MARMARA ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

ASMA YAPRAĞINDA HASTALIK TESPİTİ İÇİN DERİN ÖĞRENME TABANLI MOBİL UYGULAMA GELİŞTİRİLMESİ

SEYİT GAZİ YILDIZ

YÜKSEK LİSANS TEZİ

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Yüksek Lisans Programı

DANIŞMAN

Doç. Dr. Kazım YILDIZ

EŞ-DANIŞMAN

Dr. Öğr. Üyesi Ali SARIKAŞ

İSTANBUL, 2025

MARMARA ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

ASMA YAPRAĞINDA HASTALIK TESPİTİ İÇİN DERİN ÖĞRENME TABANLI MOBİL UYGULAMA GELİŞTİRİLMESİ

SEYİT GAZİ YILDIZ

523622011

YÜKSEK LİSANS TEZİ

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Yüksek Lisans Programı

DANIŞMAN

Doç. Dr. Kazım YILDIZ

EŞ-DANIŞMAN

Dr. Öğr. Üyesi Ali SARIKAŞ

İSTANBUL, 2025

MARMARA ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

Marmara Üniversitesi Fen Bilimleri Enstitüsü Yüksek Lisans Öğrencisi Seyit Gazi YILDIZ'ın "Asma Yaprağında Hastalık Tespiti İçin Derin Öğrenme Tabanlı Mobil Uygulama Geliştirilmesi" başlıklı tez çalışması, 23.09.2025 tarihinde savunulmuş ve jüri üyeleri tarafından başarılı bulunmuştur.

Jüri Üyeleri

Doç.Dr. Kazım YILDIZ (Danışman)

Marmara Üniversitesi ..... (İMZA).....

Doç.Dr. Önder DEMİR (Üye)

Marmara Üniversitesi ..... (İMZA).....

Dr. Öğr. Üyesi Esra ÇALIK BAYAZIT (Üye)

Fatih Sultan Mehmet Vakıf Üniversitesi ..... (İMZA).....

ONAY

Marmara Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ..... tarih ve ..... sayılı kararı ile Seyit Gazi YILDIZ'ın Bilgisayar Mühendisliği Anabilim Dalı Bilgisayar Mühendisliği Programında Yüksek Lisans derecesi alması onanmıştır.

Fen Bilimleri Enstitüsü Müdürü

Prof. Dr. Bülent MERTOĞLU

ÖNSÖZ

Bu tez çalışması, tarım alanında yapılacak çalışmalara katkı sunmak amacıyla hazırlanmıştır. Hazırlık süresinde bu alan kapsamında yapılmış çalışmalar, uygulamalar ve hazırlanmış veri setleri üzerinde kapsamlı bir yaklaşım izlenmiştir. Bu doğrultuda çevrimiçi ve çevrimdışı Android işletim sisteminde çalışabilecek mobil uygulama ile çevrimiçi çalışabilecek web uygulama geliştirmeleri yapılmıştır.

Tez çalışma sürecinde araştırma ve yürütme başta olmak üzere her aşamasında desteğini esirgemeyen değerli danışmanım Doç. Dr. Kazım YILDIZ'a, eşdanışmanım Dr. Öğr. Üyesi Ali SARIKAŞ'a, desteklerini her zaman yanımda hissettiğim aileme ve eğitim hayatımda bilgi ve tecrübeleriyle bana yol gösteren tüm hocalarıma teşekkürlerimi sunarım.

Bu tez çalışmasının, tarım alanında yapılacak araştırmalara faydalı olmasını ve yeni yapılacak çalışmalarda ilham vermesini temenni ediyorum.

Saygılarımla

EYLÜL, 2025 **Seyit Gazi YILDIZ**

İÇİNDEKİLER

	SAYFA
ÖNSÖZ	**i**
İÇİNDEKİLER	**ii**
ÖZET	**iv**
ABSTRACT	**vii**
SEMBOLLER	**x**
KISALTMALAR	**xi**
ŞEKİL LİSTESİ	**xiii**
TABLO LİSTESİ	**xv**
1. GİRİŞ	**1**
2. MATERYAL VE YÖNTEM	**9**
2.1. Veri Setleri	**9**
2.1.1. Asma Yaprağı Deneysel Özgün Veri Seti	9
2.1.2. Biber Yaprağı Hazır Veri Seti	11
2.2. Veri Seti Önişlem Süreci	**11**
2.3. Unet Tabanlı Derin Öğrenme Mimarisi	**13**
2.3.1. Epoch	16
2.3.2. Kayıp Fonksiyonu	16
2.4. Doğruluk Ölçümlemede Uygulanan Değerler	**17**
2.5. Deney Değerlendirme Performans Metrikleri	**18**
2.5.1. Duyarlılık	19
2.5.2. Kesinlik	19
2.5.3. F1 Score	19
2.5.4. Doğruluk	20
2.6. Yaprak Analizi İçin Hazırlanan Sınıflandırma Modelleri	**20**
2.6.1. PT ve ONNX Model Eğitimi	21
2.6.2. ML.NET Eğitimi	22
2.7. Yerel ve Çevrimiçi Veri Tabanları	**23**
2.7.1. MySQL Veri Tabanı	24

2.7.2. SQLite Veri Tabanı 24
2.7.3. Room Veri Tabanı 25
2.8. Uygulama Programlama Arabirimleri 25
2.8.1. Uygulama Programlama Arayüzü 26
2.8.2. Web API Servisleri 26
2.8.3. Fast API Servisleri 27
2.9. Çalışma Modelleri 27
2.9.1. Çevrimiçi Çalışma Modeli 32
2.9.2. Çevrimdışı Çalışma Modeli 34
2.10. Yeniden Uygulanabilir Eğitim ve Sürümleme 37
3. BULGULAR 39
3.1. Deney Ortamları 39
3.2. Asma Yaprağı Deneyi 40
3.3. Biber Yaprağı Deneyi 44
4. SONUÇLAR VE TARTIŞMA 49
KAYNAKLAR 52
ÖZGEÇMİŞ

ÖZET

ASMA YAPRAĞINDA HASTALIK TESPİTİ İÇİN DERİN ÖĞRENME TABANLI MOBİL UYGULAMA GELİŞTİRİLMESİ

Teknolojinin hızla gelişmesi paralelinde bilgiye erişimi de hızlandırmıştır. Mobil (taşınabilir) cihazlar hem iletişim yönünden hem de bilgiye erişme noktasında en çok tercih edilen teknolojik cihazlardır. Bunun sonucu olarak, istenilen veriye ulaşmak geçmişe göre çok daha pratik bir durum kazanmıştır. Yaygın olarak kullanılan bu cihazlar günümüzde akıllı sistemlerin gelişmesinde de önemli rol oynamaktadır. Akıllı sistemler tarım endüstrisinde de yaygın bir biçimde kullanılmaya başlanmıştır. Gerek küresel ısınma gerekse iklim değişikliği etkisiyle tarımsal ürünlerde verim olumsuz olarak etkilenmektedir. Su seviyelerindeki azalmayla birlikte tarım alanlarındaki sulama günümüzde olduğu gibi gelecekte de kritik önemini korumaya devam edecektir. Bu çerçevede tarıma yönelik atılacak her adım gelecek için büyük bir değer taşımaktadır. Tarımla elde edilecek ürünlerin ürün kalitesi ve verimliliği oldukça önemlidir. Tarım ürünlerindeki bilinen hastalıkların yanı sıra değişen çevre faktörleriyle yeni çeşitli negatif etkenler de ortaya çıkmaktadır. Bu tip tespitlerin yapılmasını sağlamak ve gerektiğinde yerinde tespit yöntemlerinin geliştirilmesini sağlamak amacıyla yeni bir model önerisi ile farklı bir bakış açısı kazandırmak bu tez kapsamında hedeflenmektedir.

Bu çalışmada sorunlar çerçevesinde son kullanıcının aktif biçimde rol alacağı, gelişmeye ve dinamik biçimde öğrenmeye açık interaktif bir sistem geliştirilmesi hedeflenmektedir. Çalışmada sunulan sistem ile birlikte asma yaprakları üzerinde hastalık tespiti yapan bir model geliştirmesi yapılmıştır. Eğitim ve test aşamalarında kullanılması için asma yaprağının görsellerinden oluşan bir veri seti elde edilmiştir. Veri seti hazırlanırken, sistemin bir mobil cihaza kurulması amaçlandığından, görseller mobil cihazlar aracılığıyla sağlanmıştır. 3122 asma yaprağı görsel verileri iPhone İşletim Sistemi (iOS) ve Android işletim sistemlerine sahip mobil cihazlarla toplanmış, böylelikle özgün bir veri seti oluşturulmuştur. Bu yapraklardan 465 tanesi model eğitim ve doğrulaması ile

birlikte test verileri olarak model geliştirme aşamasında kullanılmıştır. Ön işleme süreçlerinde ve eğitim aşamasında modelden daha verimli sonuç alınabilmesi için, yaprakların bulunduğu ortamdaki karmaşıklıkla birlikte düşük kaliteye sahip ön işlem süreci başarısız olan yapraklar model eğitim ve doğrulama adımlarında kullanılmamıştır. Gerekli önişlem ve etiketleme işlemleriyle dengeli ikili sınıfsal dağılıma sahip bu veriler ileri görüntü işlemeye hazır hale getirilmiştir. Görüntü işleme teknikleri kullanılarak Python üzerinde bir makine öğrenmesi modeli geliştirilmiştir. Aynı zamanda mobil cihazlar için de bir mobil uygulama ile sistem desteklenmiştir. Mobil cihazlarda kamera üzerinden elde edilecek görselleri, sistemin asma yapraklarının sağlıklı ya da hastalık tespitini yaparak sınıflandırması hedeflenmiştir. Python vasıtasıyla oluşturulan derin öğrenme tabanlı makine öğrenmesi modeli, mobil uygulamaya dâhil edilerek aktif kullanılabilecek bir uygulama geliştirilmiştir. Python ve mobil cihazda kullanılan uygulama ile çıktı alınan model üzerinde yapılan tahminlemelerde yakın sonuçlar çıkması sebebiyle her iki ortamda da kullanılabilecek hale getirilmiştir.

Çalışmalarımızda doğruluğu artırmak, farklı model yapılarını kullanarak kıyaslama olanağını artırabilecek bir sistem tasarlanmıştır. Bu kapsamda Python tabanlı yaptığımız modele ek olarak, Microsoft tarafından geliştirilen Makine Öğrenimi .NET (ML.NET) tabanlı makine öğrenmesi modeliyle de tasarladığımız modeli güçlendirmek hedeflenmiştir. ASP.Net Core web uygulama çerçevesi kullanılarak bir web uygulama hazırlanmıştır. ML.NET modelinde 3122 fotoğrafın tamamı ön işleme tabi tutulmadan orjinal veri yapısı ile eğitime tabi tutulmuştur. Mobil uygulama tarafında aynı resim üzerinden Python modeliyle birlikte ML.NET üzerinden de eş zamanlı sınıflandırma yapılabilmektedir. Uygulamalar arasında haberleşmenin sağlanabilmesi için Python ve ASP.Net Core tabanlı gerekli hizmet bağlantılarıyla birlikte mobil uygulamanın çevrimiçi ortamlarla iletişimi sağlanmıştır.

Üzüm yapraklarındaki bu hastalık zamanında tespit edilemezse ileriye yönelik üzümlerin küflenmesine, asma yapraklarının ticari kullanılamamasına ve asma yaprağıyla birlikte üzümün zarar görmesine yol açacağından ürünlerin tüketilemez duruma gelmesine sebep olacaktır. Uygulama tarafında tahmin sonuçlarının kullanıcıya gösterilerek, kullanıcıdan geri dönüş alınması sağlanmıştır. Bu kapsamda son kullanıcının tarım ürünlerindeki kalite ve verimi dolaylı olarak tespit etmesi amaçlanmıştır. Bu amaçla birlikte elde edilen

verilerle çalışan modelin tekrar tekrar kendini güncelleyebilir olmasına da katkı sağlaması hedeflenmiştir.

Bu çalışmadaki hazırlanan sistemin, farklı tarım ürünlerinde verdiği sonuçları da görebilmek için farklı çalışmalarda kullanılmış bir hazır veri seti üzerinde biber yapraklarıyla uygulamada kıyaslama yapılmıştır. Çalışmada yapmış olduğumuz uygulamalara benzer oranlarda eğitim, doğrulama ve test verileri verilerek sonuçlar gözlemlenmiştir. Arka planı temizlenmiş sadece yaprak görüntülerinin olduğu 2475 biber yaprağı görseli uygulamada başarılı bir şekilde eğitimi tamamlamıştır. Hazır görseller üzerinde sınıflandırma işlemini başarıyla uygulayabilmektedir. Bu analiz ile sadece ürün değişimi yapılarak uygulamanın farklı veri setleri ile çalışabilirliği deneyimlenmiştir. Bu kapsamda, tarım alanında yapılacak ilerideki çalışmalara yönelik geliştirilmesi hedeflenen modellere, diğer tarım ürünlerine de uygulanabilirliği ölçüsünde yön vermesi ve bu yönde çalışmalara katkı sağlaması beklenmektedir. Bu çalışma ile birlikte mobil uygulamanın makine öğrenmesine yönelik yapılacak çalışmalarda daha öncü rol oynayacağı düşünülmektedir.

EYLÜL, 2025 **Seyit Gazi YILDIZ**

ABSTRACT

DEVELOPMENT OF A DEEP LEARNING BASED MOBILE APPLICATION FOR DISEASE DETECTION IN GRAPEVINE LEAVES

The rapid development of technology has accelerated access to information in parallel. Mobile (portable) devices are the most preferred technological devices both in terms of communication and access to information. As a result, accessing the desired data has become much more practical than in the past. These widely used devices also play an important role in the development of smart systems today. Intelligent systems have started to be widely used in the agricultural industry. Both global warming and climate change have negatively affected the yield of agricultural products. With the decrease in water levels, irrigation in agricultural areas will continue to maintain its critical importance in the future as it is today. In this context, every step to be taken towards agriculture is of great value for the future. Product quality and productivity of the products to be obtained through agriculture are very important. In addition to the known diseases in agricultural products, various new negative factors are emerging with changing environmental factors. This thesis aims to provide a different perspective with a new model proposal in order to ensure that such determinations are made and to develop on-site detection methods when necessary.

In this study, it is aimed to develop an interactive system in which the end user will take an active role within the framework of the problems, open to development and dynamic learning. With the system presented in the study, a model for disease detection on grapevine leaves was developed. A dataset consisting of images of grapevine leaves was obtained for use in the training and testing phases. While preparing the dataset, the images were provided via mobile devices since the system was intended to be installed on a mobile device. 3122 grapevine leaf visual data were collected using mobile devices running the iPhone Operating System (iOS) and Android operating systems, thereby

creating a unique dataset. 465 of these leaves were used in the model development phase as test data along with model training and validation. In order to obtain more efficient results from the model during preprocessing and training, leaves that failed the preprocessing step due to low quality and complexity in their environment were not used in the model training and validation steps. With the necessary preprocessing and labeling, these data with a balanced binary class distribution were made ready for further image processing. A machine learning model was developed in Python using image processing techniques. At the same time, the system was supported with a mobile application for mobile devices. It is aimed for the system to classify the images to be obtained through the camera on mobile devices by detecting healthy or diseased vine leaves. A deep learning-based machine learning model created through Python was included in the mobile application and an application that can be used actively was developed. Since the predictions made on the model outputted with Python and the application used on the mobile device yielded close results, it has been made usable in both environments.

We designed a system that can increase the accuracy of our studies and increase the possibility of comparison by using different model structures. In this context, in addition to our Python-based model, we aimed to strengthen the model we designed with the Machine Learning .NET (ML.NET) based machine learning model developed by Microsoft. A web application was prepared using the ASP.Net Core web application framework. In the ML.NET model, all 3122 photos were trained with the original data structure without preprocessing. On the mobile application side, simultaneous classification can be performed on the same image via ML.NET as well as Python model. In order to provide communication between the applications, Python and ASP.Net Core based service connections were used to enable the mobile application to communicate with online environments.

If this disease in grape leaves is not detected in time, it will cause the grapes to mold, the grape leaves cannot be used commercially and the grape will be damaged along with the grape leaves, which will cause the products to become inedible. On the application side, the prediction results are shown to the user and feedback from the user is provided. In this context, it is aimed for the end user to indirectly determine the quality and yield of

agricultural products. For this purpose, it is also aimed to contribute to the fact that the model working with the data obtained together can update itself repeatedly.

In order to see the results of the system prepared in this study on different agricultural products, a comparison was made in the application with pepper leaves on a ready-made dataset used in different studies. The results were observed by giving training, validation and test data at similar rates to the applications we have made in the study. In the application, 2475 pepper leaf images with only leaf images with cleaned background completed the training successfully. It can successfully apply the classification process on ready-made images. With this analysis, only the product change was made and the application's ability to work with different datasets was experienced. In this context, it is expected that the models to be developed for future work in the field of agriculture will guide other agricultural products to the extent that they are applicable and contribute to work in this direction. With this study, it is thought that the mobile application will play a more pioneering role in future studies on machine learning.

SEPTEMBER, 2025 **Seyit Gazi YILDIZ**

SEMBOLLER

μ	:	Ortalama
σ	:	Standart Sapma
y_i	:	Görsel Sınıflandırma Etiketi
p_i	:	Sınıflandırma Olasılık Değeri

KISALTMALAR

API	**:**	Uygulama Programlama Arayüzü
CNN	**:**	Evrişimli Sinir Ağı
CPU	**:**	Merkezi İşlem Birimi
CUDA	**:**	Bilgisayar Birleşik Cihaz Mimarisi
FN	**:**	Yanlış Negatif
FPN	**:**	Feature Pyramid Network
FP	**:**	Yanlış Pozitif
GB	**:**	Gigabayt
GPU	**:**	Grafik İşlem Birimi
GPD	**:**	Görüntü Piksel Değeri
HSV	**:**	Hue, Saturation, Value (Renk Tonu, Doygunluk, Değer)
HTTP	**:**	Hiper Metin Aktarım Protokolü
IDE	**:**	Entegre Geliştirme Ortamı
IEEE	**:**	Institute of Electrical and Electronics Engineers
İHA	**:**	İnsansız Hava Aracı
iOS	**:**	iPhone İşletim Sistemi
ML.NET	**:**	Makine Öğrenimi .NET
MP	**:**	Megapiksel
MRI	**:**	Manyetik Rezonans Görüntüleme
ND	**:**	Normalleştirilmiş Değer
ONNX	**:**	Açık Sinir Ağı Değişimi
ORM	**:**	Nesne İlişkisel Eşleme
OS	**:**	İşletim Sistemi
PT	**:**	Pytorch Model
RAM	**:**	Rastgele Erişimli Bellek
RGB	**:**	Kırmızı, Yeşil, Mavi
ROI	**:**	İlgi Alanı
SegNet	**:**	Segmentation Network
SI	**:**	Severity Index (Ciddiyet Seviyesi)
SVM	**:**	Destek Vektör Makinesi

TL	**:**	Transfer Öğrenmesi
TN	**:**	Gerçek Negatif
TP	**:**	Gerçek Pozitif
URL	**:**	Tekdüzen Kaynak Bulucuları
Unet	**:**	U-Shaped Convolutional Network
U-net++	**:**	Nested Unet
VGG16	**:**	Visual Geometry Group 16-Layer Network
YOLO	**:**	You Only Look Once
YOLOv3	**:**	You Only Look Once Version 3
YOLOv3-tiny	**:**	You Only Look Once Version 3 Tiny
YOLOv4	**:**	You Only Look Once Version 4
YOLOv4-tiny	**:**	You Only Look Once Version 4 Tiny
YOLOv5-CA	**:**	YOLOv5 with Coordinate Attention
YOLOv5s	**:**	You Only Look Once Version 5 Small
YOLOv5x	**:**	You Only Look Once Version 5 Extra Large

ŞEKİL LİSTESİ

SAYFA

Şekil 1.1. Fındık yapraklarında önerilen çalışma için uygulanan model diagramı…….5

Şekil 2.1. Özgün hazırlanmış asma yaprağı veri seti örnek görselleri……...10

Şekil 2.2. Biber yaprağı için hazır veri setine ait örnek……11

Şekil 2.3. Veri setinde yer alan sağlıklı ve hastalıklı asma yaprağı görselleri……...13

Şekil 2.4. Unet tabanlı modelin çalışma mantığını anlatan görsel……15

Şekil 2.5. Derin öğrenme modeli kullanarak veri setini uçtan uca işleme akış şeması…..16

Şekil 2.6. Asma yaprağı model eğitimi kayıp fonksiyonunu gösteren grafik…….....21

Şekil 2.7. ML.NET kullanılarak yapılan çevrimiçi yaprak sınıflandırma örnekleri.....23

Şekil 2.8. HTTP isteklerinin kullanıcı ile veri tabanı arasındaki iletişimi…….27

Şekil 2.9. Mobil uygulamaya ait kullanıcı arayüz ekranlarına ait ekran görüntüleri…….30

Şekil 2.10. Canlı kamera kullanarak yapılmış tahmin örneği……...31

Şekil 2.11. Tek görsel ile kamera kullanarak yapılmış tahmin örneği……..32

Şekil 2.12. Çevrimiçi sınıflandırma akış şeması……..33

Şekil 2.13. Sağlıklı yaprak ile çevrimiçi sınıflandırma uygulama kullanım örneği……..34

Şekil 2.14. Hastalıklı yaprak ile çevrimiçi sınıflandırma uygulama kullanım örneği…...34

Şekil 2.15. Çevrimdışı sınıflandırma akış şeması…….35

Şekil 2.16. Sağlıklı yaprak ile çevrimdışı sınıflandırma uygulama kullanım örneği……36

Şekil 2.17. Hastalıklı yaprak ile çevrimdışı sınıflandırma uygulama kullanım örneği….36

Şekil 2.18. Çevrimiçi model eğitimi ve sürüm güncelleme akış şeması……...38

Şekil 3.1. Asma yaprağına ait mobil uygulama üzerindeki örnek rapor görselleri……42

Şekil 3.2. Asma yaprağına ait çevrimiçi rapor örneği.....43

Şekil 3.3. Biber yapraklarının model eğitimi kayıp fonksiyonunu gösteren grafik……...45

Şekil 3.4. Sağlıklı biber yaprağına ait sınıflandırma örneği……..45

Şekil 3.5. Hastalıklı biber yaprağına ait sınıflandırma örneği.....46

Şekil 3.6. Biber yaprağına ait mobil uygulama üzerindeki örnek rapor görselleri……46

Şekil 3.7. Biber yaprağına ait çevrimiçi rapor örneği……...47

TABLO LİSTESİ

SAYFA

Tablo 2.1. Karmaşıklık matrisi…….....18

Tablo 3.1. Uygulamada kullanılan mobil cihazlara ait teknik özellikler.....39

Tablo 3.2. Uygulamada kullanılan bilgisayarlara ait teknik özellikler.....40

Tablo 3.3. Asma yapraklarına ait işlenmemiş verilerle elde edilen başarı metrikleri.....42

Tablo 3.4. Asma yaprağına ait örnek görsellerle sonuç kıyaslama tablosu.....42

Tablo 3.5. İşlenmemiş asma yapraklarıyla elde edilen doğruluk başarı sıralaması……..43

Tablo 3.6. İşlenmemiş asma yapraklarıyla elde edilen F1 Score başarı sıralaması.....43

Tablo 3.7. Biber yapraklarına ait işlenmiş verilerle elde edilen başarı metrikleri.....44

Tablo 3.8. Asma ve biber yaprağına ait metriklerin karşılaştırılması.....44

Tablo 3.9. Biber yaprağına ait örnek görsellerle sonuç kıyaslama tablosu.....47

Tablo 3.10. İşlenmemiş biber yapraklarıyla elde edilen doğruluk başarı sıralaması.....48

Tablo 3.11. İşlenmemiş biber yapraklarıyla elde edilen F1 Score başarı sıralaması…..48

1. GİRİŞ

Tarım ürünlerindeki hastalık tespiti ve sınıflandırılması ile ilgili kaynaklarda yapılan araştırmalarla bu alanda yapılmış çalışmaların olduğu, birçok yayın yapıldığı gözlemlenmektedir. Buna benzer çalışmalarda modeller oluşturulurken genellikle görüntü işleme teknikleriyle birlikte makine öğrenmesi algoritmalarının kullanıldığı görülmektedir. Görüntü işleme teknikleri ve makine öğrenmesi algoritmaları kaynaklarda yer alan çalışmalarda geniş ölçüde kullanılmaktadır. Bu kapsamda mobil uygulamaların da makine öğrenmesi süreçlerine dâhil edildiği çalışmalar bulunmaktadır. Mobil uygulamalarda ise daha ağırlıklı olarak Android işletim sistemi (OS) bulunan mobil cihazlarla çalışıldığı görülmüştür. Android OS olan mobil cihazların kullanımının daha yaygın olması ve maliyet açısından iOS OS olan cihazlara göre daha uygun olması son kullanıcıya erişim noktasında uygulamanın kullanım sayısını etkileyeceği düşünülmektedir. Android cihazların, iOS cihazlara göre daha esnek yapıda olmasıyla birlikte geniş özelleştirilebilir olanaklara da sahip olması son kullanıcıda iOS cihazlara kıyasla daha yüksek kullanılabilirlik sağlayabileceği düşünülmektedir. Bize yön vermesi açısından ilgili alanda yapılmış kaynak taramasıyla benzer çalışmaları konu alarak uygulaması yapılmış sistemlerden esinlenerek model üzerinde çalışılmıştır.

