



İSKENDERUN TEKNİK

ÜNİVERSİTESİ

LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

**YÜKSEK
LİSANS
TEZİ**

**ROBOTİK SİSTEMLER İÇİN
GÖRÜNTÜ İŞLEME
YÖNTEMLERİ KULLANILARAK
ZEYTİN MEYVESİNİN KONUMUNUN
BELİRLENMESİ**

Demet OFLAZOGLU

**ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ
ANABİLİM DALI**

OCAK 2023



**ROBOTİK SİSTEMLER İÇİN GÖRÜNTÜ İŞLEME YÖNTEMLERİ
KULLANILARAK ZEYTİN MEYVESİNİN KONUMUNUN
BELİRLENMESİ**

Demet OFLAZOGLU

**YÜKSEK LİSANS TEZİ
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI**

**İSKENDERUN TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**

OCAK 2023

Demet OFLAZOGLU tarafından hazırlanan “ROBOTİK SİSTEMLER İÇİN GÖRÜNTÜ İŞLEME YÖNTEMLERİ KULLANILARAK ZEYTİN MEYVESİNİN KONUMUNUN BELİRLENMESİ” adlı tez çalışması aşağıdaki jüri tarafından OY BİRLİĞİ ile İskenderun Teknik Üniversitesi Elektrik Elektronik Mühendisliği Anabilim Dalında YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Danışman: Dr. Öğretim Üyesi Ersin ÖZDEMİR

Elektrik Elektronik Mühendisliği Anabilim Dalı, İskenderun Teknik Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

.....
.....

Başkan: Prof. Dr. Mustafa YÜKSEK

Elektrik Elektronik Mühendisliği Anabilim Dalı, İskenderun Teknik Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

.....
.....

Üye: Doç. Dr. Mehmet SÖNMEZ

Elektrik Elektronik Mühendisliği Anabilim Dalı, Korkut Ata Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

.....
.....

Tez Savunma Tarihi: 24/01/2023

Jüri tarafından kabul edilen bu tezin Yüksek Lisans Tezi olması için gerekli şartları yerine getirdiğini onaylıyorum.

.....

Doç. Dr. Ersin BAHÇECİ

Lisansüstü Eğitim Enstitüsü Müdürü

ETİK BEYAN

İskenderun Teknik Üniversitesi Lisansüstü Eğitim Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez üzerinde Yükseköğretim Kurulu tarafından hiçbir değişiklik yapılamayacağı için tezin bilgisayar ekranında görüntülendiğinde asıl nüsha ile aynı olması sorumluluğunun tarafıma ait olduğunu,
 - Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
 - Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
 - Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
 - Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
 - Bu tezde sunduğum çalışmanın özgün olduğunu,
- bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

İmza

Demet OFLAZOGLU

24 /01/2023

ROBOTİK SİSTEMLER İÇİN GÖRÜNTÜ İŞLEME YÖNTEMLERİ KULLANILARAK
ZEYTİN MEYVESİNİN KONUMUNUN BELİRLENMESİ
(Yüksek Lisans Tezi)

Demet OFLAZOGLU

İSKENDERUN TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

Ocak 2023

ÖZET

Teknolojinin gelişmesiyle birlikte görüntü işleme tekniklerinin ve yapay zekâ kullanımının artması, birçok alanda süreçlerin otomatik olarak gerçekleşmesine olanak sağlamıştır. Son yıllarda tarımsal ilaçlama, sulama gibi işlemlerin yanında ürün hasadı gibi zorlu işler de otomatik olarak gerçekleştirilmeye başlanmıştır. Otomatik hasat sistemlerinde hasadı gerçekleştirecek robotun meyveyi tanıması ve ona yönelebilmesi için meyvenin koordinatlarını tespit edebilmesi gerekmektedir. Meyvelerin doğru bir şekilde tespit edilebilmesi için de sistemin modellenmesi amacıyla kullanılacak veri seti çok önemlidir. Literatürde otomatik meyve hasadı ve meyvelerin otomatik olarak tespit edilebilmesi üzerine pek çok çalışma bulunmaktadır. Ancak, literatürde zeytin meyvesinin gerçek ortamda tespit edilmesi ile ilgili herhangi bir çalışma bulunmamaktadır.

Bu tez çalışmasında, 12 MP Optik görüntü sabitleyiciye (OIS) sahip kamera ile 1,3 MP CMOS fotoğraf sensörüne sahip IP kamerası kullanılarak elde edilen zeytin ağacının meyvesine ait görüntülerden bir veri seti oluşturulmuştur. Daha sonra veri setinde bulunan resimlere çeşitli görüntü işleme teknikleri ve meyve etiketleme işlemleri uygulanmıştır. Son olarak CNN, YOLOv3-tiny, YOLOv4-tiny-custom derin öğrenme modelleri kullanılarak zeytin meyvesinin gerçek ortamda tespit edilmesi işlemi gerçekleştirilmiştir. Robot kolun zeytin meyvesine erişebilmesi için mesafenin de doğru olarak ölçülmesi gerekmektedir. Bu nedenle çalışmada ayrıca meyvenin derinlik tespiti için Microsoft Kinect v1 ve stereo kamera düzenekleri kullanılmıştır. Zeytin meyvesini tespit etmek amacıyla kullanılan YOLOv4-tiny-custom modeli 0.88 F1 skoru ile en yüksek başarı oranını vermiştir. Derinlik tespitinde %98'lik başarı ile meyve derinliği ölçülmüştür.

Çalışmalar gerçek bir hasat robot sisteminin tasarlanması konusunda devam etmektedir.

