamaları için kullanılan yazılım kütüphanesidir. İlk olarak 1999 yılında Intel tarafından geliştirilmiş sonrasında ve günümüzde OpenCV.org tarafından geliştirilmeye devam etmektedir. OpenCV, görüntü ve video analizi, yüz tanıma, nesne tespiti ve izleme, hareket takibi, 3D modelleme, panorama oluşturma, ve makine öğrenmesi ile ilgili geniş kapsamlı araçlar sunar. Kütüphane, gerçek zamanlı görüntü işleme yeteneği sayesinde robotik, güvenlik sistemleri, tıbbi görüntüleme, artırılmış gerçeklik ve otonom araçlar gibi çeşitli uygulama alanlarında yaygın olarak kullanılmaktadır.

OpenCV, görüntülerin yüklenmesi, işlenmesi ve kaydedilmesi; görüntü boyutlandırma, döndürme, kırpma ve renk uzayı dönüşümleri gibi temel işlemlerin yanı sıra, gürültü azaltma ve görüntü iyileştirme araçları olarak çeşitli filtrelerle birlikte kenar algılama ve histogram işlemlerinin yapılmasına da imkan tanır.

Video dosyalarını okuma ve yazma işlemlerinin yanı sıra farklı video formatlarıyla çalışabilme ve kare kare işleme yapabilir. Her karede nesne tespiti, hareket analizi ve kareler arasındaki farkların hesaplanması gibi işlemler de gerçekleştirilebilir.

Cascade ve HOG (Histogram of Oriented Gradients) gibi algoritmalarla yüz tespiti ve tanıma işlemleri yapılabilir KCF, MedianFlow, TLD gibi çeşitli nesne izleme algoritmaları ile videolardaki nesnelerin hareketi izlenebilir. SIFT, SURF, ORB gibi algoritmalar ile görüntülerdeki belirgin özellik noktaları tespit edilebilir ve bu noktalar diğer görüntülerle eşleştirilebilir.

Sınıflandırma ve Kümeleme: SVM (Support Vector Machine), K-means, Random Forest gibi algoritmalarla görüntü sınıflandırma ve kümeleme işlemleri yapılabilir. TensorFlow, PyTorch gibi kütüphanelerle entegrasyon sağlayarak derin öğrenme modelleri eğitilebilir.

İki farklı perspektiften alınan görüntülerle derinlik haritaları oluşturulabilir ve 3D yapıları analiz edilebilir. Yapılandırılmış ışık ve hareketli kameralarla 3D modeller oluşturulabilir.

Gerçek dünya görüntülerine kameranın konumu ve yönelimi belirlenerek sanal nesnelerin gerçek dünyaya doğru şekilde yerleştirilmesi sağlanır.

4.8.5.3 Keras Python Kütüphanesi

Keras, Python ile yazılmış yüksek seviyeli bir sinir ağı API'sidir ve 2015 yılında Google mühendislerinden biri olan François Chollet tarafından geliştirilmiştir. TensorFlow, Theano ve Microsoft Cognitive Toolkit (CNTK) gibi güçlü arka plan uygulamalarıyla çalışabilen Keras, sinir ağı katmanlarını bir araya getirerek karmaşık modeller oluşturulmasına olanak tanır. Theano, hızlı sayısal

hesaplama görevleri için kullanılan bir Python kütüphanesidir. TensorFlow, sinir ağları ve derin öğrenme modelleri oluşturmak için kullanılan en ünlü sembolik matematik kütüphanesidir. CNTK, Microsoft tarafından derin öğrenme için geliştirilen bir framework'tür. Keras, görüntü tanıma, doğal dil işleme ve öneri sistemleri gibi çeşitli alanlarda yaygın olarak kullanılır ve derin öğrenme uygulamaları için optimal bir seçimdir. Genel bilgiler kısmında değinilen sınıflandırma ve derin öğrenme modellerinin tamamı Python arayüzü kullanılarak Keras kütüphanesi ile oluşturulabilir.

Keras temel bileşenleri:

- **Modeller:** Keras'taki ana yapı mimarisi olup, katmanları organize etmenin bir yoludur. En yaygın model türü, katmanların doğrusal bir yığını olan Sequential modelidir. Daha karmaşık mimariler için Keras Functional API'yi sunar.
- **Katmanlar:** Keras'taki sinir ağlarının temel yapı taşlarıdır. Keras, evrişimli, havuzlama (pooling), tekrarlayan (recurrent), gömme (embedding) ve yoğun (dense) katmanlar gibi geniş bir katman yelpazesi sahiptir.
- **Optimizatörler:** Optimizatörler arasında SGD, Adam ve RMSprop bulunur. Bu optimizatörler, kayıpları azaltmak için sinir ağlarının ağırlık ve öğrenme oranı gibi özniteliklerini değiştirmeye yönelik çeşitli yöntemler içerir.
- **Kayıp Fonksiyonları:** Beklenen çıktı ile model tarafından üretilen çıktı arasındaki farkı hesaplamak için kullanılır. Yaygın kayıp fonksiyonları arasında mean_squared_error, categorical_crossentropy ve binary_crossentropy Keras tarafından sağlanan başlıca kayıp fonksiyonlarıdır.
- **Metrikler:** Modelin performansını değerlendirmek için kullanılır. Yaygın metrikler arasında doğruluk (accuracy), kesinlik (precision) ve geri çağırma (recall) bulunur.

