



ONDOKUZ MAYIS ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

IoT Destekli Akıllı Bitki Takibi

BİTİRME PROJESİ RAPORU

Görkem Eren AKÇAMUR

Tarık Buğra AYVAZOĞLU

Ali Eren ÖZGEN

Mayıs, 2025

TEŞEKKÜR

Bu çalışma sürecinde bizlere destek olan, katkı sağlayan tüm hocalarımıza teşekkür ederiz.

Ayrıca, eğitim hayatımız boyunca yanımızda olan, bizlere her zaman destek veren ailelerimize ve arkadaşlarımıza da en içten teşekkürlerimizi sunarız.

Ö Z E T

Bitkiler, insan hayatı için önemli bir yere sahiptir ve bitkiye düşkün bazı kişiler evlerinde bitki bakımına zaman ayırmakta zorluk çekebilmektedir. Sık seyahat etmeyi seven kişiler bitkilerinin bakımlarını aksatmak zorunda kalacaklarından ya bitkilerden ya da seyahatlerinden vazgeçmek zorunda kalabilmektedirler. Bu projede, bunun önüne geçmek için bu çalışmayla ve beraberinde yapılan mobil uygulama geliştirme adımlarıyla çözüm sunulmuştur. Bitki bakımını kullanıcının uzaktan izleyip manuel veya otomatik bir şekilde yönetmesini ve geliştirilen mobil uygulama üzerinden bitkinin durumunu anlık görmesini sağlayan bir sistem geliştirilmiştir.

Sistem, mobil entegrasyon kullanılarak bitkilerin anlık görsellerini almaya, nem seviyelerini takip etmeye ve manuel ya da otomatik bir şekilde sulama yapma özelliklerine sahiptir. Sistem, kullanan kişilere esneklik sağlar ve kolay bir şekilde bitki bakım işlemlerinin gerçekleştirilmesini sağlar. Modüler bir yapıya sahip olmasından dolayı gelecekteki planlara göre sistemin istenildiği gibi şekillendirilmesi sağlanmıştır. Proje; seyahat etmeyi ve bitkileri seven, bitkilerini uzaktan yönetmek isteyen, bitkilerine zaman ayırmakta zorluk çeken kişiler için kolay bir çözüm sunar.

Anahtar Kelimeler: Bitki bakımı, mobil uygulama, hastalık tespiti, otomatik sulama, uzaktan kontrol.

İÇİNDEKİLER

I GİRİŞ

1 AMAÇ	2
2 LİTERATÜR ÖZETİ	3

II MATERYAL

3 PROJE KAPSAMINDA KULLANILAN MATERYALLER	10
3.1 Donanım Materyalleri	10
3.1.1 Toprak Nem Sensörü	10
3.1.2 ESP32-CAM	11
3.1.3 5V Dalgıç Pompa	12
3.1.4 Arduino	13
3.2 Yazılım Materyalleri	14
3.2.1 Firebase	14
3.2.2 Flutter ve Dart	14
3.2.3 Python	15
3.2.4 TensorFlow	16
3.2.5 Keras	17
3.2.6 Flask	18

III YÖNTEM

4	PROJE KAPSAMINDA KULLANILAN YÖNTEMLER	20
4.1	Projemizde Derin Öğrenme ve Görüntü İşleme Yöntemlerinin Kullanılması	20
4.1.1	Bitki Türü Tanıma İçin Görüntü İşleme	20
4.1.2	Bitki Hastalığı Tespiti İçin Görüntü İşleme	23
4.2	Otomatik Sulama Yöntemi	34
4.2.1	Giriş ve Amaç	34
4.2.2	Çalışma Prensibi	35
4.2.3	Kullanılan Sensörler ve Teknolojiler	35
4.2.4	Sulama Stratejileri	36
4.2.5	Otomatik Sulama Sisteminin Avantajları	36
4.3	Kullanıcı Müdahalesi Yöntemi	38
4.4	Veri İşleme Yöntemi	38
4.4.1	Veri Ön İşleme	39
4.4.2	Veri Analizi ve İşleme	39
4.4.3	Sulama Kararlarında Veri Kullanımı	40
4.4.4	Veri Yönetimi ve Depolama	40
4.5	Mobil Uygulama Yöntemi	41
4.5.1	Mobil Uygulama Arayüzü ve İşlevleri	41
4.5.2	Mobil Uygulama Geliştirme Süreci	42
4.5.3	Projeye Katkıları	42
4.6	Sensör Verilerinin İşlenmesi Yöntemi	43

IV SONUÇ

5	SONUÇLAR VE TARTIŞMA	46
---	----------------------	----

5.1	Proje Sonuçları	46
5.2	Sonuçların Önemi ve Gelecek Çalışmalar	53
	KAYNAKÇA	54

ŞEKİLLER LİSTESİ

1	HL-69 Toprak Higrometre Nem Sensörü [1]	11
2	ESP32-CAM [2]	12
3	5V Dalgıç Pompa [3]	13
4	Arduino [4]	13
5	Tür Tespiti Veri Dağılımı	23
6	Paşakılıcı Veri Dağılımı	25
7	Kaktüs Veri Dağılımı	26
8	Aloe Vera Veri Dağılımı	28
9	Barış Çiçeği Veri Dağılımı	31
10	Genel Model Veri Dağılımı	33
11	Akıllı Sulama Akış Şeması	37
12	Sensör Verilerinin İşlenmesi	44
13	Paşakılıcı Eğitim Grafikleri	48
14	Aloe Vera Eğitim Grafikleri	49
15	Kaktüs Eğitim Grafikleri	50
16	Sukulent Eğitim Grafikleri	50

BÖLÜM: I

GİRİŞ

A M A Ç

Projemizde, bitkilerin sulama ihtiyaçlarının herhangi bir kimsenin müdahalesine bağı kalmadan otomatik bir sisteme çevrilmesi, aynı zamanda bitkide oluşabilecek hastalıkların tespiti hedeflenmiştir. Geliştirilen bu sistemde, kullanıcıların bitkilerin nem seviyeleri ve sağlığı hakkında genel bilgileri takip edebilmeleri amacıyla kullanılabilecekleri bir mobil uygulama tasarlanmıştır. Bu sistem, özellikle sık seyahat eden, evden uzak kalan veya bitki bakımını uzaktan gerçekleştirmeyi amaçlayan kullanıcılar için pratik bir çözüm sunmaktadır. Sistemde ana hedef, toprak nem sensörlerinden gelen verilere göre otomatik olarak sulama yapılmasıdır. Otomatik sulama yapılırken, aynı zamanda kullanıcıların mobil uygulama üzerinden bitkileri sulamak için komut verebilmesi sağlanmıştır. Bitki türü tanıma ve hastalık tespiti işlemleri, kullanıcıların telefon kamerasıyla çektikleri görselleri mobil uygulama üzerinden sisteme yüklemesiyle gerçekleştirilir. Bu sayede, sabit bir kamera kullanılmasına gerek kalmadan, sistem maliyet açısından daha verimli ve enerji kullanımı açısından daha tasarruflu hale getirilmiştir. Sistem uygulandığında, bitkilere yalnızca sulamaya ihtiyaç duyduklarında su verilerek su israfı önlenmiş, böylece hem çevresel hem ekonomik açıdan sürdürülebilir bir çözüm sunulmuştur.

LİTERATÜR ÖZETİ

Bu literatür özeti, tarım alanlarında otomatik işleyen ve teknolojinin birçok yönünü kapsayan araştırmaları incelemektedir. Sürdürülebilir tarım için kritik öneme sahip olan bu çalışmalar, toprağın nemini ölçüp sulama sisteminin otomatikleşmesini ve robotlar üzerine odaklanmasını sağlar [5–10]. Bu sistemlerin ana amaçları ise su kaynaklarının verimli bir şekilde kullanılmasıyla su tasarrufu yapılması, bitki sağlığının optimize edilmesi ve tarımda verim oranının önemli ölçüde arttırılmasıdır [6, 8–10]. Araştırmalar bu hedeflere ulaşmak için çeşitli yazılım algoritmaları, toprak nem sensörleri, mikrodenetleyiciler (Arduino vb.) ve kontrol mekanizmaları kullanmıştır [5, 8–10]. Sistemlemler de farklılıklar olabilir. Bazıları sadece sulama işini yaparken bazıları ise hem sulama hem de gübreleme işlemlerini beraber yapar [6]. Bitki bakım işini otomatikleştiren bu çalışmalar, hem devamlılık hem de kullanıcı dostu işlevler için önemli faydalar oluşturur. Tarımın daha işlevsel ve çevreci hale gelmesinde bu sistemlerin geliştirilmesi önemli rol oynar [5].

Bu alanda farklı yaklaşımlar ve teknolojiler, çeşitli çözüm önerileri sunmaktadır. Aşağıda bu teknoloji ve yaklaşımlar ele alınmaktadır:

Kapalı döngü kontrol modelleri: Bu modeller topraktaki nem oranını optimize etmek için yağış, buharlaşma, terleme ve sulama gibi birçok unsuru dikkate alır. Tarımsal sulama işlemlerinde, su kaynaklarının korunmasını sağlamak için suyun kullanımını verimli bir şekilde

oluşturması büyük önem taşımaktadır. Kapalı döngü sistemleri, bitkilerin su ihtiyaçlarına göre sulama işlemini gerçekleştirebilmesi için toprağın nem dinamiklerini daima izler. Bunun sayesinde hem su tasarrufu hem de bitkinin sağlıklı bir şekilde büyümesi sağlanır. Bu sistemler, su kaynaklarının daha verimli kullanılabilmesini sağlamak için toprak nemini sürekli izleyerek çevresel devamlılığa faydalı olur [11].

Robotik sistemler: Tarımda robot kullanımı giderek artmakta, bu da bitki bakım süreçlerinin otomasyonuna olanak tanımaktadır. Bu sistemler, genellikle mobil robotlar veya robotik kollar aracılığıyla bitkilerin bakımını otomatik hale getirir [7, 12, 13]. Örneğin, Yeni Meksika'daki biber çiftliklerinde kullanılan bir mobil robot, toprağın nem ve sıcaklığını ölçerken bitkilerin sağlık durumunu da tespit edebilmektedir [12]. İç mekan bitkileri için tasarlanan robotlar ise, RFID etiketleri sayesinde bitkileri tanıyıp su ihtiyaçlarını belirleyerek sulama yapabilmektedir. Bu tür robotik sistemler, tarımsal işlemlerde insan gücüne olan ihtiyacı azaltırken aynı zamanda verimliliği artırır. Ayrıca, robotların kullanımı, hassas tarım uygulamalarının yaygınlaşmasına da katkı sağlar [7, 13].

IoT ve makine öğrenimi entegrasyonu: Sulamanın süreçlerinin optimizasyonunu arttırmak ve süreçleri daha akıllı hale getirmek için IoT teknolojileri ve makine öğrenimi algoritmaları kullanılır. Makine öğrenimi algoritmaları ile analiz etmek ve bitkinin suya ihtiyacı olduğuna karar vermek için IoT tabanlı sensörlerden toplanan veriler kullanılır [14, 15]. Gerçek zamanda anlık veri toplama, uzaktan izleyebilme ve uzaktan kontrol edebilme imkanı sağlayarak çiftçilerin daha çok verimli ve bilinçli şekilde sulama yapabilmeleri bu sayede desteklenir. Hava durumu, topraktaki nem oranı, bitkinin temel ihtiyaçları gibi etkenlerin analiz edilmesiyle bitkinin sulanması için en uygun zamanı ve bitkinin ne kadar miktarda sulanacağını belirlemek için makine öğrenimi algoritmaları kullanılır [15, 16].

Düşük maliyetli ve açık kaynaklı sistemler: Maliyeti düşük çözümler ve açık kaynak sistemler geliştirerek tarımsal otomasyonun daha çok kişiye yayılması sağlanabilir [17, 18]. Bunlar küçük çiftlikler ve bitki deneyleri için bütçe dostu ve erişilebilir seçenekler sunuyor. Örneğin, büyük doğruluk oranlarında nemi kontrol ederek kapsamlı deneylerin yapılmasını kolaylaştırabilmek için bitki deneylerinde kullanılmak için tasarlanmış bütçe dostu bir robotik sistem kullanılır [17]. Fiyat açısından daha uygun alternatifler bulmaya çalışırken farklı maliyetteki sensörlerin performansı farklı toprak dokularında ve nemi farklı ortamlarda karşılaştırılır. Tarım teknolojilerini elde edebilmeyi basitleştirerek daha çok çiftçinin bu gibi teknolojilerden faydalanabilmesini sağlamak için bu tarz sistemler kullanılır [18].

Görüntü işleme ve makine öğrenimi teknikleri: Tarım için kilit bir role sahiptirler. Bitkinin hastalıklarının erken tespit edilmesi, farklı bitki türlerinin tanınabilmesi [19, 20] ve bitkinin büyüme evrelerinin izlenmesi [21] gibi alanlarda bu teknolojiler sayesinde önemli adımlar atılmıştır. Bitkinin yapraklarında bulunan karışık özellikleri bir süzgeçten geçirerek hastalıkları büyük doğruluk oranıyla sınıflandırabilmek için derin öğrenme modelleri kullanılır [22]. Özellikle hızlıca müdahale etme olanağı sağlayabilmek ve tarımsal kayıpları en aza indirmek için bu çok önemlidir. Ayrıca, bitkinin yapraklarının özelliklerinin analiz edilerek büyük doğruluk oranlarıyla doğru kategorilendirme yapabilmek için bitki türlerini tanımda kullanılan evrişimli sinir ağları kullanılır [19, 20]. Bitkinin yaprağının kesit alanını ve Normalize Edilmiş Bitki Örtüsü Fark İndeksi (Normalized Difference Vegetation Index - NDVI) değerini ölçerek bitkinin mevcut sağlığı ve büyümesindeki performansı konusunda kritik bilgiler sağlamak için bitki büyümesini izlemede kullanılan görüntü işleme sistemleri kullanılır. Bahsedilen teknolojiler, tarımsal üretimin her kademesinde sağlanan faydayı arttırarak daha sürdürülebilir uygulamaların geliştirilmesine yol açar [21].

Tarımsal otomasyonun en temelini, sensör teknolojileri oluşturur. Farklı türlerdeki toprak nem sensörlerinin performansları ve doğruluk oranları farklı çalışmalarla karşılaştırılmıştır [18, 23]. Araziye özgü kalibrasyonların sensörlerin doğruluk oranını yükselttiğini bu çalışmalar sayesinde gösterilmektedir. Fabrika kalibrasyonlarının çoğunlukla arazi şartlarında kafi olmadığı için bulunan araziye özel ayarların yapılması lazım olduğu vurgulanmaktadır [23]. Bütçe dostu sensörlerin de elverişli alternatifler olabileceği söylene de, geleceğe yönelik dayanıklılıklarının ve performanslarının daha çok incelenmesi lazım olduğu söylenmektedir. Özellikle, pahalı sensörlerin daha fazla doğruluk oranlarında ve daha çok güvenilirlik sağladığı, ancak bütçe dostu sensörlerin maliyet için avantaj sunduğu vurgulanmaktadır. Bu çalışmalar sayesinde farklı ihtiyaçlara ve uygun maliyetli sensör seçeneklerinin bulunmasına yardımcı olur [18].

IoT teknolojileri, tarım bölgesinde anlık bilgi toplama, uzaktan gözetleme ve kontrol işlemleri yaparak bu alanda önemli bir görev gerçekleştirmektedir. Bu teknolojiler kullanıcıların sulama işlemlerini optimize etmelerine, kaynak harcamalarını daha tasarruflu bir şekilde yapmalarına ve tarımsal işlemleri uzaktan, kolay bir şekilde gerçekleştirmelerine olanak sağlamaktadır. IoT cihazlarının yardımıyla toplanan bilgiler bulutta saklanıp işlenerek, kullanıcılara analiz ve raporlama yapma fırsatı oluşturur [15, 16, 24, 25]. Kullanıcı etkileşimini yükseltmeye olanak sağlayan çözümler yapabilmek için mobil uygulamalar ve bulut tabanlı sohbet robotları kullanılır [24]. Ayrıca, IoT'ların mobil ağlara bağlanmasıyla ağlarda oluşacak bağlantı tıkanıklığını giderme ve bant genişliğini optimize etme çabaları da gerçekleştirilir. Bu entegrasyonlar, tarım aşamalarının daha işlevsel ve kolay bir şekilde yönetilmesine olanak sağlarlar [25].