Anahtar Kelimeler : YOLOv4, derin öğrenme, zeytin hasadı

Sayfa Adedi : 59

Danışman : Dr. Öğretim Üyesi Ersin ÖZDEMİR

DETERMINING THE POSITION OF OLIVE FRUIT USING IMAGE PROCESSING METHODS FOR ROBOTIC SYSTEMS

(M. Sc. Thesis)

Demet OFLAZOGLU

ISKENDERUN TECHNICAL UNIVERSITY
INSTITUTE OF GRADUATE STUDIES

January 2023

ABSTRACT

With the advancement of technology, the increase in image processing techniques and the use of artificial intelligence has made it possible for processes to be automatically carried out in many fields. For example, in the past few years, in addition to agricultural pest control and irrigation processes, challenging processes such as fruit harvesting have also been automated. In automatic harvesting systems, it is necessary for the robot responsible for harvesting to be able to identify and approach the fruit by determining its coordinates. A data set that will be used for modeling the system is a very important requirement for accurately identifying the fruit. There are many studies in the literature on automatic fruit harvesting and the automatic detection of fruit. However, there is no study on the detection of olive fruit, which is densely found on tree branches and almost the same color and size as the leaves, in real environments.

In this study, a dataset was created from images of the fruit of an olive tree obtained using 12 MP Optical image stabilizer (OIS) camera and an IP camera with a 1.3 MP CMOS photo sensor. Then various image processing techniques and fruit labeling processes were applied to the images in the dataset. Finally, the process of detecting olive fruit in real environments was carried out using CNN, YOLOv3-tiny, and YOLOv4-tiny-custom deep learning models. It is also necessary to accurately measure the distance between the robot arm and the fruit in order for the robot arm to be able to access the olive fruit. For this reason, in the study, Microsoft Kinect v1 and stereo camera configurations were also used to detect the depth of the fruit. According to the results of the models used to detect olive fruit, the YOLOv4-tiny-custom model achieved the highest success rate with a 0.88 F1 score.

Key Words : YOLOv4, deep learning, olive harvesting
Page Number : 59
Supervisor : Asst. Prof. Dr. Ersin ÖZDEMİR

TEŞEKKÜR

Bu çalışmada, gerçek ortamda elde edilmiş görüntülerden derin öğrenme algoritmaları ile zeytin meyvesinin tespiti gerçekleştirilmiştir. Bu tez çalışmasının tüm aşamalarında bana rehberlik edip, bilgi ve tecrübelerini benimle paylaşan ve tezimin düzenlenmesi ve yazımı konusunda her türlü desteği sağlayan danışman hocam Sayın Dr. Öğr. Üyesi Ersin ÖZDEMİR'e şükranlarımı sunarım.

Ayrıca, hayatım boyunca benim için en iyi şartları sağlayarak, bugünlere gelebilmem için her türlü fedakârlıkta bulunan saygıdeğer aileme, yaptığım her işte beni destekleyerek her zaman yanımda olan eşim Öğretim Görevlisi Çağlar OFLAZOGLU'na ve son olarak en büyük motivasyon kaynağım canım oğlum Toprak OFLAZOGLU'na teşekkür ederim.

İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT.....	v
TEŞEKKÜR.....	vi
İÇİNDEKİLER	vii
ÇİZELGELERİN LİSTESİ.....	ix
ŞEKİLLERİN LİSTESİ.....	x
RESİMLERİN LİSTESİ.....	xi
SİMGELER VE KISALTMALAR.....	xiii
1. GİRİŞ.....	1
2. ÖNCEKİ ÇALIŞMALAR.....	3
3. MATERYAL VE YÖNTEM	13
3.1. Veri Seti.....	13
3.2. Evrişimsel Sinir Ağları.....	13
3.2.1. Evrişimsel sinir ağının elemanları.....	14
3.2.2. Evrişimsel sinir ağları ile zeytin meyvesinin tespiti	16
3.3. YOLO Mimarisi	20
3.3.1. YOLO model eğitimi için meyve etiketlemesi	22
3.3.2. YOLO model eğitimleri	25
3.4. Kullanılan Nesne Tespit Modellerinin Performans Ölçme Parametreleri	27
3.5. Zeytin Meyvesinin Uzaklığının Tespiti	28
4. ARAŞTIRMA BULGULARI	33
4.1. CNN Modeli İle Zeytin Meyvesi Tespiti.....	33
4.2. YOLOv3-tiny ve YOLOv4-tiny-custom Model Sonuçları	38

	Sayfa
4.2.1. YOLOv3-tiny	38
4.2.2. YOLOv4-tiny-custom	42
4.3. Zeytin Meyvesinin Uzaklık Tespitine Yönelik Ön Çalışma Sonuçları	47
5. SONUÇ VE ÖNERİLER	55
KAYNAKLAR	56



ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 3.1. CNN modeline ait veri seti dağılımı.....	16
Çizelge 3.2. Güncellenen YOLO parametreleri.....	26
Çizelge 3.3. Samsung S20 FE kamerasının teknik özellikleri.....	30
Çizelge 3.4. Microsoft Kinect v1 kamerasının teknik özellikleri.....	31
Çizelge 4.1. CNN modeline ilişkin test sonuçlarının karışıklık matrisi.....	38
Çizelge 4.2. YOLOv3-tiny modeline ilişkin test sonuçlarının karışıklık matrisi.....	42
Çizelge 4.3. YOLOv4-tiny-custom modeline ilişkin test sonuçlarının karışıklık matrisi.....	46
Çizelge 4.4. Derin öğrenme modellerine ilişkin karşılaştırmalı sonuçlar.....	47
Çizelge 4.5. Zeytin meyvesinin derinlik tespitine ilişkin sonuçlar.....	53

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 3.1. CNN modelinin yapısı	14
Şekil 3.2. Keras akış diyagramı	18
Şekil 3.3. Piramit gösterimi	20
Şekil 3.4. YOLO zeytin meyvesi tespit sistemi	22
Şekil 3.5. YOLOv3-tiny mimarisi	26
Şekil 3.6. YOLOv4-tiny mimarisi	27
Şekil 3.7. Stereo kamera ile derinlik tespit düzeneği	29
Şekil 3.8. Stereo kamera ile zeytin meyvesinin uzaklık tespitine ait blok diyagramı	30
Şekil 4.1. CNN modelinin eğitimine ait öğrenme eğrisi.....	34
Şekil 4.2. CNN modelinin eğitimine ait karışıklık matrisi	35
Şekil 4.3. YOLOv3-tiny ortalama kayıp ve kesinlik eğrisi	39
Şekil 4.4. YOLOv4-tiny-custom ortalama kayıp ve kesinlik eğrisi.....	43