Çalışmamıza yön vermesi açısından Liu ve diğerleri tarafından yapılan çalışmada yapılan sistem üzerinden esinlenilerek gerçek zamanlı bir mobil uygulamanın yapılması hedeflenmiştir. Gerçek hayata dâhil edilebilecek bir mobil uygulama tasarlanmıştır. 4200 kadar görselin bulunduğu veri seti ile yapılan çalışmada, Evrişimli Sinir Ağı (CNN) sınıflandırma algoritması yöntemi kullanılarak sınıflandırma işlemi tamamlanmıştır. En yüksek doğruluk oranının GoogleNet modeliyle yakalandığı gözlemlenmiştir [1].

Alkan ve diğerleri tarafından yapılan çalışmada asmada bulunan üzümlerin önemi vurgulanarak, Türkiye'de varlığını uzun süredir devam ettiren tarım ürünü için asma yapraklarındaki hastalıkların otomatik olarak tespiti üzerine derin öğrenme modeliyle çalışılmış CNN algoritması kullanılmıştır. Suriyeli ve Türk çiftçilerinin ürünlerindeki kaliteyi korumalarına yardım etmesi açısından geliştirilmiş bu sistemde, hastalık tespit süreçlerinin hızlanması için Grafik İşlem Birimi (GPU) ve Bilgisayar Birleşik Cihaz Mimarisi (CUDA) kullanılmıştır. Dokuz farklı hastalık tipinin tanımlaması incelenmiş,

GoogleNet ve Transfer Öğrenmesi (TL)'nin birlikte kullanıldığı senaryoda %85,0 doğruluk oranı tespit edilmiştir [2].

Khaled ve diğerleri tarafından yapılmış diğer bir çalışmada, akıllı telefonlara hemen hemen herkes tarafından erişim sağlanabileceğini belirtilmiştir. Engelli bireylerin, akıllı telefonların sensörlerinde de yararlanabileceği erişebilirlik özelliklerinin bulunduğu önerilen bir sistemden bahsedilmektedir. İlgili çalışmada engelli bireylere yönelik iç mekânda yönlendirmedeki süreci kolaylaştırmak amaçlı bir yaklaşım önerilmektedir. Eğitilmiş CNN modeli kullanılarak nesne tespitinde kullanıcıya doğru yönlendirmeler yapabilmesi ve engellerde kullanıcının uyarılması için sistemde nesnelere olan uzaklık hesabı yapmışlardır [6].

Sozzi ve diğerleri tarafından yapılmış çalışmada doğruluk ve hız değerleriyle birlikte CNN'nin nesne tespitinde önerildiğine değinilmiştir. Bağcılıkta verimli ürünlerin yönetiminde ve verim tahmininde kullanıldığından bahsedilmektedir. Bu çalışmada You Only Look Once (YOLO) nesne tespit algoritmasının üzüm ürünlerinde salkım tespitinde kullanılırken, sayımında farklı YOLO sürümleri uygulanmıştır. Çalışmada sürüm 3 kapsamında (You Only Look Once Version 3 (YOLOv3) ve You Only Look Once Version 3 Tiny (YOLOv3-tiny) YOLO sürümleri uygulanmıştır. Yine aynı çalışmada sürüm 4 kapsamında You Only Look Once Version 4 (YOLOv4) ve You Only Look Once Version 4 Tiny (YOLOv4-tiny) YOLO sürümleri uygulanmıştır. Devamında sürüm 5 kapsamında You Only Look Once Version 5 Extra Large (YOLOv5x) ve You Only Look Once Version 5 Small (YOLOv5s) YOLO sürümleri uygulanmıştır. Beyaz meyvelerin arka planında yaprağın olması tanımlamayı zor hale getirdiğinden dolayı hedef olarak beyaz üzümleri tercih etmişlerdir. YOLOv4-tiny ile en iyi hız uyumu ve doğruluğu elde etmişlerdir [7].

Koklu ve diğerlerinin yaptığı çalışmada asma yapraklarının yılda bir kez hasat edildiğine, asma yaprağının türünün ürünün fiyat ve lezzeti açısından önemli olduğuna değinilmiştir. Derin öğrenme modelini kullanarak sınıflandırmayı konu alan çalışmada 5 türe ait 500 tane yaprak görselinin görüntüsü alınıp veri artırma yöntemleriyle 2500 e kadar bu sayı yükseltilmiştir. Sistemin sınıflandırma başarısı %97,60 olarak elde edilmiştir. Çalışmada CNN ve Destek Vektör Makinesi (SVM) yöntemleri kullanılmıştır [8].

Thet ve diğerlerinin yaptığı çalışmada, üzümdeki hastalıkların hızlı yayıldığına ve hastalığın yapraktan başlayarak sırayla gövde, meyve ve köke doğru ilerlediğine değinilmiştir. Uzak bölgelerde çiftçilerin üzüm yaprağındaki hastalıkları sınıflandırmada sorun yaşadığından bahsedilmiştir. Son araştırmalarda üzüm yaprağındaki hastalıkların Visual Geometry Group 16-Layer Network (VGG16) ile sınıflandırıldığına değinilmiştir. Önerilen sistemin yaprak hastalıklarını sınıflandırması için SVM ve aynı zamanda CNN mimarisinden biri olan VGG16 kullanılmıştır. Önerilen sistemde 5 tip hastalık çeşidi bulunan Myanmar Grapevine Yard'ın 6000 görüntüye sahip yaprak veri seti kullanılmıştır [9].

Math ve Dharwadkar yaptıkları çalışmada tarımdaki mahsüllerin kalitesi göz önünde bulundurularak, hastalık tehditlerine dayanıklı hastalık tespiti yapabilmesini sağlayacak bir model geliştirmeyi amaçlamışlardır. CNN sınıflandırma modelini konu alan ve derin öğrenme modelleri ile hastalık tespitini yapmanın faydalı olduğuna değinilen çalışmada üzüm ürününü konu almışlardır [10].

Zhang ve diğerlerinin yapmış olduğu üzümdeki külleme hastalığına yönelik yapılan bir çalışmada, bilgisayarla görme yöntemlerinin ve yapay zekâ teknolojileri ile birlikte gerçek zamanlı hastalık tespitinin yapılabileceğine değinilmiştir. Çalışmamızda elde ettiğimiz veri setinin bulunan ortamına benzer şekilde bağda elde edilen veri seti kullanılmıştır. Çalışmada konu alınan hastalığa yönelik en hızlı ve doğru algılama yöntemi için YOLOv5 with Coordinate Attention (YOLOv5-CA) derin öğrenme yaklaşımı önerilmiştir [11].

Jamadar ve diğerlerinin yaptıkları çalışmada farklı bir yaklaşıma değinilmektedir. Hastalıklı ve sağlıklı kısımları görsel üzerinde piksel seviyesinde etiketleme yapmak amacıyla semantik segmentasyon tercih edilmiştir. Derin öğrenme modellerinden Segmentation Network (SegNet) mimarisi ile birlikte sınıflandırmada ResNet kullanılmıştır. Farklı derin öğrenme modellerinin de bulunduğu çalışmada SegNet, Feature Pyramid Network (FPN), U-Shaped Convolutional Network (Unet) ve DeepLabv3 gibi diğer mimarilerin karşılaştırılması yapılmıştır. Kıyaslama sonucunda Segnet'in bellek açısından daha verimli olduğu söylenmiştir [12].

Uttam ve diğerlerinin yaptıkları çalışmada üzüm bitkisi üzerindeki hastalıkların sağlıklı, çürük, enfekteli olarak sınıflandırıldığı görülmektedir. Otomatik tespitin önerildiği

sistemde görüntü işleme teknikleri ve derin öğrenme çalışma yöntemi tercih edilmiş, ResNet tabanlı model kullanılmıştır. %99,58 doğruluk oranı yakalanan çalışmada üzüm yaprağındaki hastalık tespiti için otonom bir model geliştirilmesi hedeflenmiştir [13].

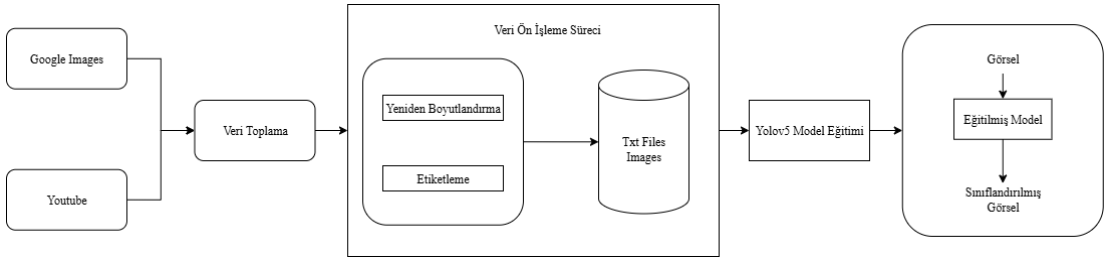
Kandel ve diğerlerinin yaptıkları çalışmada bitkilerin korunmasına yönelik hastalık teşhisinde kullanılacak bir model önerilmiştir. Derin öğrenme modellerinden CNN yöntemiyle birlikte toplanan verilerin ön işleme ve model oluşturma aşamalarıyla çalışma tamamlanmıştır [14]. Wang ve diğerlerinin yaptıkları çalışmada üzüm yaprağında hastalık tespiti üzerine konu ele alınmıştır. CNN kullanılan çalışmada Nested Unet (U-net++) tabanlı model kullanılmıştır [15].

Medikal alanda yapılmış bir çalışma kapsamında beyindeki tümör segmentasyonunda derin öğrenme tabanlı Unet ile birlikte ResNet50 kodlayıcısı kullanılmıştır. Manyetik rezonans görüntüleme (MRI) görüntüleri kullanılarak yapılan çalışmada beyin tümörü hastalarında daha hassas tanı yapılmasına ve tedavi planlamasına katkı sağlayabileceği düşünülmektedir. ResNet50 kodlayıcısı ile beyin tümörü segmentasyonunda yüksek oranda doğruluk sağladığı söylenmektedir. Bu çalışmada kullanılan Unet ve ResNet50 birlikteliği, bu tez çalışmasında benzer bir yöntemi kullanabilmemiz açısından bize pozitif anlamda katkı vermektedir. Unet mimarisine ResNet50 tabanlı kodlayıcı dâhil ederek ResUNet50 modeli elde edilmiştir. Klasik Unet'e kıyasla %60'tan fazla parametrede azalma sağlamış ve çıkarım süresinde 4.8 kat daha hızlı bir performans elde edildiği söylenmiştir [37].

Siddiqui ve Kayte yaptıkları çalışmada son zamanlarda sivrisineklerin ısırmasıyla bulaşabilen sıtma, dang humması ve chikungunya hastalıklarının bütün dünya üzerinde yayıldığından bahsedilmektedir. Sivrisinek türlerinin görsel olarak birbirlerine yakın olmaları sebebiyle sınıflandırmada zorluk yaşandığına değinilmektedir. Bu kapsamda sivrisineklerin türlerini sınıflandırmak amacıyla CNN tabanlı derin öğrenme modeli uygulanmış ve CNN tabanlı derin öğrenme modeli önerilmektedir. Aedes, Anopheles ve Culex türlerini konu alan çalışmada 1800 görseli kapsayan veri seti hazırlanıp 3 farklı sivrisinek türünün sınıflandırılması yapılmaktadır. Veri artırma yöntemleri de kullanılan modelin CNN tabanlı derin öğrenme uygulamasında doğruluk değeri %84,51 olarak gözlemlenmiştir [42].

Boyar ve diğerlerinin yaptığı farklı bir semantik segmentasyon modeli sunulan bu çalışmada fındık bahçelerinin sağlıklı ve hastalıklı bölgelerinin ikili sınıflandırılması yapılmaktadır. Bu araştırma kapsamında insansız hava araçlarından (İHA) elde edilen görüntülerin kullanıldığına değinilmektedir. Çevre koşullarının zorlukları ve engelleriyle birlikte kullanılan cihazın pil kapasitesinin ve görüntüde kararlılık gibi drone tabanlı görüntülemenin zor kısımlarına yer verilmektedir. Ağaç bazında hastalık tespitinin yapıldığı bu çalışmada yaprakların tek tek analizinin yanı sıra farklı bir sınıflandırma yöntemiyle birlikte özgün bir veri seti oluşturulduğuna değinilmektedir. Birden fazla Unet varyasyonlarıyla birlikte farklı derin öğrenme modelleriyle testlerin yapıldığı bu çalışmada en performanslı sonuç U-Net++ modelinde elde edildiği söylenmektedir. Bunun yanı sıra, modelde gelişimin devam edebilmesi için hiperparametre iyileştirmesiyle veri setinde genişletilme önerilmektedir. Bu tez çalışması kapsamında bulunan ortak hedef tarımsal üretim kayıplarında azalma sağlanarak doğru hastalık tespitiyle üretim kaybının azalması amaçlanmaktadır. İHA ve derin öğrenme tabanlı modelin potansiyelinin yüksek olduğu vurgulanmaktadır [44].

Boyar ve Yıldız yaptıkları diğer bir araştırmada ise fındık yapraklarında görülen külleme hastalığını konu alan fındık yetiştiriciliği dâhilinde önemli bir sorun üzerine çalışılmıştır. Makine öğrenmesi tabanlı YOLO modeli kullanılarak külleme hastalığında erken tespit çalışması ele alınmıştır. Yaprak odaklı sınıflandırma yapılan bu çalışmada özgün bir veri seti hazırlanmıştır. YOLOv5 modelinin hastalıklı yapraklar üzerinde %90 başarı oranıyla doğru tespit yaptığı ifade edilmektedir. Arka planı karışık görüntülere sahip çoklu yaprakların olduğu görsellerde, sağlıklı alanların tespittinde %85'lik bir değerle başarı elde edildiği söylenmektedir. Bu kapsamda kullanılacak uygulamanın fındık üretiminde verimliliği artırarak üretici zararlarını azaltabileceği gözlemlenmektedir. Bu yapılan çalışma kapsamında önerilen çalışma modeline yönelik uygulanan model diagramına Şekil 1.1'de tekrar görsellendirilerek yer verilmiştir [41].



Şekil 1.1. Fındık yapraklarında önerilen çalışma için uygulanan model diagramı [41].

Mobil uygulamaların dâhil edildiği çalışmaların, ilgili alanda yapılmış yayınlarda farklı modellerle karşımıza çıktığını gözlemleyebiliriz. Bu tip mobil uygulamaların dâhil edildiği hastalık tespitinde ve sınıflandırılmasında kullanılan çalışmalar, farklı bir metot olarak değerlendirilebilir. Çeşitliliğin olması çalışmalardaki farklılıklara yön verebildiği gibi aynı zamanda mobil uygulamaların dâhil edildiği bu çalışmalar, bu çalışmada uygulayacağımız model için yeni parametreler de eklenerek farklı bir bakış açısı sunmamız için bize rehberlik eden kaynak olarak değerlendirilebilir [1,20].

Kaynak taraması sonucunda çeşitli çalışmalar incelendiğinde görüntü işleme tekniklerinin ve makine öğrenme algoritmalarının kullanıldığı bilimsel yayınlar yer almaktadır. Bilimsel yayınların yer verildiği sciencedirect veri tabanından, Institute of Electrical and Electronics Engineers (IEEE) elektronik veri tabanından ve diğer farklı veri tabanlarından incelenen kaynaklardan elde edilen bilimsel yayınların incelemesi sonucu, görüntü işleme teknikleriyle birlikte makine öğrenmesi algoritmalarının sıklıkla kullanıma açık olduğu görülmektedir.

Yapılan bazı çalışmalarda hastalık tespiti konusunda sınıflandırma yaparken ortak olarak derin öğrenme modellerinin kullanıldığı gözlemlenmektedir. Bu çalışmada kullanmayı hedeflediğimiz görüntü işleme teknikleri ve evrişimli sinir ağı modelleri yapılan kaynak taraması sonucunda elde edilen çalışmalarda konu olarak işlendiği görülmektedir. Tarım üzerine yapılan çalışmalar incelendiğinde bitkide hastalık tespitine ve sınıflandırılmasına yönelik çalışmaların yapıldığı fark edilmektedir. Yine bazı araştırmalarda derin öğrenme yöntemlerinden CNN algoritmasının hastalık tespiti yapılan sistemlerde kullanıldığının görülmesi, çalışmamızdaki kullanacağımız bu modeli destekler niteliktedir. Bu kapsamda çalışmalarda başarılı sonuçlar elde edildiği görülmektedir.

İncelenen çalışmalarda gerçek zamanlı uygulamaların da yer aldığı gözlemlenmiştir. Hastalık tespiti üzerine yapılan araştırmalarda gerçek zamanlı mobil uygulamalarla birlikte doğruluk oranlarının kıyaslamasının yapıldığı çalışmalara yer verilmektedir. Uygulamalar çerçevesinde bakıldığında Android işletim sistemli cihazların daha yoğun kullanılmaktadır. Bu doğrultuda çalışmalardaki mobil uygulamalarda uygulanan modellerin görüntü işleme teknikleriyle birlikte makine öğrenmesi algoritmaları kullanılarak sınıflandırılmasının yapıldığı ve görüntü işleme tekniklerinin mobil uygulamalara dâhil edilebildiği gözlemlenmiştir. Güncel olarak kullanıma sahip olan

CNN yönteminin hastalık tespitinin sınıflandırılmasında yüksek doğruluk oranlarına sahip olduğu görülmektedir. Görüntü işleme tekniklerinin kullanıldığı çalışmaların veri setinde bulunan görsellerinde veri çoklama yöntemleri de kullanılmaktadır. Çalışmalarda uygulanan modeller tarımda verimliliğin artması ve ürünlerdeki kalitenin yüksek olması hedefiyle birlikte ele alınmaktadır.

Literatürdeki yapılan çalışmalar dâhilinde, tarım alanında yapılmış benzer çalışmaların olduğu gözlemlenmiştir. Bu kapsamda farklı çalışmalar kapsamında mobil uygulamaların kullanıldığı, görüntü bölütleme yönteminde YOLO teknikleri ve modellerde CNN algoritmasının kullanıldığı tespit edilmiştir. Araştırmalar kapsamında mobil cihazlarda kullanılabilen makine öğrenmesine yönelik algoritmaların başarılı olduğu uygulamalar bulunmaktadır. Başarıyla sonuçlandığı gözlemlenen çalışmalarda, yüksek doğruluk oranına sahip sınıflandırma modelleri incelenmiştir. Aynı zamanda kameranın aktif kullanılarak uygulamalara dâhil edildiği çalışmalar dikkate alınmıştır. Gerçek zamanlı yapmayı hedeflediğimiz model için yönelik bize pozitif yönden katkı sağlamıştır. Bu duruma ek olarak gerçek zamanlı tahminleme yapılabilmesi, birden fazla model üzerinden çevrimiçi tahminle ile birlikte çevrimdışı tahmin edilebilir bir alt yapıya sahip olması ve son kullanıcıya tahminleme sonuçlarının onaylatılması sistemi daha interaktif bir kullanıma açmıştır. Model ile sınıflandırılan görsellerin kaydedilerek sunucuya gönderilmesi sonucunda tekrar eğitime dâhil edilebilir olması, sistemde eğitimi tamamlanan Python modellerinin tekrar mobil tabanlı uygulamaya indirilerek mevcut modellerin güncellenmesine olanak sağlamaktadır. Bu sayede sistemin daha güncel sürümlerinin kullanılabilir olmasıyla birlikte uygulamanın geliştirilebilir bir yönünün olması sağlanmıştır. Aynı zamanda modelde veri setinin farklı görsel veri setleriyle eğitilebilir olmasına ve elde edilen modellerin mobil uygulamada kullanılabilmesine de olanak sağlayacaktır. Kullanılan güncel teknolojilerin hedeflenen sistem doğrultusunda, elde edilen doğruluk oranlarının yüksek olması ve görüntü işleme teknikleriyle birlikte sınıflandırma algoritmalarının doğru sonuçlanması uygulama performansına olumlu yönde katkı sağlamıştır.

Sunduğumuz sistemin çalışma modellerine ilişkin diagramlara, uygulanan çalışma yöntemlerine ve model eğitiminde elde edilen grafiklere yönelik görsellere çalışmada yer verilmektedir. Bu çalışmada uygulanan materyal ve yöntemlere 2 numaralı bölümde, çalışma sonucunda elde edilen bulgulara ve elde edilen değerlere uygulama metoduyla

çalışmanın hedefine 3 numaralı bölümde yer verilmiştir. Aynı zamanda sınıflandırma modellerimize göre elde ettiğimiz veriler sonuçlar bölümünde analiz edilerek değerlendirilmiştir.

2. MATERYAL VE YÖNTEM

Bu tez çalışmasında bilimsel araştırmalarda kullanılan güncel derin öğrenme yaklaşımlarıyla özgün olarak hazırlanmış deneysel veri setinin yanı sıra yine güncel olarak sıklıkla kullanılan hazır bir veri seti üzerinde sınıflandırma deneyleri uygulanmıştır. Sınıflandırma yönteminin uygulanmasında kullanılan verisetlerine, model eğitiminde kullanılan derin öğrenme mimarilerine, hazırlanan sistemin kıyaslanabilmesi amacıyla farklı model eğitimlerine, bu modellerle birlikte hazırlanan çevrimiçi ve çevrimdışı çalışma modellerine ve modelin kendi kendini eğitmesine olanak sağlayan alt yapısıyla birlikte sürümle aşamalarına ilişkin detaylı bilgilere yer verilmektedir.

2.1 Verisetleri

Çalışmamızda hedef aldığımız asma yapraklarına ait veri seti ile karşılaştırma yapabilmek amacıyla kullanılan hazır biber yaprağı veri seti kapsamında detaylı bilgilere bu bölümde yer verilmektedir. 2.1.1 numaralı asma yaprağı deneysel veri seti bölümünde özgün olarak hazırlanmış mobil fiziksel cihazlarla görselleri elde edilmiş asma yapraklarına ait detaylar sunulmaktadır. Görsellerin toplanma süreci, işlenmesi ve modelin kullanacağı şekilde hazır hale gelene kadar yapılan süreçlere değinilmiştir. 2.1.2 numaralı biber yaprağı hazır veri seti bölümünde ise çalışmada yapılan modele ait başarı oranının hazır bir veri seti üzerinde de etkisini görebilmek amacıyla biber yapraklarına ait detaylara yer verilmiştir.

2.1.1 Asma Yaprağı Deneysel Özgün Veri Seti

Ülkemizde birçok farklı noktada üzüm ve asma yaprağı üretimi yapılmaktadır. Ticari anlamda üretimi sağlanıp sarmalık olarak kullanılabilen üzümlerde bulunan asma yaprağı farklı bölgelerimizde ve illerde yetiştirilmektedir. Bu doğrultuda ön planda bulunan çok sayıda ilimiz mevcuttur. Nevşehir, bu illerimiz arasında ön plana çıkan bir ilimizdir [21]. Nevşehir ili Ürgüp ilçesi, ülkemizde en önemli bağcılık bölgelerinden biridir. Geleneksel bağcılık kültürü ile birlikte verimli toprağı ve bölgedeki iklim koşuluyla birlikte üzüm yetiştirme noktasında öne çıkmaktadır. Ürgüp Bağ Bozumu Festivali etkinliğine de konu olması, bölgenin doğal ve kültürel zenginliğinin de bir parçası olduğunu göstermektedir [22]. Ürgüp'te bağlarda ve evlerin bahçelerinde yetiştirilen üzümler, yerel halka ait geçim kaynaklarından da birisidir. Birçok faktörü göz önünde bulundurduğumuzda, çalışma

kapsamında hem ticari anlamda hem de tüketim anlamında da ön planda olan asma yaprağı ele alınmıştır. Çalışma kapsamında Nevşehir ili Ürgüp ilçesinde bölgesinde gerek bağlarda gerekse evlerde yetiştirilen asmalardan veri seti elde edilmiştir.