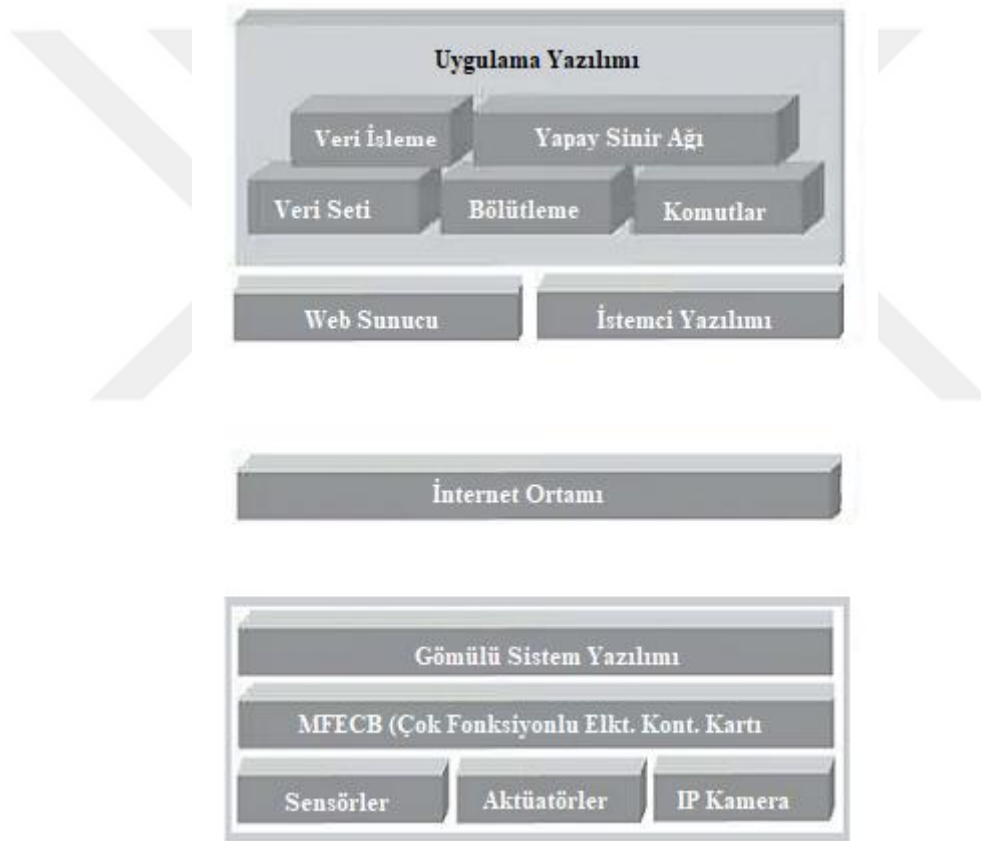
Bu bileşenler ile Keras sinir ağı modelleri oluşturmak ve optimize etmek için güçlü ve esnek bir araç olarak karşımıza çıkar.

4.9 Yöntem

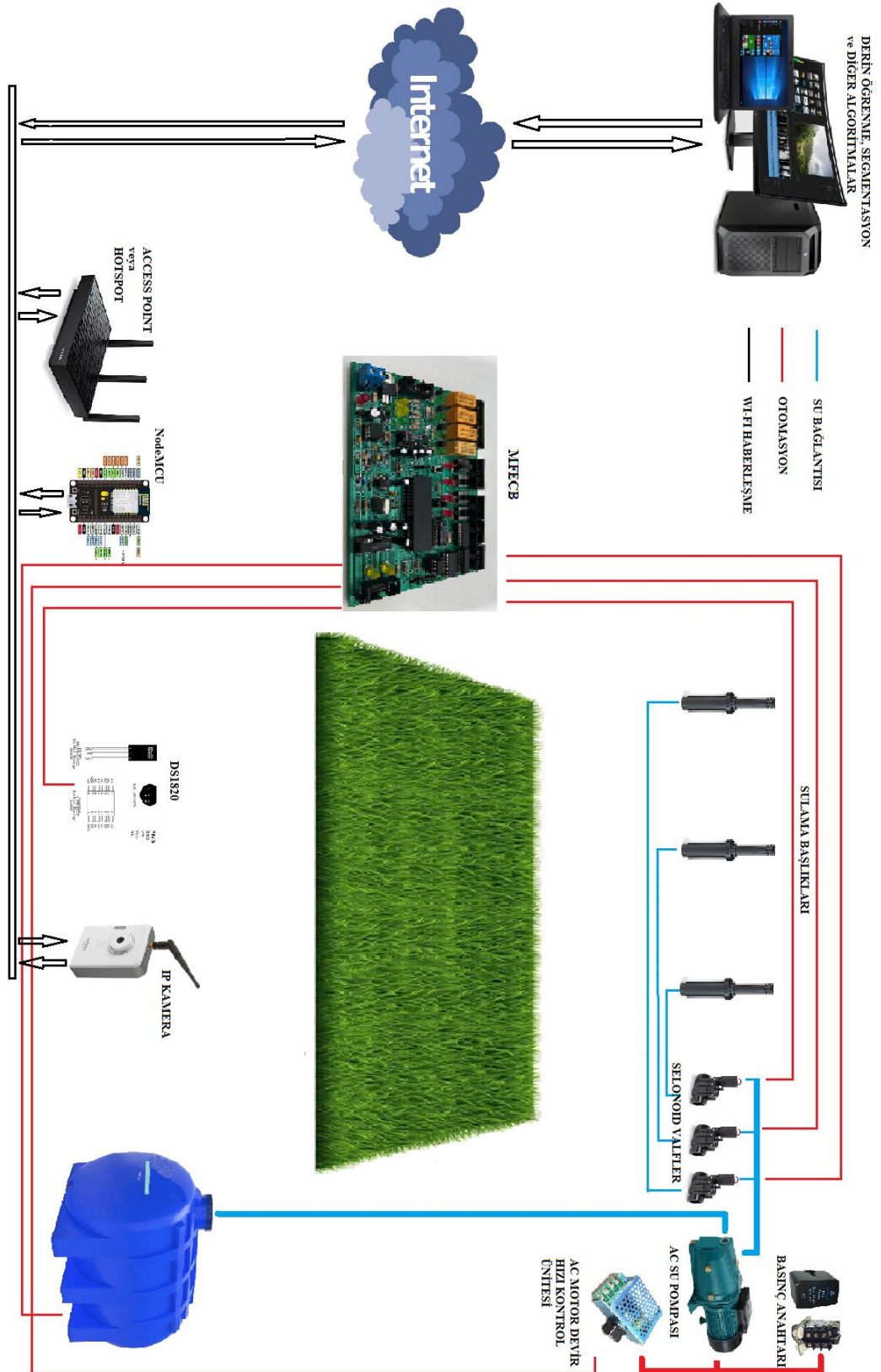
Sulama projesinin tamamı göz önünde bulundurulduğunda, sistem katmanları ve hiyerarşik yapılanması **Şekil 4.5.** de görülmektedir. Bu yapılanmada MFECB sahada merkezi görev birimi olarak konumlanmaktadır. Kart, çalışma alanında hizmet verirken ortam sıcaklığını ve ihtiyaç çeşitliliğine göre diğer sensör verilerini toplar ve ayrıca sulanan alanın görüntülerini periyodik olarak yakalayarak internet üzerinden bir iş istasyonunda çalışan derin öğrenme (DL) algoritmalarına iletir. Bu veri iletimi, istemci ve sunucu ilişkisi şeklinde aralıklı olarak gerçekleşir. Sahadan alınan kamera görüntülerinin OpenCV kütüphanesi ile veri hazırlama ve işleme süreci gerçekleştirilir sonrasında PixelLib kütüphanesi tarafından segmentasyon ve sınıflandırması yapılır. DL öğrenme modellerinin oluşturulmasında Keras