RESİMLERİN LİSTESİ

Resim	Sayfa
Resim 1.1. Zeytin meyvesi toplama yöntemleri	2
Resim 3.1. Kayan pencere yöntemi (<i>150px kaydırılmış, 300x300px pencere</i>).....	17
Resim 3.2. CNN modeline ait veri setinde bulunan resim örnekleri	17
Resim 3.3. Kayan pencere yöntemi ile resimlerin kırılması	23
Resim 3.4. BBox-Label-Tool etiketleme aracı arayüzü	24
Resim 3.5. BBox-Label-Tool ile etiketlenmiş görüntüler.....	24
Resim 3.6. Stereo kamera düzeneği.....	31
Resim 4.1. a) Orjinal ve b) renk filtresi uygulanmış görüntüler	33
Resim 4.2. CNN ile eğitilmiş modelin test sonuçları	36
Resim 4.3. CNN modelinin test sonuçları.....	37
Resim 4.4. Yolov3-tiny sonuçları-1	40
Resim 4.5. Yolov3-tiny sonuçları-2	40
Resim 4.6. Yolov3-tiny sonuçları-3	41
Resim 4.7. Yolov3-tiny sonuçları-4	41
Resim 4.8. Yolov4-tiny-custom sonuçları-1	44
Resim 4.9. Yolov4-tiny-custom sonuçları-2	45
Resim 4.10. Yolov4-tiny-custom sonuçları-3	45
Resim 4.11. Yolov4-tiny-custom sonuçları-4	46
Resim 4.12. Kinect v1 kullanılarak yakın çekim ile elde edilen a) RGB görüntü b) derinlik görüntüsü	47
Resim 4.13. Kinect v1 kullanılarak uzak çekim ile elde edilen a) RGB görüntü b) derinlik görüntüsü	48
Resim 4.14. Samsung S20 FE ile sol taraftan yapılan çekimden elde edilen görüntü ...	49

Resim	Sayfa
Resim 4.15. Samsung S20 FE ile sađ taraftan yapılan çekimden elde edilen görüntü ..	50
Resim 4.16. Sol kamera çekimi ile elde edilen görüntüden YOLOv4 modeli ile zeytin meyvesinin tespiti.....	51
Resim 4.17. Sađ kamera çekimi ile elde edilen görüntüden YOLOv4 modeli ile zeytin meyvesinin tespiti.....	52
Resim 4.18. Zeytin meyvelerinin gerçek derinlik deđerleri.....	54



SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Kısaltmalar

Açıklamalar

AI	Artificial Intelligent
ASPP	Atrous Spatial Pyramid Pooling
CCD	Charged Coupling Devices
CHT	Circular Hough Transform
CMOS	Complementary Metal Oxide Semiconductor
CNN	Convolutional Neural Network
COCO	Microsoft Common Objects
DaSNet	Detection and Segmentation Network
F-RCNN	Faster Region Based Convolutional Networks
GFPN	Gated Feature Pyramid Network
IR	Infrared
LBP	Local Binary Pattern
LIDAR	Light Detection and Ranging
LMS	Laser Measurement Sensor
LPT	Laplacian Pyramid Transform
Lw-net	Lightweight network
NMS	Non-Maximum Suppression
RGB	Red-Green-Blue
SSD	Single Shot Detector
ToF	Time of Flight
VGG	Visual Geometry Group
YOLO	You Look Only One
ZF	Zeiler – Fergus

1. GİRİŞ

Günümüzde her alanda olduğu gibi tarım alanında da gerçekleşen teknolojik gelişmeler ekim, sulama, hasat, ilaçlama gibi süreçlerin otomatize edilerek; süre, işgücü gibi konularda tasarruf sağlanmasına ve verimliliğin artmasına olanak sağlamıştır. Asırlardır önemini yitirmemiş olan zeytin bitkisinin anavatanı Doğu Akdeniz bölgesi olarak kabul edilmektedir. Zeytin yetiştiriciliğinin büyük bir kısmı Akdeniz havzası, geriye kalan kısmı ise Latin Amerika ülkelerinde yapılmaktadır. Ticaret Bakanlığı tarafından 2019 yılının nisan ayında yayımlanan “2018 YILI ZEYTİN VE ZEYTİNYAĞI RAPORU” na göre dünya genelinde zeytinyağı üretimi son beş sezon ortalamasına göre 2,91 milyon ton ve sofralık zeytin üretimi ise ortalama 2.77 milyon tonda olarak açıklanmıştır. En önemli üretici ülkeler sırasıyla AB (895 bin ton), Türkiye (423,6 bin ton), Mısır (450 bin ton), Cezayir (342,5 bin ton), Fas (130 bin ton) ve Suriye (100 bin ton)’dir. Ayrıca Türkiye genelinde zeytin alanlarının artış göstermesi ve üreticilerin verim arttırmaya yönelik uygulamaları ile birlikte zeytin üretiminin önemli ölçüde artması hedeflenmektedir (Seçer ve Ekmeksiz, 2012). Buna bağlı olarak hasat süreleri ve maliyetlerinin de artması kaçınılmazdır.