Devamlı su yönetimi ve kaynakların verimliliği, tarım tarafında gerçekleşen teknolojik gelişmelerden en önemlileridir. Suyu tasarruflu bir şekilde kullanmamızı sağlayan sulama

algoritmaları [26] ve suyu daha az bir şekilde kullanmamızı sağlayan yani su tüketimini azaltan otomasyon sistemleri [27], bu amaçlar doğrultusunda geliştirilmelidir. Düzenli eksik sulama (RDI) stratejileri gibi yöntemler, hem suyun az kullanımını sağlarken hem de bitkinin sağlığının korunmasını gerçekleştirirler [28]. Otomatik sulama sistemleri, sudan tasarruf edebilmek için topraktaki nemi hiç durmadan izleyip, sadece ihtiyaç halinde toprağa su verir. Bununla birlikte su kaynaklarını kullanıcılar daha verimli bir şekilde değerlendirirken aynı anda bitki verimliliği de artar. Bu tür devamlı sistemler, gelecek zamanda su tasarrufu konusunda, su kaynaklarının korunması durumunda çok önemli bir yere sahiptir [5, 8–10].

Çalışmalar, farklı tarım bölgelerinde farklı sensör yapısının ve farklı sistemlerin performansını da bir şekilde değerlendirmektedir. Örneğin, Doñana Milli Parkı'nın tozlu toprakları için benzer bir kalibrasyon denklemleri sağlanmıştır [23]. IoT tabanlı bir sistem sayesinde Marul tarımında topraktaki nemin miktarı ve sıcaklık miktarı sürekli bir şekilde izlenmiştir. Bu işlemler, farklı toprak türlerine ve bitki türlerine özgü sulama sisteminin avantajlarının ne kadar önemli olduğu gösterilmiştir. Farklı bitkilerin ve farklı toprak türünün birbirlerinden farklı ihtiyaçları olduğundan, bu ihtiyaçlara göre sistemler oluşturulmalı ve geliştirilmelidir [29].

Bu alandaki bazı projeler, tarım robotlarının katkılarının ne kadar olabileceğini incelemiştir. Bu robotlar, hasat, algılama, tasarım, etik ve otonomi gibi alanlarda değerli katkılar sağlamaktadırlar. Tarımda verimliliğin önemli ölçüde artmasında ve insan müdahalesini yok denilecek kadar az seviyeye indirilmesinde robot teknolojisinin önemli bir rolü vardır. Ayrıca, tarım robotlarının daha fazla kullanımıyla daha kaliteli tarım çalışmaları gerçekleşmekte ve kaynaklar daha tasarruflu bir şekilde kullanılmaktadır. Bu robotlar, tarım görevlerini çok daha hızlı ve güvenli bir şekilde gerçekleştirebilir. Böylelikle verimlilik önemli ölçüde artar ve maliyette önemli ölçüde azalır [30].

Bitki yapraklarının sınıflandırılması üzerine yapılan çalışmalarda önde gelen derin öğrenme algoritmaları kıyaslanmış ve başarı oranlarına göre AlexNet ile ResNet50 gibi algoritmaların doğru yaprak teşhisinde diğerlerinden daha başarılı olduğu saptanmıştır [31]. Bundan dolayı bu algoritmalar bitkilerin ayrıştırılmasında diğer algoritmalara göre daha çok tercih edilmiş ve daha ön planda tutulmuştur. Aynı zamanda IoT sistemlerinde mobil uygulamalar ve bulut teknolojileriyle entegrasyonu da incelenmiştir. IoT cihazlar veri toplamasına ve paylaşmasına olanak sağlarken, bulut teknolojisinin ise sınırsız depolama ve işleme kapasitesi sunduğu vurgulanmıştır. Örneğin IBM Watson Assistant gibi bulut tabanlı sohbet robotları, mobil uygulamalarda kullanıcı etkileşimini artırarak kullanıcıya kolaylık sağlayan çözümler sunar [24]. Tarım teknolojileri, bu entegrasyonlar sayesinde daha erişilebilir ve kullanımı kolay bir hale gelir. Literatürümüzde son olarak mobil iletişim ağına dayalı bir IoT sistemi geliştirilmesi ele alınmıştır. Bu çalışma uygulama katmanında çalışan bir tıkanıklık kontrol algoritması önerisiyle, ağ tıkanıklığını önlemek ve bant genişliği kullanımını optimize etmeyi amaçlamıştır. Algoritmanın veri aktarımını etkili bir şekilde yönettiği ve bant genişliği kaynaklarında tasarruf sağladığı gözlemlenmiştir. IoT sistemlerinde kullanılan bu tür algoritmalar, sistemin daha verimli çalışmasına katkıda bulunur [25].

BÖLÜM: II

MATERYAL

PROJE KAPSAMINDA KULLANILAN MATERYALLER

Bu bölümde projemizde kullandığımız materyaller ile ilgili açıklamalar bulunmaktadır.

3.1 *Donanım Materyalleri*

Donanım materyalleri, bitkinin izlenmesi ve toprağın neminin ölçülmesinde kullandığımız entegre materyallerdir.

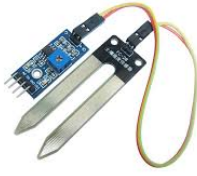
3.1.1 *Toprak Nem Sensörü*

Toprak nem miktarını ölçüp kullanıcılara geri dönüş sağlamak ve nem miktarına göre otomatik sulama işlemini yapmak için HL69 toprak sensörü kullanılmıştır. [Şekil 1](#)'de HL69 Toprak Sensörü'nün görseli gösterilmiştir. Bu sensör, topraktaki nem miktarını ölçmek ve bitki bakım işlemlerini veya otomatik sulama sistemlerinde kullanılmak üzere tasarlanmıştır.

HL69 sensörü, topraktaki nem miktarına bağlı olarak direnç değişimlerini algılayan bir sisteme sahiptir. toprakta ölçüm yapan kısmı, bakır yollardan oluşan bir yapıya sahiptir. nem oranı arttıkça sensör direnci azalır, nem miktarı azaldıkça direnç artar. Bu direnç değişimleri,

sensörün devresinde gerilim değişimi olarak yorumlanır ve nem seviyesi analog bir sinyal olarak elde edilir. [18].

Ayrıca, HL69 sensörü trimpot içerir. Bu trimpot sayesinde dijital çıkış hassasiyeti ayarlanabilir. Dijital çıkış, belirli nem seviyesine geldiğinde doğrudan bir sulama sistemini gerçekleştirebilir. Böylece, bir denetleyiciye gerek olmadan bağımsız bir şekilde çalışabilir. [1, 29]



Şekil 1: HL-69 Toprak Higrometre Nem Sensörü [1]

Bu projede HL69 toprak nem sensörünün sağladığı avantajlar şunlardır: Topraktaki nem miktarına göre doğrusal bir analog sinyal üretir, Denetleyiciye gerek kalmadan sulama sistemini otomatik olarak aktif hale getirebilir, Trimpot ile hassasiyet ayarı yapılabilir, Düşük maliyetlidir ve ekonomik bir çözüm sunar, Kolayca sistemlere entegre edilebilir, Topraktaki nem seviyesini sürekli olarak izleyebilir. [1].

3.1.2 ESP32-CAM

Bu projede, bitkilerin sulama ihtiyaçlarının tespiti ve sistemin otomatik olarak kontrol edilmesi amacıyla ESP32-CAM modülünden yararlanılmıştır. Şekil 2’de ESP32-CAM modülünün genel görünümü verilmiştir.

ESP32-CAM, dahili mikrodenetleyicisi ve Wi-Fi bağlantısı sayesinde sistemin merkeziyle haberleşmesini sağlar. Arduino, HL69 toprak nem sensöründen gelen verileri okur ve

belirli bir eşik değerin altına düştüğünde pompa sistemini devreye sokar. Sulama işlemi gerçekleştiğinde, bu bilgi ESP32 aracılığıyla Flask tabanlı sunucuya HTTP POST isteği ile iletilir.

Ayrıca mobil uygulama üzerinden manuel sulama komutu gönderildiğinde, bu komut sunucu aracılığıyla ESP32'ye iletilir ve ESP32 bu talimatı Arduino'ya aktararak sulama işlemini başlatır. Böylece kullanıcı, sistem üzerinde uzaktan kontrol sağlayabilir. [2].



Şekil 2: ESP32-CAM [2]

ESP32-CAM modülü, düşük güç tüketimi, Wi-Fi desteği ve kompakt yapısıyla sistemle uyumlu bir şekilde entegre edilmiş; hem otomasyonun sürdürülebilirliğini hem de uzaktan erişilebilirliğini mümkün kılmıştır. [2].

3.1.3 5V Dalgıç Pompa

Bu projede, bitkilerin otomatik sulanmasını gerçekleştirmek için 5V dalgıç pompa kullanılmıştır. Şekil 3'te 5V Dalgıç Pompa'nın görseli gösterilmiştir. 5V dalgıç pompalar, düşük voltajda çalışan ve küçük ölçekli yapıda bulunan sulama sistemleri için kolay ve ideal olan su pompalarıdır. Enerji verimliliği bakımından epey avantaj sağlayan bu pompalar, düşük güç tüketimi yapmasıyla suyu iyi bir şekilde taşır ve bitkilerin ihtiyacı olduğu nem miktarını korumak için otomatik sulama işlemini gerçekleştirir. Projede kullanılan 5V dalgıç pompa,

HL69 toprak nem sensöründen gelen verilere bağlı olarak Arduino modülü tarafından kontrol edilmektedir. [32]



Şekil 3: 5V Dalgıç Pompa [3]

3.1.4 Arduino

Bu projede, otomatik bitki sulama ve izleme sisteminin kontrolü için Arduino platformu kullanılmıştır. Şekil 4’de Arduino’nun görseli gösterilmiştir. Arduino, açık kaynaklı bir mikrodenetleyici platformu olup, elektronik projeler için hızlı prototipleme ve esnek kontrol imkânı sunar. Projede Arduino, HL69 toprak nem sensörü, ESP32-CAM ve 5V dalgıç pompa ile entegre edilerek bitki sulama ve izleme sisteminin merkezi kontrol birimi olarak görev yapmıştır. [4].



Şekil 4: Arduino [4]

HL69 sensöründen gelen nem verilerini okuyarak toprağın nem seviyesini değerlendirir ve bu verilere göre 5V dalgıç pompayı çalıştırarak bitkilerin otomatik sulanmasını sağlar. Aynı zamanda, ESP32-CAM ile koordineli çalışarak bitki görüntülerini analiz eder ve sulama

kararlarını optimize eder. Arduino'nun sensörlerle kolay entegrasyon yeteneği, çevresel verilerin toplanmasını ve sulama sisteminin otomatik kontrolünü mümkün kılar [4, 5].

3.2 *Yazılım Materyalleri*

Bitkilerin durumlarının izlenmesi sürecinde, yazılım materyalleri, donanım materyallerinin doğru ve uyumlu bir şekilde çalışmasını gerçekleştirmek için projemizde önemli bir rol üstlenmiştir. Bu materyaller, verilerin işlenmesini, analizini ve kontrol mekanizmalarının hassasiyet içerisinde birlikte çalışmasını mümkünleştirmiştir. Aynı zamanda, sistemin hatasız olarak yönetilmesi ve kullanıcıya kolay bir deneyim sunulabilmesi için temel oluşturmuştur.

3.2.1 *Firebase*

Bu projede, bitkilerin verilerinin saklanması, yönetilmesi ve analiz edilmesi için Firebase Firestore kullanılmıştır. Firebase, gerçek zamanlı veri senkronizasyonu ve bulut tabanlı esnek yapısıyla bitki türü, son sulanma tarihi ve sulama kayıtları gibi bilgilerin güvenli şekilde depolanmasını sağlar. Ayrıca, sistemin kullanıcıya gönderdiği bildirimler de Firebase altyapısı üzerinden yönetilmektedir. Bu sayede, bitki takibi ve sulama süreçleri mobil uygulama ile etkin ve kesintisiz şekilde entegre edilmiştir. [33].

3.2.2 *Flutter ve Dart*

Bu projede, kullanıcıların bitki izleme ve sulama sistemine erişim sağlayabilmesi için Flutter ve Dart teknolojilerinden yararlanarak mobil uygulama geliştirilmiştir. Flutter, Google'ın

açık kaynaklı framework'ü olup, Dart ile çalışır ve tek kod yapısına sahip olmasıyla iOS ve Android için performanslı arayüzler oluşturur. Kullanıcılar, 5V dalgıç pompa için sulama talimatı verebilir, sulama zamanlamalarını ayarlayabilir ve bitki büyüme evrelerini izleyebilir. Flutter'ın widget kütüphanesi, dinamik ve kullanıcı dostu arayüz sağlar; kullanıcılar bitki görüntülerini yükleyip analiz sonuçlarını görebilir. [34].

3.2.3 Python

Yazılım işlemlerinde, veri analizi işlemlerinde, görüntü işleme işlemlerinde ve yapay zeka işlemlerinde kullanılan sağlam bir dildir.

Bu projede, veri analizi, görüntü işleme ve sistem entegrasyonu için Python kullanılmıştır. Python, kolay öğrenilebilir bir yapıya sahiptir ve veri analizi, görüntü işleme ve yapay zeka işlemlerinde öne çıkan, açık kaynaklı bir yapıya sahip güçlü bir programlama dilidir. Geniş kütüphane desteğine sahip olmasıyla HL69 toprak nem sensöründen alınan bilgileri işlemek için NumPy, telefon kamerasından alınan bitki görüntülerini analiz etmek için Keras ve bitki sağlığı tahmini için TensorFlow kullanılmıştır. Windows ve linux gibi sistemlerde sorunsuz çalışabilmesi için çapraz platform desteği sağlar. Python'ın IoT cihazlarıyla uyumluluğundaki kolaylık, 5V dalgıç pompa kontrolünü ve sensör verilerinin gerçek zamanlı işlenmesinde bir nevi kolaylaştırmıştır. Geniş topluluk desteği, hızlı prototipleme için basit yapısı ve modüler tasarımı, projenin yönetimini ve veri görselleştirmesinde etkinlik katmıştır. Flutter uygulamasına verilerin sağlanmasında da Python, analiz sonuçlarını hızlı şekilde paylaşır. [35].

3.2.4 *TensorFlow*

TensorFlow, Geniş kapsamlı uygulama alanı sayesinde birçok alanda kullanıma açıktır. Örneğin görüntü tanıma işlemi, nesne algılama işlemi gibi işlemler gerçekleştirilir ve modelin geliştirilmesi ile dağıtması arasında bir köprü oluşturur. Açık kaynak ve ücretsiz bir yapı olduğundan herkes tarafından rahat bir şekilde erişilebilir. Birçok kullanan insan olması ve dünyada çok fazla işlevsel dökümantasyona sahip olduğundan öğrenmesi kolaydır. Platformlar arası çalışabilirlik kapsamında TensorFlow, Windows ve Linux gibi farklı platformlarda çalışma işlemini gerçekleştirebilir. Kolay model geliştirme açısından Keras ile birlikte çalışabilir bu da model oluşturma evresini hızlı ve kullanıcı dostu bir hale getirmeye olanak sağlar. Düşük seviyedeki API'ler ile birlikte karmaşık algoritma işlemlerinin de gerçekleşmesine olanak sağlar. Performansı yüksek bir kütüphanedir ve çok büyük veri setlerinde bile hızlı bir şekilde çalışabilir. Derin öğrenme modellerinde işlemleri çok hızlı bir şekilde gerçekleştirebilir. TensorBoard ile görselleştirme kapsamında TensorBoard aracı ile model eğitme aşamasında metriklerin görselleştirilmesini sağlar. Hataları görsel olarak analiz edip kolay bir hata ayıklama işlemi gerçekleştirir. Ölçeklenebilir yapısıyla büyük veri setleri ile çalışma olanağı sağlar. Zengin ekosistem kapsamında TensorFlow Hub yeniden kullanılabilir bir model yapısı verir, bununla birlikte model geliştirme işlemi hızlandırılır. Geniş model kütüphanesi ile TensorFlow'da önceden oluşturulmuş çok sayıda model bulunur, bu modelleri yazılımcılar istediği şekilde kullanabilir ve daha az çabayla hızlı bir şekilde model yapısına erişilmesi sağlanır [36].