kütüphanesinden faydalanılmıştır. DL algoritmaları bu verileri ve görüntü segmentasyonunu kullanarak sulanacak alanları, sulama süresini ve çapını belirler. Hangi sulama başlığının etkin olacağı ve konumu bu işlemlerden sonra belirlenir. Sulama başlığı verileri, sulama çapı ve çalışma süresi bilgileri internet üzerinden MFECB'ye iletilir. Alınan bilgilere dayanarak, kart ilgili sulama başlığını belirtilen süre boyunca açar ve uygun sulama çapını gerçekleştirmek için AC su pompasının hızını sulama çapı bilgisine göre ayarlar. DL algoritmaları bu tekrarlanan süreçlerden öğrenme gerçekleştirir ve optimal bir sulama pratiğine ulaşırlar. DL algoritmaları, ya uzaktaki bir iş istasyonunda IoT üzerinden veya sulama uygulamasının yapılacağı bir PC de doğrudan bağlantı ile hizmet verebilirler.

Sulama sisteminin tamamının organizasyonu ve yapılanmasının bütününi içeren görsel şema, **Şekil 4.6.**'da gösterilmiştir.



Şekil 4.5. Sistem katmanları ve hiyerarşik yapılanması.



Şekil 4.6. Sulama sisteminin tam yapılanması ve organizasyon şeması.

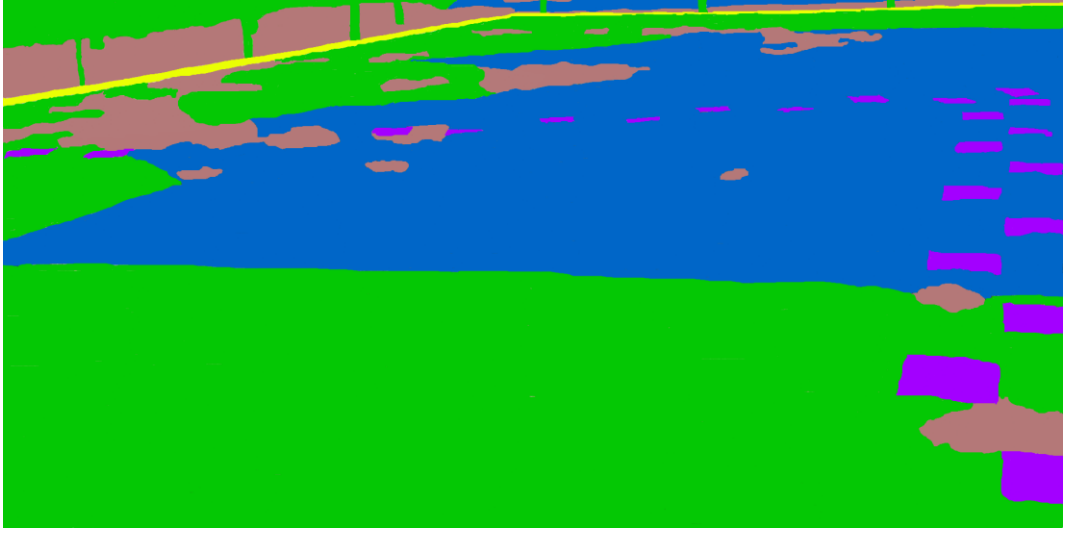
5. BULGULAR

Uygulama sahası olarak Ankara ili Gölbaşı ilçesi sınırları içerisindeki bir çiftlik evi seçilmiştir. Sulama yapılan alan, 240 m² çim alandan oluşmaktadır. Veriler, 2023 Nisan-Mayıs ayı ile 2024 Nisan-Mayıs ayları karşılaştırılarak incelenmiştir. 2023 yılında sulama sistemi, zamanlama saati ile belirlenen gün ve sürelerde zaman çizelgesini uygulamıştır. Bu yöntemde sulamanın başlama ve bitiş saati, aktif olacak sulama başlıklarının sayısı ve bu başlıklara ait sulama çapları sabittir. Bu sabit değerler günlük sabit bir su tüketimine denk düşmekte olup, hava sıcaklıkları artarsa bu zaman çizelgesi kullanıcı tarafından tahmini bir değerle güncellenmektedir. Bu durumda çim sahasının teorik olarak eşit sulanması sağlanmakta, fakat su kıtlığı çeken bölgelerin oluşmasına da neden olmaktadır. Bu etkiyi, verilen görsellerde de görmek mümkündür.