Günümüzde zeytin hasadı elle, sırkla vurarak, titreşimli veya sensörlü teleskopik hasat makinaları gibi yöntemlerle yapılmaktadır. Ancak, bu yöntemlerin her birinin ayrı bir zorluğu ve olumsuzluğu bulunmaktadır. Örneğin, sırkla vurarak toplama (Resim 1.1.a) hem yorucu hem de çoğu zaman zeytin meyvesinin ve ağaç dallarının zarar görmesine neden olan bir yöntemdir. Elle toplama yöntemi ise yüksek ağaçların yüksek noktalarına tırmanmayı gerektirir ve bu da yorucu olmakla beraber ağaçtan düşme vb. kazalarla yaralanmalara neden olabilmektedir. Ayrıca hasadın sonbahar mevsiminde, kısa süreli insan işgücü ihtiyacı gerektirmesi gibi nedenler ile işçi ve üretim maliyetlerinin ciddi oranlarda artmasına sebep olmaktadır.

Titreşimli hasat makinaları ile zeytin hasadı ağaç boyundan bağımsız ve hızlı bir yöntemdir. Ancak zeytin arazilerinin genellikle eğimli ve titreşimli makinaların çok yüksek maliyetli olması birer dezavantajdır. Işık ve Ünal (2003) tarafından yapılan çalışmaya göre, mekanik titreşimli hasat makinası ile yapılan ortalama hasat kapasitesi, elle yapılan hasattan yaklaşık 6,7 kat daha fazladır. El ile hasat maliyet değeri de makineyle hasada göre 6,5 kat daha fazladır. Ancak yapılan deneylerde makineyle hasatta %1,80 oranında yaprak kaybı

gözlemlenirken el ile hasatta %1,02 oranında yaprak kaybı meydana gelmiştir. Bununla beraber sofralık zeytinlerin zarar görmemesi için hassas yöntemlerle toplanması gerekmektedir. Teleskopik hasat makinaları (Resim 1.1.b) ile hasat ise ağaçlarda fazla yaprak kaybına ve sonraki yıllarda ise ürün verecek olan yeni filizlenmiş dalların zarar görmesine dolayısıyla da sonraki senelerin ürün veriminin azalmasına neden olmaktadır. Ayrıca hasat esnasında zeytin tanesi dışında düşen dal, filiz ve yaprak gibi materyallerin temizlenmesi için ayrı bir işlem gerekmektedir. Aygün, Urkan, Alayunt, Yalçın ve Tekin (2019), teleskopik hasat makinası kullanarak yapmış oldukları çalışmada zeytin tanesi dışındaki materyal (dal+filiz+yaprak) oranının %7 ile % 10,71 aralığında olduğunu ve ağaç boyutu arttıkça makine ile yüksek dallara ulaşımın zorlaştığını belirtmişlerdir (Ereyon, 2022).



(a)



(b)

Resim 1.1. Zeytin meyvesi toplama yöntemleri: a) Sırıkla vurarak, b) Sensörlü teleskopik zeytin hasat makinası

Görüntü işleme ve sınıflandırma yöntemlerinin yapay zekâ algoritmaları ile desteklenmesi ile birlikte otomatik meyve hasadı gibi birçok konu güncelliğini ve önemini sürdürmektedir. Literatürde meyvelerin tespiti ve meyve hasadının otomatik olarak gerçekleştirilebilmesi konusunda birçok çalışma yer almakla beraber otomatik zeytin meyvesi hasadı konusunda henüz bir çalışma yapılmadığı gözlemlenmiştir.

2. ÖNCEKİ ÇALIŞMALAR

Tarımsal ürünlerin insan işgücüne ihtiyaç olmadan maksimum verim ve minimum maliyetle hasat edilmesi geçmişten günümüze süregelen bir problemdir. Bu amaçla hasat işlemi için birçok yöntem geliştirilmiştir.

Li, Lee ve Hsu'nun (2011) yapmış olduğu araştırmaya göre 1960'ların başında mekanik hasat yöntemleri araştırılmaya başlanmış ve uygulanmıştır. Mekanik olarak, uzuv çalkalayıcı, gövde çalkalayıcı, hava üflemeli, saçak çalkalayıcı gibi yöntemler geliştirilmiş ancak bu yöntemlerde yaprak ve/veya meyvenin zarar görmesi ve olgunlaşmamış meyvelerin hasat edilmemesi gibi problemlerle karşılaşmıştır. Bazı çalışmalarda mekanik hasat işleminde meyvelerin gevşeyip daha hızlı düşmesini sağlamak amacıyla hasat öncesinde Ethephon ve 2-kloroetil fosforik asit gibi absisyon (kesme) kimyasalları kullanılmıştır. Örneğin, Sumner ve Hedden (1975) portakal hasadı için dal silkme ile birlikte iki çeşit kesme kimyasalı kullanmışlardır. Ancak bu kimyasalların kullanılması birlikte ile aşırı yaprak dökülmesi ve meyvelerin hasattan önce dökülmeye başlaması gibi problemlerin ortaya çıktığını bildirmişlerdir.

Mekanik hasat sistemleri üzerine yapılan çalışmalarda, mekanik hasat sistemlerinin insan görüşünün sağlayabildiği kalite, boyut ve olgunluk gibi parametreleri doğru olarak belirleyemediği ortaya konulmaktadır. Bu nedenle Schertz ve Brown (1968) mekanik hasat makinesine alternatif olabilecek bir otomatik hasat makinesi önermişlerdir. İki ayırma cihazı, meyveyi döndürüp çevirerek ayıran büyük kauçuk boru ve bir döner kesme cihazından oluşan bu hasat makinesi, meyvelerin otomatik olarak tespit edilmesini sağlayan ilk çalışmadır. Çalışmada, meyvelerin konumunun fotometrik bilgilerle, özellikle elektromanyetik spektrumun görünür veya kızılötesi kısmında yapraklar ve meyveler arasındaki ışık yansıtma farkları kullanılarak belirlenebileceğini ancak, aydınlatmanın düzgün olmaması ve meyveleri sınırlayan yeşillik konsantrasyonu gibi problemlere dikkat edilmesi gerektiği öne sürülmüştür.