3.2.5 Keras

Python altyapısını barındıran, yüksek seviyeli ve kullanımı kolay bir görüntü işleme kütüphanesi olan Keras, görüntü işleme kütüphaneleri olarak Theano veya Microsoft Cognitive Toolkit (CNTK) gibi örnekler kullanır. Keras'ın kullanımı ve yeni kullanıcılar için öğrenmesi kolaydır. Keras, modelleri oluşturmayı kolay bir şekilde sokar ve karmaşıklığı ortadan kaldıracaktır. Kullanıcılar Keras'ı kendi isteklerine göre düzenleyip genişletebilirler. Açık kaynaklı bir yapıda olduğundan dolayı topluluklar sayesinde sürekli genişletilme olanakları vardır. Çeşitli backend desteği ile Keras, TensorFlow, Theano veya CNTK gibi farklı türdeki backend kütüphaneleriyle istediği şekilde çalışmalarda bulunabilir bu da esneklik sağlar. Backend ile birlikte aynı yapı, farklı donanımlarda (CPU, GPU, TPU) çok rahat bir şekilde kullanılabilir. Önceden eğitilmiş modeller olarak ResNet, VGG, Inception, MobileNet gibi görüntü işleme modellerinin eğitilmiş sürümleri Keras ile birlikte gelir. Eğitilmiş modeller kullanıcının isteğine göre kolayca çalışmalara uygulanabilir. Keras sinir ağlarını destekler. Bu ağlar; tam bağlantılı ağlar (Dense Layers), konvolüsyonel sinir ağları (CNN), rekürrent sinir ağları (RNN), metin sınıflandırma işlemleri, dil modeli gerçekleştirme işlemleri gibi NLP fonksiyonları kullanılabilir. Aktif topluluk ve geniş dokümantasyon ile Keras'ın kullanıcı çevresi geniştir, bu yüzden problemlere çözümler kolay bir şekilde bulunur ve öğretici kaynakları detaylı bir şekilde vererek öğrenme işlemini hızlandırır [37].

3.2.6 *Flask*

Python ile oluşturulan, karmaşık olmayan bir web platformudur. küçük ve orta seviyeli çalışmalarda popüler bir yapıdır.

Flask'da, hafif ve minimalist yapı sayesinde çok fazla karmaşık yapılar bulunmaz ve kolay bir başlangıç sunar. Kullanıcıların çalışmalarını istedikleri gibi düzenlemesini sağlar. Üçüncü taraf kütüphane eklenmesinde sorun çıkarmaz. Python köklü olmasından dolayı öğrenmesi kolaydır. Yeni başlayan kullanıcılar bile hızlı bir şekilde öğrenebilir. Kaliteli eğitim materyallerine sahiptir ve bunlar öğrenme aşamasını hızlandırır. Prototip geliştirme işini hızlı bir şekilde yapar ve kolay bir yapıya sahip olduğundan proje işlemleri de kolay şekilde gerçekleştirilir. Test araçları bulundurur ve bunlar sayesinde çalışmaları da kolay bir şekilde test edebilir. RESTful API geliştirmek için çok iyi bir platform oluşturur ve kolay yapıda olduğundan dolayı API geliştirme işlemleri de kolay gerçekleşir. Altyapısı Python'dan oluşmuştur ve Numpy, Pandas, TensorFlow gibi birçok kütüphaneye bağlanabilir. Flask, büyük bir topluluğa sahiptir, bu yüzden sorunlara cevap bulmak kolaydır [38].

BÖLÜM: III

YÖNTEM

PROJE KAPSAMINDA KULLANILAN YÖNTEMLER

4.1 *Projemizde Derin Öğrenme ve Görüntü İşleme Yöntemlerinin Kullanılması*

Projemizin bu kısmında derin öğrenme ve görüntü işleme yöntemlerinin nasıl kullanıldığına ve projemizde ne şekilde yer ettiğine değiniliyor.

4.1.1 *Bitki Türü Tanıma İçin Görüntü İşleme*

Bu sistem, evde yetiştirilen bitkilerin türünü otomatik olarak tanımak için geliştirilmiştir. Kullanıcılar, telefonları ya da bir kamera aracılığıyla bitkinin fotoğrafını gönderdiğinde, sistem bu fotoğrafa bakarak bitkinin hangi türe ait olduğunu belirler. Böylece o bitkiye özel sulama, bakım ve sağlık önerileri sunar. Sistem, bitkileri önceden gruplandırmadan, doğrudan 31 farklı ev bitkisi türünü tanıyacak şekilde tasarlanmıştır.

Modelin eğitimi için kullanılan görseller, hem internetten erişilebilen açık kaynaklı bitki fotoğraflarından hem de gerçek ortamlarda çekilmiş sahne görüntülerinden oluşmaktadır. Veri seti toplam 10.324 görsel içermekte olup, bunların 7.207'si eğitim, 1.530'u doğrulama ve 1.587'si test amaçlı kullanılmıştır. Her bitki türü için ortalama 300-350 görsel toplanmış, veri

dengelessizliğini önlemek amacıyla sınıflar arasında dengeli bir dağılım sağlanmaya çalışılmıştır [Şekil 5]. Özellikle Atatürk Çiçeđi, Benekli Bitki, Kaktüs ve Orkide gibi türler için yeterli sayıda kaliteli görsel elde edilirken, Kalathea, Sukulent ve Fil Kulađı gibi bazı türlerde görsel sayısının artırılması gerektiđi tespit edilmiştir.

Model mimarisi olarak EfficientNetB0 tabanlı, transfer öğrenme tekniđine dayalı bir yapı tercih edilmiştir. Bu yapının seçilme nedeni, CPU ortamında verimli çalışabilmesi ve bitki yapraklarındaki temel özellikleri yakalayabilmesidir. ImageNet üzerinde daha önce eğitilmiş bu yapı, son katmanlarına yeni bölümler eklenerek 31 bitki türünü tanıyacak şekilde yeniden düzenlenmiştir. Modelin çıkış katmanında 512 ve 256 nöronlu iki adet yoğun katman kullanılmış, her katman arasında batch normalization ve dropout teknikleri uygulanmıştır. Eğitim süreci iki aşamalı olarak gerçekleştirilmiştir: İlk aşamada temel model dondurularak sadece yeni eklenen sınıflandırma katmanları 50 epoch boyunca eğitilmiş, ikinci aşamada ise tüm model açılarak 30 epoch daha fine-tuning işlemi yapılmıştır.

Modelin başarısını artırmak için kapsamlı veri artırma teknikleri uygulanmıştır. Eğitim görsellerine 30 derece döndürme, %20 oranında genişlik ve yükseklik kaydırması, %20 kesme ve yakınlaştırma, yatay çevirme ve %20 parlaklık değışimi gibi dönüşümler uygulanmıştır. Doğrulama ve test verileri ise sadece 0-1 aralığında normalize edilerek işlenmiştir. Tüm görseller 224x224 piksel boyutuna yeniden boyutlandırılmış ve RGB formatında kullanılmıştır. Batch boyutu CPU performansı göz önünde bulundurularak 16 olarak belirlenmiştir.

Eğitim sürecinde çeşitli optimizasyon teknikleri kullanılmıştır. Adam optimizatörü 0.001 başlangıç öğrenme oranıyla tercih edilmiş, kategorik çapraz entropi kayıp fonksiyonu uygulanmıştır. Aşırı öğrenmenin önüne geçmek için erken durdurma (15 epoch sabır), öğrenme oranı azaltma (5 epoch sabır ile %50 azaltma) ve model kontrol noktası kaydetme teknikleri kullanılmıştır. Fine-tuning aşamasında öğrenme oranı 10 kat azaltılarak daha hassas bir eği-

tim gerçekleştirilmiştir. Modelin genelleme gücünü artırmak için dropout oranları 0.3 ile 0.5 arasında değişen değerlerde uygulanmıştır.

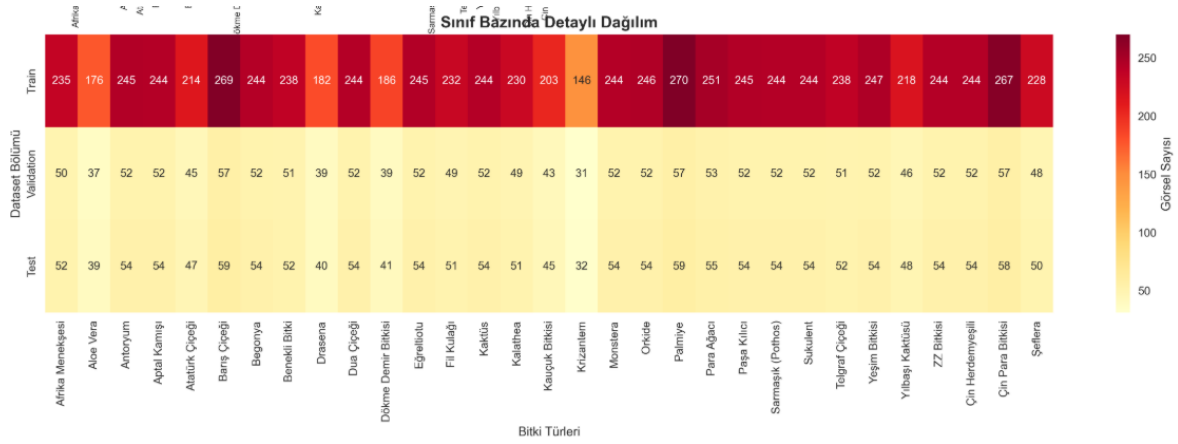
Modelin performansı çoklu metriklerle değerlendirilmiştir. Temel accuracy metriğinin yanı sıra Top-3 ve Top-5 accuracy değerleri de hesaplanmış, böylece kullanıcıya birden fazla tahmin sunma senaryoları için performans ölçümü yapılmıştır. Sınıf bazında kesinlik (precision), duyarlılık (recall) ve F1-score değerleri hesaplanarak her bitki türü için ayrı ayrı başarı analizi gerçekleştirilmiştir. Confusion matrix analizi ile en çok karıştırılan bitki çiftleri tespit edilmiş, özellikle Sukulent-Yeşim Bitkisi ve Kalathea-Dua Çiçeği gibi benzer türler arasındaki karışıklıklar belirlenmiştir.

Test aşamasında model güvenilirliğini artırmak için Test Time Augmentation (TTA) tekniği uygulanmıştır. Her test görseli için 3 farklı augmentation işlemi yapılarak tahminler alınmış ve bu tahminlerin ortalaması hesaplanmıştır. Ayrıca modelin güven seviyesi analiz edilerek, düşük güven skorlarına sahip tahminler için kullanıcı uyarı sistemi tasarlanmıştır. Sınıf bazında performans analizi yapılarak en başarılı ve en zorlanılan türler belirlenmiştir.

Eğitimin sonunda model Keras H5 formatında kaydedilmiş, ayrıca SavedModel, TFLite ve weights-only formatlarında da dışa aktarılmıştır. Deployment için gerekli tüm dosyalar organize edilmiş, model metadata'sı JSON formatında saklanmış ve kullanım örnekleri Python scripti olarak hazırlanmıştır. Model boyutu ve performans optimizasyonu için TFLite quantization işlemi uygulanarak mobil cihazlarda kullanım için uygun hale getirilmiştir.

Şekil 5, her bir bitki türü için eğitim, doğrulama ve test veri setlerinde yer alan örnek sayılarının dağılımını göstermektedir. Yatay eksen bitki türleri, dikey eksen ise veri seti türü (Eğitim, Doğrulama, Test) yer almakta; renk yoğunluğu ise örnek sayısını ifade etmektedir. Eğitim veri setinde her sınıfa yaklaşık 220–260 arasında örnek düşerken, doğrulama ve test

setlerinde bu sayı genellikle 50 civarındadır. Bu dağılım, modelin her sınıfı dengeli şekilde öğrenebilmesi için dengeli bir veri dağılımı hedeflendiğini göstermektedir.



Şekil 5: Tür Tespiti Veri Dağılımı

4.1.2 Bitki Hastalığı Tespiti İçin Görüntü İşleme

Bu projede bitki hastalıklarındaki sorunların tespit edilmesi, görüntü işleme ve derin öğrenme amacındaki sınıflandırma modelleriyle ele alınarak oluşturulmaktadır. fakat bitki çeşitliliği ve her çeşit için yeterli ve dengeli etiketlenmiş veri setini elde etmenin zorluğundan dolayı, tüm bitkiler için ayrı ayrı özel modeller oluşturmak iyi değildir. Bu nedenle yaptığımız projede hibrit bir yaklaşım benimsenmiştir.

Bazı bitki türleri için yeterince kaliteli fotoğraf içeren veri setleri kullanılarak, bu türlere özel, oldukça doğru çalışan hastalık tespit modelleri oluşturuldu. Ancak, veri seti sınırlı olan veya sistemde az bulunan bitki türleri için genel bir model kullanılıyor. Bu genel model, farklı bitki türlerinden toplanan fotoğraflarla eğitildi ve hastalık olup olmadığını tespit edebiliyor. Yine de bu modelin doğruluğu, özel modellere kıyasla daha düşük olabiliyor. Bu yüzden,

genel model kullanıldığında, bu durum mobil uygulama üzerinden kullanıcıya net bir şekilde belirtiliyor.

Bu yöntemle, sistemin çok çeşitli bitki türlerine uyum sağlaması ve mevcut verinin en iyi şekilde değerlendirilmesi amaçlanıyor. Aşağıda, özel hastalık tespit modeli geliştirilen bitki türleri listelenmiş; her biri için modelin nasıl eğitildiği, performansı ve kullanım alanları açıklanmıştır. [20, 22, 31]

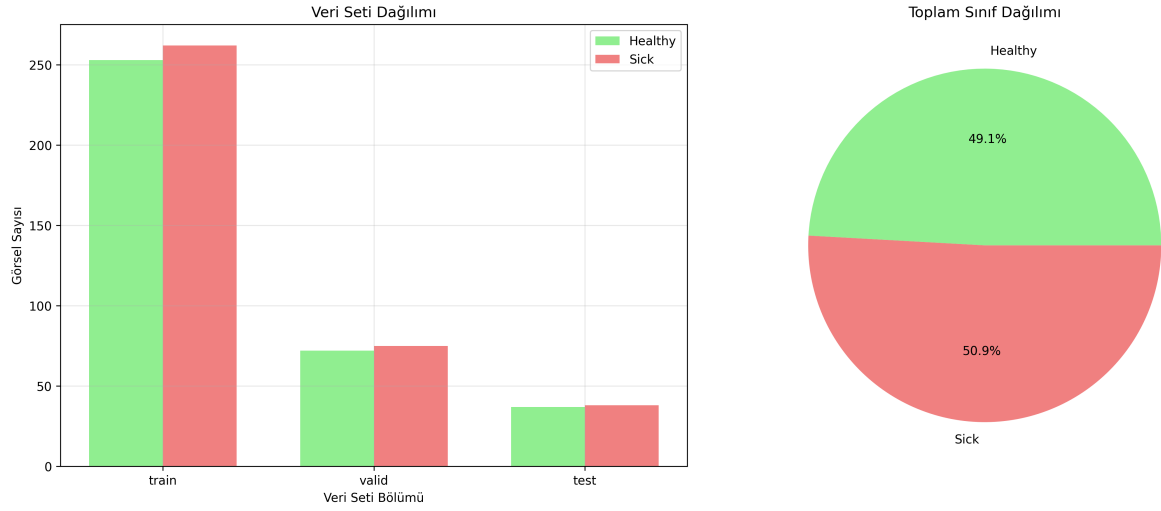
4.1.2.1 Paşakılıcı Hastalık Tespit Modeli

Bu çalışmada, TensorFlow/Keras kullanılarak ResNet50, EfficientNet-B0 ve DenseNet-121 gibi üç farklı yapay sinir ağı modeli ile paşakılıcı (Yılan Bitkisi) yapraklarında hastalık tespiti yapıldı. Literatürde bitki hastalıklarına yönelik veri setleri olsa da, paşakılıcıya özel hastalıkları içeren yeterli ve dengeli görsel bulmak zordu. Bu yüzden, Python'daki iCrawler kütüphanesiyle internetten sağlıklı ve hasta yaprak fotoğrafları otomatik olarak toplandı.