Çalışma konusu olan “derin öğrenme tabanlı geniş yüzey alanlarının sulanması” yönteminde, çim sahasının kamera görüntüsü işlenerek sahanın segmentasyonu oluşturulmuş ve segmente edilen bölgelerin derin öğrenme algoritmaları tarafından ihtiyaç duydukları su miktarı öngörülmüştür. Bu sayede tüm yüzeye homojen bir şekilde su vermek yerine, heterojen bir sulama yapılmıştır. İhtiyaç duyulan sulama başlıkları aktif edilerek, değişken sulama çapı ve sulama süresi ile heterojenlik sağlanmıştır. Bir önceki gün yapılan sulama ile bir sonraki gün yapılan sulama sabit kalmamış ve algoritmalar parametreleri günlük ihtiyaca göre değiştirdiği için zaman çizelgesi yönteminde olduğu gibi çizelgeyi tahmini bir değer ile güncelleme ihtiyacı doğmamıştır. Yine saha görsellerinden de görüleceği üzere, çim sahasında homojen bir bitki gelişimi yakalanmış ve su kıtlığı yaşayan bölgeler ortadan kalkmıştır.



Fotoğraf 5.1. Uygulama sahasına ait 02 Nisan 2024 tarihli görsel.

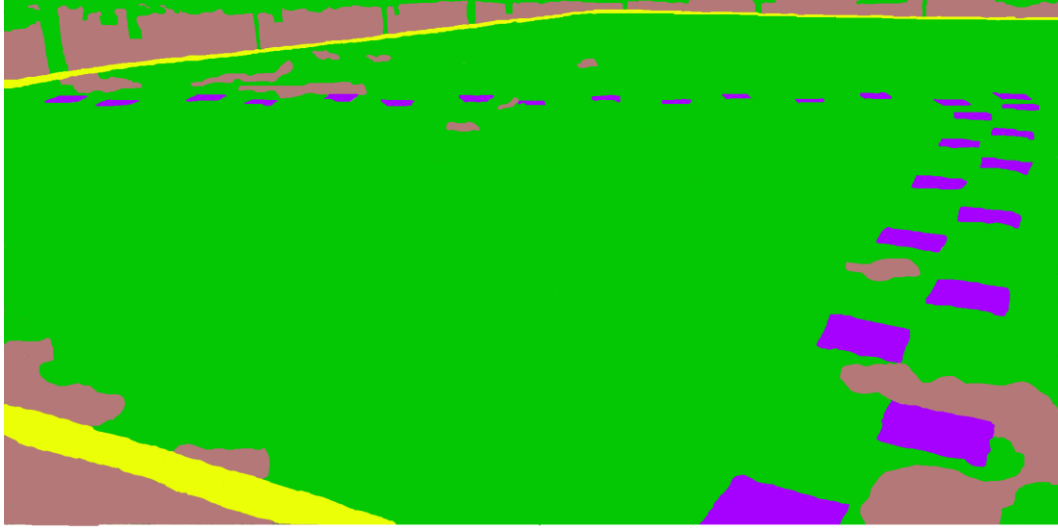


Fotoğraf 5.2. Uygulama sahası 02 Nisan 2024 segmentasyon haritası.

Fotoğraf 5.2. deki segmentasyon haritasında, 2023 yılında zaman çizelgeli yöntem ile yapılan sulama sonucunda su kıtlığı yaşayan bölgeler kahverengi olarak işaretlenmiştir. Gölge alanlar ise mavi renk ile işaretlenmiştir. Homojen sulamanın gölge alanlarda yeterli olduğu, fakat daha fazla güneşe maruz kalan bölgelerde yeterli olmadığı ve buna bağlı olarak çim kaplı alanlarda yer yer kurumaların olduğu gözlenmektedir. Gölge alanlarının gün içerisinde güneşin pozisyonuna bağlı olarak hareket halinde olması da homojen sulamayı yetersiz kılmaktadır.



Fotoğraf 5.3. Uygulama sahasına ait 28 Nisan 2024 tarihli görsel.

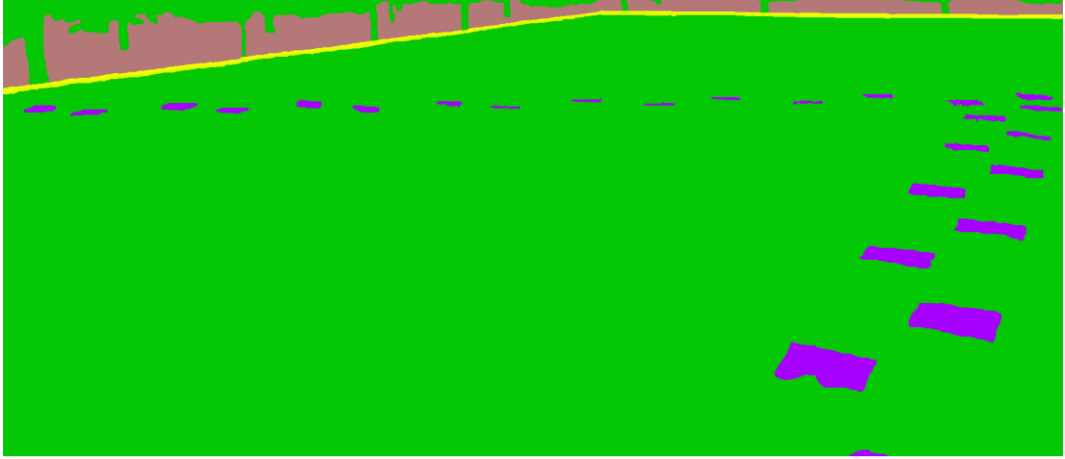


Fotoğraf 5.4. Uygulama sahası 28 Nisan 2024 segmentasyon haritası.

Fotoğraf 5.4. deki 28 Nisan tarihli segmentasyon haritasından da görüleceği üzere, derin öğrenme tabanlı geniş yüzey alanlarının sulanması yöntemi ile sulaması yapılan uygulama sahasında, 26 gün sonra çim alanındaki kurumaların yavaş yavaş küçülmeye başladığı ve diğer bölgelerin canlılık kazandığı gözlenmektedir.



Fotoğraf 5.5. Uygulama sahasına ait 19 Mayıs 2024 tarihli görsel.



Fotoğraf 5.6. Uygulama sahası 19 Mayıs 2024 segmentasyon haritası.