Jimenez, Ceres ve Pons (2000) yaptıkları çalışmada bir lazer mesafe bulucu kullanarak meyve rengi veya olgunluk aşamasından bağımsız olarak portakal, şeftali ve elma gibi küresel şekilli herhangi bir meyveyi tespit etmek için şekil analizine dayanan bir yöntem

önermişlerdir. Çalışmada; 3D konum, yarıçap ve yansıtma gibi küre parametrelerini tahmin edebilmek için şekil, tepe, bombelik, yansıtma katsayısı gibi dört farklı bileşen kullanmışlardır.

Li ve diğerlerinin (2011) yapmış olduğu araştırmaya göre otomatik hasat sistemlerinin temel fikirleri ilk olarak Montpellier Fransa'da geliştirilmiştir. Buna göre 1984'te laboratuvar ortamında meyvelerin tespit ve hasat edilmesini sağlayan bir prototip geliştirilmiştir. Geliştirilen modelde meyveye yatay olarak ulaşmaya çalışan bir teleskop kolun üzerine bir siyah beyaz kamera yerleştirilmiştir. Daha sonra yapılan çalışmalarda renkli kameralar, sensörler, kendinden tahrikli ve/veya birden fazla serbestlik derecesine sahip robotlar kullanılmaya başlanmıştır.

Tanigaki ve Fujiura, Akase ve Imagawa (2008), 3 boyutlu kızılötesi lazer diyotlardan oluşan bir görüş sensöründen elde ettikleri anlık görüntülerden meyve ve engellerin yerini tespit ederek 4 eksenli bir robot ile kiraz hasadı gerçekleştirmişlerdir.

Scarfe, Flemmer, Bakker ve Flemmer (2009) dallar arasına bakmak, meyveyi ve meyvenin koordinatlarını tespit edip binoküler görüntü elde etmek için 8 web kamerasından oluşan 4 kollu bir otonom kivi toplama robotu geliştirmişlerdir.

Yasukawa, Li, Sonoda ve Ishii (2017) domates hasadı için 6 eksenli bir robot kol, bir uç efektör ile tüm bu aksamı taşıyan ve raylar üzerinde hareket eden özel bir mekanizma tasarlamışlardır. Yasukawa ve diğerleri, domateslerin kızılötesi ışığa karşı hem saplar hem de yapraklar için yüksek yansıtma oranına sahip olması özelliğinden yola çıkarak kızılötesi görüntü ve aynasal yansıma yöntemiyle meyve konumunu tespit etmişlerdir.

Lehnert, English, McCool, Tow ve Perez (2017), bir uç efektörde bulunan RGB-D kamera ve hareketli bir tabana yerleştirilmiş 7 eksenli özel bir araç kullanarak biber hasadı gerçekleştirmişlerdir. Biberlerin konumu ve her bir biberin 3 boyutlu şeklini belirleyerek kesme ve kavrama noktalarını algılamak için Kinect Fusion (Gerçek Zamanlı Yoğun Yüzey Haritalama ve İzleme) algoritması kullanmışlardır.

Mu ve diğerleri (2017) Kinect sensöründen alınan bilgilerden görüntü işleme tekniklerini kullanarak meyve koordinatları ve kartezyen koordinatlarını hesaplamışlardır. Parmaklara

takılan fiber optik sensörlerle de uç efektörün optimum konumu ve doğru tutma pozisyonunu belirleyerek kivi meyvesinin hasadını gerçekleştirmişlerdir.

Xiong, Ge, Grimstad ve From (2020), çilek hasadı için RGB-D kamera ile donatılmış tek raylı çift kollu manipülatör, iki gelişmiş tutucu ve bir robotik platformdan oluşan bir hasat robotu kullanmışlardır. Konum algılaması için robotun önüne bir Hokuyo marka LIDAR (Light Detection and Ranging) sensörü yerleştirmişlerdir.

Arad ve diğerleri (2020) RGB-D kamera, grafik işleme ünitesi ve bir uç efektörden oluşan 6 eksenli bir endüstriyel robot kol kullanarak tatlı biber hasadı gerçekleştirmişlerdir.

Hannan, Burks ve Bulanon (2007); Sony Cybershot, DSC-P8 dijital renkli kamera ile elde ettikleri iç gölgelik, yakın ve uzak görüşleri kullanarak narenciye tespit için bir yapay görme algoritması önermişlerdir. Önermiş oldukları tespit algoritması bölütleme (segmentasyon), bölge etiketleme, boyut filtreleme, çevre çıkarma ve dairelerin algılanması aşamalarından oluşmaktadır. Yapmış oldukları çalışma ile test edilen görüntülerde meyvelerin %90'dan fazlasını tespit edilebildiğini göstermişlerdir.

Bulanon, Burks ve Alchanatis (2009), ABD'nin Florida kenti Ocala bölgesinde bulunan bir korudan Olympus D-490ZOOM ve FLIR System ThermoCAM P65HS termal kamera ile portakal resimleri çekerek bir veri seti oluşturmuşlardır. Portakal meyvesini algılamak için LPT (Laplacian Pyramid Transform) ve bulanık mantık (Fuzzy Logic) yöntemlerini kullanmışlardır. Yapmış oldukları çalışmada, termal kameradan almış oldukları görüntüler ile normal kameradan aldıkları görüntülere birleştirme işlemi uygulanınca meyve tespit performansının iyileştiğini ve görüntü birleştirme değerlendirme endekslerine göre bulanık mantık yönteminin LPT yönteminden daha iyi sonuç verdiğini ortaya koymuşlardır.