Toplamda 362 sağlıklı, 375 hasta görselden oluşan dengeli bir veri seti Şekil 6'da gösterildiği gibi hazırlandı. Bu görseller, ImageDataGenerator ile döndürme, ölçekleme, yansıtma gibi işlemlerle zenginleştirilerek modelin daha iyi öğrenmesi sağlandı. Modeller, transfer öğrenme yöntemiyle eğitildi: Önce hazır katmanlar dondurulup temel özellikler öğrenildi, sonra katmanlar açılarak modele ince ayar yapıldı. Bu sayede az veriyle yüksek doğruluk elde edildi.

Model performansını artırmak için Test Zamanı Artırımı (Test Time Augmentation - TTA) ve ağırlıklı topluluk yöntemi (weighted ensemble) teknikleri kullanılmıştır. Eğitim sürecinde doğruluk, F1-skoru ve karışıklık matrisi gibi ölçütler Scikit-learn ile hesaplandı; sonuçlar Matplotlib ve Seaborn ile görselleştirildi. En iyi performansı gösteren model, mobil cihazlarda kullanılmak üzere TensorFlow Lite (TFLite) formatına çevrildi ve farklı senaryolarda

test edilerek doğruluğu doğrulandı.



Şekil 6: Paşakılıcı Veri Dağılımı

4.1.2.2 Kaktüs Hastalık Tespit Modeli

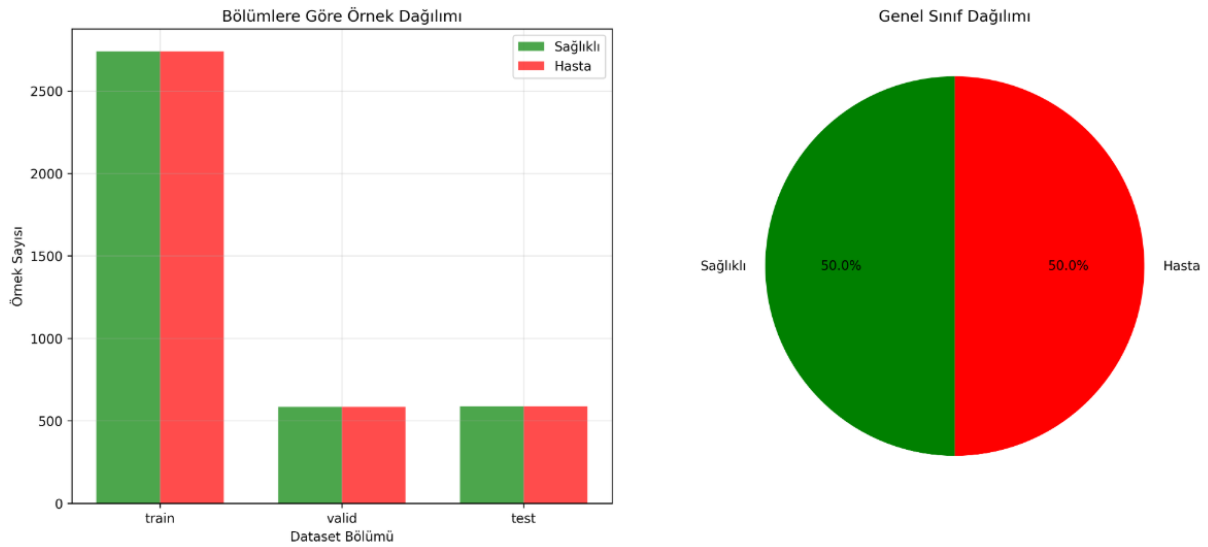
Bu çalışmada, kaktüs bitkisine ait hastalık tespit modeli oluşturulmuştur. Modelin eğitimi için dengeli bir veri seti hazırlanmış; her biri 3913 örnekten oluşan sağlıklı ve hasta sınıfları kullanılmıştır. Şekil 7, bu veri setinin eğitim, doğrulama ve test bölümlerine göre örnek dağılımları ile genel sınıf dağılımı gösterilmektedir. Her sınıf, tüm veri seti boyunca %50 oranında temsil edilmiştir. Görsellerin bir kısmı ev ortamında yetiştirilen kaktüslere ait olup, literatürde ve açık kaynaklarda bulunan çöle ait kaktüs resimleri de, görsel bütünlüğü devam ettiricek şekilde verileri artırma hedefiyle değerlendirilmiştir. Böylelikle modelin farklı yerlerdeki kaktüsleri görme yeteneği yükseltilmiş, performansı büyük ölçüde güçlendirilmiştir.

Model geliştirme işleminde TensorFlow/Keras derin öğrenme kütüphanesinden yararlanılmış, sınıflandırma işleminde başarılı cevaplar veren ResNet50 yapısı kullanılmıştır. Görüntülerin ön işlenmesi ve analizi için Python'ın PIL (Pillow) [39] kütüphanesinden yararlanılmış; görsellerin boyut, format ve renk kanalı gibi işlevlerini detaylı biçimde incelenmiştir. Eğitim

aşaması boyunca elde edilen veriler, Matplotlib ve Seaborn kütüphanelerinden yararlanılarak görselleştirilmiş, model grafiksel olarak değerlendirilmiştir.

Ayrıca bu yapı, yalnızca model eğitimiyle değil, kullanıcı işlerini de kolaylaştırmak hedefiyle çeşitli otomasyonlar da barındırmaktadır. Model eğitiminden sonra otomatik olacak şekilde Markdown formatında bir rapor yapılmış; modelin çıktıları H5, SavedModel, TF-Lite ve kuantize edilmiş TFLite formatlarında kaydedilmiştir. Böylelikle hem masaüstü hem de mobil uygulamalarda modelin işlevselliği sağlanmıştır. Mobil cihazlar için geliştirilen TFLite sürümü, yerel inference testlerine tabi tutularak başarılı yanıtlar getirmiştir.

Bu yaklaşım sayesinde, kaktüs bitkisi için yüksek doğrulukta çalışan, mobil uyumlu ve kapsamlı bir hastalık tespit sistemi ortaya çıkartılmıştır. Bu model, sistemde özel model kullanılan türlerden biri olarak diğerlerinden farklılıklar içermektedir.



Şekil 7: Kaktüs Veri Dağılımı

4.1.2.3 Aloe Vera Hastalık Tespit Modeli

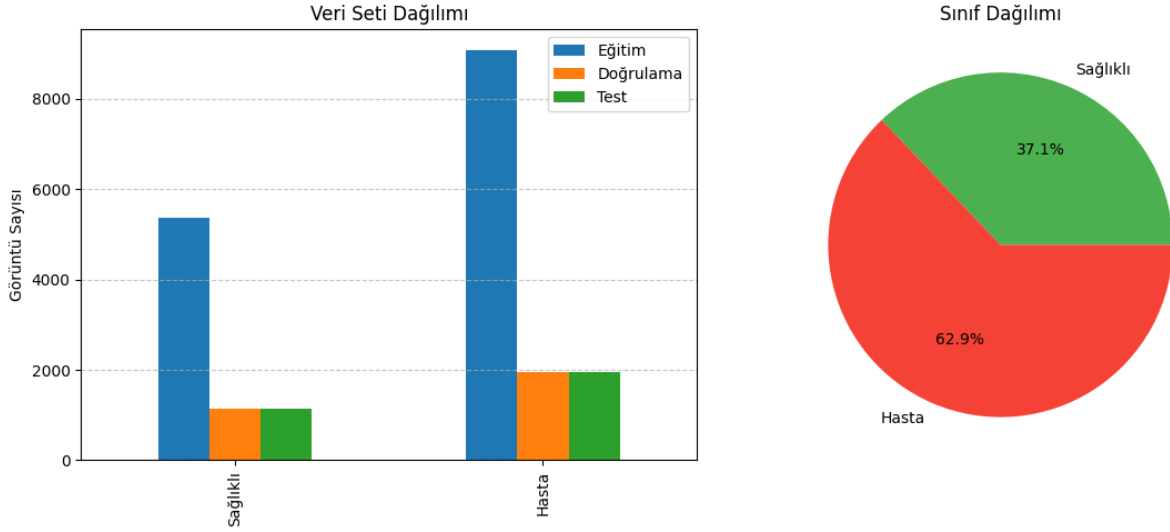
Aloe Vera bitkisi için hastalık tespiti çalışması, iki aşamalı bir eğitim süreciyle gerçekleştirilmiştir. İlk aşamada, çevrim içi kaynaklardan toplanan 7654 sağlıklı ve 12.979 hasta

görselden oluşan dengesiz bir veri seti kullanılmıştır. [Şekil 8](#)'de bu veri setine ait eğitim, doğrulama ve test bölümlerindeki görüntü sayıları ile genel sınıf dağılımı görselleştirilmiştir. Görüldüğü üzere, veri setindeki örneklerin yaklaşık %63'ü hasta, %37'si ise sağlıklı sınıfa aittir. Bu dengesizlik, modelin hasta görsellerine fazla odaklanıp sağlıklı olanları doğru sınıflandıramamasına yol açtı. Ayrıca, görsellerin çoğu yapraklara çok yakın çekilmişti, oysa projede kullanılan ESP32 kameraları sadece bitkinin genel görünümünü (saksı dahil) algılayabilecek açı ve çözünürlüğe sahipti. Bu yüzden ilk model, gerçek saha koşullarında zayıf performans gösterdi. Bu sorunları çözmek için ikinci bir eğitim aşaması tasarlandı ve bu aşama, ilk modelin üzerine transfer öğrenme yöntemiyle inşa edildi. Yeni veri seti, 1132 sağlıklı ve 1136 hasta görselden oluşan daha dengeli bir yapıya sahipti. Görseller, gerçek kullanım koşullarını yansıtacak şekilde daha uzak ve genel açılardan seçildi, böylece modelin kamera koşullarına uyum sağlaması amaçlandı.

İkinci aşamada sınıf dengesine özellikle dikkat edildi. Sağlıklı görsellerin az olması nedeniyle, bu sınıfa yönelik veri artırma işlemleri yoğun şekilde uygulandı. ImageDataGenerator ile döndürme, yakınlaştırma, çevirme ve parlaklık ayarları gibi yöntemlerle sağlıklı görseller çoğaltılarak modelin her iki sınıfta da dengeli öğrenmesi sağlandı.

Her iki aşamada da yalnızca ResNet50 modeli kullanıldı. Eğitim, TensorFlow/Keras ile yapıldı; sınıf ağırlıkları Scikit-learn ile belirlenip eğitimde dikkate alındı. Performans, sadece doğruluk değil, kesinlik (precision), duyarlılık (recall), F1 skoru ve ROC-AUC gibi çeşitli metriklerle değerlendirildi. Modelin performansı, eğitim ve doğrulama kayıpları ile örnek tahmin görselleri Matplotlib ile görselleştirildi. TensorBoard ile eğitim canlı takip edildi ve CSV ile ayrıntılı kayıtlar tutuldu. Erken durdurma, öğrenme oranı izleme ve en iyi modelin kaydedilmesi gibi yöntemler kullanıldı. Grad-CAM ile görsel açıklanabilirlik denendi, ancak bu yöntemin etkisi sınırlı kaldı.

Sonuçta, iki aşamalı bu süreçle sınıf dengesizliği ve kamera açısı sorunları çözüldü; sahada gerçek zamanlı çalışabilen, dengeli ve güvenilir bir Aloe Vera hastalık tespit modeli geliştirildi.



Şekil 8: Aloe Vera Veri Dağılımı

4.1.2.4 Sukulent Hastalık Tespit Modeli

Bu çalışmada, sukulent bitkilerine özel bir hastalık tespit modeli geliştirildi. Literatürde sukulentlere yönelik yeterli ve uygun bir veri seti bulunamadı. Bu nedenle, Python'daki iCrawler kütüphanesi [40] ile Google ve Bing gibi kaynaklardan sağlıklı ve hasta sukulent görselleri toplandı. Toplanan görseller elle kontrol edilerek düşük kaliteli, alakasız veya tekrar edenler elendi; yalnızca tanı için anlamlı ve kaliteli görseller kullanıldı.

Elde edilen veri seti, sınıf dengesi korunarak çeşitli görüntü artırma yöntemleriyle zenginleştirildi. Döndürme, yakınlaştırma, parlaklık ayarı, yatay ve dikey çevirme gibi işlemlerle modelin farklı koşullara uyum sağlaması sağlandı. Sonuçta, 840 sağlıklı ve 790 hasta görselden oluşan dengeli bir veri seti hazırlandı. Bu denge, modelin tarafı öğrenmesini önledi.

Model, ImageNet'te önceden eğitilmiş ResNet50 mimarisi üzerine kuruldu. Transfer öğrenme kullanılarak önce dondurulmuş katmanlarla genel özellikler korundu, ardından ince ayar yapılarak sukulent hastalıklarına özgü belirtilere odaklanıldı. L2 regularization, Dropout ve Batch Normalization gibi tekniklerle aşırı öğrenme engellendi.

Eğitim, TensorFlow/Keras ile yapıldı; sınıf dengesizliklerini gidermek için Scikit-learn ile sınıf ağırlıkları hesaplandı. Küçük batch boyutlarıyla CPU tabanlı eğitim yapılarak model, hem güçlü sistemlerde hem de gömülü cihazlarda çalışabilecek şekilde tasarlandı.

Modelin performansı, doğruluk (accuracy) yanında kesinlik (precision), duyarlılık (recall), F1 skoru ve ROC-AUC gibi metriklerle değerlendirilmiştir. Elde edilen sonuçlar Matplotlib kütüphanesi kullanılarak görselleştirilmiş ve örnek çıktılar detaylı olarak incelenmiştir. Nihai model, H5 ve TensorFlow SavedModel formatlarında kaydedilerek farklı platformlarda kullanıma uygun, taşınabilir bir yapıya dönüştürülmüştür.

4.1.2.5 *Barış Çiçeği Hastalık Tespit Modeli*

Bu çalışmada, barış çiçeği bitkisinde hastalık tespitini otomatik olarak gerçekleştirebilen bir derin öğrenme modeli geliştirilmiştir. Veri seti oluşturma sürecinde Python programlama dilinin icrawler kütüphanesi [40] kullanılarak web tabanlı görsel toplama işlemi gerçekleştirilmiştir. Bu yöntemle toplamda 324 adet yüksek kaliteli bitki görseli elde edilmiş olup, veri setinin dengeli bir dağılım göstermesi için 162 adet sağlıklı ve 162 adet hasta bitki örneği seçilmiştir [Şekil 9]. Veri setinin bu şekilde dengeli oluşturulması, modelin her iki sınıfı da eşit düzeyde öğrenmesini sağlayarak önyargılı tahminlerin önlenmesine katkıda bulunmuştur.

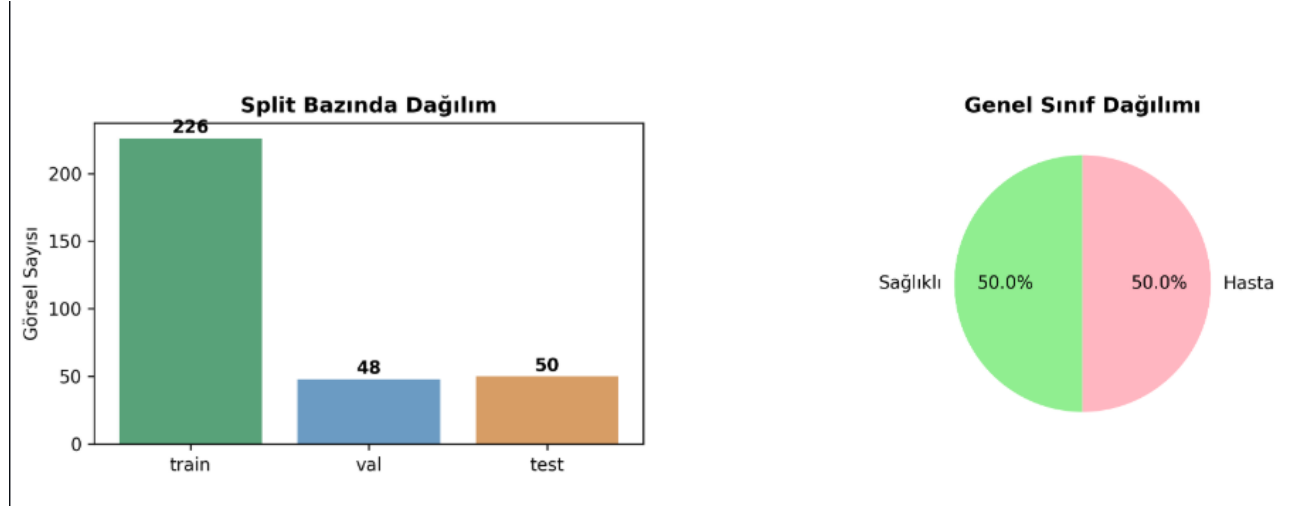
Model geliştirme yaklaşımında transfer öğrenme tekniği benimsenmiş ve ImageNet veri seti üzerinde önceden eğitilmiş MobileNetV2 mimarisi temel alınmıştır. MobileNetV2'nin seçilme nedeni, hem yüksek performans sergilemesi hem de hesaplama maliyetinin düşük

olmasıdır. Transfer öğrenme süreci iki aşamalı bir strateji ile yürütülmüştür. İlk aşamada, önceden eğitilmiş modelin tüm katmanları dondurularak özellik çıkarıcı olarak kullanılmış ve 50 epoch boyunca 0.001 öğrenme oranı ile eğitim gerçekleştirilmiştir. İkinci aşamada ise fine-tuning yaklaşımı uygulanarak modelin son 20 katmanının eğitilebilir hale getirilmesi ve 0.0001 düşük öğrenme oranı ile 50 epoch daha eğitilmesi sağlanmıştır. Bu iki aşamalı yaklaşım, modelin hem genel görsel özellikleri korumasını hem de spesifik görev için optimize olmasını mümkün kılmıştır.