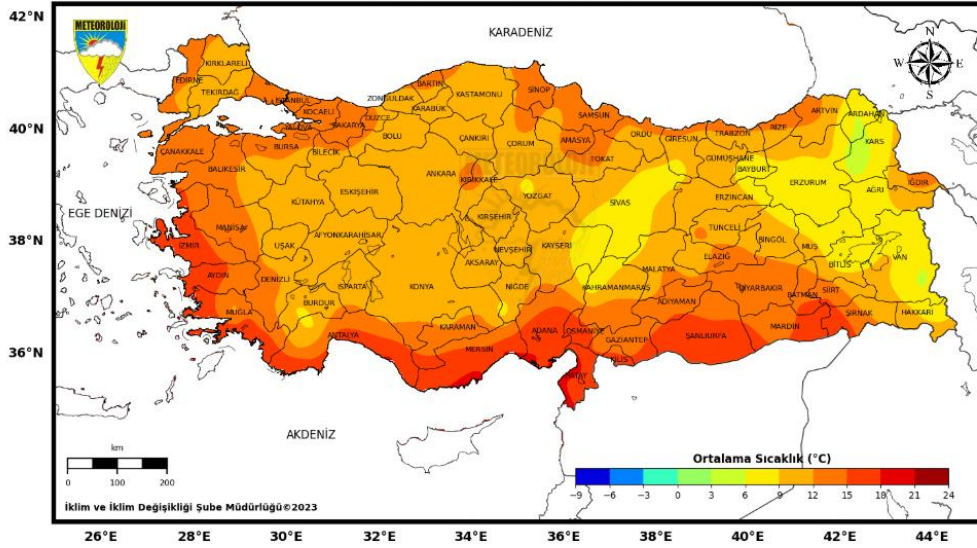
Fotoğraf 5.5. deki 19 Mayıs tarihli uygulama sahasına ait görselde, uygulamanın başlangıcından itibaren geçen 47 günlük süre sonunda tam bir homojen gelişim sağlandığı ve optimal seviyeye ulaşıldığı gözlenmektedir.



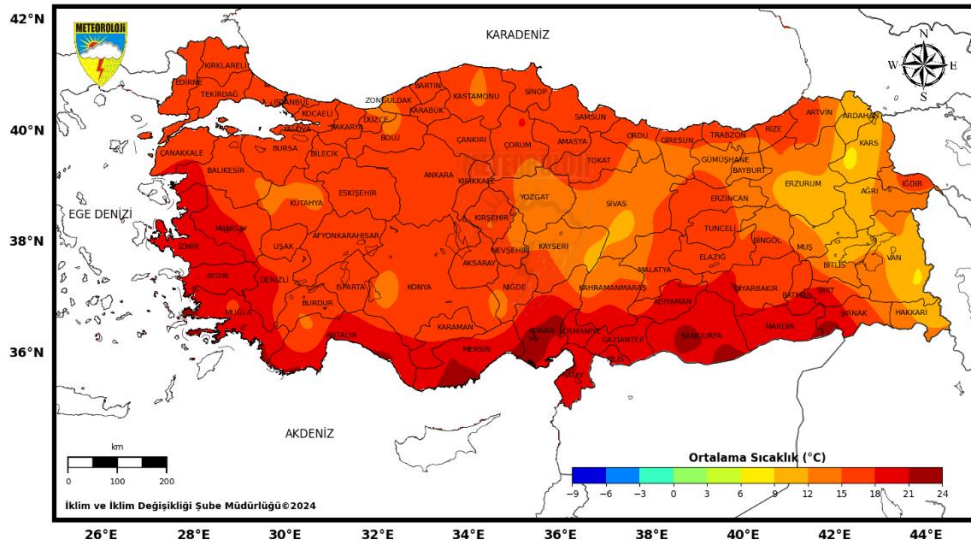
Fotoğraf 5.7. Sitemin çalışma süresi içindeki bir zaman dilimine ait görsel.

Fotoğraf 5.7.'de, mevcut 22 adet sulama başlığından o an için aktif olan 4 adet başlık ve sulama çapları, fotoğraf çekimi sırasında fotoğrafa yansıdığı kadarıyla aktarılmaya çalışılmıştır.

Su tüketiminde önemli faktörlerden biri de mevsimsel sıcaklık değerleridir. Ne yazık ki, uygulama sahasının 2023 yılı Nisan-Mayıs aylarına ait günlük ortalama sıcaklık verisi kayıtları bulunmadığından, 2024 sıcaklık verileri ile kıyaslama yapılamamıştır. Bu nedenle, bu faktörü de göz önünde bulundurmak amacıyla Meteoroloji Genel Müdürlüğü verilerine başvurulmuştur. Sıcaklık faktörünün sonuca olan etkisinin değerlendirilmesinde bu veriler kullanılmıştır. Sulama pratiğinin yapıldığı aylara ait ortalama sıcaklık haritaları şu şekildedir:

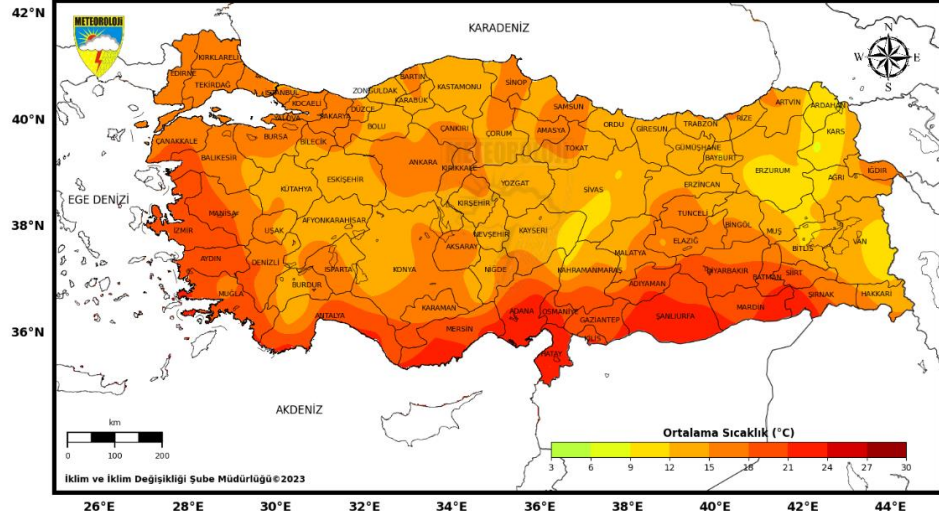


Şekil 5.1. 2023 Nisan ayı sıcaklık ortalaması haritası.