Li, M. Wang ve N. Wang (2010), gerçek zamanlı bir ananas tanıma sistemi için iki CMOS sensörlü kamera, bir tripod, bir dürbün tripod kafası ve bir bilgisayardan oluşan hasat robotu görüş platformu geliştirmişlerdir. Geliştirmiş oldukları platform ile Temmuz 2009 tarihinde Çin'in Guangdong eyaleti, Zhanjiang şehrinde bulunan bir ananas tarlasından farklı aydınlatma koşullarında (açık gökyüzü, kapalı gökyüzü, gölgesiz ve kısmen gölgeli) olgunlaşmış ananas görüntüleri toplamışlardır. Daha sonra bu görüntülere öznelik çıkarma, segmentasyon, meyvenin konum tespiti gibi görüntü işleme algoritmaları uygulamışlardır.

Yapmış oldukları çalışmalar sonrasında gökyüzünün açık olduğu günlerde meyve tanıma başarısının %90, bulutlu günlerde ise %60 olduğunu ortaya koymuşlardır.

Feng, Liu, Wang, Zeng ve Ren (2012), CCD (Charged Coupling Devices) ve CMOS (Complimentary Metal Oxide Semiconductor) kameraların kullanıldığı meyve tanıma sistemlerinin, dolaylı olarak elde edilen derinlik bilgisi ve aydınlatma koşullarına karşı hassasiyet problemlerini çözmek için bulunduğu alanın yelpaze şeklinde iki boyutlu taramasını yapabilen SICK LMS211 (laser measurement sensor) kullanarak yeni bir 3 boyut görüntü (elma) tanıma sistemi geliştirmişlerdir. Tanıma sistemi, LMS211'in ışık kaynağının (darbeli kızılötesi lazer 905 nm) bandında bir bitkinin her bir parçasının yansıtma oranının görünür bantlardan daha yüksek olması ve meyvenin yansıtma oranının da yaprak ve daldan daha yüksek olması prensibine dayanmaktadır. VC++6.0'da veri topladıktan sonra MATLAB R2010b ile derinlik görüntüsü oluşturulmuş ve ön işleme yapılarak meyvelerin tanınması gerçekleştirilmiştir. Yapmış oldukları çalışmada ağaçtaki elmaların doğru tanınma oranının %93'e ulaştığını belirtmişlerdir.

Makine öğrenme algoritmalarının çeşitliliğinin ve yaygın kullanımının artması ile birlikte bu algoritmalar literatürde yer alan son çalışmalarda daha fazla kullanılmaya başlanmıştır. Meyve hasadı ile ilgili makine öğrenme algoritmalarını kullanan çalışmalar şu şekilde sıralanabilir;

Gatica, Best, Ceroni ve Lefranc (2013), yapay sinir ağları ve temel bileşenler analizini kullanarak en iyi hasat zamanını tahmin etmek için zeytin ağacından alınan RGB görüntüleri üzerinden zeytin tanıma çalışması yapmışlardır. Bu çalışmada araştırmacılar, sabit bir zemin üzerine yerleştirdikleri zeytin dalı ve meyvesinin resimlerini kullanarak elde ettikleri veri setinden yararlanmışlardır.

Puttemans, Vanbrabant, Tits ve Goedemé (2016), bir makine öğrenme algoritması olan AdaBoost ile zayıf sınıflandırıcıların art arda sıralanmasına dayanan bir nesne sınıflandırma çerçevesi geliştirerek 2 boyutlu bilginin nesne tespiti için yeterli olabileceğini kanıtlamak amacıyla bir dedektör tasarlamışlardır. Tasarlamış oldukları bu dedektör, hem çilek hem de elma meyvelerinin otomatik hasat ve mahsul tahmini konusunda başarılı olmuştur.

Wang, Lee, Zou, Choi, Gan ve Diamond (2018), yeşil narenciye meyvelerini tespit etmek için ABD' de Florida Üniversitesi'nde bulunan bir narenciye bahçesinden Canon SD880IS

model dijital renkli kamera 2015 yılının çeşitli zamanlarında hem ön hem de arka aydınlatma şartlarında elde ettikleri veri setini kullanmışlardır. Wang ve diğerleri yapmış oldukları çalışmada, meyve ve meyve olmayan nesnelere doku özelliklerini LBP (Local Binary Pattern) ile çıkararak elde ettikleri bu özelliklerle AdaBoost algoritması tabanlı bir sınıflandırıcıyı eğitmişlerdir. Meyve genellikle yeşil ve dairesel olduğundan renk tabanlı K-ortalama (k-means) kümeleme ve şekil tabanlı CHT (Circular Hough Transform) algoritmalarını kullanmışlardır. Potansiyel meyve bölgeleri tespit edildikten sonra LBP'ye dayalı sınıflandırıcı ve CHT tespit sonuçları ile meyve olup olmadığı belirlenmiştir. Wang ve diğerleri önermiş oldukları bu yöntemin test seti için %85,6 oranında doğruluk elde ettiğini ortaya koymuşlardır.

Ge, Xiong, Tenorio ve From (2019) bir masa ve şerit arasında güvenli bir bölge belirleyerek robot kolun engellere çarpmadan güvenli bölgede konumlanmasını ve çileklerin hasat edilmesini sağlayan bir çalışma gerçekleştirmişlerdir. Ge ve arkadaşları yapmış oldukları çalışmada, RGB-D kamera ile elde ettikleri görüntülerden meyve, masa ve şeritlerin tespit edilmesinde ve ayrıştırılmasında, her bir örnek için hem sınırlayıcı kutu hem de maskeler üretebilen Mask-RCNN derin öğrenme algoritmasını kullanmışlardır. Öznelik çıkarıcı olarak da ResNet101 (Residual Networks) temel evrimsel sinir ağını kullanmışlardır.

Onishi ve arkadaşları (2019), stereo kamera ile donatılmış bir robot kol kullanarak Miyagi İl Tarım ve Bahçe Bitkileri Araştırma Merkezi'nde yetiştirilen "Fuji" elmasının otomatik tespit ve hasadı için bir yöntem önermişlerdir. Elmaların tespiti için CNN (Convolutional Neural Networks) tabanlı SSD (Single Shot Detector) yöntemini kullanmışlardır. Onishi ve diğerleri önermiş oldukları bu yöntem ile meyveleri %90 oranında doğru tespit edebildiklerini ve bir meyvenin 16 saniyede toplanabildiğini bildirmişlerdir.