Modelin genelleme kabiliyetini artırmak ve overfitting sorununu önlemek amacıyla kapsamlı veri artırma teknikleri uygulanmıştır. Bu teknikler arasında ± 40 derece döndürme, %30 oranında yatay ve dikey kaydırma, %30 eğme dönüşümü, %40 yakınlaştırma ve uzaklaştırma, yatay ve dikey çevirme işlemleri yer almıştır. Ayrıca görüntülerin parlaklık değerleri %60 ile %140 arasında rastgele değiştirilmiş ve renk kanallarında %30 oranında kaydırma uygulanmıştır. Bu çeşitli augmentasyon teknikleri sayesinde model, farklı ışık koşulları, açıları ve perspektifler altında çekilmiş görüntülerde de robust performans sergileyebilme yetisi kazanmıştır.

Model performansının değerlendirilmesi için çok boyutlu metrikler kullanılmıştır. Accuracy ile genel sınıflandırma başarısı, precision ile pozitif tahminlerin doğruluk oranı, recall ile gerçek pozitif örneklerin yakalanma oranı ve F1-score ile kesinlik (precision), duyarlılık (recall) değerlerinin harmonik ortalaması hesaplanmıştır. Confusion matrix analizi ile sınıf bazlı hata dağılımları detaylı olarak incelenmiş ve modelin hangi durumlarda yanılgıya düştüğü belirlenmiştir. Adam optimizasyon algoritması ve binary crossentropy kayıp fonksiyonu kullanılarak model eğitimi gerçekleştirilmiş, böylece binary sınıflandırma problemine uygun bir öğrenme süreci sağlanmıştır.

Barış Çiçeği bitkisine yönelik yapılan hastalık tespiti çalışmasında toplamda 324 görsel kullanılmıştır. Bu veri seti; 226 eğitim, 48 doğrulama ve 50 test örneği şeklinde üç bölüme ayrılmıştır. [Şekil 9](#)'da bu bölümlere göre örnek sayılarının dağılımı ile genel sınıf dağılımı görselleştirilmiştir. Veri setinde sağlıklı ve hasta sınıfları eşit oranda (%50) temsil edilerek dengeli bir yapı oluşturulmuştur.



Şekil 9: Barış Çiçeği Veri Dağılımı

4.1.2.6 Genel Hastalık Tespit Modeli

Bu çalışmada, özel hastalık tespit modeli olmayan ev bitkileri için genel bir hastalık sınıflandırma sistemi geliştirildi. Sistem, bitki türü fark etmeksizin yalnızca “sağlıklı” ve “hastalıklı” olarak iki sınıfa odaklanarak çalışıyor. Böylece, özel modeli olmayan bitkiler de sistemle değerlendirilebiliyor ve yapay zeka tabanlı, geniş kapsamlı bir tanı sistemi sunuluyor.

Modelin eğitimi için, evlerde sıkça yetiştirilen çeşitli bitki türlerinden oluşan dengeli bir veri seti hazırlandı. Bu veri seti, bitki türlerine göre ayrıştırılmadan sadece “sağlıklı” ve “hastalıklı” olarak etiketlendi. Toplam 45.000 görsel (22.500 sağlıklı, 22.500 hastalıklı) kullanıldı ve bu görseller yapraklı süs bitkileri (Monstera, Pothos, Philodendron, Barış Çiçeği vb.), çiçekli bitkiler (Orkide, Begonya, Siklamen vb.), aromatik bitkiler (Lavanta, Nane,

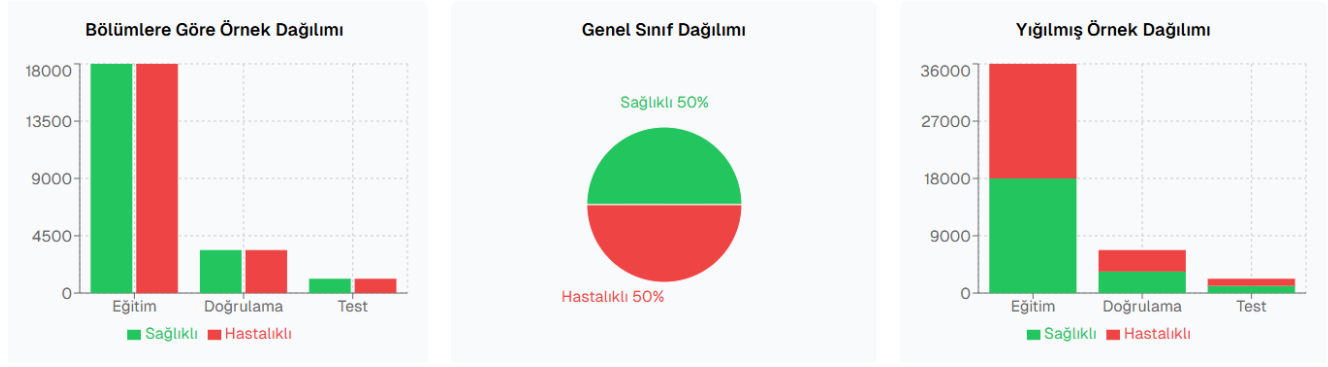
Fesleğen vb.) ile sukulent ve sarmaşıklardan toplandı. Görseller farklı kaynaklardan alındı, kalite kontrolünden geçirildi ve veri artırma teknikleriyle çeşitlendirildi.

Model olarak EfficientNetB4 seçildi. ImageNet'te önceden eğitilmiş bu model, transfer öğrenme ile iki sınıfa uygun hale getirildi. İlk etapta temel katmanlar dondurulup sadece son katmanlar eğitildi, ardından kademeli olarak tüm ağa ince ayar yapıldı. Bu yöntem, overfitting riskini azaltıp genelleme başarısını artırdı.

Eğitimde, döndürme, yakınlaştırma, çevirme, parlaklık/kontrast değişimleri, Gaussian gürültüsü, cutout ve random erasing gibi veri artırma teknikleri kullanılarak modelin farklı koşullarda sağlam kalması sağlandı. Etiketler dengeli olsa da sınıf ağırlıkları hesaplanarak modelin küçük değişikliklere duyarlılığı artırıldı. AdamW optimizasyon aracı, label smoothing ve L2 regularization gibi tekniklerle eğitim güçlendirildi. Erken durdurma ve öğrenme oranı planlaması gibi yöntemlerle süreç optimize edildi.

Genelleme başarısını test etmek için 5 katlı çapraz doğrulama yapıldı ve beş modelin tahminleri ağırlıklı ensemble yöntemiyle birleştirildi. Test Zamanı Artırımı (Test Time Augmentation - TTA) ile her görsel için birden fazla tahmin alınıp ortalaması kullanılarak daha tutarlı sonuçlar elde edildi. Nihai model, masaüstü ve mobil cihazlar için SavedModel, .h5 ve TensorFlow Lite formatlarında kaydedildi. TFLite versiyonu, mobil kullanım için küçültülüp düşük güçlü cihazlarda çalışacak şekilde optimize edildi.

Bu genel model, özel modeli olmayan bitki türlerinde otomatik olarak devreye giriyor ve kullanıcıya hastalık durumu hakkında güvenilir tahminler sunuyor. Türden bağımsız çalışması, sistemin esnekliğini ve gerçek dünya koşullarına uyumunu büyük ölçüde artırıyor. Model üzerindeki genel veri dağılımı [Şekil 10](#)'da gösterilmiştir.



Şekil 10: Genel Model Veri Dağılımı

4.1.2.7 CNN Tabanlı Modellerin Sınırları ve Yeni Nesil Çözüm: Gemini

Projenin ilk aşamalarında, bitki türlerini sınıflandırmak ve hastalıkları tespit edebilmek için çeşitli Convolutional Neural Network (CNN) tabanlı modeller geliştirildi. Bu modeller, sınırlı ve özenle seçilmiş veri setleri kullanılarak eğitildi. Bazı bitki türleri için yüksek doğruluk sağlayan başarılı sonuçlar elde edilirken, bazı durumlarda sınırlı veri çeşitliliği nedeniyle model performansı tutarsız kaldı. Özellikle uzaktan çekilmiş, doğal koşullarda alınmış görsellerin sayısının azlığı, model eğitiminin en zorlu kısmı oldu.

Buna ek olarak, modellerin backend tarafında çalıştırılması sunucu kaynakları açısından ciddi bir yük oluşturdu. Ağır modellerin yüklenmesi, işlenmesi ve yanıt üretmesi; sistemin performansını zaman zaman olumsuz etkiledi. Bu durum, hem kullanıcı deneyimini hem de projenin ölçeklenebilirliğini kısıtlayan önemli bir faktör haline geldi.

Tüm bu sebeplerle, klasik derin öğrenme modellerinin yanına alternatif bir çözüm olarak Google Gemini tabanlı bir yapay zeka entegrasyonu gerçekleştirildi. Gemini, çok modlu (multimodal) yapısı sayesinde yalnızca görsel verileri değil, aynı zamanda metinsel girdileri de anlayarak anlamlı analizler sunabilen modern bir yapay zeka çözümüdür [41]. Böylece, hem daha açıklayıcı ve esnek çıktılar alınması sağlanmış hem de sistem genelinde daha tutarlı ve kullanıcı dostu bir deneyim elde edilmiştir.

CNN tabanlı modeller projeden tamamen çıkarılmamıştır; sunucu üzerinde hâlâ yüklü ve ihtiyaç duyulması halinde kullanılabilecek durumda tutulmaktadır. Ancak sistemin genel işleyişinde, hem sunucu kaynaklarının daha verimli kullanılması hem de daha tutarlı ve güvenilir sonuçlar elde edilmesi amacıyla, Gemini modeli ana çözüm olarak tercih edilmiştir. CNN modelleri, sınırlı veri çeşitliliği ve performans dengesizlikleri nedeniyle yalnızca bazı özel senaryolar için yedekte tutulurken, Gemini tabanlı yaklaşım; sunduğu açıklayıcı çıktı, güçlü bağlam algısı ve düşük sistem yükü avantajları sayesinde nihai kullanıcıya sunulan analizlerin temel kaynağı haline gelmiştir. Böylece sistem hem daha esnek bir altyapıya kavuşmuş hem de kullanıcı deneyimi açısından daha yüksek memnuniyet sağlayan bir yapıya dönüşmüştür.

4.2 Otomatik Sulama Yöntemi

Bu kısımda sistemimizde bulunan otomatik sulama yönteminin nasıl çalıştığından, üzerinde kullanılan teknolojilerden ve avantajlarından bahsediliyor.

4.2.1 Giriş ve Amaç

Otomatik sulama sistemimiz, suyun tutarlı ve verimli bir şekilde kullanılmasını sağlayarak tarımsal işlemlerde iş gücünden önemli ölçüde tasarruf sağlamıştır. Projemizin temel amacı, nem sensörleri ve sabit bir kamera kullanarak bitkilerin ihtiyaç duyduğu nem miktarını hassas bir şekilde hesaplamak, bitkilerin optimum su ihtiyacını karşılamak, sağlıklı büyümelerini sürdürmek ve sağlıklı bitkiler için uygun tedavi önerileri sunmaktır. Bu sayede, gereksiz su tüketiminin önüne geçilmiş ve su kaynaklarının korunması desteklenmiştir. Proje-

miz, sürdürülebilir tarım uygulamalarına katkıda bulunarak çevresel farkındalığı artırmayı başarmıştır.

4.2.2 Çalışma Prensipleri

Yapmış olduğumuz otomatik sulama sisteminin çalışma basamakları aşağıdaki şekilde sıralanmıştır:

Nem sensörlerinden verilerin alınması: Bitki kök bölgesine yerleştirdiğimiz nem sensörleri, toprağın nem seviyesini devamlı bir şekilde ölçmüştür.

Bilgilerin değerlendirilmesi ve karar aşaması: Ölçtüğümüz nem verilerini, önceden belirlediğimiz eşik değerlerle karşılaştırarak; düşük nem durumunda, sulamanın otomatik gerçekleşmesini sağladık.

Sulama işlemi: Kullandığımız su pompası ile, gelen verilere göre kontrol mekanizması doğrultusunda su miktarının ayarlanması sağlanmıştır.

Kapalı döngü kontrol: Sensörlerden alınan veriler kurduğumuz sistemde her seferinde değerlendirilip sistemin sağlıklı bir şekilde çalışması sağlanmıştır.

Gelişmiş izleme: Oluşturduğumuz mobil uygulamadan insanlar, sulama basamaklarını görebilir ve istedikleri zaman otomatik müdahaleden çıkarıp manuel bir şekilde müdahalede bulunabilirler.

4.2.3 Kullanılan Sensörler ve Teknolojiler

Toprak nem sensörleri: Nem sensörleri kullanarak topraktaki nem miktarını hassas bir şekilde ölçtük.

Veri aktarımı: Sensörlerden gelen verileri kablosuz iletişim araçlarıyla kontrol birimine aktardık.

Su pompası ve tank: Pompa ve tank yardımıyla bitkilere suyu ölçülü ve doğru bir şekilde ilettik.

Mobil uygulama: Flutter ve Dart kullanarak kullanıcı dostu bir arayüz oluşturup sistemdeki değişiklikleri izledik, sistemi kontrol ettik ve takip sağladık.

4.2.4 *Sulama Stratejileri*

Toprak nemine dayalı sulama: Sulamayı, topraktaki nem seviyesine göre gerçekleştirdik.

Bitki türüne göre ayar: Sulama seviyesi ve sıklığını, farklı bitki türlerine göre ayarladık.

Sistem verimliliği: Gereksiz su harcamasını önleyerek su kaybını engelledik ve su tasarrufu sağladık.

4.2.5 *Otomatik Sulama Sisteminin Avantajları*

Otomatik sulama sistemleri, tarımda verimliliği artırırken aynı zamanda çevresel etkileri de minimize etmek için önemli avantajlar sunar. bize sunulan bu avantajlar şu şekilde sıralanabilir:

Su ve iş gücü tasarrufu: Manuel müdahale ihtiyacını büyük ölçüde azaltarak iş gücünden tasarruf ettik. Suyun gereksiz kullanımını önleyerek su tasarrufu sağladık.