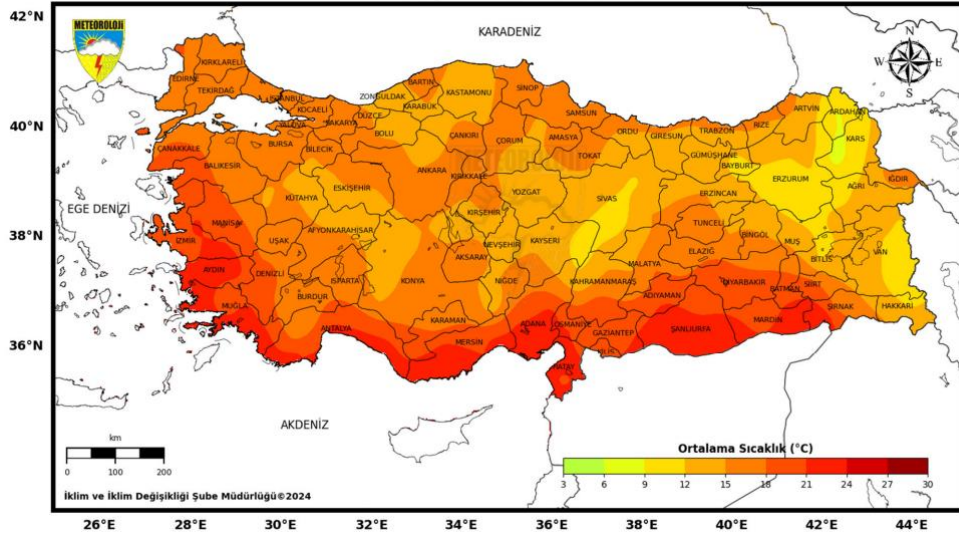


Şekil 5.2. 2024 Nisan ayı sıcaklık ortalaması haritası.

İki harita kıyaslandığında, Ankara'da 5°C'lik sıcaklık artışı olduğu görülmektedir.



Şekil 5.3. 2023 Mayıs ayı sıcaklık ortalaması haritası.



Şekil 5.4. 2024 Mayıs ayı sıcaklık ortalaması haritası.

Su tüketimi verileri açısından, 2024 yılı Nisan-Mayıs aylarına ait ayrıntılı günlük tüketim tablosunun oluşturulabilmesi için sisteme ek ölçüm ve kayıt cihazlarının eklenmesi gerekmektedir. Ancak, 2023 yılı Nisan-Mayıs aylarına ait veriler mevcut olmadığından kıyaslama yapılamayacaktı. Ayrıca, sistemin başarısını değerlendirmede bu kadar ayrıntılı bir veriye ihtiyaç duyulmadığı için her iki yıla ait su tüketimi verisi günlük yerine aylık bazda sisteme bağlı su sayaçlarından okunarak elde edilmiştir.

Tablo 5.1. Aylık bazda su tüketimi (m³/Ay).

	2023			2024		
Aylar	Ortalama Sıcaklık	Sulama Süresi	Tüketim	Ortalama Sıcaklık	Sulama Süresi	Tüketim
Nisan	11.3°C	210 dk	14.7 m ³	16.6°C	250 dk	15 m ³
Mayıs	16.1°C	300 dk	21 m ³			

Tablo 5.1. deki veriler incelendiğinde, Nisan 2023 ayında zaman çizelgeli sistem ay boyunca 210 dakika sulama yapmış ve bu sürede 14.7 m³ su tüketimi olmuştur. Nisan 2024'te ise aylık ortalama sıcaklık 5.3°C artış göstererek 2023 Mayıs ayı şartlarına yaklaşmıştır. Buna rağmen, derin öğrenme tabanlı sistem 250 dakika sulama yaparak 14 m³ su tüketmiştir. Benzer bir sıcaklık artışı, 4.8°C ile 2023 Nisan-Mayıs ayları arasında yaşanmış ve bu dönemde zaman çizelgeli sistemin çalışma süresi 300 dakikaya çıkmış ve su tüketimi 21 m³ olmuştur.

Bu veriler referans alındığında, 2024 Nisan ayında en az 300 dakika sulama süresi ve 21 m³ su tüketimi beklenmekteydi. Oysa, bu ayda derin öğrenme tabanlı sistem 250 dakika sulama yaparak 15 m³ su tüketmiştir. Bu durumda, Nisan 2024 ayı için derin öğrenme tabanlı sistemin en az 50 dakika sulama süresinden ve 6 m³ sudan tasarruf sağladığı sonucuna varılabilir. Mevsimsel sıcaklık faktörlerinin olumsuz yönde gelişmesine rağmen, derin öğrenme tabanlı sistemin yaklaşık olarak en az %28.5 su tasarrufu sağlandığını söyleyebiliriz.

6. SONUÇ VE ÖNERİLER

Günümüzde ve yakın gelecekte temiz su kaynaklarına erişim ve bu kaynakların verimli kullanımı konusundaki stratejiler geliştirme ihtiyacı uzun süredir hissedilmektedir. Bu çalışma kapsamında, su tüketiminin en fazla olduğu alanlardan biri olan geniş yüzey alanlarının sulanması konusu ele alınmıştır. Bu alanlardaki tüketimi optimize ve verimli hale getirmeye katkı sağlayacağı öngörülen derin öğrenme tabanlı bir sistemin çözüm olup olmayacağı araştırılmış ve değerlendirilmiştir.