Çalışmada hedeflenen uygulamanın çalışma ortamı Android tabanlı mobil cihazlar oluğu için, ihtiyaç duyulan veri seti mobil cihazlar üzerinden sağlanmıştır. Farklı cihazlarla birlikte her cihazın kamera çözünürlüğü birbirinden farklı olduğu için farklı çözünürlüklerde görseller toplanmıştır. 3122 fotoğraf gerçek ortam ile birlikte cihazın kendi kamerası yardımıyla görselleştirilmiştir. İkili sınıflandırmaya hazır olacak şekilde yapraklar hastalıklı ve hastalıksız olacak şekilde ayrıştırma yapılmıştır. Gruplanan fotoğraflar çalışmada kullanılan model kapsamında kullanılmak üzere gerekli ön işleme süreçleri de tamamlandıktan sonra kullanıma hazır bir hale getirilmiştir. Bu kapsamda yapılan ön işlem süreçlerine ait bilgilere 2.2 numaralı bölümde yer verilmiştir. Elde edilen görsellere ait örnek asma yaprağı görsellerine Şekil 2.1'de yer verilmiştir.



Şekil 2.1. Özgün hazırlanmış asma yaprağı veri seti örnek görselleri.

2.1.2 Biber Yaprağı Hazır Veri Seti

Çalışma kapsamında hazırlanan çalışma modelinin ikili sınıflandırmaya uygun veri setleri ile çalışabilirliğinin deneyimlenebilmesi için, PlantVillage hazır veri seti [4] kullanılarak mevcut sistem üzerinde uygulanmıştır. Elde edilen sonuçların asma yaprağında kıyaslanabilirliği mevcut görsellerin farklı sınıflandırılması çalışmaya uygun olacak şekilde ikili sınıflandırmaya çevrilmiştir. Çalışma kapsamında farklı hastalık tiplerinin sınıflandırılmasından ziyade mevcutta bulunan görselin hastalıklı mı sağlıklı mı olduğunun analizi yapıldığı için, farklı hastalık tipleri tek bir dosyada toplanarak hasta olacak şekilde kullanılmıştır. Asma yaprağı verilerinden farklı olarak arka planları temiz bir şekilde tutulmuş farklı yaprak görselleri bulunmaktadır. Biber yaprağı görsellerinden farklı olarak asma yaprağı verilerinde, gerçek ortam görsellerinin olması ve ön işlem süreçlerin sonucunda arka planı temiz görseller elde edilmesi hedeflenmiştir. Veri setinde 2475 adet biber yaprağı görseli bulunmaktadır. Hazır veri setinde bulunan biber görselleri farklı hastalık tiplerine göre sınıflandırılmıştır. Bu kapsamda kullanılan hazır veri setine ait örnek biber yaprağı görsellerine Şekil 2.2'de yer verilmiştir.



Şekil 2.2. Biber yaprağı için hazır veri setine ait örnek görseller.

2.2 Veri Seti Önişlem Süreci

Bu bölümde çalışmada kullanılan özgün hazırlanan asma yaprağı verilerinin ön işleme süreçleri hakkında bilgi verilmektedir. Gerçek ortamda görselleştirilerek 3122 adet çekilen fotoğraf içerisinden 465 görsel arka plandan ayrılarak sadece yaprak kalacak

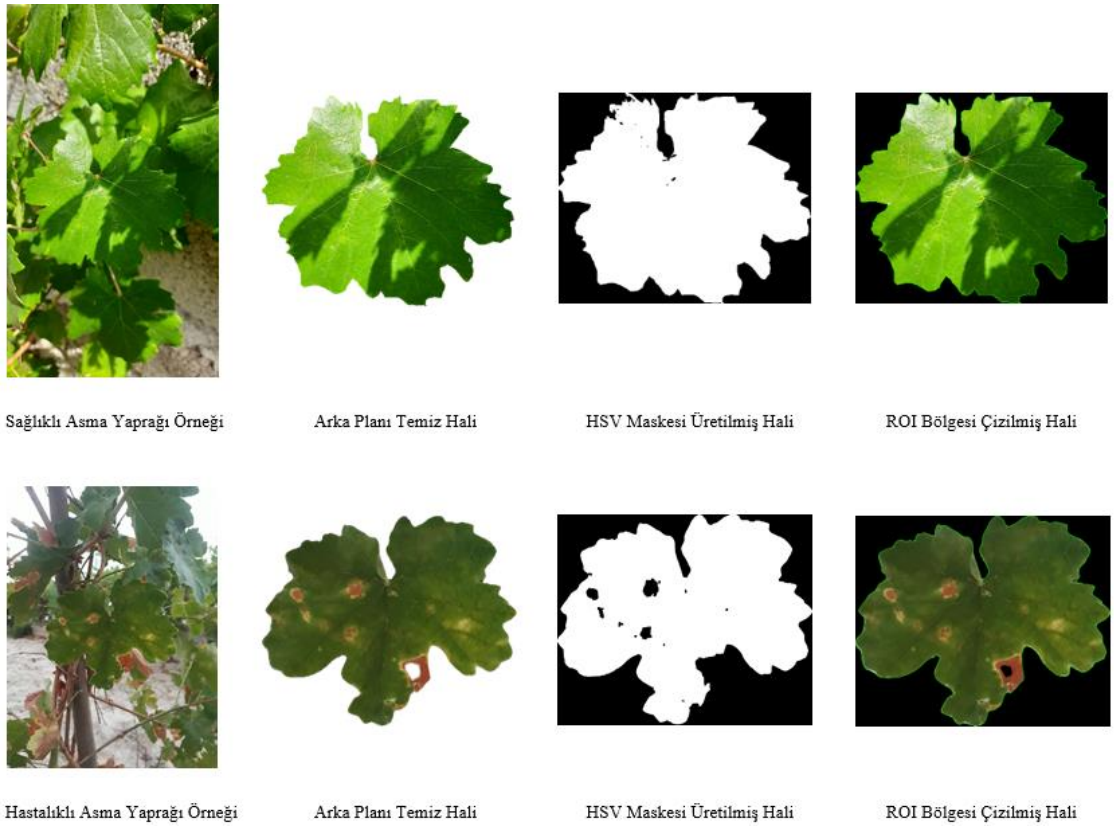
şekilde ön çalışması yapılmıştır. Arka plandan ayrılmış görseller çevrimiçi ve çevrimdışı çalışan Pytorch Model (PT) ve Açık Sinir Ağı Değişimi (ONNX) modellerinin eğitimi kapsamında kullanılmıştır. 465 görsel için arka plandan ayırma işlemi tamamlandıktan sonra orijinal Kırmızı, Yeşil, Mavi (RGB) görüntüler, Hue, Saturation, Value (HSV) renk uzayına dönüştürülmüştür. HSV maskeleri orijinal görselde hangi yaprak üzerinde eğitime dâhil edilmesi gerektiğini belirtmek amacıyla yeşil yaprağı ayırma noktasında etkinliği artıracak formata dönüştürülmüştür. Orijinal ve HSV maskeler ile model eğitimi devam ederken değerlendirme aşamasında kullanılmak üzere orijinal resimde tahminlemesi yapılan yapraklar için İlgi Alanı (ROI) maskeleri çıkarılmıştır [23]. Bu sayede elimizde orijinal görsellerle birlikte maske ve ROI alanı çizilmiş görsellerde yer almaktadır. Model eğitimi sırasında 465 görsel, görüntü işleme tekniklerinden veri artırma yöntemiyle birlikte çoğalmaktadır [24].

Her bir resim için %50 olasılık ile yatay, % 50 olasılık ile dikey ve %50 olasılık ile rastgele 90 derece çevirme parametreleriyle birlikte, her resim için ortalama ve standart sapma değerleri kullanılmıştır. Patates yapraklarında görüntülerin normalizasyonunda da kullanılan VGG16 kapsamında belirlenmiş standart sapma ve ortalama değerleri kullanılmaktadır. Standart sapma için renk değerleri standart sapma değeri σ = [0.229, 0.224, 0.225] ve ortalama için renk değerleri ortalama değeri μ = [0.485, 0.456, 0.406] şeklindedir [25]. Aynı zamanda görsel üzerindeki piksel normalleştirme de kullanılan σ, μ ve Normalleştirilmiş Değer (ND) değerlerinin bulunduğu Görüntü Piksel Değeri (GPD) formülü Denklem (2.1) de gösterilmektedir [26].

$$ND = \frac{GPD - \mu}{\sigma} \quad (2.1)$$

Çevirme ve piksel normalleştirme işlemleriyle birlikte uygun formata çevrilerek model eğitimi sırasında veri çoğaltma işlemi tamamlanmaktadır. Bu çalışma kapsamında yapılmış olan çevrimiçi çalışma modelinde 3122 görsel hiçbir veri çoğaltma işlemine tabi tutulmadan, doğrudan orijinal çekilmiş formatları da korunarak ML.NET model üzerinde çalışacak şekilde hazırlanmıştır. ML.NET model üzerinde çevrimiçi çalışacak model için görseller sağlıklı ve sağlıksız olmak üzere iki farklı dosya yolu ile modelin kullanabileceği şekilde hazırlık işlemi tamamlanmıştır. PT ve ONNX üzerinde çevrimdışı çalışacak modeller için görseller orijinal resimler, HSV maskeler ve arka planı temizlenmiş ROI maskeler ile her birisi de kendi içerisinde sağlıklı ve sağlıksız olmak

üzere dosya yolları modelin kullanabileceği şekilde hazırlık işlemi tamamlanmıştır. Veri setinde bulunan asma yaprağı fotoğraflarına ait sağlıklı ve hastalıklı görsellerden kameradan doğrudan fotoğrafı çekilmiş hali ile birlikte arka planı temizlenmiş, HSV maskesi üretilmiş ve ROI alanı çizilmiş içeriklerinde yer aldığı örnek görsellere Şekil 2.3'te yer verilmiştir.



Şekil 2.3. Veri setinde yer alan sağlıklı ve hastalıklı asma yaprağı görselleri.

2.3 Unet Tabanlı Derin Öğrenme Mimarisi

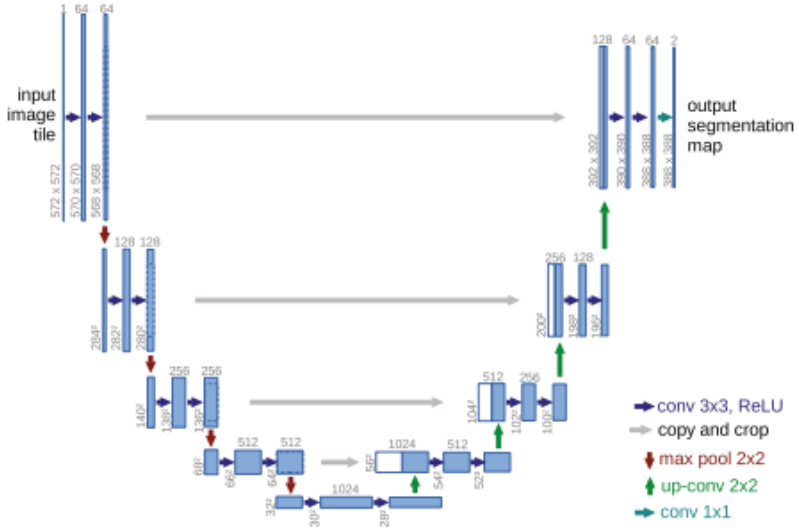
Bu bölümde çalışmada kullanılacak PT ve ONNX modelleri kapsamında Unet tabanlı derin öğrenme mimarisine yönelik detaylara yer verilmektedir. Görüntü segmentasyonu, nesneleri algılayarak görev adımlarına bilgi sağlanmak hedefiyle görüntüleri farklı pikseller halinde gruplara bölen bilgisayarlı görme tekniklerinden biridir [34]. Görüntü segmentasyonları için tasarlanmış atlama bağlantılarıyla simetrik bir yapıya sahip olan Unet, bu çerçevede yüksek doğruluk sonuçları veren bir derin öğrenme mimarisidir. Çalışma kapsamında Unet in kodlayıcı bölümünde ResNet50 kullanılarak model eğitimleri yapılmıştır. Derin ağlar kullanılarak yapılan eğitimlerde bozulma sorunlarını çözebilen artık öğrenme yapısında bir mimari yapıya sahiptir. Kodlayıcı olarak Unet

modelinde kullanılmasıyla daha derin katman yapılarından gradyan geçişi sağlar. Medikal alanda kullanılan bir çalışma kapsamında, tümör dokularla normal dokuların ayrımında benzer bir yapı kullanılmıştır [37]. Simetrik olarak çalışan Unet iki bölümden oluşmaktadır. Contracting path, CNN sinir ağına benzer şekilde 3x3 evrişim katmanı uygulayarak max pooling işlemleriyle çalışan, düşük ve yüksek seviyeli özellikler elde edebilmektedir. Özellik çıkarımlarını detaylı bir şekilde yapan kısım olarak söylenebilmektedir. Expansive path ise sınıflandırma haritası oluşturabilmek için çözünürlüklerin geri kazanıldığı bölümdür. Aynı seviye bulunan contracting path katmanlarına ait özellik haritalarının kopyalanmasıyla expansive path'e aktarım yapmaktadır. Bu sayede Unet bilgi kaybını minimuma indirecek şekilde çalışmaktadır. Bu sayede Unet detaylı ve doğru segmentasyon sunabilmektedir. Bu bölümler encoder ve decoder alt yapısında çalışmaktadır. Encoder kısmı giriş görüntüsünü kullanarak anlamlı özellik çıkararak her katmanda boyut küçülten yapıdır. Pooling işlemleriyle gerçekleşen bu sürece aşağı örnekleme (downsampling) adı verilmiştir. Encoder kısmında efficientnet-b4, resnet50, mobilenet_v2 gibi sınıflandırma modelleri çalışabilmektedir. Diğer bir ifadeyle, encoder kısmından gelen soyut bilgilerle kademeli bir şekilde orjinal boyuta görsele geri dönmek, özellikleri geri kazanmak için çalışan yapı olarak adlandırılabilir.

Transpoz konvolüsyonlar ya da interpolasyon yöntemleri kullanılarak yapılan bu işlem yukarı örnekleme (upsampling) adı verilmektedir. Decoder kısmında bu çok katmanlı harita kanallarının çalışmasıyla birlikte segmentasyona dâhil olan girdiyi kayıpları önleyerek yüksek çözünürlüğe dönüştürmektedir. Yukarı örnekleme işlemleri kullanarak özellik haritalarını büyütmektedir. Konum bilgilerini tekrar kazanmaktadır. Bağlantıları atla (Skip connections) bölümü, encoder adımında oluşan ilk katmanlardaki yüksek çözünürlüğe ait özellik haritalarından doğrudan decoder katmanlarına iletişim kurarak, detay özelliklerin kaybını en aza indirmeye ve doğru sınıflandırma yapmaya yardımcı olan atlamalı bağlantılar olarak adlandırılır.

Bu şekilde model hangi aşamada olduğunu öğrenebilmektedir. Bu birleşim kısmı kanallar ile gerçekleşmektedir. Unet, segmentasyon adımında oluşacak kanal sayısı, encoder sınıfı, ağırlık dosyası çıktı sınıf sayısıyla birlikte aktivasyon fonksiyonuyla parametrik olarak ayarlanıp metoda dâhil edilebilir. Unet modeli, giriş kısmıyla eşit boyutta ve sınıf olasılıklarını görseldeki piksel dâhilinde bulunduran bir sınıflandırma haritası

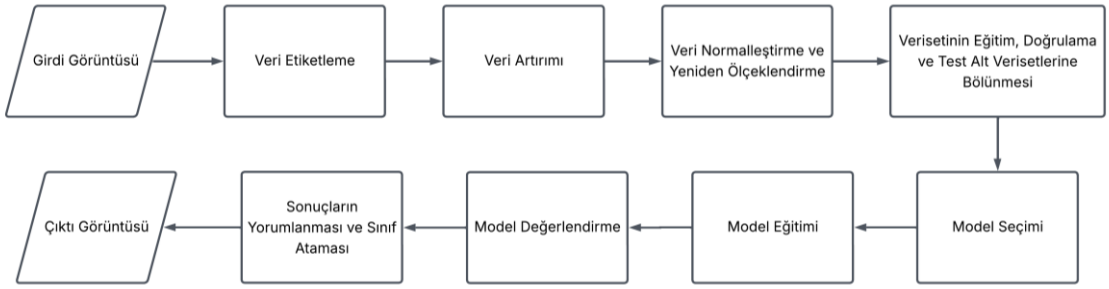
üretmektedir. Bu kapsamda Unet modeliyle örnek bir girdiden segmentasyonu yapılmış bir oluşan maske üretilmektedir. Bu kapsamda Unet modelinin aşama aşama hangi adımlarla ilerlediği, bununla birlikte encoder ve decoder yapılarının çalışma mantıkları Şekil 2.4'te gözlemlenebilmektedir.



Şekil 2.4. Unet tabanlı modelin çalışma mantığını anlatan görsel [30].

Bölüm 2.6.1 de ONNX ve PT çıktıları için kullanılmak üzere hazırlanmış 465 görsele ait veri seti; yüzde 70 oranında eğitim kümesi, yüzde 15 oranında doğrulama kümesi ve yüzde 15 oranında test kümesi olarak ayrılmıştır. Model eğitimleri için Pycharm IDE ortamı ve Python kodlama dili kullanılmaktadır. ResNet50 tabanlı bir Unet modeli oluşturularak encoder parametresi resnet50 belirlenmiştir. Önceden eğitilmiş bir biçimde yüklemesini yapabilmek için encoder ağırlıkları parametresi imagenet olarak belirlenir. Modelin daha hızlı ve doğru öğrenebilmesini sağlamak amacıyla bu şekilde ayarlanmıştır. 3 kanallı (RGB) olarak modelin girdi alacağı ifade edilmiştir. Sınıf değeri 1 yapılarak tek kanallı bir çıktı üretileceği bildirilmiştir. Çıktı değerinin 0 ile 1 arasında değerler alabilmesini sağlamak amacıyla activation parametresi sigmoid olarak belirlenip piksel odaklı sınıflandırmada kullanılacak şekilde model parametreleri tanımlanmıştır. Epoch değeri 20 tekrar olarak belirlenmiştir. Çalışmamızdaki veri setinin model eğitimi sırasında artırılması amacıyla, veri artırma metodu kullanılarak literatürde yapılan çalışma incelenmiştir [5]. Bu çalışma kapsamında yapılan veri artırma işleminden esinlenerek ek olarak model eğitimi sırasında program yolu ile veri artırma işlemi

yapılmaktadır. Tasarlanan sisteme ait derin öğrenme modelinin kullanılarak örnek bir veri setinin uçtan uca işlenmesini anlatan akış şeması Şekil 2.5'te görselleştirilmiştir.



Şekil 2.5. Derin öğrenme modeli kullanarak veri setini uçtan uca işleme akış şeması.

2.3.1 Epoch

Model eğitimi aşamasında tüm versitesinde bulunan örnek kayıtları modelin baştan sona tamamını bir kez görmesi durumu epoch olarak ifade edilmektedir. Bu çalışmada epoch değeri 20 olarak belirlenmiştir. Sağlıklı ve hastalıklı bütün yaprak verileri model eğitim aşamasını bu çalışmada 20 tekrar şeklinde tamamlamaktadır. Son tekrar işleminin sonucunda epoch tamamlanmış kabul edilerek eğitim aşaması sonlandırılmaktadır. Eğitim aşamasında epoch sayısını artırmak eğitilen modelin tüm veri setini baştan sona her seferinde tekrar görmesine öğrenme yeteneğini geliştirmesine yardım etmektedir. Modelin sınıflandırma aşamasında daha doğru tahmin yapabilmesine olanak sağlamaktadır [38].

2.3.2 Kayıp Fonksiyonu

Sinir ağlarındaki eğitimin temelinde modelin yapmış olduğu tahminlemeler ile gerçek değerlerin arasında bulunan farkı en az değere azaltmak amacı bulunmaktadır. Bu farkın ölçümlenmesi amacıyla kullanılmakta olan yönteme kayıp fonksiyonu (loss function) denilmektedir. Bu fonksiyonun çıktısında bize modelin eğitim aşamasındaki öğrenme sürecine ilişkin bilgi vermektedir. Eğitilen modelde öğrenim süreci için başarılı olup olmadığını sayısal olarak grafik üzerinde gözlemlememize olanak sağlamaktadır. Bu kapsamda model eğitiminde analiz ve iyileştirme sürecinde bize yardımcı olabilmesi için kullanılmaktadır [38].

Logloss kayıp fonksiyonunun eşitliğini temsil etmektedir. Eşitliğin devamında formül içerisinde N toplamda tüm veri setinde bulunan kayıt sayısını temsil etmektedir. Her bir örneğin sırasını i sembolü ve doğal logaritmayı da ln sembolü temsil etmektedir. Bununla birlikte çalışmada kullanılan modelin hastalıklı ve sağlıklı sınıflandırması yönünde y_i görselin etiketini temsil etmektedir. Sağlıklı yaprak y_0 ile hastalıklı yaprak y_1 ile sembolize edilerek iki sonuçlu sınıflandırma seçeneği için kullanılmaktadır. Formülde bulunan diğer bir parametre p_i ise i ye karşılık gelen değerde bulunan örnek görselin hastalıklı yaprak olup olmama olasılığı için verdiği değere karşılık gelmektedir. Bu durumda 0 ile 1 arasında bir değere sahip olmaktadır [35]. Formül eşitliği için kayıp fonksiyonu için Denklem (2.2) de verilmiştir [39].

$$Logloss = -\frac{1}{N}\sum_{i=1}^{N} y_i \ln(p_i) + (1 - y_i) \ln(1 - p_i) \quad (2.2)$$

2.4 Doğruluk Ölçümlemede Uygulanan Değerler

Bu bölümde deney değerlendirme performans metriklerinde kullanılacak Gerçek Pozitif (TP), Gerçek Negatif (TN), Yanlış Pozitif (FP) ve Yanlış Negatif (FN) ifadelerine yer verilmektedir. Doğruluk ölçümlerinde kullanılan değerlere ilişkin hesaplanma mantığına yönelik detaylar kapamlı bir şekilde ifade edilmektedir. Modelin sınıflandırma sonucunda TP, doğru şekilde sağlıklı tahmin sınıflandırılması yapılan sağlıklı yaprağı ifade eder. TP yüksek olması durumunda erken teşhis açısından modelin hasta yaprakları doğru bir şekilde yakaladığı ifade edilebilmektedir. Modelin sınıflandırma sonucunda TN, doğru şekilde hastalıklı tahmin sınıflandırılması yapılan hastalıklı yaprağı ifade eder. TN yüksek olması durumunda model sağlıklı yaprakların sınıflandırmasını doğru bir şekilde yapabilmektedir. Modelin sınıflandırma sonucunda FP, hatalı şekilde sağlıklı tahmin sınıflandırılması yapılan hastalıklı yaprağı ifade eder. FP yüksek olması durumunda model sağlıklı yaprakları hasta olarak sınıflandırıyor olarak değerlendirilebilmektedir. Modelin sınıflandırma sonucunda FN, doğru şekilde hastalıklı tahmin sınıflandırılması yapılan sağlıklı yaprağı ifade eder. FN yüksek olması durumunda model hasta olan yaprakların sınıflandırmasını doğru yapamıyor ve yanlış sınıflandırma yapıyor şeklinde ifade edilebilmektedir [37].

Metriklerin hesaplanmasında kullanılan denklemler için TN, TP, FN ve FP değerleri çevrimdışı kullanım için yerel veri tabanından, çevrimiçi kullanım için çevrimiçi veri tabanından kaydedilen sınıflandırma sonuçlarına göre hesaplanır.

Yaprak verilerinin gerçek ve tamin durumları üzerinden doğruluk değerleri bulunmaktadır. Yaprağın gerçekte hastalık mı sağlıklı mı olduğunu etiketleyen durum gerçek durum olarak ifade edilmektedir. Modelin sınıflandırma sonucunda verdiği sonuç tahmin olarak ifade edilmektedir. Duyarlılık, kesinlik, F1 Score ve doğruluk performans metriklerinin hesaplanmasında doğruluk ölçümleme değerleri kulanılmaktadır. Bu kapsamda karmaşıklık matrisi olarak karşılaştırmaları Tablo 2.1'de gösterilmektedir [38].

Tablo 2.1. Karmaşıklık matrisi [38].