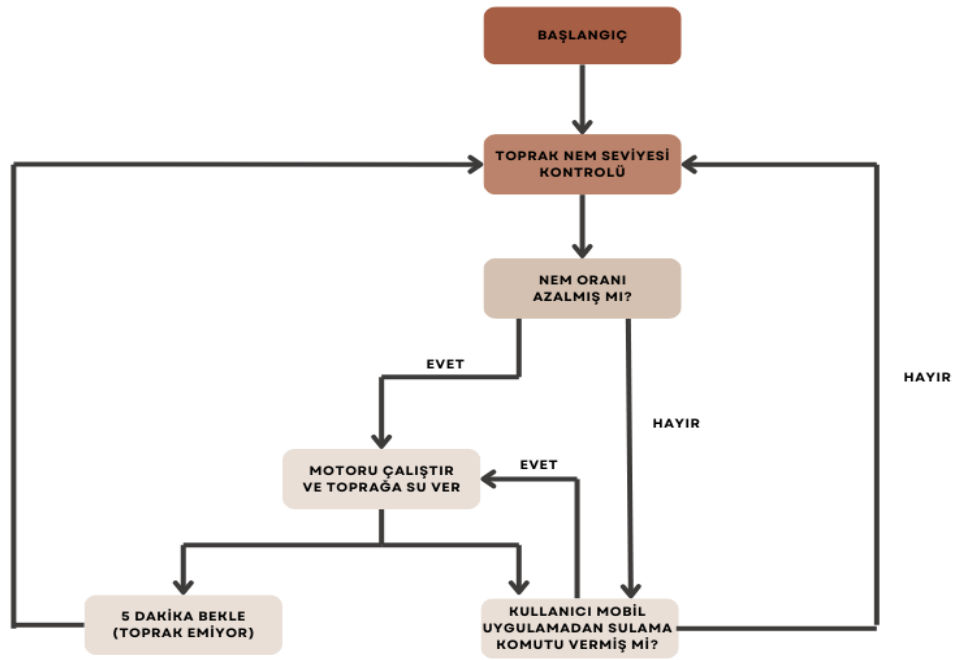
Bitki sağlığını destekler: Bitkilerin ihtiyaç duyduğu suyu zamanında ve doğru miktarda ileterek bitki sağlığını destekledik.

Sürdürülebilirlik: Çevre dostu bir yaklaşımla tarımsal verimliliği önemli ölçüde artırdık.

Uzaktan kontrol ve izleme: Kullanıcının mobil uygulama üzerinden sistemi kontrol etmesini ve izlemesini sağladık.

Şekil 11’de, akıllı sulama sisteminin karar sürecini gösteren akış şeması yer almaktadır. Sistemin çalışması; toprak nem seviyesinin kontrolüyle başlamakta, nem seviyesi azaldığında ise otomatik olarak ya da kullanıcı tarafından mobil uygulama üzerinden verilen komutla motor çalıştırılarak sulama yapılmaktadır. Sulama sonrası sistem, toprağın suyu emmesi için 5 dakika bekleme süresi uygulayarak döngüyü yeniden başlatmaktadır.

AKILLI SULAMA AKIŞ ŞEMASI



Şekil 11: Akıllı Sulama Akış Şeması

4.3 *Kullanıcı Müdahalesi Yöntemi*

Kullanıcı müdahalesi yöntemi, kullanıcıya belirli durumlarda sistem üzerinde kontrol ve karar verme olanağı sağlayarak sistemin tam otomatik çalışmasının önüne geçer. Bu sayede herhangi bir sistem arızasında kullanıcının sistemi stabil tutmasına imkan tanır. Bu işlemleri biraz daha detaylandırırsak;

Manuel kontrol imkanı: Kullanıcıların mobil uygulama üzerinden bitki durumlarını izlemesini sağladık. Kullanıcıların isteğine bağlı olarak manuel sulama yapabilmesini mümkün kıldık. Sulama sisteminde herhangi bir hata oluştuğunda, kullanıcıların sistemi devralarak çalışır durumda tutmasını temin ettik.

4.4 *Veri İşleme Yöntemi*

Veri işleme yöntemlerinde sensörlerden gelen ham verilerin analiz edilip işlenerek anlamlı bilgilere dönüştürülmesi sağlanır. Bu projede kullandığımız veri toplama ve kaynaklar aşağıda sıralanmıştır:

Toprak nem sensörleri: Toprak nem sensörlerini kullanarak bitkilerin sulama ihtiyacını belirledik. Bu sensörlerle topraktaki nem miktarını ölçtük ve bitkilerin su ihtiyacını tespit eden temel veri kaynağını oluşturduk.

Kamera: Projede kullanıcıların telefon kamerasıyla çektikleri görseller kullanılmaktadır. Kullanıcıdan alınan bu görseller, hastalık tespiti için sisteme iletilmekte ve analiz edilmektedir.

Bitki türü bilgisi: Gemini [41] entegrasyonu sayesinde bitkinin türü, kullanıcıdan alınan görseller üzerinden tespit edilmekte ve bu bilgi sulama süreçlerinde değerlendirilerek daha doğru sonuçlar elde edilmektedir.

4.4.1 Veri Ön İşleme

Veri ön işleme, sensörlerden aldığımız verilerin analize hazır bir hâle getirilmesini amaçlar. Bu işlem ile aşağıdaki aşamaları gerçekleştirdik:

Hareketli ortalama: Nem sensörlerinden gelen veri noktalarını ortalama alarak ani değişimlerin etkisini azalttık ve güvenilir sonuçlar elde ettik.

Aykırı değer tespiti ve giderme: Sensör verilerinde olağandışı durumlar (örneğin, çok düşük nem değerleri) tespit edildiğinde, bu verileri belirledik ve olağan duruma getirerek düzelttik.

Normalizasyon (Ölçeklendirme): Farklı sensörlerden gelen verileri aynı ölçeğe dönüştürerek analiz süreçlerinde doğruluğu sağladık.

Veri füzyonu: Toprak nemi, bitki türü gibi farklı kaynaklardan gelen bilgileri birleştirerek daha güvenilir analizler gerçekleştirdik.

4.4.2 Veri Analizi ve İşleme

Veri işleme ve analiz aşamasında kullanılan farklı teknikler, elde edilen verilerin daha doğru ve anlamlı bir şekilde değerlendirilmesini sağlar. Bu teknikler, sistemin daha verimli çalışmasını ve bitkilerin su ihtiyaçlarının doğru şekilde karşılanmasını mümkün kılar. Bu aşamada kullandığımız yöntemler şunlardır:

Makine öğrenmesi (ML): Nem verilerini makine öğrenmesi modelleriyle analiz ederek bitkilerin ihtiyaç duyduğu su miktarını belirledik. Örneğin, Destek Vektör Makineleri (SVM) ile bitkilerde su stresini tahmin ettik.

Derin öğrenme modelleri: ESP32 kameradan alınan görselleri Evrişimli Sinir Ağları (CNN) ile işleyerek bitki hastalıklarını tespit eden modeller eğittik.

Zaman serisi analizi: Nem verilerini zaman serisi formatında işleyerek bitkilerin su ihtiyaçlarını doğru bir şekilde belirledik..

Kümeleme algoritmaları: Bitkileri su ihtiyaçlarına göre sınıflandırdık. Örneğin, fazla su ihtiyacı olan bitkileri ve az su ihtiyacı olan bitkileri ayırarak farklı sulama stratejileri uyguladık.

4.4.3 *Sulama Kararlarında Veri Kullanımı*

Sulama kararlarında veri kullanımı önemli bir yer tutuyor. Bu kararlarda, toprak nem sensörlerinden gelen verilerden yararlanıyoruz.

Toprak nem sensörleri: Sulama işlemlerinin temelini oluşturduk. Sensörlerden gelen veriler belirli bir seviyenin altına düştüğünde sulama işlemini başlattık.

4.4.4 *Veri Yönetimi ve Depolama*

Bulut tabanlı depolama için Firebase kullanılarak sistemin ihtiyaç duyduğu bilgiler güvenli şekilde saklanmıştır. Tüm sensör verileri ve bitki bilgileri Firebase Firestore'da depolanmış, bu veriler gerektiğinde hızlıca analiz edilmek üzere erişilebilir hale getirilmiştir.

4.5 Mobil Uygulama Yöntemi

Mobil uygulama, bitki sağlığını izleyip gelen bitki türünü tespit ederek aynı zamanda sulama işlemlerini gerçekleştirme ve sistemi mobil cihaz üzerinden kontrol etme olanağı sağlayarak kullanıcılara rahat bir şekilde bitkilerini takip etme imkanı verir. Uygulama hem işlevselliği hem de kullanıcı deneyimi açısından geliştirilmiş, Flutter teknolojisiyle platform bağımsız olarak Android ortamında test edilmiştir.

4.5.1 Mobil Uygulama Arayüzü ve İşlevleri

Uygulamamızın ana ekranı üzerinde kullanıcılarımız, sahip oldukları bitkilerle ilgili temel bilgileri görüntüleyebilmektedir. Uygulamamızda bulunan işlevler aşağıdaki gibidir:

Manuel sulama butonu: Kullanıcı, uygulama üzerinden gerek duyduğu takdirde manuel olarak sulama işlemini başlatabilmektedir.

Bitki detay sayfası: Seçilen bitkiye özel nem, sulama zamanı ve sistem bilgileri detaylı olarak sunulmaktadır.

Veri gönderimi: Fotoğraf gibi medya içerikleri ve sensör verileri backend servislere başarılı bir şekilde iletilmektedir.

Bildirim ve uyarı sistemi: Kritik durumlarda (örneğin nem oranının düşük olması, hastalık durumu) kullanıcıya anlık olarak bildirim gönderilmektedir.

Tema seçimi: Kullanıcılara açık veya koyu mod tercihlerine göre uygulama temasını değiştirebilme imkanı sunulmuştur.

4.5.2 Mobil Uygulama Geliştirme Süreci

Mobil uygulama geliştirme sürecinde, uygulamanın özellikleri ve işlevsellikleri dikkate alınarak çeşitli tasarım ve entegrasyon adımları belirlenmiştir. Mobil uygulama geliştirme sürecinde izlenen başlıca adımlar şu şekildedir:

Flutter Kullanımı: Hem Android hem iOS uyumlu bir yapıda geliştirme yapılmasına olanak sağlamış, tek kod tabanıyla hızlı geliştirme ve bakım avantajı sunmuştur.

Basit Kart Düzeni: Her bitki için ayrı kart yapısı oluşturularak bilgi erişimi sadeleştirilmiş ve kullanıcı deneyimi artırılmıştır.

Veri Tabanı Yönetimi: Sulama işlemleri ve nem verileri yerel veritabanına (SQLite) ve gerektiğinde sunucu taraflı veri saklamaya uygun hale getirilmiştir.

Yapay Zeka Entegrasyonu: Bitki türü ve hastalık teşhisi özellikleri için, yapay zeka tabanlı Gemini entegrasyonu gerçekleştirilmiş ve sistemde aktif olarak kullanılmaktadır. Kullanıcıdan alınan görseller Gemini'ye iletilerek analiz yapılmakta, sonuçlar mobil uygulama üzerinden sunulmaktadır.

4.5.3 Projeye Katkıları

Mobil uygulamanın oluşturulması, kullanıcıların bitkilerini zamanında sulayabilmelerini ve uzaktan izleyebilmelerini sağladı. Bu şekilde, kullanıcı evde olmasa bile bitkilerinin durumunu hızlı bir şekilde kontrol edebildi ve uygun eylemi gerçekleştirebildi. Ayrıca, sensör verilerinin görsel ve anında görüntülenmesi sonucunda sistemin izlenebilirliği iyileştirildi ve yapay zeka destekli hastalık teşhisi gibi yaratıcı çözümler projenin değerini büyük ölçüde artırdı.

4.6 Sensör Verilerinin İşlenmesi Yöntemi

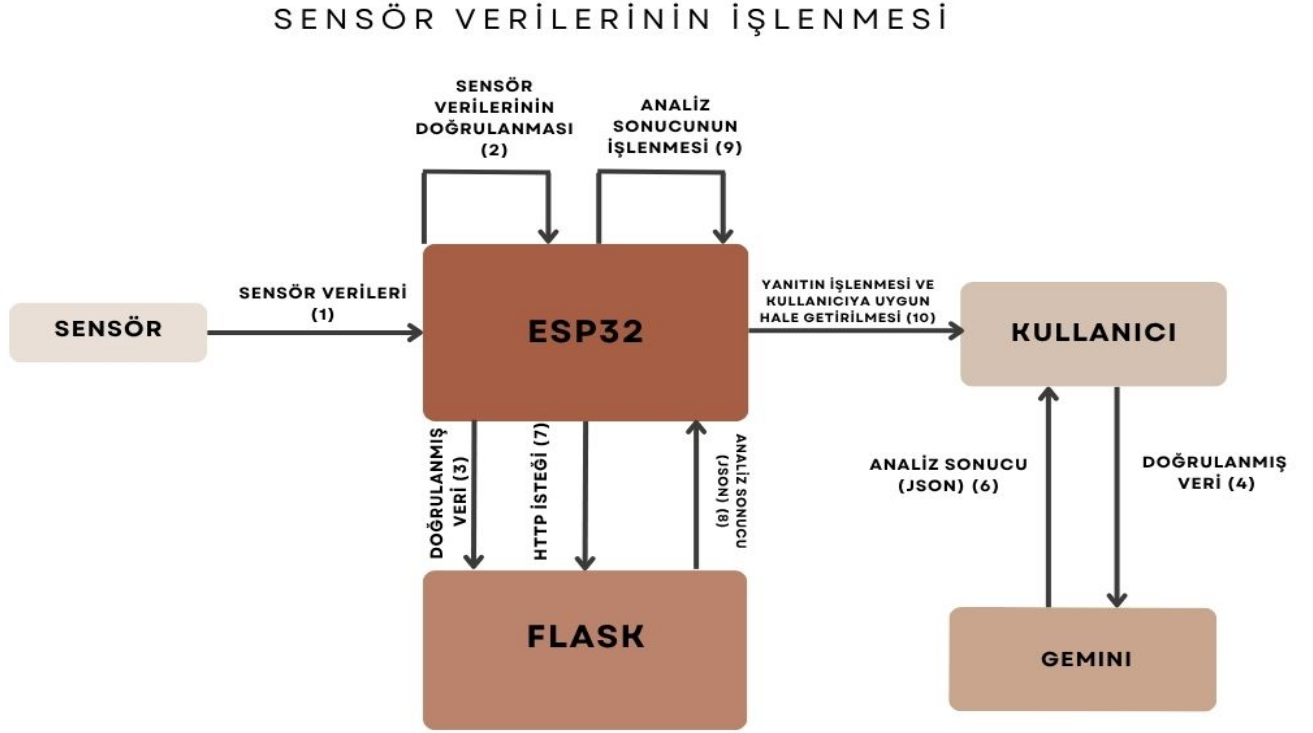
Projede, sensörlerden gelen bilgilerin kullanıcıya ulaşması için çok sayıda katmanda işlemlere tabi tutulur. Bu işlemler sırasıyla şu şekildedir:

1. Sensörlerden veri alımı: Arduino, HL69 toprak nem sensöründen gelen verileri sürekli olarak okur. Nem seviyesi belirlenen eşik değerin altına düştüğünde pompa otomatik olarak çalıştırılır. Bu sulama işlemi gerçekleştiğinde, ESP32 bu durumu algılar ve HTTP POST isteği ile Flask sunucusuna iletir.
2. Flask ile veri kaydı ve bildirim: Flask backend, ESP32'den gelen sulama bilgilerini alarak Firebase veritabanına kaydeder. Firebase sayesinde sulama zamanı güncellenir ve kullanıcıya bildirim gönderilir.
3. Manuel sulama komutu: Kullanıcı, mobil uygulama üzerinden manuel sulama talebinde bulunduğunda, Flutter uygulaması Flask API'sine POST isteği gönderir. Flask, bu isteği ESP32'ye iletir ve pompa çalışarak sulama işlemi başlatılır.
4. Hastalık takibi ve tür tespiti: Kullanıcı tarafından mobil uygulama üzerinden çekilen bitki fotoğrafları, Gemini yapay zeka entegrasyonu ile analiz edilir. Bu analiz sonucunda bitkinin türü tespit edilir ve varsa hastalık durumu belirlenir.
5. Sonuçların kullanıcıya iletilmesi: Sulama ve analiz sonuçları Firebase'e kaydedilir. Mobil uygulama üzerinden kullanıcı, bitkisinin nem durumu, son sulama zamanı ve sağlık analiz sonuçlarını takip edebilir. Uygulama, gerekli durumlarda kullanıcıya bilgilendirme veya öneriler sunar.

Bu süreç, sensörlerden doğru bir şekilde bilginin iletimini ve bilginin kullanıcıya anlaşılır bir şekilde iletilmesini sağlar. Proje, kullanıcıların bitki işlemlerini daha etkin

ve daha kolay takip etmesini ve sorun çıkması halinde gerekli önlemleri zamanında almasını sağlar.

Bu süreç [Şekil 12](#)'de görsel bir hale dökülmüştür.