Geniş yüzey alanlarından kastedilen, bina ve kent peyzajında, park ve yeşil alanlarda, golf ve futbol sahalarında karşımıza çıkan çim sahalarıdır. Çim, canlılığını koruyabilmek için doğası gereği çok fazla suya ihtiyaç duyar. Bu faktöre bilinçsiz sulama da eklendiğinde, bu alanlardaki su tüketimi göz ardı edilemeyecek boyutlara ulaşmaktadır.

Bu çalışmada, geniş yüzey alanlarının sulanmasında kullanılan klasik yöntem olan zaman çizelgeli sabit menzilli sulama başlıkları ile yapılan homojen sulama, bu amaç için geliştirilen ve özelleştirilen çok fonksiyonlu elektronik kart ve derin öğrenme tabanlı heterojen sulama yapan sistem ile karşılaştırılmıştır. Değerlendirme sonucunda, derin öğrenme tabanlı sistemin bitkinin suya bağlı gelişimini sağlıklı bir şekilde sürdürdüğü ve hem sulama süresinden hem de su tüketiminden tasarruf sağladığı, bu tasarrufun da en az %28.5 olduğu sonucuna varılmıştır.

Derin öğrenme tabanlı sistem, ortam sıcaklık verisini ve sulanacak yüzeyin canlı kamera görüntülerini girdi olarak almakta ve görüntü segmentasyonu yaparak değişken sulama çapı ve su yoğunluğu ile sulama işlemini gerçekleştirmektedir. Aynı zamanda, oluşan gölge alanlar ve güneşe daha uzun süre maruz kalan alanları da ayırt ederek bu faktörü de göz önünde bulundurmaktadır.

1. Derin öğrenme tabanlı sistemin başarılı olduğu gözlemlenmiş, hedeflenen sağlıklı bitki gelişimini yakaladığı ve hem enerjiden hem de su tüketiminden tasarruf sağladığı görülmüştür.
2. Farklı hava koşulları, sıcaklık dalgalanmaları ve ışık değişimlerinin neden olduğu sonuçları otomatik olarak analiz ederek uyumlanmış sulama işlemini optimize etmiştir.

3. Geliştirme ve iyileştirmelere açık olup, daha başarılı sonuçların alınması mümkündür.

6.1 Öneriler ve Gelecek Çalışmalar

1. Yüksek seviyeli programlama dilleri ile yazılan derin öğrenme algoritmaları, daha düşük seviyeli diller kullanılarak gömülü sisteme uyarlanabilir ve MFECB kartı üzerinde çalışır hale getirilebilir. Bu sayede, sistemin dağınık ve karmaşık yapısı ortadan kalkar ve tak-çalıştır bir cihaz haline dönüşebilir. Bu yaklaşım, sistemin performansını artırarak daha hızlı ve verimli çalışmasını sağlar ve aynı zamanda kullanıcılar için daha kolay bir kullanım sunar. Ayrıca, daha düşük seviyeli dillerin kullanımı, sistemin daha az kaynak tüketmesi ve daha stabil çalışması anlamına gelir.
2. Bu çalışmada, çim saha görüntüsü ve ortam sıcaklık değeri günde bir kez alınmış ve algoritmalar bu verilerle işlem yapmıştır. Sulama işlemi ise buharlaşmayı minimuma indirmek için gece saatlerinde gerçekleştirilmiştir. Günün aydınlık olduğu saatlerde, bir kez veri toplama yerine o güne ait çok fazla kez veri toplanarak güneşin hareketine bağlı olarak değişen gölge alanların izlediği rota ve alanlar öğrenilmeye dahil edilmiş olur. Gölge alanlarından sağlanan su tasarrufu çok daha yüksek seviyelere çıkarılabilir.
3. Gün içerisinde daha sık okunan ortam sıcaklık verileriyle daha sağlıklı bir çarpan elde edilebilir ve sulama işleminin başlama saati, daha optimal bir saatte sistem tarafından belirlenebilir.
4. Sisteme eklenecek nem sensörü verisi ve bu verinin gün içindeki dağılımından elde edilecek bilgilerle ikinci bir çarpan bilgisi elde edilerek daha fazla su ve zaman tasarrufu sağlanabilir. Bu sayede sistemin başarısı daha da ileri bir noktaya taşınabilir.

7. KAYNAKLAR

(Bu tez çalışmasında APA atıf sistemi kullanılmıştır)

- A.P. Dempster, N. L. (1977). Maksimum likelihood from incomplete data via the Em algorithm. *J. Royal Statiscal*, 1-38.
- Albawi, S. M.-Z. (2017). Understanding of a convolutional neural network. *international conference on engineering and technology (ICET)*, 1-6.
- Aleksandr D.Knyž. (1999). *Ibn 'Arabi in the Later Islamic Tradition: The Making of a Polemical Image in Medieval Islam*. SUNY Press.
- Al-Jazari. (2015). *Al-Jazari*. Encyclopædia Britannica: <http://www.britannica.com/biography/al-Jazari> adresinden alındı
- Allen, J. F. (2003). Natural language processing. *Encyclopedia of computer science*, 1218-1222.
- Allen, R. G. (1998). Crop evapotranspiration-Guidelines for computing crop water requirements. *FAO Irrigation and drainage paper*, D05109.
- B Boser Le Cun, J. S. (1990). Handwritten digit recognition with a back-propagation network. *Advances in neural information processing systems*.
- Bau, D. Z. (2018). Gan dissection: Visualizing and understanding generative adversarial networks. *arXiv preprint*, arXiv:1811.10597.
- Bezdek, J. C. (1980). A convergence theorem for the fuzzy ISODATA clustering algorithms. *Bezdek, J. C. (1980). A convergence theorem for the IEEE transactions on pattern analysis and machine intelligence*, 1-8.
- Bishop, C. M. (2006). *Pattern recognition and machine learning (Information science and statistics)*. Springer-Verlag New York, Inc.
- Breiman, L. (2001). Random forests. *Machine Learning*, vol. 45, no. 1, 5-32.
- Brynjolfsson, E. &. (2017). The business of artificial intelligence. *Harvard Business Review*, 1-20.
- Comaniciu, D. &. (2002). Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on pattern analysis and machine intelligence*, 24(5), 603-619.
- Crevier, D. (1993). *AI: Yapay Zekanın Çalkantılı Arayışı*. New York: NY: BasicBooks. ISBN 0-465-02997-3.
- Dela Cruz, J. R. (2017). Water usage optimization of Smart Farm Automated Irrigation System using artificial neural network. *2017 5th International Conference on Information and Communication Technology (ICoICT)*, 1-5.
- Dutta, U. &. (2021). *The Internet of Things Using NODEMCU*. Blue Rose Publishers.
- Ester, M. K. (1996, August). A density-based algorithm for discovering clusters in large spatial databases with noise. *kdd Vol. 96, No. 34*, , 226-231.
- Fix, E. H. (1951). Discriminatory analysis, nonparametric discrimination: Consistency properties. *Technical Report 4, USAF School of Aviation Medicine*.