	Pozitif Tahmin	**Negatif Tahmin**
Gerçek Pozitif	Doğru Pozitif	Yanlış Negatif
Gerçek Negatif	Yanlış Pozitif	Doğru Negatif

2.5 Deney Değerlendirme Performans Metrikleri

Model eğitimi sonucunda yapılan tahmin işlemlerinde ve gerçek ortamda alınan görsellerle elde edilecek tahmin sonuçlarını analiz amacıyla kullanılan metriklere yönelik detaylara bu bölümde yer verilmiştir. Modellerin karşılaştırılmasında kullanılmak için elde edilen sınıflandırma bilgileri ile kullanıcının geri dönüşü ile saklanan veriler yardımıyla başarı metrikleri hesaplanmaktadır. Bu kapsamda TP, TN, FN ve FP değerleri performans metriklerinin hesaplanmasında kullanılmaktadır. Bu değerler kullanılarak değerlendirme ölçütlerinin hesaplanması için kullanılan denklemler detaylandrılmaktadır. Sağlıklı ve hastalıklı yaprakların analizinde hastalığın doğru tahmininde kullanacağımız duyarlılık metriği 2.5.1 numaralı bölümde, hastalıklı yaprakların gerçekten de hastalıklı olması oranını değerlendirebileceğimiz kesinlik metriği 2.5.2 numaralı bölümde detaylandırılmıştır. Bununla birlikte kesinlik ve duyarlılık değerlerinin birlikte değerlendirildiği modelin genel başarısının yorumlanmasında kullanacağımız F1 Score metriği 2.5.3 numaralı bölümde ve hem sağlıklı hem de hastalıklı yaprakların doğru sınıflandırma yapma oranını gözlemleyebileceğimiz doğruluk metriği 2.5.4 numaralı bölümde hesaplanma formülleri ile birlikte detaylıca açıklanmaktadır.

2.5.1 Duyarlılık

Duyarlılık, model eğitimi ve ya deney sonucu çıktılarında verinin doğru tespitinin oranı olarak ifade edilebilmektedir. Asma yaprağı üzerinden tahmin sonucunda gerçek anlamda hasta olarak bilinen yaprakların model aracılığıyla doğru tespit edebilmesi oranı olarak söylenebilmektedir. Duyarlılık değerinin yüksek olması modelin tahmin gücünün kuvvetli olduğunu ve modelin hasta yaprakları gözden kaçırmadığını göstermektedir. Gerek tarımda gerekse medikal alanlarda önemli bir veri olarak düşünülebilmektedir. Hasta verinin gözden kaçmaması hastalığın yayılmasında ya da önlem alınması noktasında değerli bir nokta olarak tanımlanabilmektedir. TP değerinin TP ve FN değerlerinin toplamına bölümünde ortaya çıkan sonuç ile duyarlılık değeri hesaplanmaktadır. Duyarlılık değeri Denklem (2.3)'de ifade edilmiştir [37].

$$\text{Duyarlılık} = \frac{TP}{TP+FN} \tag{2.3}$$

2.5.2 Kesinlik

Model eğitimi sonucunda, model üzerinden tahminleme yapıldığında modelin hastalıklı olarak tahmin ettiği yaprak görsellerinin doğru olarak gerçekten hasta olması oranı olarak ifade edilebilmektedir. Kesinlik değerinin yüksek olması yanlış tahminleme yapmadığını göstermektedir. Sağlıklı yaprak görsellerini hasta olarak tahmin etmemesi olarak söylenebilmektedir. Bu kapsamda kesinlik değeri de tarımsal uygulamalarda önemli bir veri olarak gözlemlenebilmektedir. Bu gibi uygulamalarda sağlıklı yapraklara hasta demesi yanlış teşhis olarak ortaya koyacağı veriyle birlikte gereksiz ilaçlama da yapılabilmesini sağlar. TP değerinin TP ve FN değerlerinin toplamına bölümünde ortaya çıkan sonuç ile duyarlılık değeri hesaplanmaktadır. Kesinlik değeri Denklem (2.4)'de ifade edilmiştir [37].

$$\text{Kesinlik} = \frac{TP}{TP+FP} \tag{2.4}$$

2.5.3 F1 Score

F1 Score başarı metriği kesinlik ve duyarlılık değerlerinin dengeli bir biçimde birleşmesi sonucunda ortaya çıkmaktadır. Eğitilen model çok dikkatli biçimde analiz ederek bu değer kapsamında değerlendirme yapabiliyorsa, sadece yaprakların kesin hastalıklı olduğunda işaretleme yapıp kesinlik değerini yüksek, duyarlılık değerini düşük

verebilmektedir. Duyarlılık ve kesinlik değerlerinin ikisini birlikte değerlendirerek modelin tahminlemesindeki genel başarıyı yorumlamamızı sağlamaktadır. TP değerinin iki katının TP değerinin iki katı ile false positive ve false negative değerlerinin toplamına bölümünde ortaya çıkan sonuç ile F1 Score değeri hesaplanmaktadır. F1 Score değeri Denklem (2.5)'de ifade edilmiştir [37].

$$\text{F1 Score} = \frac{2xTP}{2xTP+FP+FN} \tag{2.5}$$

2.5.4 Doğruluk

Doğruluk performans metriği eğitim sonucunda elde edilen modelin, hem sağlıklı hem de hastalıklı yaprakları doğru sınıflandırma yapma oranını ifade etmektedir. Doğruluk değerinin yüksek olması, sistemde kullanılan modelin sınıflandırmada genel olarak doğru bir yaklaşım ortaya koyduğunu, hastalıklı ve sağlıklı yaprakları doğru tanımlayabildiğini ifade etmektedir. Model eğitiminde kullanılan veri setinde hastalıklı yaprak sayısının az olması durumunda, model tahminlerinde yalnızsa sağlıklı diyerek yüksek doğruluk alma durumu bulunmaktadır. Bu gibi durumlarda yanıltıcı bir veri elde edilebileceği için, veri setinde bulunan orantılı dağılım bu değerin hesaplanmasında önemli rol oynamaktadır. TP ve TN değerlerinin toplamının true positive, true negative, false positive ve false negative değerlerinin toplamına bölümünden elde edilen sonuç ile doğruluk değeri hesaplanmaktadır. Doğruluk değeri Denklem (2.6)'da ifade edilmiştir [37].

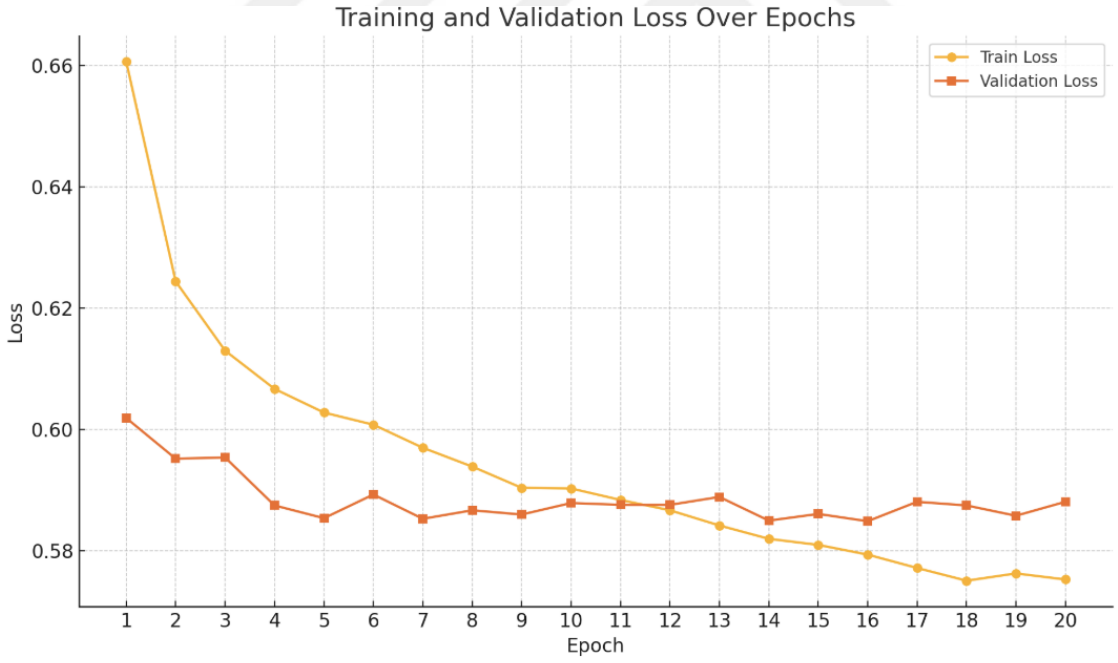
$$\text{Doğruluk} = \frac{TP+TN}{TP+TN+FP+FN} \tag{2.6}$$

2.6 Yaprak Analizi İçin Hazırlanan Sınıflandırma Modelleri

Bu bölümde hazırlanan önişlem süreçleri tamamlanmış veri seti ile birlikte derin öğrenme tabanlı Unet mimarisinin model eğitim süreci hakkında bilgiye yer verilmektedir. Bu kapsamda sınıflandırma ve analiz aşamasında kullanılan yöntemlere ilişkin detaylar açıklanmaktadır. Uygulamada çevrimiçi ve çevrimdışı çalışma modellerde ortak olarak kullanılacak olan PT ve ONNX modellerinin hazırlanma sürecine 2.6.1 numaralı bölümde yer verilmiştir. Uygulamada sadece çevrimiçi çalışma modelinde kullanılacak olan ML.NET modelinin hazırlanma sürecine 2.6.2 numaralı bölümde yer verilmiştir. Çalışma kapsamında hazır biber veri seti için de benzer eğitim aşamaları uygulanarak asma yaprağındaki süreçler ortak olarak uygulanmıştır.

2.6.1 PT ve ONNX Model Eğitimi

Bu kısımda PT ve ONNX model dosyalarının hazırlanma sürecine dair bilgilendirmeler yer almaktadır. Eğitim aşamasında kullanılan veri seti için 2.1 ve 2.2 numaralı bölümlerde verilerin hazırlanma süreci hakında detaylı bilgilere yer verilmiştir. Çalışmamızda hazırlanmış eğitime hazır hale getirilmiş veri setinde yer alan orijinal, HSV ve ROI maskelerine ait görsellerin eğitimi sırasında Python kodlama dili kullanılmıştır. Model eğitim sürecinde görsel işleme tekniklerinden Unet tabanlı derin öğrenme mimarisi kullanılmaktadır. En düşük 32x32 piksele ait bir görsel için çalışmaktadır. Girdi olarak kullanılan görseli çok katmanlı bir yapıda çıktı olarak sınıflandırmaktadır. Unet medikal alanlarda yapılan görüntü sınıflandırma konusunda başarılı bulunmaktadır [37]. Bu kapsamda model çıktılarımızı elde etmemizde kullandığımız Unet tabanlı derin öğrenme modeline dair detaylara 2.3 numaralı bölümde kapsamlıca açıklanmıştır. Unet modeli hazırlanan yaprak veri seti üzerinde uygulanmış ve model eğitimi sonucunda elde edilen kayıp fonksiyonu grafiği Şekil 2.6'daki gibi gözlemlenmiştir.



Şekil 2.6. Asma yaprağı model eğitimi kayıp fonksiyonunu gösteren grafik.

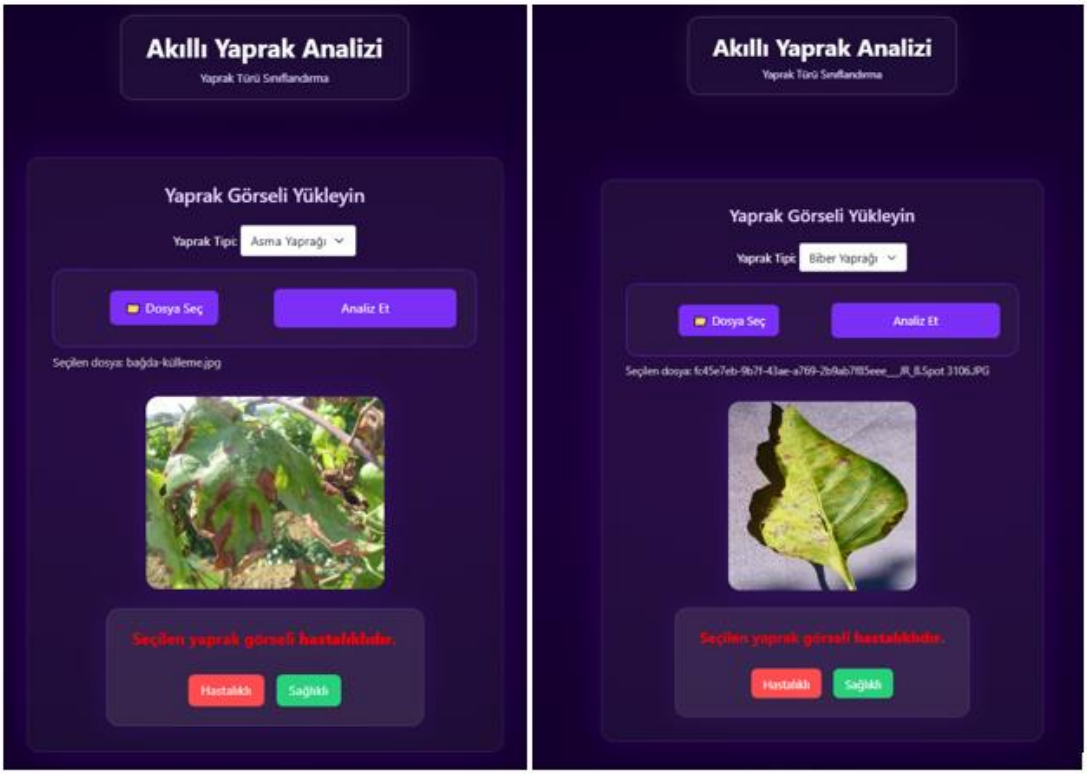
Grafikte, eğitim ve doğrulama kayıplarının eğitim boyunca her tekrarda düzenli olarak azaldığı izlenmiştir. Doğrulama kaybı 4 ile 7 epoch aralığında en düşük seviyelerine ulaşırken devamında sabitlenmiştir. Test veri seti ile model üzerinde sınıflandırma yapılmasıyla birlikte bazı metrikler elde edilmiştir. Doğruluk değeri 0.7594, kesinlik

değeri 0.8475, duyarlılık değeri 0.7113 ve F1 Score değeri 0.7734 şeklinde gözlemlenmiştir.

2.6.2 ML.NET Eğitimi

Bu kısımda ML.NET model dosyasının hazırlanma sürecine dair bilgilendirmeler yer almaktadır. Daha önceki bölümlerde verilerin hazırlanma süreci detaylandırılmış olup, ML.NET model çıktısında kullanılmak üzere hazırlanmış 3122 görsele ait veri seti sağlıklı ve hastalıklı olacak şekilde iki farklı eğitim kümesine ayrılmıştır. Tamamen orijinal resimlerin içerikleri ve boyutları da korunarak saf haliyle eğitim sürecine dâhil edilmiştir. Microsoft tarafından geliştirilen ML.NET bu çalışma kapsamında bize farklı modeller ile kıyaslayabilme olanağını sunmaktadır. Python gibi ek platformlara ihtiyaç duymadan doğrudan makine öğrenmesi modeli üretmemize olanak sağlamaktadır. ML.NET model eğitimi için Visual Studio Entegre Geliştirme Ortamı (IDE), .Net Core çatısı altında C# dili kullanılarak Web Uygulama katmanlarıyla birlikte tasarlanmıştır. Çalışmamız kapsamında görüntü sınıflandırma yöntemiyle ImageClassificationMulti eğiticisi kullanılmıştır. ImageClassificationMulti TL temelli bir yaklaşım olup büyük veri kümeleri üzerinde eğitilmiş son katmanları tekrar eğitim aşamasına dâhil ederek sınıflandırma adımının yapılmasını sağlar. Resnet50 gibi sınıflandırma yöntemlerini kullanmaktadır. Visual Studio IDE yardımıyla model eğitimi aşaması tamamlanmıştır. Logaritmic kayıp azalışı (Logarithmic loss reduction) metriği kullanılmıştır. Logaritmic kayıp azalışı, modelin tahmin olasılıklarının doğruluğu yönünde ölçüm yaparak başarılı bir değerlendirme sürecinin tamamlanması için imkân sunmaktadır. Tamamlanan çalışmada ImageClassificationMulti eğiticisi ile yapılan eğitim sürecinin sonucunda 0,0508'lik bir log loss azalması elde edilmiştir. Bu değeri modelin sınıflandırma başarısında pozitif yönde bir iyileşme sürecinin olduğu düşünülmektedir [31-32].

Seçilmiş yaprak tipi üzerinden dosya seç butonu kullanılarak cihaz hafızasında bulunan bir görsel seçilerek analiz et butonuyla birlikte yaprağın sınıflandırma sonucu elde edilmektedir. Elde edilen sonucun doğru mu yanlış mı olduğu bilgisi tekrar kullanıcıdan geridönüş yoluyla hastalıklı ve ya sağlıklı seçeneklerinden biriyle toplanabilmektedir. Bu bilgiler seçimin yapılmasıyla birlikte çevrimiçi veri tabanında doğrudan saklanabilmektedir. ML.NET model için web arayüzü kullanılarak yapılmış sınıflandırma örneğine ait görseller Şekil 2.7'de gösterilmiştir.



Şekil 2.7. ML.NET kullanılarak yapılan çevrimiçi yaprak sınıflandırma örnekleri.

2.7 Yerel ve Çevrimiçi Veri Tabanları

Bu bölümde çalışmada kullanılan model ile birlikte tahminlemesi yapılan sağlıklı ve hastalıklı yaprak verilerinin saklanmasına yönelik detaylar verilmektedir. Hem özgün deneysel verilerle hem de hazır veri seti için ortak şekilde çalışabilmektedir. Verilerin saklanmasında mevcut modelin sürümleme yöntemiyle kendisini geliştirmesi amaçlanmaktadır. Bu doğrultuda yapılan tahminelemelerin sonuçları kullanıcıdan alınacak geri dönüş bildirimiyle birlikte orijinal halini korumaktadır. Analiz ve iyileştirme süreçlerinde de faydalı olmasıyla birlikte geri dönüş bildirimlerinin pozitif kullanılması durumunda sistemin kendini yenileme süreçlerine faydalı olacağı düşünülmektedir. Bu kapsamda tahminlemesi yapılan yaprak verilerinin mobil cihazlarda yerel olarak saklanabilme özelliğinin olmasıyla birlikte çevrimiçi olarakta saklanabilmesine olanak sağlamaktadır. İlgili veriler Uygulama Programlama Arayüzü (API) yardımlarıyla çevrimiçi veri tabanına gönderilmektedir. Bu sayede hem yerel veri tabanında hem de çevrimiçi veri tabanında kayıtlar saklanabilmektedir. Hem çevrimiçi hem de çevrimdışı yerel bir cihazda kurulu olarak çalışabilen MySQL veri tabanı

hakkında 2.7.1 numaralı bölümde bilgi verilmektedir. Aynı zamanda yerel veri tabanı kapsamında mobil cihazlarda Android uygulamalarda kullanılabilen SQLite veri tabanı hakkında 2.7.2 numaralı bölümde ve yine mobil cihazlarda Android uygulamalarda kullanılabilen Room veri tabanı hakkında 2.7.3 numaralı bölümde bilgi verilmektedir.

2.7.1 MySQL Veri Tabanı

Bu tez çalışmasında çevrimiçi sınıflandırma verilerini saklamak, analiz ve iyileştirmelerde bulunmak amacıyla MySQL veri tabanı kullanılmaktadır. MySQL çeşitli uygulamalarda tercih edilen performansı yüksek ve güvenirliği olan sunucuyla istemciler arasında ilişkisel bir yönetim sistemi olarak ifade edilen veri tabanı olarak tanımlanmaktadır. Çevrimiçi sınıflandırmayı da hedef alan çalışmamız açısından fazla veri akışı durumunda MySQL veri tabanının düşük bellek kullanımıyla yüksek bir sorgu işleme kapasitesine sahip yapısı hakkında bilgi verilmektedir. Modelimizin kullanılabilirliği ve sürdürülebilirliği açısından verimlilik gerektiren ortamlarda önemli bir özelliğe sahip olduğu düşünülmektedir. Farklı donanıma sahip ortamlarda da çalışabilen esnek yapısına değinilmektedir [43].

Yapılan bir araştırmada veri tabanının hacmindeki artışın performans ve kararlılıktaki etkisinin önemli bir yer edindiği söylenmektedir. Bu kapsamda bir içme suyu şişeleme şirketindeki veri tabanı yönetim sistemlerinden Oracle, MySQL ve PostgreSQL veri tabanlarının yanıt süreleri karşılaştırılmıştır. Bu doğrultuda en iyi yanıt süresini MySQL, devamında Oracle ve PostgreSQL takip etmektedir. Çalışma sonucunda MySQL veri tabanının yanıt süresi açısından Oracle ve PostgreSQL veri tabanlarından daha performanslı bir sonuç verdiği gözlemlenmiştir [48].

2.7.2 SQLite Veri Tabanı

Sunucu ihtiyacı olmadan büyük ölçekli web uygulamalarının yerine küçük ve orta ölçekte web uygulamalarında kullanılan hafif bir veri tabanı olan SQLite ilişkisel bir veri tabanı yönetim sistemidir. Arka planda etkin bir şekilde kullanılabilen SQLite birçok farklı bilginin bünyesinde depolanmasına ve verilerin yönetilmesine olanak sağlamaktadır. SQLite veri tabanı tek bir dosya şeklinde saklanıp başka ortama kolayca aktarılabilmektedir [47]. Ruh sağlığıyla ilgili geliştirilmiş bir mobil uygulama kapsamında bu kapsamda yapılmış uygulamaların kullanıcı deneyimleri ve ihtiyaçları

doğrultusunda değerlendirme yapılmaktadır. Bu kapsamda yapılmış mobil uygulama Android Studio IDE geliştirme aracı kullanılarak veri tabanı tarafında SQLite kullanılarak gözlem yapıldığı söylenmektedir [40].

2.7.3 Room Veri Tabanı

Room veri tabanı, bir tür Nesne İlişkisel Eşleme (ORM) kütüphanesi olarak ifade edilmektedir. ORM ise veri tabanıyla kod bloğı arasında bir köprü niteliği görevi görmektedir. Mobil uygulamalar için kullanılabilen bir veri tabanı türüdür. Veri tabanı üzerinde tanımlanan alanları Java nesnelerine dönüştürme işlemini sağlayan bir tür kütüphane olarak tanımlanmaktadır. Yaygın kullanım amaçlarından birisi olan verilerin önbelleğe alınmasını sağlarken cihazın internete bağlı olmasına gerek kalmadan içeriklere erişim sağlama imkânı sunmaktadır. Diğer bir avantajı ise tanımlanan sorguların hatalı olması durumunda derleme aşamasında doğrulama yaptığı için uygulamanın çalıştığı durumlarda çökmesini engelleyerek erken tespit edilmesini sağlamaktadır. Android bileşenlerine kolayca dâhil edilebilmektedir. Bu sağladığı özelliklerle Android tabanlı mobil uygulamalarda SQLite veri tabanı yerine Room veri tabanı kullanımının tercih edilebileceği önerisi yapılmaktadır [49].

Yapılan farklı bir araştırmada da SQLite veri tabanının genelde küçük gereksinime sahip olan projelerde tercih edildiğinde avantajlı olacağı söylenmektedir. Bununla birlikte Room veri tabanı için ise geniş kapsamı olan projeler için gelişmiş niteliklere ihtiyacı olan uygulamalar için daha avantajlı olabileceği ifade edilmektedir. Her iki veri tabanınında kendine özel güçlü taraflarının olduğu değerlendirilmektedir [50].