Wang, Luo, Zhao, Tang ve Zou (2019), sebze ve meyvelerin değişken aydınlatma, tıkanıklık, yüzey rengi ve dokusu gibi kontrol edilemeyen faktörler nedeniyle tespit edilme zorluğuna karşılık F-RCNN (Faster Region Based Convolutional Networks) algoritmasına dayalı bir pencere yakınlaştırma tabanlı eşleştirme yöntemi önermişlerdir. Bu amaçla, veri seti için iki adet CCD görüntü sensörü ve bir tripodtan oluşan bir sistem kullanarak sırasıyla Ekim 2017'de narenciye, Haziran 2016'da litchi, Eylül 2018 tarihlerinde de biber ve patlıcan görüntüleri toplamışlardır. Veri setindeki görüntüler F-RCNN eğitim formatına dönüştürüldükten sonra ağ eğitilmiş ve normalleştirilmiş çapraz korelasyonun benzerlik

ölçümü yöntemine göre eşleştirme tamamlanmıştır. Önermiş oldukları yöntemin meyve ve sebze için arka plandan ayırabileceğini ve 6 farklı koşulda tanıma oranının %96,33 değerine ulaştığını öne sürmüşlerdir.

Kang ve Chen (2019), elma ve dalların gerçek zamanlı tespit edilmesi için Çin'in Qingdao şehrinde bulunan bir meyve bahçesinden Kinect-v2 kullanarak elde ettikleri veri seti ile çok işlevli bir derin evrişimli sinir ağı olan DaSNet (Detection and Segmentation Network) mimarisini önermişlerdir. Ağın özellik çıkarma yeteneğini geliştirmek için GFPN (Gated Feature Pyramid Network) ve ASPP (Atrous Spatial Pyramid Pooling) kullanmışlardır. Ağın hesaplama hızını arttırmak için ise artık ağ mimarisine dayalı lw-net (lightweight-net) geliştirmişlerdir. Önermiş oldukları lw-net'li DaSNet'in iyi bir algılama ve segmentasyon performansı ve elmaların tespitinde 0,821 oranında F1 puanı elde ettiğini belirtmişlerdir.

Koirala, Walsh, Wang ve McCarthy (2019), bir çiftlik arazi aracına monte edilmiş 5 MP- RGB kamera ve 720 W LED gıda aydınlatması kullanarak gece vakti meyve bahçesinden mango meyvesi görüntüsü toplamışlardır. Elde ettikleri veri seti ile mevcut derin öğrenme mimarilerinden altı tanesinin (Faster R-CNN(VGG-Visual Geometry Group), Faster R-CNN(ZF-Zeiler ve Fergus), YOLOv3(You Only Look One-v3), YOLOv2, YOLOv2-tiny ve SSD) mango meyvesi tespit performanslarını karşılaştırmışlardır. YOLOv3 yalnız 512x512 piksel; Faster R-CNN(VGG), Faster R-CNN(ZF), YOLOv2, YOLOv2-tiny ve SSD ise hem orjinal hem de 512x512 piksel çözünürlük ile eğitilerek toplam on bir model elde etmişlerdir. Ayrıca YOLOv3 ve YOLOv2-tiny tabanlı 'MangoYOLO' mimarisini oluşturmuşlardır. Oluşturmuş oldukları MangoYOLO mimarisini; öncelikle COCO veri seti ile daha sonra 1300 görüntüden oluşan mango veri seti ve son olarak da literatürde yer alan bir çalışmanın gündüz çekilmiş görüntülerden oluşan veri setini kullanarak eğitmişlerdir. Yapmış oldukları çalışmaya göre MangoYOLO mimarisi bellek kullanımı, hız ve doğruluk açısından diğer mimarilerden (YOLOv3, YOLOv2-tiny, Faster R-CNN (VGG) ve Faster R-CNN (ZF)) daha üstün olduğunu belirtmiş ve meyve tespitinde 0,97'lik bir F1 puanı elde etmişlerdir.

Häni, Roy ve Isler (2020), Samsung Galaxy S4 cep telefonu kamerasını kullanarak Haziran 2015 ile Eylül 2016 tarihleri arasında Minnesota, Eden Prairie' deki Minnesota Üniversitesi Bahçe Bitkileri Araştırma Merkezi'nden elde ettikleri veri setini kullanarak bir elma meyvesi tespit ve sayım sistemi tasarlamışlardır. Elmaların tespit ve sayımı için üretilen tüm veri noktalarının, sonlu Gauss dağılımlarının bir karışımından türetilen bir model olan GMM

(Gaussian Mixture Model), U-Net¹ ve F-RCNN olmak üzere üç farklı eğitim modelini karşılaştırmışlardır. Yapmış oldukları çalışmaya göre Gauss Karışım modeli meyvenin tespiti için diğer yöntemlerden daha iyi sonuç vermiştir. Meyve sayımında ise derin öğrenmeye dayalı yöntemler tüm veri setlerinde daha iyi performans göstermiştir. Dolayısıyla bu iki yöntemi birleştirmiş ve %95,56 ile %97,83 arasında değişim gösteren verim tahmini doğruluğu elde etmişlerdir.