- Garbade, D. M. (2018, 09 14). *Clearing the Confusion: AI vs Machine Learning vs Deep Learning Differences*. Medium (İngilizce): <https://towardsdatascience.com/clearing-the-confusion-ai-vs-machine-learning-vs-deep-learning-differences-fce69b21d5eb> adresinden alındı
- García, S. &. (2008). An extension on “statistical comparisons of classifiers over multiple data sets” for all pairwise comparisons. *Journal of Machine Learning Research*, 9(89), 2677–2694.
- Gavilan, V. a.-S.-P. (2019). Seasonal crop water balance using harmonized Landsat-8 and Sentinel-2 time series data. *Water*, 2236.
- Gimpel, H. a.-D. (2021). Designing smart and sustainable irrigation: A case study. *Journal of Cleaner Production*, 128048.
- Goodfellow, I. B. (2016). Deep learning. *The MIT Press*.
- Goyal, P. P. (2018). *Deep learning for natural language processing*. New York: Apress.
- Grigorescu, S. T. (2020). A survey of deep learning techniques for autonomous driving. *Journal of Field Robotics*, 37(3), 362–386.
- Guevara, S. a. (2020). Development of a pilot smart irrigation system for Peruvian highlands. *Journal of Contemporary Water Research & Education*, 49-62.
- Gürçan, M. (1998). Lojistik regresyon analizi ve bir uygulama. Yayınlanmamış yüksek lisans tezi. *Ondokuz Mayıs Üniversitesi, Fen Bilimleri Enstitüsü*.
- Heinrich, K. Z. (2021). Process data properties matter: Introducing gated convolutional neural networks (GCNN) and key-value-predict attention networks (KVP) for next event prediction with deep learning. *Decision Support Systems*, 143, 113494.
- Hinton, G. E. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, 1527-1554.
- Hochreiter, S. &. (1997). Long short-term memory. *Neural computation*, 9(8), 1735-1780.
- Hu, J., Niu, H., Carrasco, J., Lennox, B., & Arvin, F. (2020). Voronoi-Based Multi-Robot Autonomous Exploration in Unknown Environments via Deep Reinforcement Learning. *IEEE Transactions on Vehicular Technology*, 14413-14423.
- Kish, K. M. (2006). A low-cost microprocessor and infrared sensor system for automating water infiltration measurements. *Computers and Electronics in Agriculture*, 122-129.
- Kotsiantis, S. B. (2006). Machine learning: A review of classification and combining techniques. *Artificial Intelligence Review*, 26(3), 159–190.
- LeCun, Y. B. (2015). Deep learning. *Nature*, 521(7553), 436–444.
- MacQueen, J. (1967, June). Some methods for classification and analysis of multivariate observations. *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, Vol. 1, No. 14, pp. 281-297.
- Nilsson, N. (1965). *Learning Machines*. McGraw Hill.
- Rabiner, L. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *P. IEEE vol 77 no 2*, 257-286.
- Raei., E. (2022). A deep learning image segmentation model for agricultural irrigation system classification. *Computers and Electronics in Agriculture*, 106977.

- Roberts, J. (2016). Thinking machines: The search for artificial intelligence. *Distillations*, 14-23.
- Ruspini, E. (1969). A new approach to clustering. *Inform Contr*, vol 19, 22-32.
- Ruspini, E. (1970). Numerical methods for fuzzy clustering. *Information Sciences*, 319-350.
- S. Albawi, T. M.-a. (2017). Understanding of a convolutional neural network. *Proceedings of 2017 International Conference on Engineering & Technology (ICET'2017)*, 274-279.
- Samuel, A. (1959). Some Studies in Machine Learning Using the Game of Checkers. *IBM Journal of Research and Development*, 210-229.
- Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*. Elsevier, 61, 85-117.
- Shaikh, R. &. (2015). Data classification for achieving security in cloud computing. *Procedia computer science*, 45, 493-498.
- Shrestha, Y. R. (2021). Augmenting organizational decision-making with deep learning algorithms. *Journal of Business Research*, 123, 588-603.
- Staudemeyer, R. C. (2019). Understanding LSTM--a tutorial into long short-term memory recurrent neural networks. *arXiv preprint*, arXiv:1909.09586.
- Turing, A. M. (1950). Computing Machinery and Intelligence. *Mind*, 433-460.
- VAPNIK, V. N. (1995). The Nature of Statistical Learning Theory. *Springer-Verlag*.
- Wang Y, W. I. (2002). Modeling for optimal probability prediction. *Proceedings of the 19th International Conference in Machine*, 650-657.
- Welchman, G. (2018,2015,1997). *Hut Six Story: Breaking the Enigma Codes*. M.& M.Baldwin.
- Young, T. H. (2018). Recent trends in deep learning based natural language processing [review article]. *IEEE Computational Intelligence Magazine*, 13(3), 55–75.
- Zhang, Y. &. (2018). A strategy to apply machine learning to small datasets in materials science. *npj Computational Materials*, 4(1).