2.8 Uygulama Programlama Arabirimleri

Bu bölümde çalışmada kullanılan model ile birlikte tahminlemesi yapılan sağlıklı ve hastalıklı yaprak verilerinin saklanmasına yardımcı olacak uygulama programlama arabirimlerine yönelik detaylar verilmektedir. Yerel veri tabanında saklanan mobil cihazdan alınmış verilerin çevrimiçi veri tabanına aktarılmasında yardımcı olan hizmetlerin çalışmasına yönelik bilgi verilmektedir. Uygulama programlama arabirimleriyle sınıflandırılması yapılan yaprak verilerinin yeni bir veri seti oluşturma olanağıyla birlikte, mevcut modele sürekli veri aktarımı sağlaması uygulamanın sürümlenebilmesine ve kendini geliştirmesine katkı sağlayacaktır. Bu hizmetler ile mobil

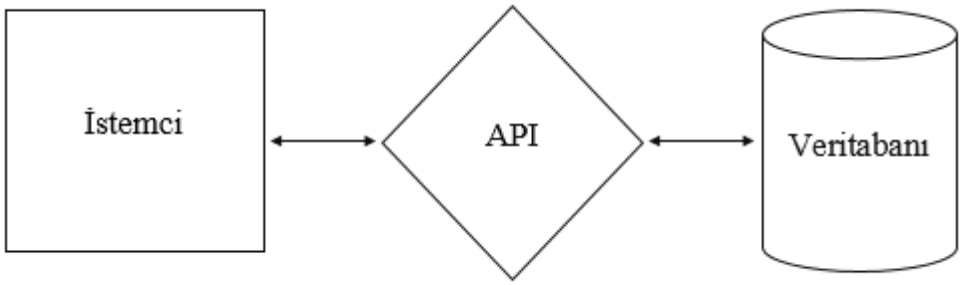
uygulama ile modelin tekrar eğitilebilmesini sağlayacak bir katman olması sağlanıp uygulamaya pozitif anlamda katkı sağlaması hedeflenmektedir. Analiz ve iyileştirme doğrultusunda mevcut veri akışını sağlayan bu arabirimlere yönelik iki farklı çalışma mantığına Web API servisleri için 2.8.1 numaralı bölümde ve Fast API servisleri için 2.8.2 numaralı bölümde detaylıca yer verilmektedir.

2.8.1 Uygulama Programlama Arayüzü

Bir uygulama için tasarlanmış işlevlerin farklı bir uygulama tarafından da kullanılmasına olanak sağlayan yapı için API ifadesi kullanılmaktadır. Farklı uygulamalar tarafından kullanılabilen arayüz olarak ifade edilebilmektedir. İki farklı yazılım ve ya veri tabanına ait sistemlerin birbirleri arasında hatasız çalışabilmesi ve birbirleri arasında iletişime geçilmesini API uygulamaları sağlamaktadır [36].

2.8.2 Web API Servisleri

Hiper Metin Aktarım Protokolü (HTTP), sunucu ve son kullanıcıların birbirleri arasındaki iletişimi sağlayan, birbirleri arasında verilerin nasıl aktarılması gerektiğine ait yöntem ve kuralları düzenleyerek uygulama seviyesinde faaliyet gösteren bir haberleşme protokolü olarak tanımlanmaktadır. Bu protokol yardımıyla kullanıcılar bir kaynağın üzerinde hangi işlemin yapılması gerektiğini kullanıcı istekleriyle belirleyebilmektedir. Bu istekler dört temel tür ile GET, POST, PUT ve DELETE olacak şekilde sınırlandırılarak HTTP istekleri olarak adlandırılabilmektedir. HTTP isteklerindeki 4 temel istek; GET okuma eylemlerinde, POST kayıt oluşturma işlemlerinde, PUT verilerin güncelleştirilmesinde ve DELETE silme işlemlerinde kullanılmaktadır. Bu çalışma üzerinde HTTP isteklerinin çalışma mantığı görselleştirilerek bir örnek ile çalışma içerisinde sunulmaktadır. Çalışma mantığı kapsamında görselde, istemci tarafından iletilen isteklerin API tarafında karşılanacağı aktarılmaktadır. API tarafına gelen istek gerekli işleme süreciyle birlikte süzülerek veri tabanına verilerin işlendiği, veri tabanından sorgu sonucunda gerekli veri setinin tekrar API tarafına iletildiği ve API vasıtasıyla tekrar istemciye veri setinin uygun formatta iletildiği bilgisine yer verilmektedir. Bu tez çalışmasında da kullanılan benzer bir mantık olduğu düşünülerek tekrar görselleştirilerek Şekil 2.8'de API isteklerinin çalışma mantığına yer verilmektedir [46].



Şekil 2.8. HTTP isteklerinin kullanıcı ile veri tabanı arasındaki iletişimi [46].

2.8.3 Fast API Servisleri

Bu çalışma kapsamında çevrimiçi çalışma mantığında hizmet sağlayıcısı olarak kullandığımız Fast API, Python 3.7 ile ve ya daha üzerindeki sürümlerle çalışabilen, performansı hızlı ve modern bir yapıda API oluşturmaya olanak sağlayan web çatısı olarak adlandırılmaktadır. Hızlı geliştirebilme imkânıyla birlkite düşük hata riskini barındıran, aynı zamanda kolay öğrenilebilir olmasıyla ve otomatik dokümantasyona destek vermesi nedeniyle tercih edilebilmektedir. Kaynak araştırması kapsamında ilgili çalışmada Fast API hizmet sağlayıcısı kullanılarak veri iletişimi sağlanmıştır. İlgili Fast API servisleri Tekdüzen Kaynak Bulucuları (URL) tarafından erişilebilir duruma getirildikten sonra, çalışma kapsamında yapılan mobil tabanlı uygulama tarafından URL'lere istek göndererek istenilen sonuçlar elde edilmiştir. Veri iletişimi sonucunda alınan cevaplar ve atılan istekler SQLite3 veri tabanı kullanılarak işlendiği aktarılmaktadır [45].

2.9 Çalışma Modelleri

Bu bölümde çalışmada yapılmış olan uygulamanın çalışma modelleri hakkında bilgi verilmektedir. Uygulamada çevrimiçi ve çevrimdışı çalışma modeli olmak üzere iki farklı çalışma modeli bulunmaktadır. Çalışmada sunulan uygulama yöntemlerinde modelin çalışma tercihlerine yönelik 2.9.1 numaralı bölümde çevimiçi ve 2.9.2 numaralı bölümde çevrimdışı modellerde kullanılmak üzere tanımlanmış sistem bilgileri hakkında detaylıca bilgi verilmektedir. Bu çalışma modelleri Android tabanlı mobil bir uygulamada çalışmaktadır. Uygulama Kotlin dilinde Android Studio IDE ortamında geliştirilmiştir. Uygulama tarafında çevrimiçi ve çevrimdışı çalışabilecek bir sınıflandırma yöntemi

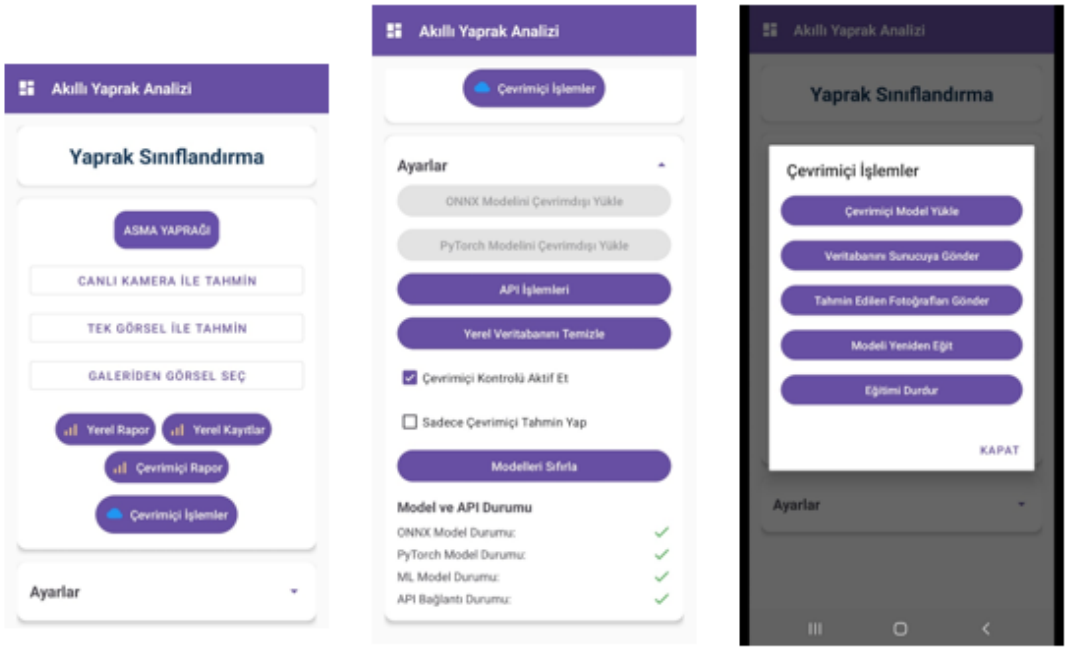
tasarlanmıştır. Görsel sınıflandırma için tüm kategoriler düşünülerek ilerlenmiştir. Bu kapsamda mobil uygulama içerisinde cihazın kendi kamerası kullanılarak canlı görsel yakalama ile canlı görsel sınıflandırma, çekilen tekli fotoğraf ve ya galeriden seçilebilen görsel ile doğrudan sınıflandırma yapabilecek iki farklı yöntem bulunmaktadır. Canlı kamera üzerinde görseller Image Proxy yöntemiyle görsel haline çevrilerek, tekli resim yöntemleriyle ise doğrudan yakalanan görseller kullanılarak tahmin yöntemlerine iletilmektedir.

Yakalanan görsel için ilk olarak HSV maskeli görsel oluşturulur. HSV maskesi oluşturulan görsel için devamında ROI bölgesinin çizimi gerçekleştirilir. Bir görselin hastalıklı mı sağlıklı mı olduğunun kararına, yaprağın ROI bölgesi içerisinde kalan kısımlarında yapılan hesaplamalar sonucunda elde edilen severity index (SI) değerine göre karar verilmektedir. Bu alandaki SI alanı hesaplaması yapılırken, görüntüde belirli büyüklükteki kareler oluşturup her bölge kendi içerisinde hastalıklı olup olmadığının kontrolü yapılır. Seçilen bölge ROI bölgesinde tespit edilen yaprak görüntüsü ile sınırlandırılmıştır. Eğer seçilen bölgedeki en az bir piksel ROI bölgesindeyse, bu bölge ROI içerisinde kabul edilir. Bu doğrultuda pikseldeki renk aralıkları kontrol edilir. Piksellerden en az bir tanesi belirlenmiş renk aralığında ise o bölge hasta olarak kabul edilir ve hastalıklı bölge olarak işaretlenir. Renk aralığı sağlıklı yapraklar üzerinden seçilen renklerle bir renk paketi oluşturulmuştur. Eğer bir bölge hem ROI bölgesi içerisinde yer alıp hem de en az bir hastalıklı piksel barındırırsa, o bölgenin merkezi temel alınarak koordinatı bir listede saklanır. Bu liste hastalıklı olarak işaretlenen alanları temsil edecektir. Yaprak üzerindeki hastalıklı alanın saydam kırmızı rengiyle işaretlenmesine yardımcı olmaktadır. Ek olarak ROI bölgesindeki piksel sayısı ile bu piksellerden hastalıklı olmayan piksel sayılmaktadır. Bu bilgiler yapraktaki hastalık oranının tespit edilmesinde ve SI oranının hesaplanmasında kullanılmaktadır. SI hesaplaması, yaprağın ROI bölgesinde bulunan yeşil oranına göre belirlenmektedir. SI oranı 0.95 üzerinde ise yaprak sağlıklı olarak değerlendirilir. Sistem yaprak üzerinde yeşil renk aralıkları tek tek hesaplamaktadır. Bazı yaprakların karanlıkta kalmasıyla ışık almayan kısımların farklı renklerde olması ya da bazı yaprakların güneşte kalmasıyla ışık vuran kısımlarının farklı renklerde olması sonucunda, sistemin sağlıklı yaprağı hastalıklı olarak tanımlayabilme olasılığı bulunmaktadır. Bu yüzden 0.05'lik bir pay bu kapsamda düşünülerek 0.95 oranı

belirlenmiştir. Sınıflandırması yapılan yaprak yerel veri tabanında saklanmaktadır. Yine uygulama altındaki paket altında orijinal ve maskeli görseller dosyalanmaktadır.

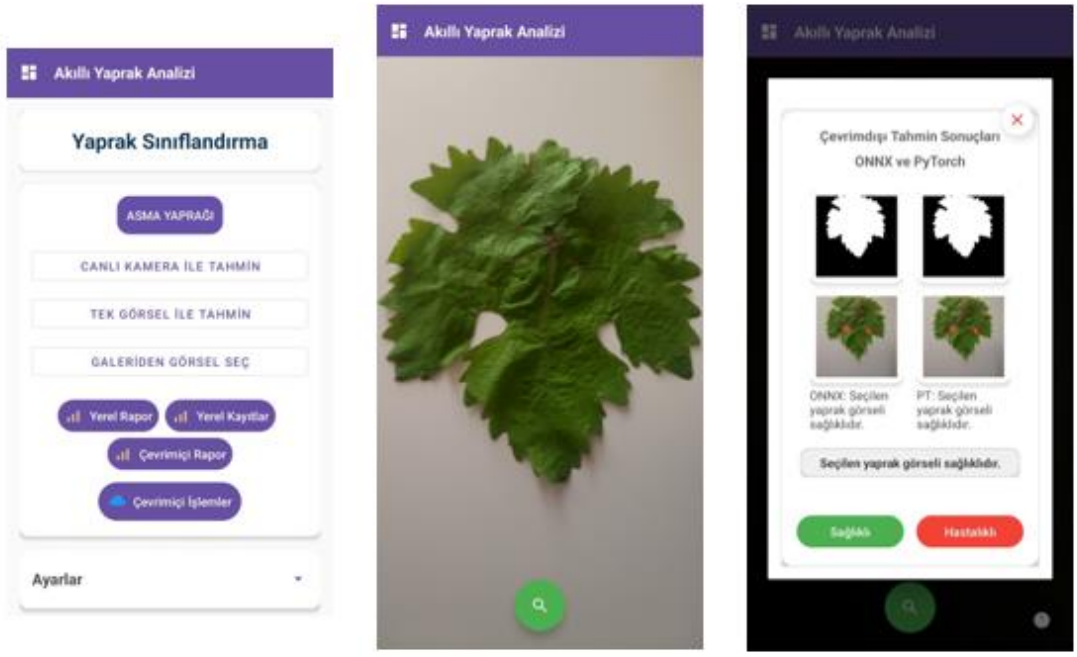
Sınıflandırılan görseller çevrimiçi tutulan veri tabanına iletilebilmektedir. Çevrimiçi bağlantının sağlanabilmesi için API ve HTTP yöntemleri kullanılmaktadır. Servislerle iletişim sağlanırken Python tarafında PT ve ONNX modelleri için Fast API yoluyla, ML.NET model için .NET Core Web Uygulamasında bulunan ilgili HTTP servisleriyle tahmin sonuçları çevrimiçi veri tabanında saklanabilmektedir. Aynı şekilde görseller de ilgili klasör yapısına uygun biçimde tutulabilmektedir. Saklanan görseller ile tekrar ML.NET, ONNX ve PT modelleri eğitilebilmektedir. Güncel sürümleri mobil uygulamaya API yardımıyla servislerle indirilerek güncel model ile çalışma olanağı sağlanmıştır. Bu kapsamda 2.9.1 numaralı bölümde çevrimiçi ve ve 2.9.2 numaralı bölümde çevrimdışı çalışma modellerinde galeri üzerinden görsel seçerek sınıflandırma yöntemlerinin uygulanmasına yönelik kullanım örnekleri görseller ile birlikte detaylandırılmıştır.

Uygulama kapsamında ayarlar bölümünde API bağlantı durumları ve modellerin kullanılabilirliği izlenebilmektedir. Yine ayarlar bölümünde çevrimiçi ya da çevrimdışı sınıflandırma ayarı, API ve URL değişiklikleri, modellerin sıfırlanması ve yerel veri tabanının temizliği gibi işlemler yapılabilmektedir. Uygulama arayüzünden çevrimiçi model yükleme, sunucuya yerel veri tabanını gönderme, tahmin edilen fotoğrafları sucunuya gönderme, modelin tekrar eğitilmesi ve eğitimin durdurulması gibi çevrimiçi işlemler tetiklenebilmektedir. Yerel ve çevrimiçi rapor çıktılarıyla birlikte yerel tahminlemesi yapılan sınıflandırma sonuçlarına anasayfa üzerinden erişim sağlanabilmektedir. Yine anasayfa üzerinden hangi yaprak türünde sınıflandırma yapılabileceğinin seçimi de ayarlanabilmektedir. Bu kapsamda uygulama üzerinden açılan anasayfaya, yapılandırılabilen ayarlar bölümüne ve çevrimiçine yönelik kullanım özelliklerine Şekil 2.9’da yer verilmiştir.



Şekil 2.9. Mobil uygulamaya ait kullanıcı arayüz ekranlarına ait ekran görüntüleri.

Canlı kamera ile tahmin seçeneği seçildikten sonra açılan canlı kamera ekranında bulunan yeşil tarama butonuyla ilgili görsel taraması yapılarak sonuçlar kullanıcıya gösterilmektedir. Sınıflandırma tamamlandıktan sonra açılan pencere üzerinde taranan yaprak görselinin tahmin sonuçları PT ve ONNX modelleriyle birlikte sonuçlandırılmaktadır. Yaprağın her iki modele göre de sınıflandırma sonucu ekranda gösterilmektedir. Bununla birlikte yaprağın ROI içerisinde kalan kısımlarında hastalıklı bölgeler kırmızı renk ile işaretlenerek kullanıcı bilgilendirilmektedir. Yine sınıflandırması yapılan yaprağın gerçekte hastalıklı mı sağlıklı mı olduğunun geribildirimi de kullanıcıya seçenek olarak sunulmaktadır. Gelen geri bildirim ile birlikte yaprağın tekrar versiyonlama işlemlerinde ve analiz aşamalarında değerlendirilebilmesi için yerel veri tabanında kayıt işlemleri yapılabilmektedir. Yerel yapılan kayıtlar çevrimiçi veri tabanına ve sunucu üzerindeki dosyalara API hizmetleri kullanılarak aktarılabilmektedir. Kamera üzerinden canlı görsel tahminine yönelik kullanım biçimi ve örnek sınıflandırma görselleri Şekil 2.10'da gösterilmektedir.

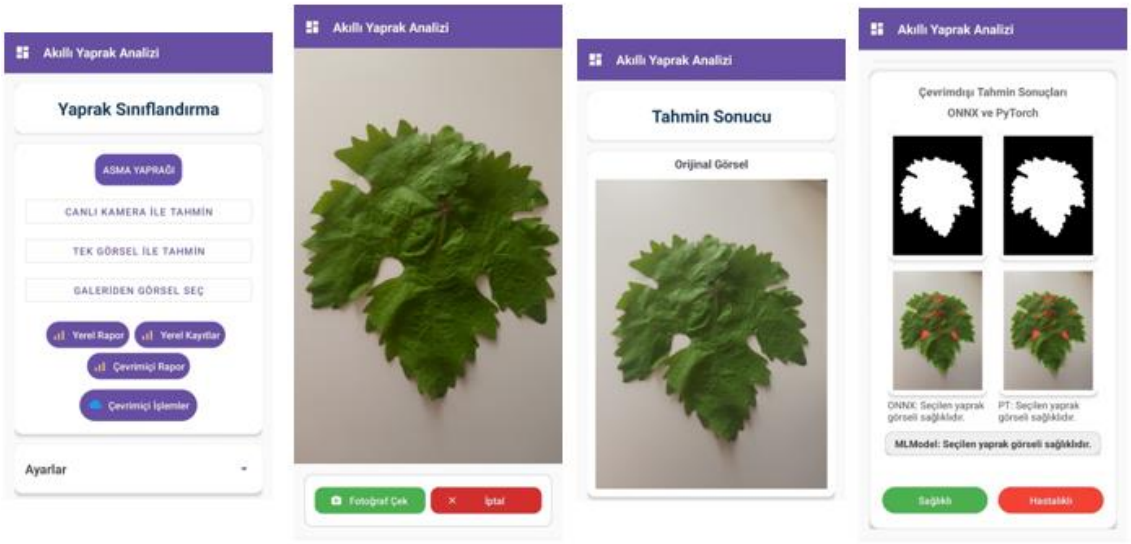


Şekil 2.10. Canlı kamera kullanarak yapılmış tahmin örneği.

Benzer bir kullanım şekliyle kamera üzerinden tek fotoğraf çekilerek tahmin sonucu ekranına yönlendirilen bir görselin de sınıflandırma işlemi yapılabilmektedir. Tek görsel ile tahmin seçeneği seçildikten sonra açılan canlı kamera uygulamasında yeşil fotoğraf çek butonuna tıklanmaktadır. Yakalanan görsel için sınıflandırma sonuçları tahmin sonuçları ekranında kullanıcıya gösterilmektedir. Sınıflandırma tamamlandıktan sonra tahmin sonuçları ekranında çevrimdışı ve ya çevrimiçi yöntemleriyle yaprak görselinin tahmin sonuçları PT ve ONNX modelleri kullanılarak değerlendirilebilmektedir. Bununla birlikte çevrimiçi olarak eğer tahmin ayarları yapılandırıldıysa ML.NET model sonuçları da kullanıcıya gösterilmektedir. PT ve ONNX modelleri için de yine çevrimiçi tahmin yapabilmesi için çevrimiçi ayarlarının yapılandırılması gerekmektedir. Yaprağın üç modele göre de sınıflandırma sonuçları ekranda gösterilebilmektedir.

Bununla birlikte yaprağın ROI içerisinde kalan kısımlarında hastalıklı bölgeler kırmızı renk ile işaretlenerek kullanıcıya bilgi verilmektedir. Yine kullanıcıya sınıflandırması yapılan yaprağın gerçekte hastalıklı mı sağlıklı mı olduğunun geribildirimi de kullanıcıya sunulmaktadır. Gelen geri bildirim ile birlikte yaprağın tekrar versiyonlama işlemlerinde ve analiz aşamalarında değerlendirilebilmesi için yerel veri tabanında kayıt işlemleri yapılabilmektedir. Yerel yapılan kayıtlar çevrimiçi veri tabanına ve sunucu üzerindeki

dosyalara API hizmetleri kullanılarak aktarılabilmektedir. Canlı kamera ile tahmin işlemlerine ek olarak tek kamera üzerinde ML.NET model sınıflandırmasının sonuçları da gösterilebilmektedir. Bu yöntem ile birlikte üç modelin sınıflandırma sonuçları kıyaslanabilmektedir. Galeriden görsel seç yöntemiyle birlikte cihazın hafızasında olan bir yaprak görseli kullanılarak, kameradan tek görsel ile tahmin yapma adımlarında kullanılan yöntemlerin aynı şekilde uygulanmasıyla sınıflandırma işlemi yapılabilmektedir. Kamera üzerinden fotoğraf çek işlemi ile tek görsel üzerinden tahmin yapılabilen kullanım biçimi ve örnek sınıflandırma Şekil 2.11'de gösterilmiştir.

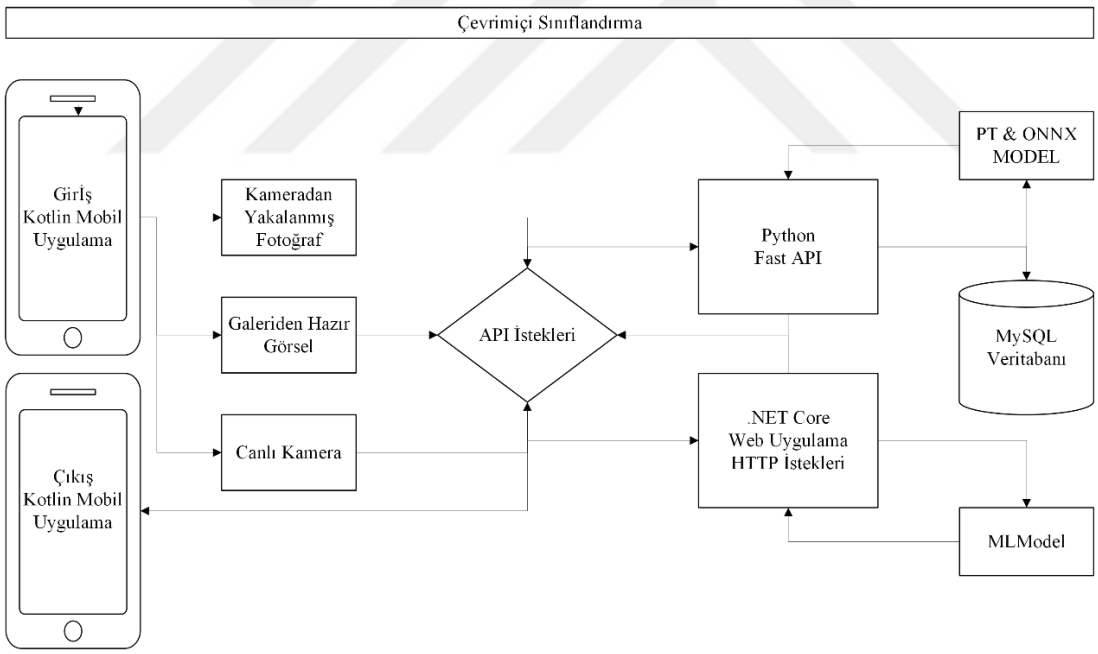


Şekil 2.11. Tek görsel ile kamera kullanarak yapılmış tahmin örneği.