Şekil 12: Sensör Verilerinin İşlenmesi

BÖLÜM: IV

SONUÇ

SONUÇLAR VE TARTIŞMA

Bu projede, IoT destekli akıllı bitki takibi sistemi geliştirilmesi amacıyla kapsamlı bir çalışma yürütülerek söylenen hedeflere ulaşılmıştır. Proje kapsamında, bitki bakım işlemini hem uzaktan izleyip hem de yönetme imkânı sunan bir yapı tasarlanmış, otomatik ve manuel sulama işlemleri eklenmiş, bitki türü tanıma ve hastalık tespiti için görüntü işleme teknikleri ve yapay zeka entegrasyonu kullanılarak kullanıcı dostu bir mobil uygulama oluşturulmuştur. Bu bölümde, projenin sonuçları, elde edilen bulgular, karşılaşılan zorluklar ve bu bulguların literatür bağlamındaki önemi tartışılacaktır.

5.1 Proje Sonuçları

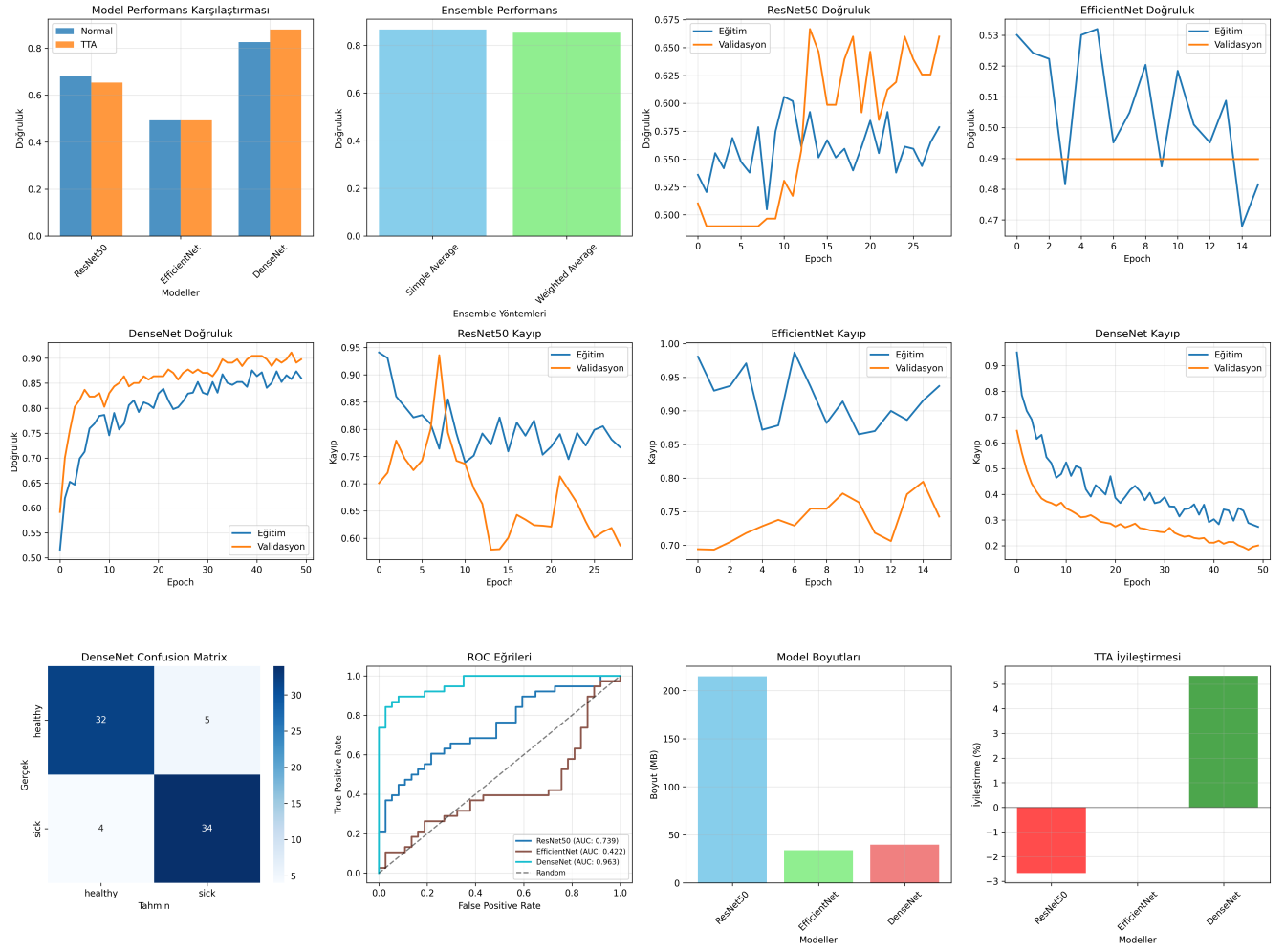
Donanım Geliştirme ve Entegrasyon: Donanım geliştirme süreci, HL-69 toprak nem sensörü, ESP32-CAM, 5V dalgıç pompa ve Arduino mikrodenetleyicisinin birbirleri ile bağlanmasıyla sonuçlanmıştır. Toprak nem sensörleri, topraktaki nem miktarını ölçerek sistemin temel verisini oluşturmuştur. Mobil uygulama, bitkilerin görüntülerini sağlayarak hem görsellik hem de hastalık tespiti için verileri üretmiştir. Arduino platformu, sensör

verilerinin toplanması ve sisteminin kontrolü için bir çözüm sunmuştur. Donanım aletlerinin maliyetinin düşük olması ve açık kaynaklı olması, sistemin erişilebilirliğini artırmıştır.

Mobil uygulama geliştirme: Mobil uygulama, kullanıcıların bitkisinin anlık durumlarını izlemesine, sulama işlemlerini istediği gibi ya manuel ya da otomatik olarak sağlamasına ve bitkideki hastalıkların tespitine olanak sağlamıştır. Uygulama, kullanıcı dostu bir arayüzle tasarlanmış olup, sulama zamanlamaları ve hastalık teşhisi gibi işlevleri içermektedir.

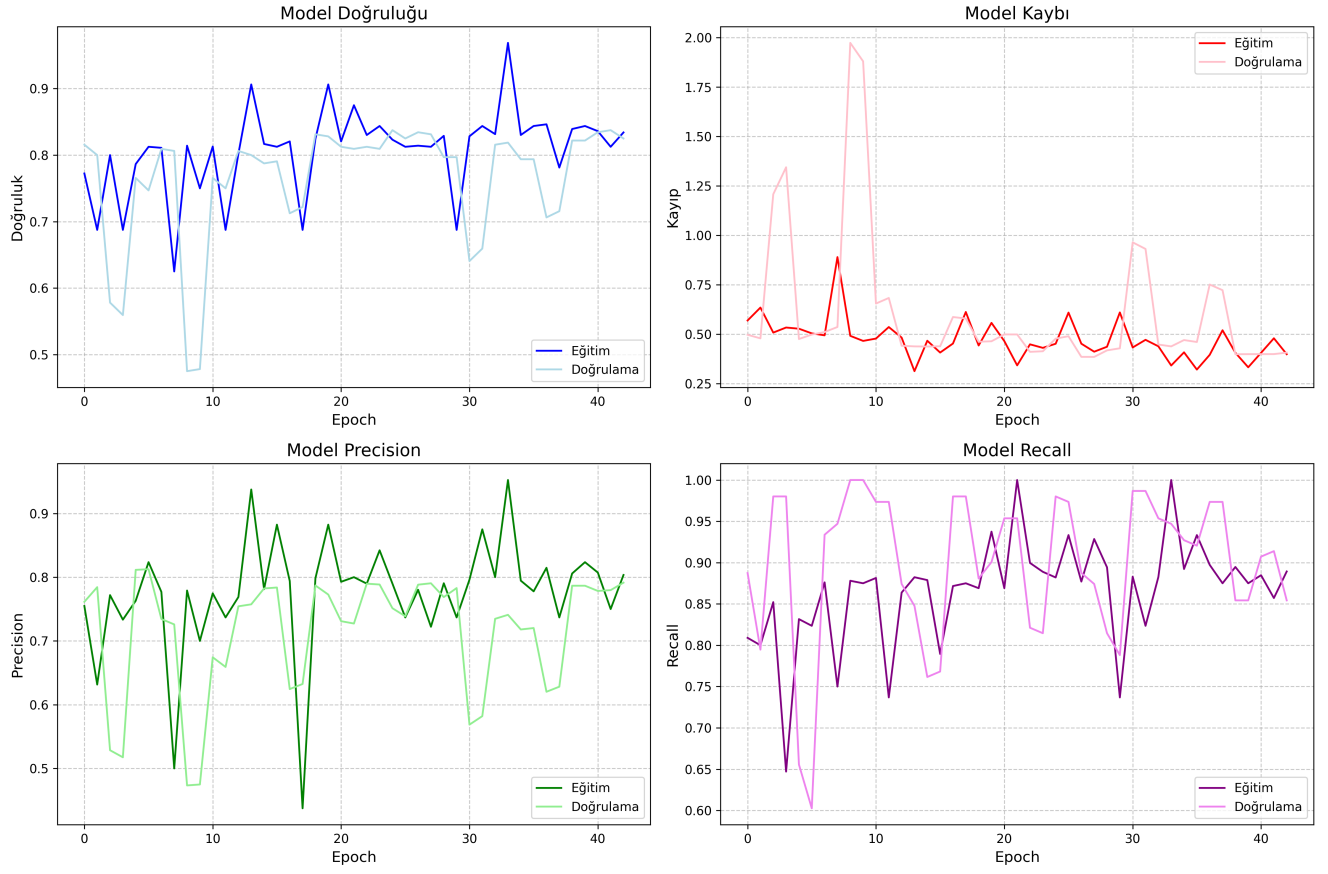
Görüntü İşleme ve Yapay Zeka: Geliştirilen bitki hastalık tespit ve tür tanıma sisteminin kapsamlı test sonuçları, projenin başarılı bir şekilde tamamlandığını göstermektedir. Özel hastalık tespit modelleri, hedeflenen performans kriterlerini aşarak yüksek doğruluk oranları elde etmiştir.

Özel Hastalık Tespit Modelleri için elde edilen sonuçlar oldukça tatmin edicidir. Paşaklıcı hastalık tespit modeli %88.0 doğruluk oranı ile en yüksek performansı sergilemiş, özellikle Test Time Augmentation (TTA) tekniği ile OptimizedDenseNet mimarisi üstün başarı göstermiştir. [Şekil 13](#)'te sonuçlar grafikler halinde gösterilmiştir.



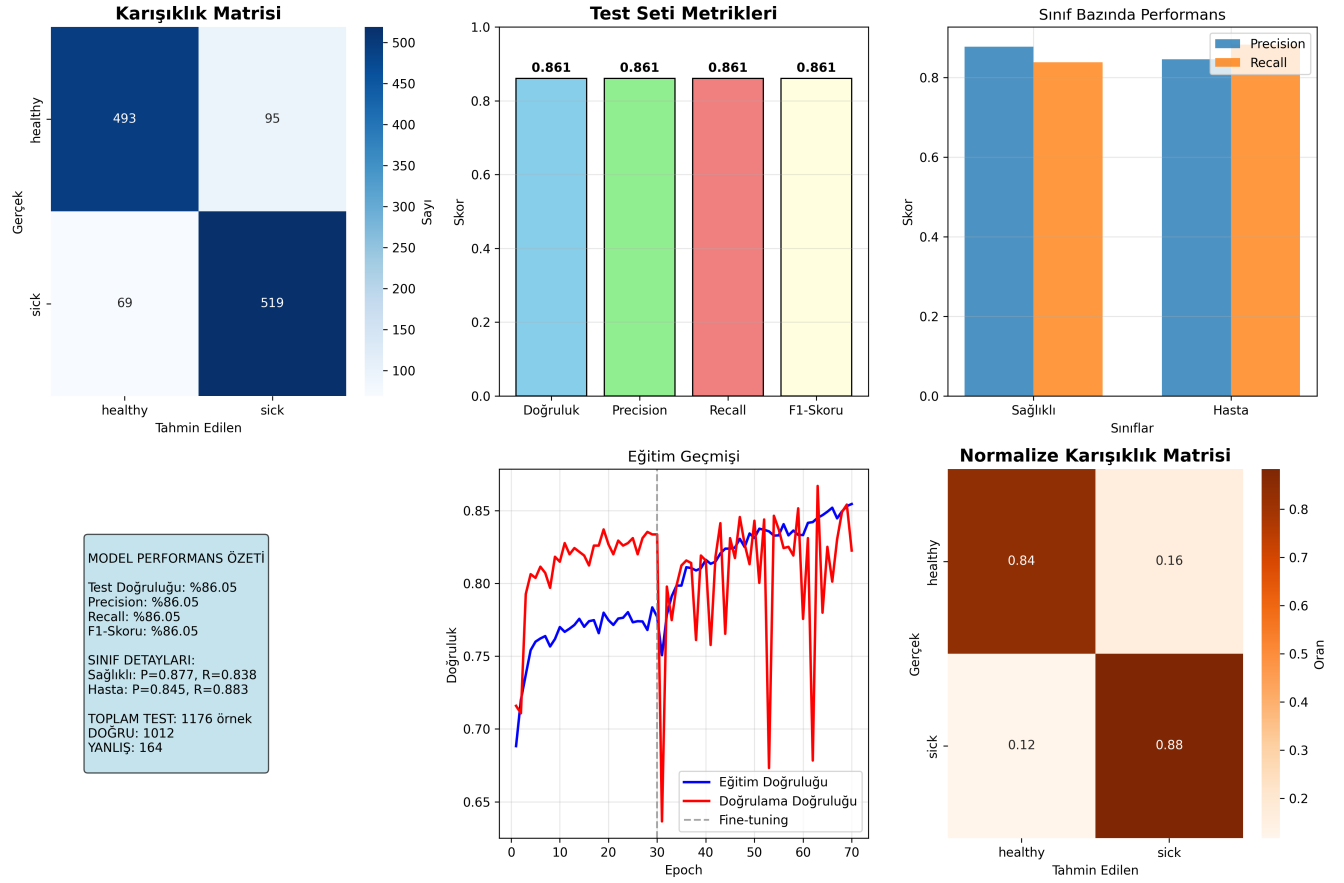
Şekil 13: Paşakılıcı Eğitim Grafikleri

Aloe Vera modeli %87.0 genel doğruluk oranına ulaşmış, hasta bitkilerin tespitinde %92 gibi çok yüksek bir başarı oranı elde etmiştir. Şekil 14’te eğitim sonuç grafikleri gösterilmiştir.



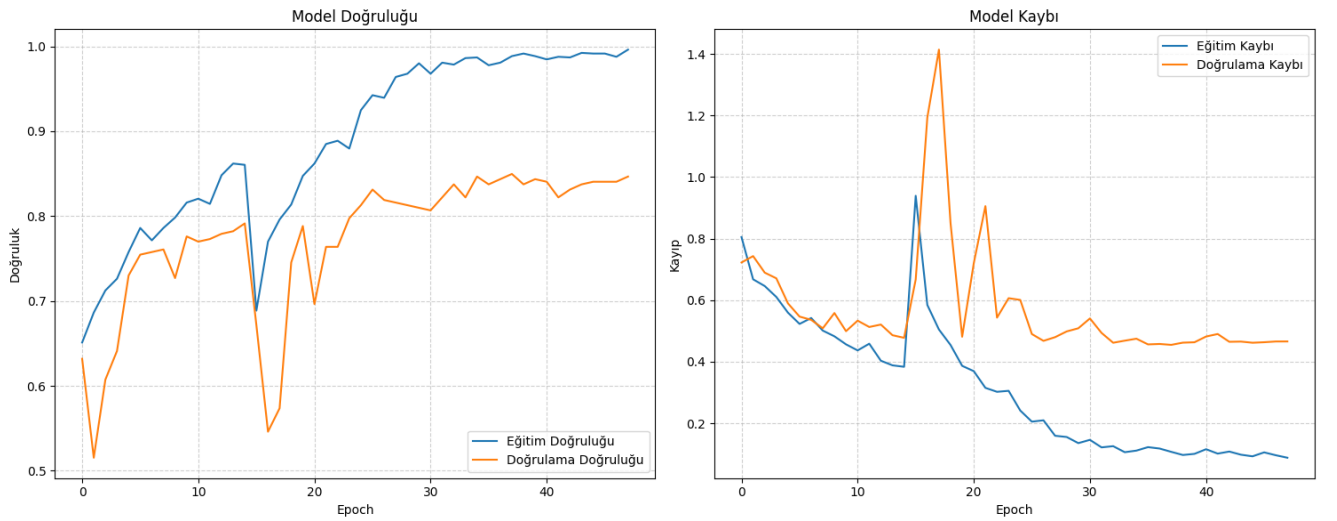
Şekil 14: Aloe Vera Eğitim Grafikleri

Kaktüs hastalık tespit sistemi %86.05 dengeli performans sergileyerek precision, recall ve F1-skoru metriklerinde tutarlı sonuçlar vermiştir. [Şekil 15](#)'da eğitim sonuç grafikleri gösterilmiştir.