2.9.1 Çevrimiçi Çalışma Modeli

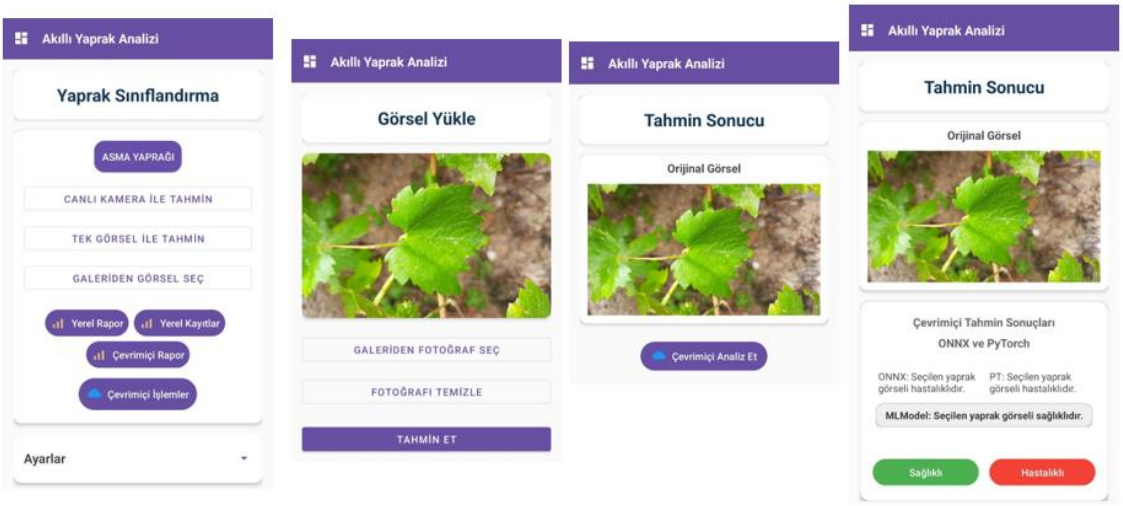
Bu bölümde önceki bölümlerde detaylı bilgisi verilen mobil uygulama üzerinden yine daha önce detayları verilen sınıflandırma modelleri kapsamında hazırlanmış olan PT, ONNX ve ML.NET modelleri kullanılarak çevrimiçi sınıflandırma modeline yönelik bilgi verilmektedir. PT, ONNX çıktıları ile çevrimdışı tahmin de yapılabilmektedir. Her iki sınıflandırma yöntemiyle de yerel ve çevrimiçi olarak sınıflandırma sonuçları veri tabanında tutulabilmektedir. Uygulamada yerel olarak room veri tabanı üzerinde tutulmaktadır. Yerel veri tabanında tutulan tahmin sonuçları MySQL tabanlı veri tabanına mobil uygulama vasıtasıyla API hizmetleriyle iletilebilmektedir. Ek olarak Python ve .Net web tabanlı yapılmış uygulamalar da bu uygulamalar üzerinden yapılan sınıflandırma sonuçlarını MySQL veri tabanına çevrimiçi olarak kayıt gönderilmektedir.

Bu doğrultuda tahmin edilmiş yaprakların maskeleri ve orijinal görüntüleri yerel ve çevrimiçi olarak saklanabilmektedir. API bağlantıları ile çevrimiçi sunuculara aktarım yapılabilmektedir. Çalışma kapsamında nihai hedeflerimizden biri de modelin tekrar tekrar eğitilebilir ve güncelliğini korumasını sağlamaktır. Bu çerçevede mobil uygulama üzerinden tahminleri yapılmış resimleri, oluşturduğumuz veri setine dâhil ederek tüm modeli tekrar eğitebilir şekilde tasarlamak hedeflenmiştir. Çevrimiçi çalışma modelinde mobil cihaz üzerinden üç farklı yöntem ile görsel sınıflandırma işlemine dâhil edilebilmektedir. Kameradan canlı resim yakalayarak, tek görsel ile canlı kameradan fotoğraf çekilerek ya da galeriden hazır bir görsel seçilerek API hizmetleriyle Web API ve ya Fast API tarafında işlenebilmektedir. Çevrimiçi analiz et butonuyla bu işlemlerin çalışma metotları tetiklenebilmektedir. Alınan çıktı sonuçları çevrimiçi veri tabanına kaydedilerek tekrar API hizmetiyle mobil uygulamaya çıktı olarak cevap gönderilmektedir. Bu kapsamda çevrimiçi çalışma modelinin görselleştirilmiş akış şemasına Şekil 2.12'de yer verilmiştir.



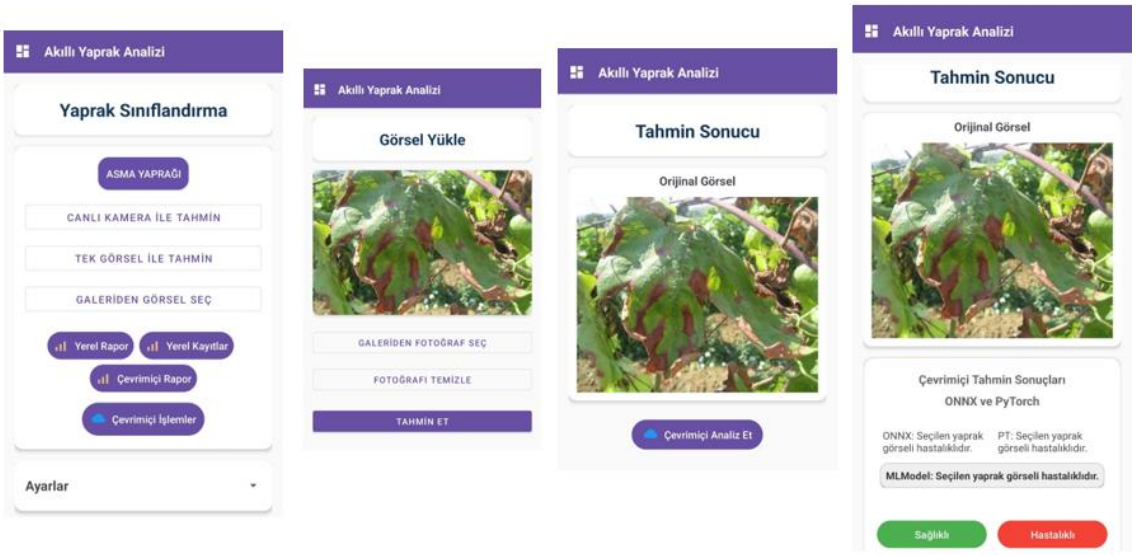
Şekil 2.12. Çevrimiçi sınıflandırma akış şeması.

Bu doğrultuda, çevrimiçi sınıflandırma örneğinin yapıldığı sağlıklı yaprak ile mobil uygulama üzerinde kullanım şekline yönelik arayüz örneklerine Şekil 2.13'te yer verilmiştir.



Şekil 2.13. Sağlıklı yaprak ile çevrimiçi sınıflandırma uygulama kullanım örneği.

Aynı şekilde çevrimiçi sınıflandırma için hastalıklı yaprak görseliyle yapılmış bir örneğine de Şekil 2.14'te yer verilmiştir.

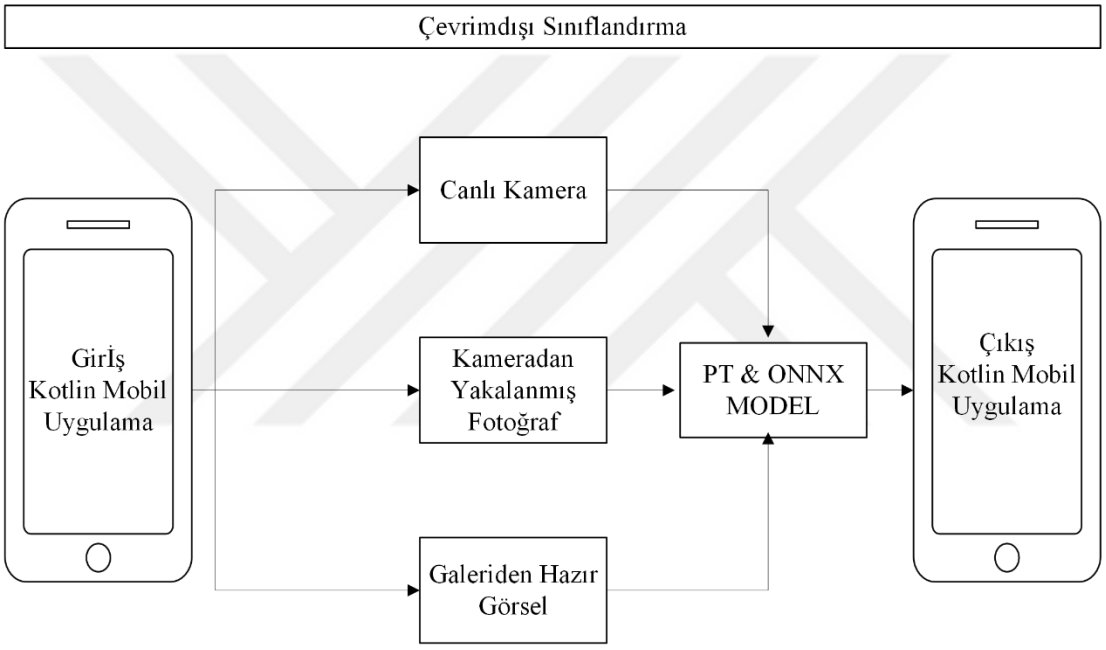


Şekil 2.14. Hastalıklı yaprak ile çevrimiçi sınıflandırma uygulama kullanım örneği.

2.9.2 Çevrimdışı Çalışma Modeli

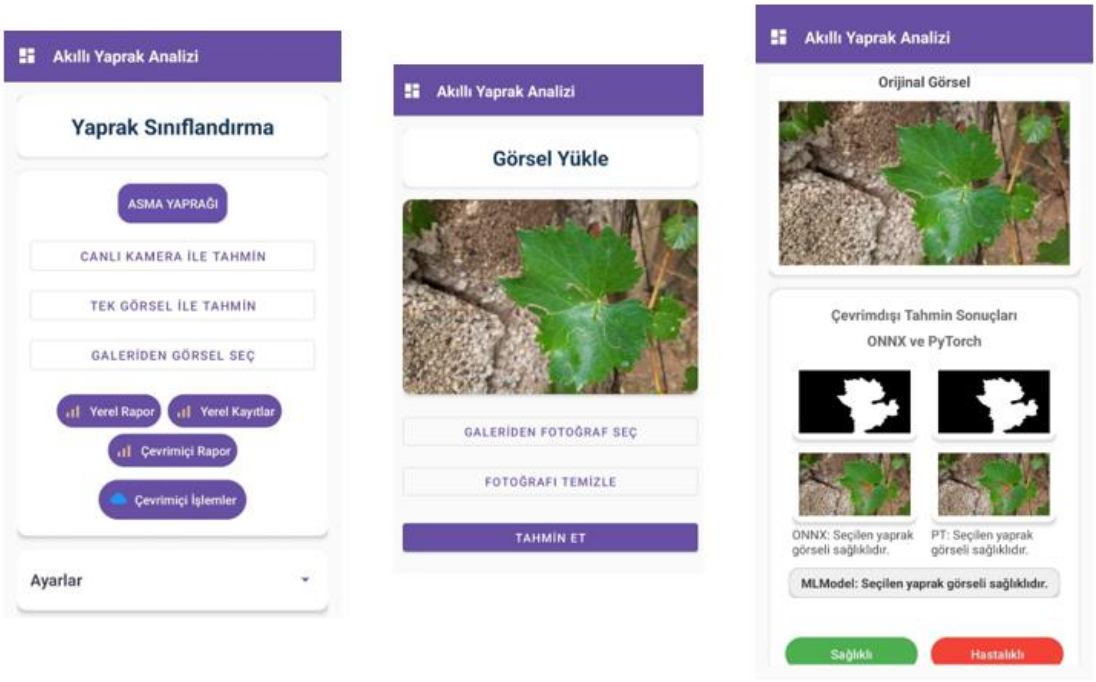
Bu bölümde daha önceki bölümlerde detaylı bilgisi paylaşılan mobil uygulama üzerinden yine daha önce detayları verilen sınıflandırma modelleri kapsamında hazırlanmış olan PT ve ONNX modelleri kullanılarak çevrimdışı sınıflandırma modeline yönelik bilgi verilmektedir. PT ve ONNX model çıktıları ile çevrimdışı sınıflandırma

yapılabilmektedir. Çevrimdışı sınıflandırma yöntemi kullanılarak çekilen görselin sınıflandırma sonuçları veri tabanında tutulmaktadır. Uygulamada yerel olarak room veri tabanı üzerinde kaydedilmektedir. İnternet erişiminin sağlanması durumunda, yerel veri tabanında tutulan çevrimdışı kayıtların, yerel veri tabanından çevrimiçi MySQL veri tabanına aktarılması sağlanabilmektedir. Bölüm 2.9.1'de detaylandırdığımız gibi çevrimiçi veri tabanında tüm kayıtların tutulabilmesi mümkündür. Bu doğrultuda tahmin edilmiş çevrimdışı yaprakların HSV maskeleri ve orijinal görüntüleri API bağlantıları ile çevrimiçi sunuculara aktarımı sağlanabilmektedir. Çevrimdışı çalışma modelinin görselleştirilmiş akış şemasına Şekil 2.15'te yer verilmektedir.



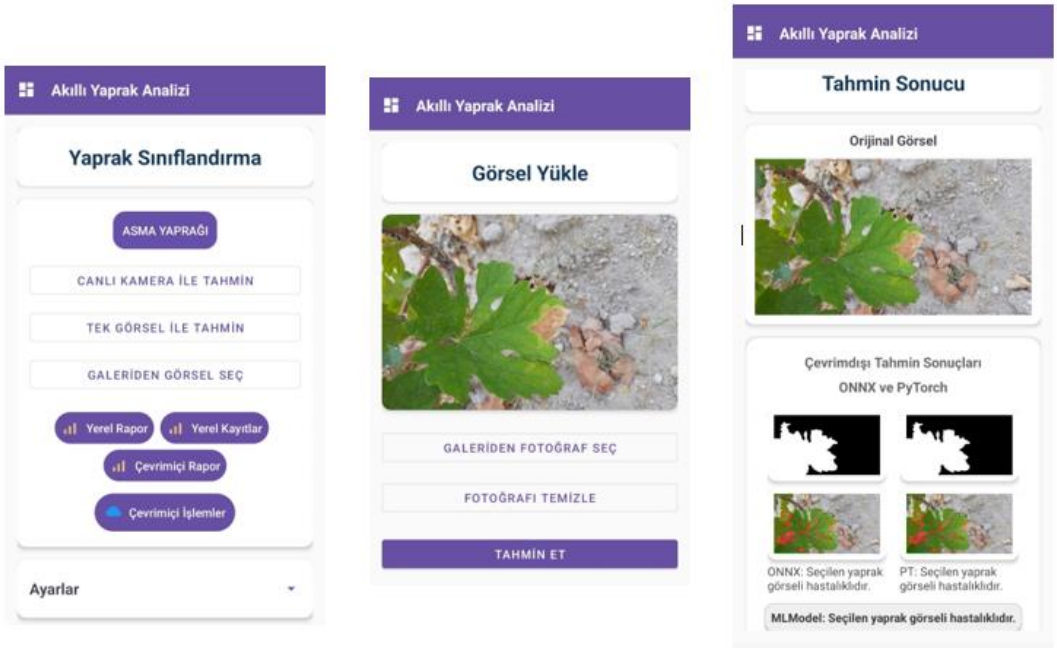
Şekil 2.15. Çevrimdışı sınıflandırma akış şeması.

Bu doğrultuda, çevrimdışı sınıflandırma kapsamında sağlıklı yaprak ile mobil uygulama üzerinde kullanım şekline yönelik arayüz örneklerine Şekil 2.16'da yer verilmiştir.



Şekil 2.16. Sağlıklı yaprak ile çevrimdışı sınıflandırma uygulama kullanım örneği.

Aynı şekilde çevrimdışı sınıflandırma için hastalıklı yaprak görseliyle yapılmış bir örneğine de Şekil 2.17'de yer verilmiştir.



Şekil 2.17. Hastalıklı yaprak ile çevrimdışı sınıflandırma uygulama kullanım örneği.

2.10 Yeniden Uygulanabilir Eğitim ve Sürümleme

Bu bölümde sistemin yeniden eğitilebilirliği ve sürümlenmesine yönelik bilgiler verilmektedir. Önceki bölümlerde detayları paylaşılan çalışmakta olan çevrimiçi ve çevrimdışı modellerin sonucunda elde edilen sınıflandırılması yapılmış yaprak görsellerinin son kullanıcının geridönüşüne göre dosyalanmış formatları, PT, ONNX ve ML.NET model eğitimleri yapılan sunuculara tekrar iletilebilmektedir. PT ve ONNX modelleri için kullanıcı tarafından sağlıklı ve ya hastalıklı onaylanmış görseller, model eğitiminde kullanılan 465 adet asma yaprağı görsellerinin bulunduğu orijinal ve maskelenmiş dosya yollarına eklenmesi API hizmetleriyle sağlanabilmektedir. ML.NET model için ise yine kullanıcı tarafından sağlıklı ve ya hastalıklı onaylanmış görseller, model eğitiminde kullanılan 3122 adet asma yaprağı görsellerinin bulunduğu ilgili dosya yollarına eklenmesi yapılabilmektedir.

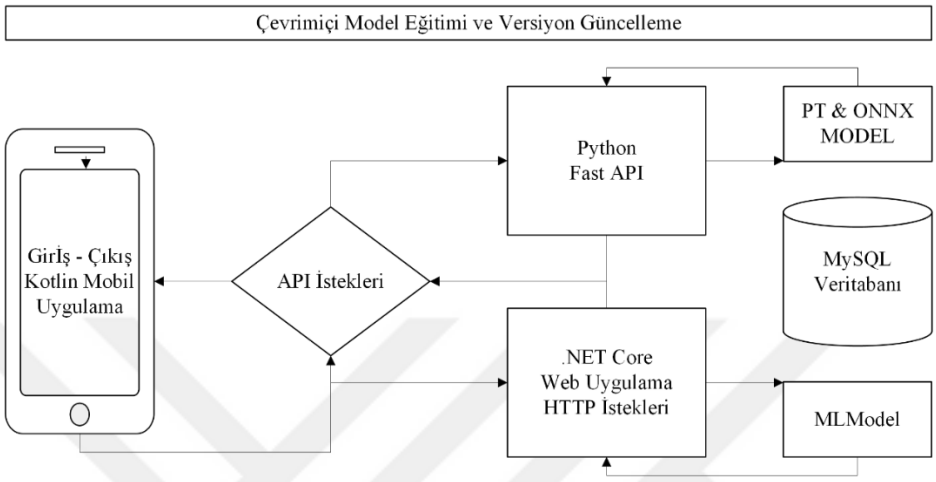
Çevrimiçi model eğitiminde Web API ve Fast API üzerinde tanımlı URL'ler kullanılarak API istekleriyle ilgili metotlar tetiklenerek model eğitim ve versiyonlama işlemleri yapılabilmektedir. Bununla birlikte tekrar sürümlenen modeller mobil uygulamada yine API hizmetleriyle güncellenebilmektedir. Eğitim sürecinin mobil uygulama tarafından başlatılmasıyla ilgili sunucularda eğitim süreçleri yapılabilmektedir. Eğitim süreçlerinin tamamlanmasının ardından çevrimiçi veri tabanına yeni sürüm kaydedilmektedir.

Kullanıcı mobil uygulama üzerinden yeniden eğitim işlemini başlatabildiği gibi, mevcutta başlamış bir eğitimin durması için de tekrar sunucuya istek gönderebilmektedir. Bu durumda mevcut sürümdeki modeller korunarak sadece eğitim aşaması sonlandırılmaktadır. Tamamlanmamış bir eğitim ile üretilen bir modelin eski sürümü güncelleme ihtimali engellenmiştir. Mobil uygulama üzerinden tetiklenebilen bu süreçler analiz edilerek çevrimiçi bir şekilde manuel olarak alternatif olarak tetiklenebilmektedir. Hatalı görsellerin ya da yanlış bir geridönüş yapılmış görsellerin çıkarılmasıyla daha manuel olarak eğitim yinelenebilmektedir.

Mobil uygulama üzerinden kullanıcı çevrimdışı kullanılan eski modelleri silebilmektedir. Bunun beraberinde çevrimiçi olarak yeni sürümlerin indirilip çevrimdışı kullanılabilir olması sağlanmıştır. Eğer kullanıcı eski modeli silme işlemi yapmamışsa ve yeni modelleri indirme işlemi yapmışsa mevcutta yeni sürüm varsa bu modelleri indirecek bir

mekanizma tasarlanmıştır. Yeni sürüm bulamadığı durumda ise eski sürümler üzerinden uygulamanın çalışmabilmesi sağlanmıştır.

Şekil 2.18'de tekrar eğitim sürecine ilişkin ve sürümleme adımlarının sonrasında mobil uygulamaya dâhil edilme süreçlerine ait detaya yer verilmiştir.



Şekil 2.18. Çevrimiçi model eğitimi ve sürüm güncelleme akış şeması.

3. BULGULAR

Bu bölümde veri seti önişleme süreçlerinin uygulandığı, model eğitiminin uygulandığı, model eğitim süreçlerinin analizinin ve iyileştirmelerinin yapıldığı, bununla birlikte modelin kullanılarak sınıflandırma işlemlerinin yapıldığı deney ortamlarından 3.1 numaralı bölümde bahsedilmektedir. Özgün deneysel veri seti kullanılarak yapılan asma yaprağı deney sonuçlarından 3.2 numaralı bölümde ve hazır veri seti biber yaprağı deney sonuçlarından 3.3 numaralı bölümde bahsedilmektedir. Her iki bölümde de çalışmada uygulanmış sistemle birlikte elde edilen modelin sınıflandırma sonuçlarıyla birlikte başarı metriklerininde yer aldığı nicel ve nitel verilere yer verilmektedir.

3.1. Deney Ortamları

Bu bölümde veri setinin elde edilmesi süreçlerinde, önişleme süreçlerinde ve sınıflandırma kapsamında kullanılan cihazlar hakkında bilgi verilmektedir. Çalışma kapsamında kullanılmak üzere veri seti özgün bir şekilde hazırlanmıştır. Bu kapsamda birden fazla mobil cihaz tarafından asma yaprağı görselleri toplanmıştır. Kullanılan mobil cihazlardan Android OS özelliğine sahip Samsung M31 ve Samsung J7 Prime cihazlarla birlikte iOS OS özelliğine sahip olan iPhone 7 cihaza Tablo 3.1'de yer verilmiştir. Tablo 3.1'de veri seti hazırlamada ve uygulamanın kullanıldığı sınıflandırma işlemlerinde kullanılan cihazlar hakkında bazı teknik özelliklere yer verilmiştir. Cihazların model bilgileriyle birlikte OS ve Merkezi İşlem Birimi (CPU) bilgileri listelenmektedir. Bununla birlikte Rastgele Erişimli Bellek (RAM) kapasiteleri Gigabayt (GB) türünden ve cihazların arka kameralarının Megapiksel (MP) bilgileri tabloda kıyaslanabilmektedir.

Tablo 3.1. Uygulamada kullanılan mobil cihazlara ait teknik özellikler.

Mobil Cihaz Model	OS	CPU	RAM	Kamera
Samsung Galaxy M31	Android 12	Samsung Exynos 9611 8-core 2.3 GHz	6 GB	64 MP
Galaxy J7 Prime	Android 8.1.0	Samsung Exynos 7870 8-core 1.6 GHz	3 GB	13 MP
iPhone 7	iOS 15.8.3	Apple A10 Fusion 2.3 GHz	2 GB	12 MP

Çalışmanın devamında ön işleme süreçlerinde Tablo 3.2'de bulunan Mac OS özelliğine sahip bilgisayar kullanılmıştır. Görsellerin arka planlarının temizlenme süreçleri, veri

çoğaltma işlemleri ve gruplanması aşaması bu cihaz kullanılarak yapılmıştır. Tablo 3.2'de veri seti önişlem süreçlerinde, model eğitiminde, uygulama hazırlık aşamasında, sınıflandırma işlemlerinde ve sunucu hizmetlerinde kullanılan cihazlar hakkında bazı teknik özelliklere yer verilmiştir. Bilgisayarlara ait cihaz modelleri, OS, CPU ve RAM bilgileri tablo üzerinde kıyaslanabilmektedir.

Tablo 3.2. Uygulamada kullanılan bilgisayarlara ait teknik özellikler.

Bilgisayar Model	**OS**	**CPU**	**RAM**
Asus X550VC	Windows 10	Intel Core i5-3230M 4-core 2.60 GHz	12 GB
MacBook Air M2	macOS	Apple M2 8-core 2.42 GHz	16 GB

Hazırlanan veri seti ile birlikte PT ve ONNX model eğitim süreçleri yine Mac OS özelliğine sahip Macbook cihaz kullanılarak yapılmıştır. ML.NET model eğitim süreci ve modelin çıktısı Windows OS özelliğine sahip Asus bilgisayar kullanılarak yapılmıştır. Çevrimdışı ortamda testlerin yapılması aşamasında her iki cihazda sunucu hizmeti sağlaması amacıyla kullanılmıştır. Aynı zamanda çevrimiçi ortam testlerinin de yapılması için yine her iki cihazdan yararlanılmıştır. Görsellerin yerel dosyalardan okunarak tahminlemesinin yapılması, tez çalışmasında sunulan çevrimiçi ve çevrimdışı modellerin Android tabanlı mobil uygulama çerçevesinde gerekli testleri ve sınıflandırma işlemleri Samsung M31 cihazı kullanılarak yapılmıştır. Yeniden versiyonlama işlemleri de aynı mobil cihazda kullanılarak kontrolleri tamamlanmıştır. Çalışma kapsamında elde edilen verilere ait detaylı analiz sonuçları asma yaprağı deneyleri 3.2 numaralı bölümde ve biber yaprağı deneyleri 3.3 numaralı bölümde detaylıca sunulmaktadır.

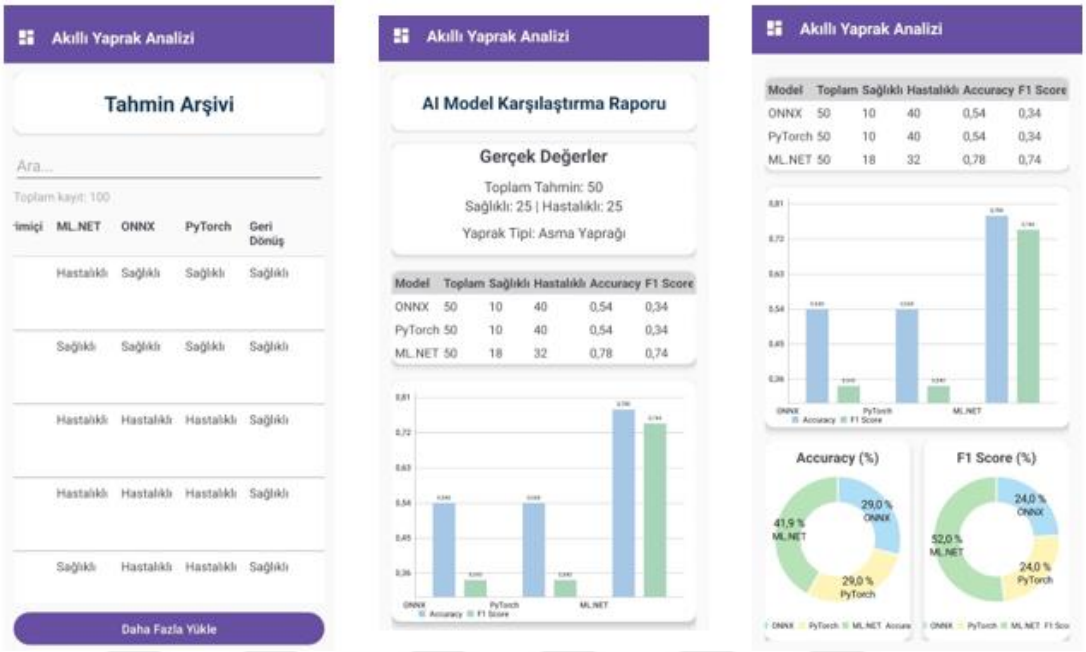
3.2 Asma Yaprağı Deneyleri

Uygulama çalışacak şekilde mobil uygulama Samsung M31 cihaza kurularak testleri yapılmıştır. ML.NET model için hazırlanan ASP .Net Core Web Uygulaması Windows OS özelliğine sahip bir cihazda sunucu görevi görecek şekilde çalıştırılmıştır. Aynı şekilde Mac OS özelliğine sahip bir cihaz da ONNX ve PT modellerinin tekrar eğitim adımında çalışabilmesi için Fast API ortamı sunucu görevi görecek şekilde çalıştırılmıştır. Tüm ortam kurulumlarının çalışma durumu kontrol edilerek, mobil uygulama üzerinde canlı kamera, tekli görsel yakalama ve galeriden fotoğraf çekme yöntemleriyle çevrimdışı sınıflandırma yapılması sağlanmıştır. Aynı şekilde çevrimdışı sınıflandırma yaparken ML.NET model içinde çevrimiçi tahmin yapması sağlanmıştır.

Aynı görsel için çalışma kapsamında yapılan sınıflandırma sonuçlarında yer alan ekran çıktılarıyla birlikte tüm modeller için sağlıklı ve hastalıklı çıktılar ortak bir tahmin sonucu verdiği gözlemlenmiştir.

Yerel veri tabanına kaydedilen kayıtlar yardımıyla maskeli ve orijinal görseller sunuculara gönderilebilmektedir. Gönderilen görsellerle birlikte yeniden eğitim başlatılıp, güncellenen sürümlerle birlikte modeller mobil uygulamaya tekrar indirilebilmiştir. Bu sayede güncel sürüm ile çalışabilirlik kontrol edilmiştir. Sınıflandırılması yapılan sonuçlar yine çalışma kapsamında yapılan sınıflandırma sonuçlarında yer alan ekran çıktılarında, HSV maskelerinin ve ROI bölgelerinin işaretlenmiş görsellerin başarılı bir şekilde sonuç verdiği gözlemlenmiştir. Hastalıklı bölgelerin şeffaf kırmızı ile renklendirilmesi başarıyla tamamlanmıştır. Çalışmada sunulan sistemin başarılı bir şekilde çalıştığı gözlemlenmiştir.

Çevimdışı tahmin sonuçlarının kaydedildiği veri tabanı üzerinden tahmin arşivi ekranından daha önce yapılmış sınıflandırmaların tahmin sonuçları her modeli için yan yana kıyaslanabilecek şekilde sunulmaktadır. Aynı zamanda yine mobil uygulama dâhilinde kullanılan yerel veri tabanı üzerinden kaydedilen veriler kullanılarak, karşılaştırmalı rapor ekranından daha önce yapılmış tüm tahminlerin nitel ve nicel verilerine erişilebilmektedir. Yaprakların sınıflandırma işlemleri sonrasında toplam tahmin sayısıyla birlikte kaç yaprağın sağlıklı tahmin edildiği, kaç yaprağın sağlıksız tahmin edildiği, kullanıcının geri dönüşü üzerine gerçekte kaç tanesinin sağlıklı ya da hastalıklı yaprak olduğu gibi bilgilere ulaşılabilmektedir. Bu değerler üzerinden her model için bu verilerin analizi yapılabilmektedir. Üç model için de grafikler oluşturularak kolay analiz edilebilir ekranlar kullanıcıya sağlanmaktadır. En yüksek doğruluk oranıyla birlikte üç model kendi arasında yüzdelik dilimlerle birlikte sıralaması yapılabilmektedir. Bu doğrultuda çevrimdışı çalışma modellerinin kullanabileceği rapor ekranlarıyla mobil uygulama kapsamında yerel veri tabanı üzerinden hesaplanan doğruluk ve F1 Score değerlerine ait karşılaştırma örnekleri, tahmin arşivi ve karşılaştırmalı rapor ekranları Şekil 3.1’de gösterilmektedir.



Şekil 3.1. Asma yaprağına ait mobil uygulama üzerindeki örnek rapor görselleri.

Ön hazırlık işlemleri yapılmadan gerçek veriler korunarak işlenmemiş asma yapraklarının model eğitimi sonucunda elde edilen başarı metriklerine Tablo 3.3'te yer verilmiştir.

Tablo 3.3. Asma yapraklarına ait işlenmemiş verilerle elde edilen başarı metrikleri.

Veri Seti	Doğruluk	F1 Score	Kesinlik	Duyarlılık
Asma Yaprağı	0.7594	0.7734	0.8475	0.7113

Tablo 3.4'te 50 görselden 25'i gerçekte sağlıklı, 25'i ise gerçekte sağlıksız olan asma yaprağı test veri setinin, sınıflandırma modelleri ile değerlendirme ölçütleriyle birlikte başarılı tahmin sayılarının karşılaştırmalı sonuç tablosuna yer verilmiştir. ML.NET modelinin diğer modellere kıyasla daha başarılı bir sonuç verdiği gözlemlenmiştir.

Tablo 3.4. Asma yaprağına ait örnek görsellerle sonuç kıyaslama tablosu.

Model	Doğruluk	F1 Score	Sağlıklı Yaprak Sayısı	Hastalıklı Yaprak Sayısı
ONNX	0.54	0.34	10	40
PT	0.54	0.34	10	40
ML.NET	0.78	0.74	18	32

Tablo 3.5'te 50 görsel ile elde edilen her model için toplam doğruluk değerlendirme ölçütüne göre başarı sıralammasını gösteren sonuçlara yer verilmiştir. En başarılı modelin ML.NET olduğu gözlemlenmiştir.

Tablo 3.5. İşlenmemiş asma yapraklarıyla elde edilen doğruluk başarı sıralaması.

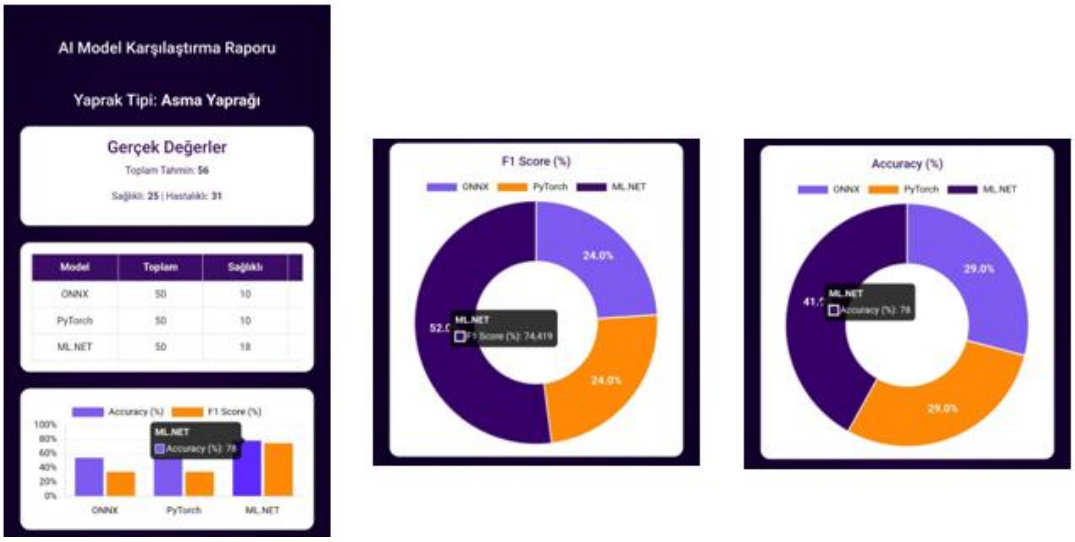
Model	Doğruluk	Kullanılan Yaprak Sayısı
ML.NET	41.9	50
PT	29.05	50
ONNX	29.05	50

Tablo 3.6'da 50 görsel ile elde edilen her model için toplam F1 Score değerlendirme ölçütüne göre başarı sıralammasını gösteren sonuçlara yer verilmiştir. Yine ML.NET modelinin F1 Score sıralamasında da diğer modellere göre daha iyi sonuç verdiği tespit edilmiştir.

Tablo 3.6. İşlenmemiş asma yapraklarıyla elde edilen F1 Score başarı sıralaması.

Model	F1 Score	Kullanılan Yaprak Sayısı
ML.NET	52.0	50
PT	24.0	50
ONNX	24.0	50

Aynı zamanda çevrimiçi rapor oluşturarak MySQL veri tabanı üzerinden uygulama üzerinde yapılmış tüm tahminlerin sonuçlarına ait raporlar da gösterilmektedir. Şekil 3.1'de detaylandırılmış rapor ekranlarıyla birebir aynı mantıkta çalışacak şekilde çevrimiçi veri tabanı üzerinden de son güncel veriler üzerinden raporlar sunulmaktadır. Bu sayede sistem üzerinde en güncel verilerle geçmiş bütün kayıtlar üzerinden kıyaslama raporlarıyla birlikte analiz edilebilmektedir. Yapılmış örneklem üzerinden elde edilen verilere göre çevrimiçi rapor örnekleri Şekil 3.2'de gösterilmektedir.



Şekil 3.2. Asma yaprağına ait çevrimiçi rapor örneği.

3.3 Biber Yaprağı Deneyleri

Hazırladığımız çalışma modeli kapsamında hedeflediğimiz diğer bir nokta ise farklı veri setleri ve hazırlanmış modeller ile geniş çaplı çalışabilir bir sistem gerçekleştirmektir. BU hedef doğrultusunda biber yapraklarının olduğu PlantVillage veri seti [4] ile daha önce yapılmış çalışmalar incelenmiştir. Bu doğrultuda benzer veri seti, DSC-TransNet mimarisini kullanarak hazırlanmış olan model dâhilinde ele alınmıştır. Bu çalışmada yapılan sistem dâhilinde %99,94 F1 Score ve %99,97 doğruluk başarı metrikleri elde edilmiştir [33].

Bu çalışmada yapmış olduğumuz modelin farklı bir veri seti ile nasıl davranacağını analiz etmek amacıyla arka planı temizlenmiş biber yapraklarının bulunduğu PlantVillage veri seti [4] ele alınmıştır. Bu veri setinde, özgün olarak hazırladığımız veri setinden farklı olarak arka planı tek renk olacak şekilde ayarlanmıştır. Bu kapsamda resimlerin boyutları da çalışma kapsamında haz ırlanan veri setinden düşüktür. Model eğitim noktasında bu gibi noktaların eğitim süresi ve başarı oranları noktasında artısı olacağı söylenebilir. Ön hazırlık aşaması yapılmış biber yapraklarının model eğitimi sonucunda elde edilen başarı metriklerine Tablo 3.7'de yer verilmiştir.

Tablo 3.7. Biber yapraklarına ait işlenmiş verilerle elde edilen başarı metrikleri.

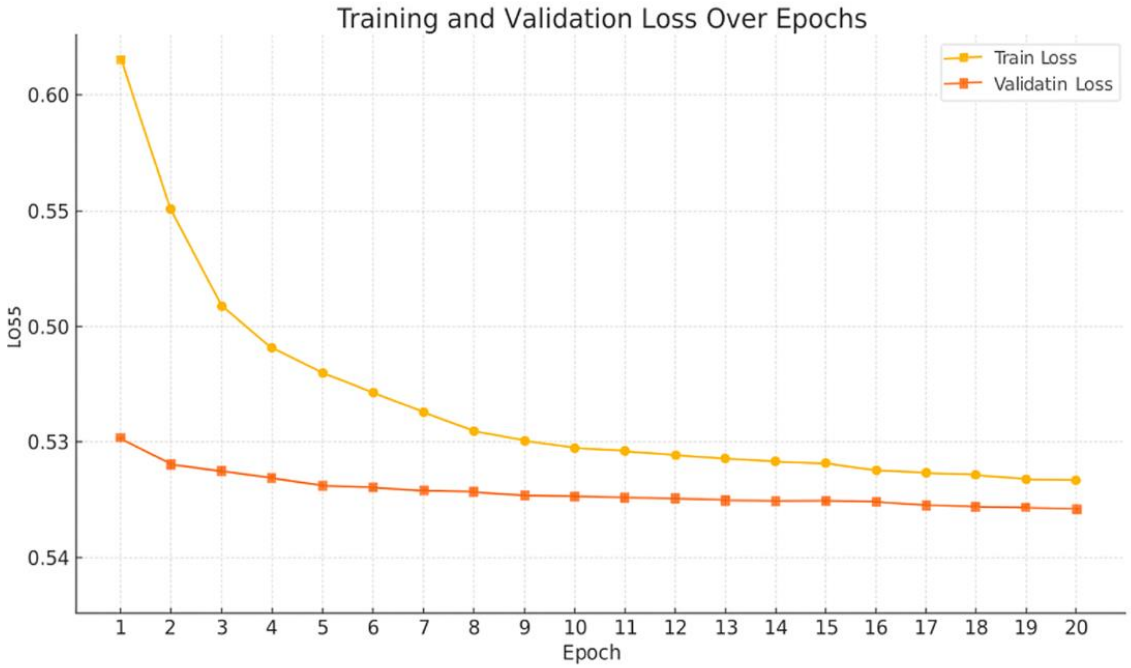
Veri Seti	Kesinlik	Duyarlılık	F1 Score	Doğruluk
Asma Yaprağı	0.8475	0.7113	0.7734	0.7594

Tablo 3.7'de eğitim sonucunda elde edilen başarı metrikleri göz önünde bulundurulduğunda arka planı temizlenmiş orjinal resimlerin doğrudan eğitim veri seti olarak verilmesi, başarı metriklerinin değerlerindeki artışı doğrudan etkilediği net olarak gözlemlenmektedir. Bütün başarı metriklerinde pozitif anlamda artış olduğu tespit edilebilmektedir. Asma yaprağı ve biber yaprağı kullanılarak model eğitimi sonucunda elde edilen başarı metrikleri Tablo 3.8'de karşılaştırılmıştır.

Tablo 3.8. Asma ve biber yaprağına ait metriklerin karşılaştırılması.

Veri Seti	Kesinlik	Duyarlılık	F1 Score	Doğruluk
Asma Yaprağı	0.8475	0.7113	0.7734	0.7594
Biber Yaprağı	0.9806	0.9594	0.9391	0.9647

Şekil 3.3'te gözlemlenen kayıp fonksiyonu grafiğinde belirli bir tekrardan sonraki eğitimlerde loss değerinin dalgalı şekilde sabit kaldığı görülmektedir.



Şekil 3.3. Biber yapraklarının model eğitimi kayıp fonksiyonunu gösteren grafik.

Bu grafikteki sonuçlar ONNX ve PT modellerinin hazırlanmasında elde edilmiştir. Eğitimi tamamlanmış model, hazırladığımız mobil ve web uygulamaya dinamik olarak çalışacak şekilde dâhil edilmiştir. Sisteme dâhil ettiğimiz ML.NET model için de eğitim aşaması başarıyla tamamlanmıştır. Benzer ekranlar kullanılarak farklı modellerin kullanımına olanak sağlayan bu çalışma modelinde başarılı bir sonuç elde edilmiştir. Şekil 3.4'te ve Şekil 3.5'te sınıflandırmaları yapılmış görseller için hastalıklı ve sağlıklı yaprağın doğru olarak analiz edildiği gözlemlenmektedir.

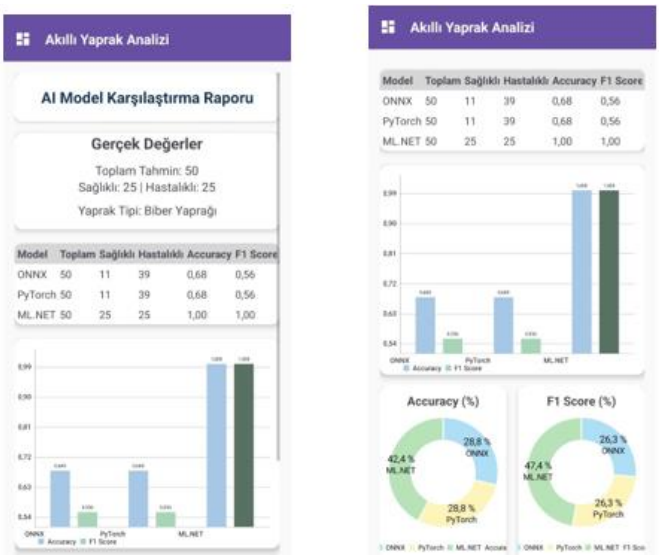


Şekil 3.4. Sağlıklı biber yaprağına ait sınıflandırma örneği.



Şekil 3.5. Hastalıklı biber yaprağına ait sınıflandırma örneği.

50 adet farklı biber yaprağı yapılacak değerlendirmeler için test verisi olarak hazırlanmıştır. Bu veri setinde seçilen biber yaparklarından 25 tanesi gerçekte sağlıklı diğer 25 tanesi gerçekte hastalıklı olarak sınıflandırılmıştır. F1 Score ve doğruluk hesaplamalarında bize gerekli olacak geri bildirimler, sınıflandırma sonrasında kullanıcıdan sağlıklı mı hastalıklı mı tercihlerinden birisi seçtirilerek alınabilmektedir. Yerel ve çevrimiçi veri tabanında saklanabilecek şekilde tahmin sonuçları yine sistemde tutulabilmektedir. Arka planı temizlenmiş yaprak verilerin kullanıldığı ML.NET modelinde sistemin %100 bir başarı elde ettiği gözlemlenmiştir. ML.NET tüm başarı metriklerinde ONNX ve PT modellerine göre daha iyi bir sonuçla çıktı sağlamıştır. Kullanılan test veri setine göre sınıflandırma sonuçlarında elde edilen sonuçlar grafik olarak Şekil 3.6'da görülmektedir.



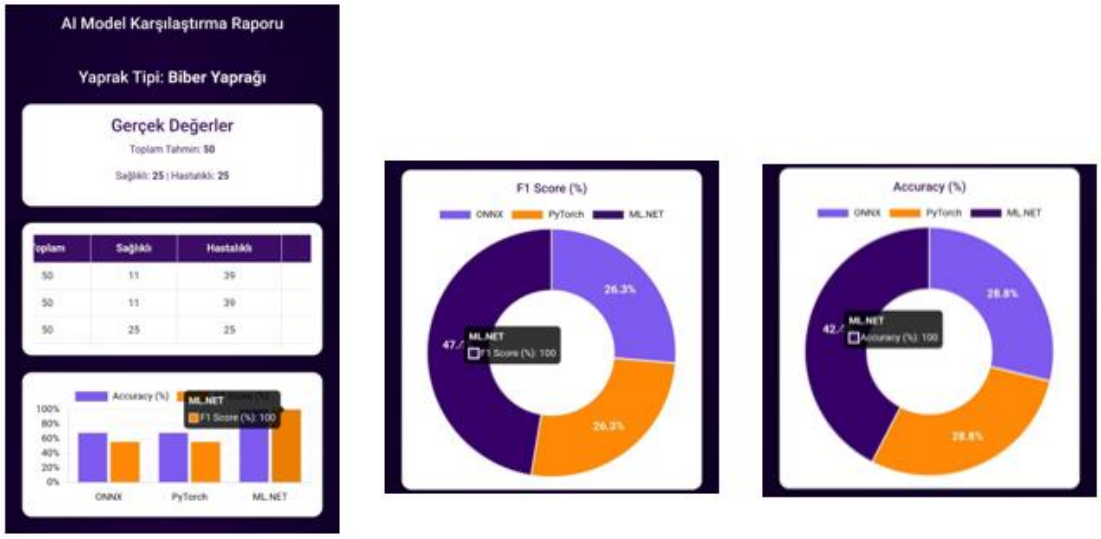
Şekil 3.6. Biber yaprağına ait mobil uygulama üzerindeki örnek rapor görselleri.

50 görselden oluşan 25'i gerçekte sağlıklı 25'i ise gerçekte sağlıksız olan biber yaprağı test veri setinin sınıflandırılması sonucunda elde edilen çıktılar incelendiğinde, ML.NET modelindeki hatasız sınıflandırma sonuçları diğer modellere kıyasla dikkat çekmektedir. Sınıflandırma modellerinin de yer aldığı elde edilen değerlendirme ölçütlerinin ve başarılı tahmin sayılarının bulunduğu karşılaştırmalı sonuç tablosuna Tablo 3.9'da yer verilmiştir.

Tablo 3.9. Biber yaprağına ait örnek görsellerle sonuç kıyaslama tablosu.