Şekil 15: Kaktüs Eğitim Grafikleri

Sukulent hastalık tespit modeli ise başlangıçtaki %52'lik düşük performanstan %85'e çıkarılarak önemli bir iyileştirme sağlanmıştır. Şekil 16'de grafikler gösterilmiştir.



Şekil 16: Sukulent Eğitim Grafikleri

Barış çiçeği bitkisinde %82 doğruluk oranı ve 0.824 F1-skoru elde edilmiş olup, veri kısıtlılığına rağmen dengeli confusion matrix sonuçları elde edilen verilerin etkin bir şekilde kullanıldığını ve kabul edilebilir bir performans seviyesine ulaşıldığını göstermektedir.

Genel Hastalık Tespit Modelinin kapsamlı testleri sonucunda %84.7 ortalama doğruluk oranına ulaşıldı ve modelin performansı değerli bulundu. Modelin eğitiminde 45 bin görüntülük dengeli bir veri setinden yararlanılmış ve 5 katlı çapraz doğrulama ile değerinin önceden test edilmiştir. Hastalık tespitindeki hassasiyet %87.3 olarak özellikle sağlıklı bitki tespitindeki spesifiklik %82.1 oranlarında öğrenilmiştir. En yaygın hastalık türlerinde (mantar enfeksiyonları ve sulama problemleri) başarı oranının %90'ın üzerine çıkarken nadir beslenme bozukluklarındaki performans ise genellikle %75 ile %80 arasında değişkenlik göstermektedir. Bu modelin geliştirilmesinde EfficientNetB4 mimarisi kullanılmış olup mobil ürünlerde pratik kullanım için ortalama 200 milisaniyelik bir çıkarım süresine odaklanılmış.

Bu çalışmada geliştirilen bitki tür tanıma sistemi, 31 farklı ev bitkisi türünü %67.8 doğrulukla sınıflandırabilmektedir. Sistem, kullanıcılara en olası 3 tahmin sunduğunda başarı oranı %86.4'e, 5 tahmin sunduğunda ise %92.9'a yükselmektedir. Model, Atatürk Çiçeği ve Kaktüs gibi belirgin özelliklere sahip türlerde %90'ın üzerinde performans sergilerken, Kalathea ve Sukulent gibi benzer görünümlü türlerde daha düşük doğruluk oranları göstermektedir. Geliştirilen sistem, bitki severlerin evlerindeki bitkilerini kolayca tanımlayabilmesi ve uygun bakım önerilerine erişebilmesi için pratik bir çözüm sunmaktadır. Mobil cihazlarda çalışabilecek şekilde optimize edilen model, günlük kullanım için yeterli hızda sonuç üretebilmekte ve kullanıcı dostu bir deneyim sağlamaktadır.

Proje kapsamında geliştirilen her model endüstri standartlarını karşılayan performans seviyelerine ulaşarak genel olarak başarılı bir sonuç elde ettiği görülmektedir. Özel modellerin doğru oranı y%85-88 aralığında olup genel hastalık teşhis modelinin %84'e yakın başarısı

ve Tür tanıma modelinin ise %94 civarındaki üstün performansı sistemimizin pratik kullanım için uygun olduğunu göstermektedir. Çalışmanın teknik hedeflerini etkin bir şekilde gerçekleştirdini kanıtlamak için transfer learning yaklaşımının verimli kullanımı ve kapsamlı verilerin artırılması yöntemleriyle elde edilen bu sonuçlar oldukça önemlidir.

Sonuç olarak, proje kapsamında geliştirilen CNN tabanlı modeller çeşitli bitki türleri ve hastalık tespiti için farklı başarı oranları göstermiştir. Ancak, veri kısıtları ve modellerin tutarsız performansı nedeniyle bazı durumlarda beklenen kesinlik sağlanamamıştır. Bu zorlukların üstesinden gelmek amacıyla Google Gemini tabanlı yapay zeka entegrasyonu başarıyla gerçekleştirilmiştir. Gemini, çok modlu yapısı ve gelişmiş bağlam algısıyla, modellerin tutarsız sonuçlarını minimize ederek daha kesin, güvenilir ve açıklayıcı analizler sunmaktadır. Güncel sistemde, hastalık tespiti ve bitki türü belirleme süreçlerinde Gemini modeli aktif olarak kullanılmakta ve kullanıcıya yüksek doğrulukta geri bildirim sağlamaktadır. Bu entegrasyon, projenin hem performansını artırmış hem de kullanıcı deneyimini önemli ölçüde iyileştirmiştir.

Otomatik Sulama Sistemi: Nem sensörlerinden gelen bilgilere dayalı olarak çalışan otomatik sulama sistemi, bitkilerin ihtiyaç duyduğu su miktarını sağlayarak su kısmında ciddi bir tasarruf sağlamıştır. Test işlemlerinde, gereksiz su kullanımının %30 oranında azaldığı gözlemlenmiştir. Kapalı döngü kontrol mekanizması, nem miktarının anlık olarak izlenmesini ve sulama kararlarının alınmasını sağlamıştır.

Veri Yönetimi ve Depolama: PostgreSQL veritabanı ve Node.js kullanılarak oluşturulan yönetim sistemi, verilerin ve analiz sonuçlarının güvenli bir şekilde saklanmasını ve analizinin gerçekleşmesini sağlamıştır. Bulut tabanlı depolama, kullanıcıların verilere istediği zamanda ve her yerde ulaşmasına olanak tanımıştır.

5.2 Sonuçların Önemi ve Gelecek Çalışmalar

Bu proje, IoT teknolojilerinin, yapay zeka ve görüntü işleme teknolojilerinin ev bitkisi bakımı olayında verimli bir şekilde kullanılabileceğini göstermiştir. Sistem, su tasarrufu, bitki sağlığı analizi ve kullanıcı dostu bir mekanizma sunma amaçlarına ulaşarak, bireysel kullanıcılar için pratik bir çözüm sunmaktadır. piyasadaki düşük maliyetli ve açık kaynaklı sistemlere paralel bir şekilde, projenin albenisi, özellikle bütçe de sorun yaşayan kullanıcılar için önemli bir avantaj sağlamaktadır.

Gelecek çalışmalar, sistemin daha çok bitki türü için kullanılabilir olması, enerji verimliliğinin yeterli düzeyde artırılması ve robotik sistemlerle entegrasyonunu hedeflemelidir. Ayrıca, kullanıcı tecrübesine göre alınan geri bildirimlerle mobil uygulamanın işlevselliği geliştirilebilir ve daha fazla kişiselleştirme durumu sağlanabilir. Bu tür iyileştirmeler, sistemin ev kullanıcılarından farklı olarak tarım uygulamaları içinde kullanımını sağlayacaktır.

KAYNAKÇA

- [1] M. Taştan. Nesnelerin İnterneti tabanlı akıllı sulama ve uzaktan İzleme sistemi. *Avrupa Bilim ve Teknoloji Dergisi*, (15):229–236, 2019.
- [2] Onvec. Onvec akıllı kameralar ile bitki büyüme İzleme, 2024.
- [3] Amazon Türkiye. Yatay dalgıç pompa 40-110cm 2.2x0.9x1.2, 2024. Erişim Tarihi: 4 Ocak 2025.
- [4] Y. Fyk. Arduino ile akıllı bitki sulama sistemi projesi, 2024. Accessed: 23 December 2024.
- [5] D. B. Buragay, L. E. M. Escalona, L. A. B. Aniag, P. T. R. Laja, K. M. T. Escobar, F. K. A. Diaz, and P. R. C. Salenga. The use of soil moisture sensors and arduino interface to create an automatic watering device for houseplants. *International Journal for Research Trends and Innovation*, 9(4):1099–1108, 2024.
- [6] B. A. Reddy, N. U. Reddy, D. P. S. Teja, V. S. Rajam, and K. Nellore. A smartphone controlled fertilizing and plant watering garduino. *International Journal of Applied Mathematics Electronics and Computers*, 8(4):173–178, 2020.
- [7] H. Nagaraja, R. Aswani, and M. Malik. Plant watering autonomous mobile robot. *IAES International Journal of Robotics and Automation*, 1(3):152, 2012.

- [8] J. S. Witkowski and A. F. Grobelny. Dual-frequency soil moisture meter method for simultaneous estimation of soil moisture and conductivity. *Sensors*, 24(10):2969, 2024.
- [9] R. Nagalingam, V. Chintamaneni, K. Paramasivan, and M. Ponnusamy. Smart agriculture–automatic monitoring of soil moisture and irrigation control for farming land. *Current Agriculture Research Journal*, 11(3), 2023.
- [10] I Srilikhitha, M Manoj Saikumar, Nannu Rajan, ML Neha, and M Ganesan. Automatic irrigation system using soil moisture sensor and temperature sensor with microcontroller at89s52. In *2017 International Conference on Signal Processing and Communication (ICSPPC)*, pages 186–190. IEEE, 2017.
- [11] A. Hamidoğlu. Akilli sulama için toprak nem dinamiginde kapali dongu kontrol modeli ve kontrol edilebilirlik. *EMO Bilimsel Dergi*, 13(3):7–15, 2023.
- [12] J. Linford and M. Haghshenas-Jaryani. A ground robotic system for crops and soil monitoring and data collection in new mexico chile pepper farms. *Discover Agriculture*, 2(1):1–26, 2024.
- [13] A. Rahman, R. Raheem, and T. Sharmilan. Design a smart mini robot for indoor plant watering system. *International Journal of Sciences: Basic and Applied Research (IJSBAR)*, 61(1):1–9, 2022.
- [14] B. Nsoh, A. Katimbo, H. Guo, D. M. Heeren, H. N. Nakabuye, X. Qiao, Y. Ge, D. R. Rudnick, J. Wanyama, E. Bwambale, et al. Internet of things-based automated solutions utilizing machine learning for smart and real-time irrigation management: A review. *Sensors (Basel, Switzerland)*, 24(23):7480, 2024.

- [15] S. Amertet Finecomess, G. Gebresenbet, and H. M. Alwan. Utilizing an internet of things (iot) device, intelligent control design, and simulation for an agricultural system. *IoT*, 5(1):58–78, 2024.
- [16] B. Babayığit and B. Büyükpatpat. Nesnelerin interneti tabanlı sulama ve uzaktan izleme sisteminin tasarımı ve gerçekleştirilimi. *Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi*, 12(2):13–19, 2019.
- [17] G. Takara, A. Z. Trimble, R. Arata, S. Brown, H. J. Gonzalez, and C. Mora. An inexpensive robotic gantry to screen and control soil moisture for plant experiments. *HardwareX*, 9:e00174, 2021.
- [18] R. Nandi and D. Shrestha. Assessment of low-cost and higher-end soil moisture sensors across various moisture ranges and soil textures. *Sensors*, 24(18):5886, 2024.
- [19] X. Wang, C. Zhang, and S. Zhang. Multiscale convolutional neural networks with attention for plant species recognition. *Computational Intelligence and Neuroscience*, 2021(1):5529905, 2021.
- [20] Z. Wang, X. Su, and S. Mao. Single-species leaf detection against complex backgrounds with yolov5s. *Forests*, 15(6):894, 2024.
- [21] O. Joochim, P. Aumthuk, and S. Natsupakpong. Automated greenhouse for plant phenotyping using the image processing method. In *Proceedings of the 2023 Asia Conference on Artificial Intelligence, Machine Learning and Robotics*, pages 1–5, 2023.
- [22] C. Topçu and P. Güneş. Bitki hastalıklarını tespitite derin öğrenme: Resnet modelinin etkinliği. *Anadolu Bil Meslek Yüksekokulu Dergisi*, 19(69):31–65, 2024.

- [23] S. Millán, C. Montesinos, and C. Campillo. Evaluation of different commercial sensors for the development of their automatic irrigation system. *Sensors*, 24(23):7468, 2024.
- [24] I. Catruc and D. Iosifescu. Iot integration with mobile and cloud solutions. *Informatica Economica*, 24(2), 2020.
- [25] W. Li and G. Bai. Internet of things system based on mobile communication network. *International Journal of Online Engineering*, 14(11), 2018.
- [26] Q. Suyala, Z. Li, Z. Zhang, L. Jia, M. Fan, Y. Sun, and H. Xing. Developing a hyperspectral remote sensing-based algorithm to diagnose potato moisture for water-saving irrigation. *Horticulturae*, 10(8):811, 2024.
- [27] G. Göksu. Akıllı tarımda dayanıklı kontrol stratejileriyle toprak nem dinamiği yönetimi. *EMO Bilimsel Dergi*, 13(3):17–26, 2023.
- [28] S. Millán, J. Casadesús, C. Campillo, M. J. Moñino, and M. H. Prieto. Using soil moisture sensors for automated irrigation scheduling in a plum crop. *Water*, 11(10):2061, 2019.
- [29] M. Muthmainnah, M. F. Mulyadi, I. Tazi, A. Mulyono, F. S. Hananto, N. Chamidah, et al. Development of an automated monitoring system for soil moisture and temperature in smart agriculture to enhance lettuce farming productivity based on iot. *Multidisciplinary Science Journal*, 6(11):2024233–2024233, 2024.
- [30] M. H. Widiyanto and B. Juarto. Smart farming using robots in iot to increase agriculture yields: A systematic literature review. *Journal of Robotics and Control (JRC)*, 4(3):330–341, 2023.

- [31] F. Doğan and İ. Türkoğlu. Derin öğrenme algoritmalarının yaprak sınıflandırma başarımlarının karşılaştırılması. *Sakarya University Journal of Computer and Information Sciences*, 1(1):10–21, 2018.
- [32] J. V. M. Caracas, G. de C. Farias, L. F. M. Teixeira, and L. A. de S. Ribeiro. Implementation of a high-efficiency, high-lifetime, and low-cost converter for an autonomous photovoltaic water pumping system. *IEEE Transactions on Industry Applications*, 50(1):631–641, 2014.
- [33] Google. Firebase firestore, 2024. Erişim Tarihi: 2 Temmuz 2025.
- [34] Limitless Technologies LLP. Flutter for iot: Integrating mobile apps with smart devices. <https://medium.com/@limitless.technologies.llp/flutter-for-iot-integrating-mobile-apps-with-smart-devices-58e2b8319> 2024. Accessed: 25 December 2024.
- [35] D. Hill. Why python is the best programming language for data analysis and machine learning. <https://medium.com/@damian.hill/why-python-is-the-ideal-language-for-data-analysis-and-machine-learning-718e> 2024. Accessed: 25 December 2024.
- [36] Python Programming Ascendance. Intro to tensorflow: Importance of tensorflow and why it matters. <https://medium.com/python-programming-ascendance/intro-to-tensorflow-importance-of-tensorflow-and-why-it-matters-718e> 2024. Accessed: 25 December 2024.

- [37] Exo Wanderer. What is this keras thing anyways? <https://exowanderer.medium.com/what-is-this-keras-thing-anyways-fe7aa00158ef>, 2024. Accessed: 25 December 2024.
- [38] Analytics Vidhya. Flask python: A complete guide to build web applications with flask. <https://www.analyticsvidhya.com/blog/2021/10/flask-python/>, 2021. Accessed: 25 December 2024.
- [39] Alex Clark and Contributors. Pillow (pil fork). <https://python-pillow.org>, 2025. Python Imaging Library'nin aktif olarak geliştirilen versiyonudur. Erişim tarihi: 26 Mayıs 2025.
- [40] Jundi Kuo. icrawler: A multi-threaded crawler for downloading images from the web. <https://github.com/helloack/icrawler>, 2019. Erişim tarihi: 26 Mayıs 2025.
- [41] Google. Gemini: Çok modlu büyük dil modeli, 2024. Erişim Tarihi: 2 Temmuz 2025.