Model	Doğruluk	F1 Score	Sağlıklı Yaprak Sayısı	Hastalıklı Yaprak Sayısı
ONNX	0.68	0.56	11	39
PT	0.68	0.56	11	39
ML.NET	1.00	1.00	25	25

ONNX ve PT modelleriyle birlikte yine ML.NET modeli de kullanılarak biber yapraklarının performansı test edilmiştir. Hazır veri seti üzerinden model eğitimi yapılarak ML.NET model çıktısı elde edilmiştir. ML.NET model ile web uygulama üzerinden yapılan sınıflandırma işlemlerinde de modelin başarılı tahmin yaptığı gözlemlenmiştir. Bu kapsamda çevrimiçi çalışma modeli kapsamında hazırlamış olduğumuz web uygulama üzerinde bulunan rapor ekranında sonuçlar görülebilmektedir. ML.NET model, biber yaprakları ile de başarılı sınıflandırma performansı göstermiştir. Test veri seti ile elde edilen sonuçlar Şekil 3.7'de rapor ekranlarında gözlemlenmektedir.



Şekil 3.7. Biber yaprağına ait çevrimiçi rapor örneği.

Tablo 3.10'da 50 görsel ile elde edilen doğruluk değerlendirme ölçütlerine her model için ayrı ayrı yer verilmiştir. Başarı sıralamasını gösteren sonuçlara göre kıyaslama

yapılabilmektedir. Üç modelin kullanıldığı bu çalışma kapsamında en yüksek skor ML.NET ile elde edilmiştir. Doğruluk pasta grafiğinde %42.4 lük bir alana sahip olan ML.NET'in başarı kıyaslamasında ilk sırada olduğu gözlemlenmektedir.

Tablo 3.10. İşlenmemiş biber yapraklarıyla elde edilen doğruluk başarı sıralaması.

Model	Doğruluk	Kullanılan Yaprak Sayısı
ML.NET	42.4	50
PT	28.8	50
ONNX	28.8	50

Tablo 3.11'de 50 görsel ile elde edilen F1 Score değerlendirme ölçütlerine her model için ayrı ayrı yer verilmiştir. Başarı sıralamasını gösteren sonuçlara göre kıyaslama yapılabilmektedir. Yine F1 Score pasta grafiğinde %42.6'lık bir alana sahip olan ML.NET'in başarı kıyaslamasında ilk sırada olduğu gözlemlenmektedir.

Tablo 3.11. İşlenmemiş biber yapraklarıyla elde edilen F1 Score başarı sıralaması.

Model	F1 Score	Kullanılan Yaprak Sayısı
ML.NET	42.6	50
PT	26.3	50
ONNX	26.3	50

4. SONUÇLAR VE TARTIŞMA

Literatür kapsamında yapılmış çalışmalar dâhilinde yoğunlukla görsellerin orijinal görselleri temizlenerek sadece sınıflandırılması yapılacak haliyle eğitime dâhil edildiği gözlemlenmiştir. Bu çalışmada geliştirilmiş model ile kaynaklarda uygulama odaklı bulunan çalışmalara katkı vermesi amacı gözetilmiştir. Çalışmamızdaki en önemli noktalardan birisi gerçek zamanlı alınmış fotoğrafların kullanılmasıdır. Veri setindeki yapraklar görselleri, asma üzerinde doğal ortamında iken kaydedilmiştir. Fotoğraflar bağ ve bahçelerde bulunan asmalardan tek tek fotoğraflanmıştır. Aynı zamanda sistemin bir parçası olarak mobil uygulamanın yapılması hedefiyle iOS ve Android OS özelleriklere sahip farklı mobil cihazlarla elde edilmiştir. Araştırmalar sonucunda çalışmamızı kaynalarda bulunan diğer çalışmalardan farklı kılacak en önemli özelliklerden birisi veri setinin özgünlüğüdür. Birçok çalışmada veri seti hazırlanırken yoğunlukla yaprak verisinin düzgün bir zeminde fotoğraflanıp ardından segmentasyon süreçlerine tabi tutulduğu gözlemlenmiştir. Yapılan çalışmalardan farklı olarak model eğitimlerinde orijinal görseller ayrıştırılmadan, HSV ve ROI maskeleriyle birlikte eğitime ve değerlendirilme kısmına dâhil edilmesi sağlanmıştır. Farklı bir nokta olarak ele alınan bu kısım için sınıflandırma aşamasında sistemin güvenirliğine katkı sağlayacağı düşünülmektedir. Tarım üzerine yapılmış çalışmalarda genel olarak temizlenmiş görsellerle çalışmışmıştır. Bu doğrultuda uygulama kapsamında veri setinin orijinal görsellerle ve arka planı temizlenmeden eğitim aşamasında sürece dâhil edilmesi hazırlanan sistem için farklı bir yön olarak sunulmaktadır.

Android OS özelliğine sahip cihaz üzerinde Python tarafında üretilmiş ONNX ve PT modelleri ile çevrimiçi sınıflandırma yapılabilmektedir. Bununla birlikte Microsoft tabanlı ML.NET üzerinde çevrimiçi görsellerin sınıflandırması yapılabilmektedir. Aynı zamanda internet erişiminin olmadığı zamanlarda çevrimdışı çalışma yöntemiyle uygulama içerisinden ONNX ve PT modelleri kullanılarak sınıflandırma yapılabilmektedir. Veri setinin çalışma kapsamında özgün olması, orijinal resimlerin doğrudan eğitim aşamasında kullanılması modelin sınıflandırma sonuçlarında elde edilen başarı metriklerinde negatif yönde eğilimini artırdığı gözlemlenebilmektedir. Bununla birlikte eğitim ve test aşaması sonucunda elde edilen başarı metriklerine yönelik yaprak veri setlerinin eğitim aşamasındaki başarılı olma durumları kıyaslanmıştır. Elde edilen çıktılar sonucunda sistemimizin, asma yaprağı kapsamında hazırlamış olduğumuz renk

paketindeki HSV kodlarına göre analiz etmesi sebebiyle, PT ve ONNX çıktılarındaki başarı oranı yapmış olduğumuz örneklemde düşük kalmıştır. Test verileriyle arka planı temizlenmiş biber yaprakları için de yakın sonuçlar elde edilmiştir. Bu durumun tersi yönde eğitim aşamasında biber yaprağında 0.98 kesinlik değeri elde edilirken, asma yaprağında 0.84 kesinlik değeri elde edilmiştir. Tüm sınıflandırma modelleri üzerinde 50 adet özgün hazırlanan asma yaprağı fotoğrafı, hem çevrimiçi hem de çevrimdışı çalışma modelleri kapsamında test edilmiştir. Bu doğrultuda çalışma kapsamında hazırlanmış olan modeller kapsamında ML.NET modelin 0.78 doğruluk ve 0.74 F1 Score çıktısıyla en başarılı sonuç verdiği tespit edilmiştir. Bununla birlikte 50 adet biber yaprağı da hem çevrimiçi hem de çevrimdışı çalışma modellerinde test edilmiştir. Bu kapsamda çevrimiçi modelin ML.NET modeli kapsamında doğruluk ve F1 Score metriklerinin çıktısında %100 başarı sağladığı gözlemlenmiştir. ML.NET sınıflandırma sonuçlarının %100 başarı sağlaması mevcut sistemimizde kullandığımız veri setinin daha iyileştirilebilir olduğunu ortaya koymaktadır. Buradaki farkın fotoğraflanan gerçek verilerin herhangi bir görüntü işleme işlemine dâhil edilmeden doğrudan eğitim aşamasında kullanılması bu noktada belirgin farklardan birisi olmuştur. Biber veri setinde bulunan arka planı temizlenmiş yaprakların mevcut ML.NET modelin eğitiminde kullanılmasının elde edilen model ile birlikte analiz sonuçlarında hatasız sınıflandırma yapması, bu noktada sunulan sistemin arka planı temizlenmiş veri setlerinde başarılı çalışacağını bize göstermektedir. İşlenmemiş gerçek ortam verileri kullanılarak elde edilen model ile sınıflandırma yapıldığında ML.NET modelin başarılı olduğu yapılan örnemlemdeki rapor sonuçlarına göre gözlemlenmektedir. Fakat temizlenmiş arka planı bulunan bir veri setine göre geride kaldığı da tespit edilmektedir.

Elde edilen sonuçların diğer çalışmalardaki başarı oranlarına yakın olması doğrultusunda modelin olumlu noktada çalıştığı düşünülmektedir. Aynı zamanda birden fazla modelin aynı anda çevrimiçi ve çevrimdışı çalışma modelleriyle desteklenebiliyor olması, farklı modellerdeki benzer çıktıların alınmasıyla birlikte güvenirliğini artırdığı düşünülmektedir. Uygulamanın gerçek ortamda çekilmiş veri setinden oluşturulması, çevrimiçi ve çevrimdışı kullanılabilirliği ve tahmin edilen görsellerin gerçekte hangi sınıfta olduğunu son kullanıcıdan geri dönüş almasıyla uygulamadaki modelin güvenirliğinin artırılması hedeflenmiştir. Aynı zamanda son kullanıcı uygulamaya dâhil edilerek interaktif bir model tasarlanmıştır. Bu doğrultuda model ile birlikte kullanıcının

onayını da alarak, tahmin sırasında yapılan maskelenmiş ve orijinal fotoğrafların kullanıcının seçimi de göz önünde bulundurularak kaydedilmektedir. Bu verilerle birlikte mevcut verilerin de birleşimiyle tasarlanmış olan derin öğrenme tabanlı model tekrar daha fazla veriyle eğitilebilmektedir. Eğitilen model tekrar mobil cihaza dâhil edilerek sürüm odaklı bir uygulama geliştirilmiştir. Bu kapsamda uygulamanın mevcutta kullanılabilirliğinin yanı sıra güncellenebilir, gerçek zamanlı sınıflandırma yapabilmesi modeli olumlu yönde etkilediği ve çalışmamızda sunulan sistem için özgünlük kattığı düşünülmektedir.

Çalışmanın hedeflerinin arasında yer alan sistemin kendini güncelleyebilir olması ve sürdürülebilirliğinin sağlanması önemli noktalar arasında görülmektedir. Yeniden sürümleme yönteminin kullanılabilir olması ve sistemin güncel sürümlerle birlikte çalışabilir şekilde tasarlanmış olması modelin gelişimine katkı sağlayacak özellikler arasında yer aldığı düşünülmektedir. Kullanıcının sınıflandırılmış fotoğrafları gerçek sonuçlarıyla birlikte geri dönüş olarak kaydedebiliyor olması sistemin bu yanını güçlü tutmaktadır. Sisteme son kullanıcının geri dönüş yapması durumuyla dâhil olması, sistemin güvenirliğini düşürebilek negatif yönlerden birisi olarak düşünülmektedir. Bununla birlikte orijinal verilerin doğrudan kullanılması yine sistemin gelişmesini yavaşlatabilecek parametrelerden birisi olarak gözlemlenmektedir. Bu gibi negatif yönlerin iyileştirilmesi yönünde sistemin kendini güncelleyebilir olması, yalnızca modelin güncellenmesiyle uygulamanın farklı veri setleriyle de çalışabilme özelliğinin olması bu gibi durumların çözümü olarak düşünülmektedir. Bu doğrultuda yapılacak her düzenlemenin yeni sürümlerde desteklenebilir olması yapılacak diğer çalışmalar içinde farklı bir bakış açısı kazandıracağı söylenebilmektedir.

Bu bağlamda hemen hemen herkesin mobil cihazlara sahip olduğu, teknoloji çağının içerisinde olduğumuz bu süreçte çalışma sonucundaki ürünün hedef kitlesi son kullanıcı olan tarım üreticisi çiftçilerimizdir. Bu sistemle birlikte üreticilerin üzüm ve asma yapraklarını yüksek kalitede hasat edebilmelerine pozitif katkı sağlaması amaçlanmıştır. Uygulamanın doğrudan olarak gerçek zamanlı bir şekilde kullanılabilir olması bu noktada çalışmamızı destekleyen önemli noktalardan birisidir. Ürünlerden numuneler alınarak incelenmeye gönderilmeden, daha dinamik bir yöntemle bu sistemin sürece hız açısından katkı sağlaması öngörülmektedir.

KAYNAKLAR

[1] Liu, Y., Su, J., Xu, G., Fang, Y., Liu, F., Su, B. (2020). Identification of grapevine (Vitis vinifera L.) cultivars by vine leaf image via deep learning and mobile devices. Res. Sq, 8, 1-21.

[2] ALKAN, A., ABDULLAH, M. U., ABDULLAH, H. O., ASSAF, M., Zhou, H. (2021). A smart agricultural application: automated detection of diseases in vine leaves usinghybrid deep learning. Turkish Journal of Agriculture and Forestry, 45(6), 717-729.

[3] Sabancı, K., Aydın, C., Ünlerşen, M. F. (2012). Görüntü işleme ve yapay sinir ağları yardımıyla patates sınıflandırma parametrelerinin belirlenmesi. Journal of the Institute of Science and Technology, 2(2 Sp: A), 59-62.

[4] Tairu, O. E. (2018). PlantVillage Dataset, Kaggle, https://www.kaggle.com/datasets/emmarex/plantdisease (01.06.2025)

[5] Mikołajczyk, A., Grochowski, M. (2018). Data augmentation for improving deep learning in image classification problem. In 2018 international interdisciplinary PhD workshop (IIPhDW), 9-12 Mayıs, Świnoujście, Poland.

[6] Khaled, N., Mohsen, S., El-Din, K. E., Akram, S., Metawie, H., Mohamed, A. (2020). In-door assistant mobile application using cnn and tensorflow. In 2020 International Conference on Electrical, Communication, and Computer Engineering (ICECCE), 12-13 June, İstanbul, Turkey.

[7] Sozzi, M., Cantalamessa, S., Cogato, A., Kayad, A., Marinello, F. (2022). Automatic bunch detection in white grape varieties using YOLOv3, YOLOv4, and YOLOv5 deep learning algorithms. Agronomy, 12(2), 319.

[8] Koklu, M., Unlersen, M. F., Ozkan, I. A., Aslan, M. F., Sabanci, K. (2022). A CNN-SVM study based on selected deep features for grapevine leaves classification. Measurement, 188, 110425.

[9] Thet, K. Z., Htwe, K. K., Thein, M. M. (2020). Grape leaf diseases classification using convolutional neural network. In 2020 International Conference on Advanced Information Technologies (ICAIT), 04-05 November, Yangon, Myanmar.

[10] Math, R. M., Dharwadkar, N. V. (2022). Early detection and identification of grape diseases using convolutional neural networks. Journal of Plant Diseases and Protection, 129(3), 521-532.

[11] Zhang, Z., Qiao, Y., Guo, Y., He, D. (2022). Deep learning based automatic grape downy mildew detection. Frontiers in Plant Science, 13, 872107.

[12] Jamadar, R. A., Sharma, A. (2023). Semantic Segmentation Based Leaf Disease Severity Estimation Using Deep Learning Algorithms. In 2023 International Conference on Emerging Smart Computing and Informatics (ESCI), 01-03 March, Pune, India.

[13] Uttam, A. K., Kumar, A. (2022). Grape Leaves Disease Identification using Transfer Learning Techniques. In 2022 3rd International Conference on Smart Electronics and Communication (ICOSEC), 20-22 October, Trichy, India.

[14] Kandel, A., Prakhar, U., Kaur, M., Nand, P., Rakesh, N. (2023). Plant Leaves Disease Detection using Deep Learning.

[15] Wang, G., Wang, J., Wang, J., Sun, Y. (2022). Grape Leaf Disease Classification Combined with U-Net++ Network and Threshold Segmentation. Computational Intelligence and Neuroscience, 11, 1042737.

[16] Abade, A., Ferreira, P. A., de Barros Vidal, F. (2021). Plant diseases recognition on images using convolutional neural networks: A systematic review. Computers and Electronics in Agriculture, 185, 106125.

[17] Picon, A., Alvarez-Gila, A., Seitz, M., Ortiz-Barredo, A., Echazarra, J., Johannes, A. (2019). Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild. Computers and Electronics in Agriculture, 161, 280-290.

[18] Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis. Computers and electronics in agriculture, 145, 311-318.

[19] Sachdeva, G., Singh, P., Kaur, P. (2021). Plant leaf disease classification using deep Convolutional neural network with Bayesian learning. Materials Today: Proceedings, 45, 5584-5590.

[20] Toseef, M., Khan, M. J. (2018). An intelligent mobile application for diagnosis of crop diseases in Pakistan using fuzzy inference system. Computers and Electronics in Agriculture, 153, 1-11.

[21] Bozkurt, A., Yağcı, A., Akgül, D. S. (2024). Bağcılıkta Sarmalık Asma Yaprağı Kullanımına Yönelik Çeşit Geliştirme. Meyve Bilimi, 11(1), 18-26.

[22] Ürgüp Belediyesi. 53. Uluslararası Ürgüp Bağ Bozumu Festivali, https://urgup.bel.tr/tr/etkinlik/53-uluslararasi-urgup-bag-bozumu-festivali/13 (02.06.2025).

[23] Luo, J., Wang, Y., Qin, H. (2024). Super-high-fidelity image compression via hierarchical-roi and adaptive quantization. arXiv preprint arXiv:2403.13030.

[24] Güven, H., Saygılı, A. (2024). COMPUTER-AIDED DETECTION OF BRAIN TUMORS USING IMAGE PROCESSING TECHNIQUES. Kahramanmaraş Sütçü İmam Üniversitesi Mühendislik Bilimleri Dergisi, 27(3), 999-1018.

[25] Karadeniz, A. T. (2024). Patates Yaprak Görüntüleri İle Hastalık Teşhisi: Vision Transformer Ve VGG16 Derin Öğrenme Yöntemlerinin Karşılaştırması. Pratik Yapay Zeka Uygulamaları 2, Bölüm VI, 128-152.

[26] Medium. Image Normalization in PyTorch: From Tensor Conversion to Scaling, https://medium.com/@piyushkashyap045/image-normalization-in-pytorch-from-tensor-conversion-to-scaling-3951b6337bc8 (02.06.2025).

[27] Medium. U-Net Mimarisi Nedir?, https://medium.com/@elifbeyzaozkan/u-net-mimarisi-nedir-9730cb1fe763 (02.06.2025)

[28] Segmentation Models. Segmentation Models Documents, https://smp.readthedocs.io/en/latest/models.html#unet (02.06.2025)

[29] Segmentation Models. Segmentation Models Documents: Encoders, https://segmentation-modelspytorch.readthedocs.io/en/latest/index.html?highlight=encoder#encoders (Erişim Tarihi: 02.06.2025)

[30] Ronneberger, O., Fischer, P., Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, 5-9 October, Munich, Germany.

[31] C# Corner. AI-Powered Upskilling and Growth Platform for Software Developers: Image Classification in Web Applications with ML.NET, https://www.c-sharpcorner.com/article/image-classification-in-web-applications-with-ml-net/ (01.01.2025).

[32] Microsoft Learn. Tutorial: ML.NET Classification Model to Categorize Images, https://learn.microsoft.com/en-us/dotnet/machine-learning/tutorials/image-classification (02.06.2025)

[33] Mathew, M. P., Elayidom, S., Jagathy Raj, V. P., & Abubeker, K. M. (2025). Development of a handheld GPU-assisted DSC-TransNet model for the real-time classification of plant leaf disease using deep learning approach. Scientific Reports, 15(1), 3579.

[34] IBM. Image Segmentation, https://www.ibm.com/think/topics/image-segmentation (31.08.2025).

[35] Bugbee, E., Wilber, J. Logistic Regression, https://mlu-explain.github.io/logistic-regression/ (31.08.2025)

[36] Medium. REST / SOAP API Nedir? Farkları Nelerdir?, https://medium.com/android-t%C3%BCrkiye/rest-apikavramlar%C4%B1-soap-ile-farklar%C4%B1-4c6f19ddbc6c (31.08.2025)

[37] Saifullah, S., Yudhana, A., Suryotomo, A. P. (2025). Automatic Brain Tumor Segmentation: Advancing U-Net With ResNet50 Encoder for Precise Medical Image Analysis. IEEE Access, 13, 43473-43489.

[38] Cil, A. E., Yildiz, K., Buldu, A. (2021). Detection of DDoS attacks with feed forward based deep neural network model. Expert Systems with Applications, 169, 114520.

[39] Korkmaz, G. (2025). Karar Ağaçlarının Aşırı Uyumla Sınavı: Karşılaştırmalı Bir Performans Analizi. Bilişim Teknolojileri Dergisi, 18(3), 253-268.

[40] Salelew, E., Srahbzu, M., Diges, T., Yeniakal, H. G., Begashaw, G., Tadele, T., Hailu, T., Degef, W., Bazezew, F., Demilew, D., Belachew, M. (2025). Developing and testing preliminary effectiveness and acceptability of a mental health application in Ethiopia. Discover Mental Health, 5(1), 1-8.

[41] Boyar, T., Yıldız, K. (2022). Powdery mildew detection in hazelnut with deep learning. Hittite Journal of Science and Engineering, 9(3), 159-166.

[42] Siddiqui, A. A. I., Kayte, C. (2023, May). Convolution Neural Network-based Mosquito Classification System. In Proceedings of the 3rd International Conference on ICT for Digital, Smart, and Sustainable Development (Vol. 449, p. 60).

[43] Loa, J., Soewito, B. (2025). Seamless high availability in MySQL database systems: An advanced implementation of ProxySQL, Orchestrator, and Keepalived. In International Conference on Advancement in Data Science, E-learning and Information System (ICADEIS 2025), 3–4 February, Bandung, Indonesia (pp. 1–6). IEEE.

[44] Boyar, T., Turan, S. C., Çıplak, Z., Yıldız, S. G., Sarıkaş, A., Yıldız, K., Demir, Ö. (2025). Recognition of leaf diseases in hazelnut orchards with drone imagery by deep learning models: Effectiveness of U-Net model variants. In Proceedings of the 26th International Conference on Computer Systems and Technologies (CompSysTech'25), 27–28 June, University of Ruse, Bulgaria. IEEE.

[45] Esen, F. A. (2022). Derin öğrenme tabanlı akıllı tarım ve uygulamaları. Master's Thesis, Izmir Katip Çelebi University, Turkey.

[46] Serdar, Ö., & Yücedağ, D. İ. (2020) Web API Tasarımı. Lisans Bitirme Tezi, Düzce Üniversitesi Teknoloji Fakültesi, Düzce, Türkiye.

[47] Mathew, A. C., Suresh, A., Biju, A. M., Shaji, S., Jacob, B. M. (2024). A Comprehensive Web-Based Solution Using Django Framework and SQLite Database.

[48] Setiawan, R., Setiawan, J. Performance Comparison Oracle, Postgresql, and Mysql Database Using Jmeter Tools. Postgresql, and Mysql Database Using Jmeter Tools. Available at SSRN: https://ssrn.com/abstract=5275984 or http://dx.doi.org/10.2139/ssrn.5275984

[49] Hanife, D. Android Room Database Kullanımı, https://hanifedev.medium.com/android-room-database-kullan%C4%B1m%C4%B1-7291ab6a6677 (04.09.2025)

[50] Huawei Developers. RoomDB vs SQLite: Exploring Database Options for Android Development, https://medium.com/huawei-developers/roomdb-vs-sqlite-exploring-database-options-for-android-development-1120151e6737 (04.09.2025)

ÖZGEÇMİŞ

Adı Soyadı **:** Seyit Gazi YILDIZ

Öğrenim Durumu

Derece	**Bölüm/Program**	**Üniversite/Lise**	**Mezuniyet Yılı**
Lisans	Bilgisayar Mühendisliği	Ankara Yıldırım Beyazıt Üniversitesi	Haziran 2018
Lise	Anadolu Öğretmen Lisesi	Nevşehir Anadolu Öğretmen Lisesi	Haziran 2013

İş Deneyimi

Yıl	**Firma/Kurum**	**Görevi**
2020 – Halen	Profe Bilgi Sistemleri	Mobile Software Engineer
2018-2019	Docuart Bilgi Teknolojileri	Bilgisayar Mühendisi

Bilimsel Eserler:

1. Yıldız, S. G., & Yıldız, K. (2023). ANN based early detection of Alzheimer disease on selected features. Mühendislik Bilimleri ve Tasarım Dergisi, 11(4), 1508-1516.
2. Çıplak, Z., Yıldız, S. G., & Doğan, B. (2024). Bilişim Sektörü ve Yakın Alanlardaki Çalışanların; Hibrit, Uzaktan ve Yerinde Çalışma Modelleriyle İlgili Görüşlerinin Değerlendirilmesi. Bilgi Yönetimi, 7(1), 123-139.
3. Boyar, T., Turan, S. C., Çıplak, Z., Yıldız, S. G., Sarıkaş, A., Yıldız, K., Demir, Ö. (2025). Recognition of leaf diseases in hazelnut orchards with drone imagery by deep learning models: Effectiveness of U-Net model variants. In Proceedings of the 26th International Conference on Computer Systems and Technologies (CompSysTech'25), 27–28 June, University of Ruse, Bulgaria. IEEE.