Jia ve arkadaşları (2020), elma hasat robotlarının hasat verimliliğini arttırmak amacıyla Jiangsu eyaleti, Fengxian ilçesi, Dashahe Kasabasındaki elma üretimi deneme bahçesi ve Jiangsu Üniversitesi elma hasadı robot deneyi bölümünden (robot experiment bases) oluşturdukları veri seti ile darbe bağlantılı sinir ağı (Pulse-CNN) ve genetik Elman sinir ağına (GA Elman) dayalı bir elma tanıma yöntemi önermişlerdir. Veri seti için toplanan görüntüler bilgisayara bağlı bir AFT-0814MP kamera ile güneşli, bulutlu ve arkadan aydınlatma gibi farklı koşullarda elde edilmiştir. Önermiş oldukları yöntemde, Pulse-CNN görüntüler parçalara ayrılır ve sırasıyla R, G, B, H, S ve I olmak üzere altı renk özelliği, dairesel varyans, yoğunluk, çevre kare oranının 10 şekil özelliği ve parçalanmış görüntülerin Hu değişmez momenti ile birlikte 16 öznitelik Elman sinir ağına giriş olarak verilmiştir. Elman sinir ağının genetik algoritma ile optimize edilmesiyle, ağın yakınsama hızının ve genelleştirme yeteneğinin geliştiğini göstermişlerdir. Sinir ağlarının bağlantı ağırlıklarını ve yapısını optimize etmek için önerilen GA-Elman algoritması ile tanıma oranının %94,88'e ulaştığını belirtmişlerdir.

Kuznetsova, Maleva ve Soloviev (2020), Nikon Nikkor AF-P DXF 18-55mm ve Nikon D3500 AF-S 18-140 mm fotoğraf makineleri ile bir elma bahçesinden (All-Russian Fruit Crop Breeding Araştırma Enstitüsü, Zhilina, Oryol Bölgesi, Rusya) 2019 hasat sezonunda VIM merkezi çalışanları tarafından çekilen, kırmızı ve yeşil elma resimlerinden oluşan bir veri seti oluşturarak bir meyve tespit sistemi gerçekleştirmişlerdir. Bu amaçla histogram normalleştirme, sınırların kalınlaştırılması, medyan filtre kullanarak bulanıklaştırma gibi ön işleme; yuvarlak boşluklar ve sarı yapraklar gibi yanlış tespit edilen nesnelere elemek için büyük kenarının küçük kenarına oranı 3'ten fazla olan nesnelere çıkarılması gibi son işlemler uygulayarak bir YOLOv3 algoritması geliştirmişlerdir. Yapmış oldukları çalışmaya

¹ *U-NET, Freiburg Üniversitesi Bilgisayar Bilimleri Bölümü'nde biyomedikal alanlardaki görüntü işleme çalışmalarında bölümlenme yapmak için geliştirilmiş bir evrimsel sinir ağıdır.

göre ön ve son işlemlerle YOLOv3 algoritmasında elma sanılan nesnelere oranı %7,8 olurken tanınmayan elmaların oranı %9,2 olarak bulunmuştur. Ancak algoritmaya ön-son işleme teknikleri eklendikten sonra meyvelerin doğru tespit oranı %90,8'e yükselirken, tespit hızı 46 milisaniyeden 7 milisaniyelere düşmüştür.

Xu, Jia, Sun, Liu ve Cui (2020), yapay zekâ (AI) özellikli Hikvision IR kamera kullanarak Çin'in Hainan kentinde bulunan Changjiang mango ekim üssünden Temmuz 2019 yılında yeşil mango meyvesi görüntüleri toplayarak bir yeşil mango algılama sistemi önermişlerdir. Önermiş oldukları sistem ile, YOLOv3 ile kendi tasarlamış oldukları Light-YOLOv3 algoritmalarının performanslarını karşılaştırmışlardır. Geliştirmiş oldukları Light-YOLOv3 modelinde mangoların üst üste örtüşme problemini çözmek için NMS (Non-Maximum Supression) yerine Soft-NMS algoritmasını kullanmışlardır. Yapmış oldukları çalışmada Light-YOLOv3' ün daha az bellek kullanımı, daha yüksek doğruluk (%97,7 F1 skoru), daha hızlı algılama ve karmaşık şartlarda bile yeşil mangoları tespit edebilme gibi avantajları olduğunu belirtmişlerdir.

Xu, Jia, Liu, Zhao ve Sun (2020), iPad Pro 11 (12 megapiksel) kamerası ile 22 Temmuz ve 25 Ağustos tarihleri arasında Çin'in Qingdao kentinde bulunan tarım deneme bahçesinden gündüz, akşam ve gece saatlerinde elde ettikleri resimleri kullanarak bir veri seti oluşturmuşlardır. Bu veri seti ile geliştirilmiş YOLOv3-tiny algoritmasını kullanarak domates meyvesi algılama yöntemi önermişlerdir. Modelin eğitim ve tespit aşamasında önermiş oldukları görüntü iyileştirme algoritması ile yüksek algılama hızı ve algılama hassasiyeti (%91,92 F1 skoru) elde ettiklerini belirtmişlerdir.

Afonso ve arkadaşları (2020), Enza Zaden'in Hollanda Enkhuizen'deki üretim serasında, sera boyunca hareket eden bir arabanın farklı yükseklikteki noktalarına yerleştirilmiş 4 adet Intel Realsense D435 kamera ile Mayıs ve Haziran 2019 tarihleri arasında gece almış oldukları görüntülerden oluşan veri setini kullanarak Mask-RCNN (Mask Region Convolutional Neural Network) algoritmasına dayalı bir domates algılama yöntemi önermişlerdir. Araştırmacılar arasından bir grup, etiketleme aracı (Labelme) ile 123 görüntüye ekledikleri açıklamaları COCO (Microsoft Common Objects) formatına göre JSON dosyasında birleştirmişlerdir. Daha sonra meyveleri olgunlaşmış ve olgunlaşmamış olarak iki temel sınıfa ayırabilmek için Matlab ortamında son işleme adımı uygulamışlardır. Elde etmiş oldukları görüntü verilerini eğitim (üçte ikisi) ve test seti (üçte bir) olarak ayırarak bir derin

öğrenme algoritması olan Mask-RCNN ağını eğitmişlerdir. Afonso ve diğerleri yapmış oldukları hasat ve verim tahmini gibi pratik uygulamalarla ilgilenen bu çalışma ile domateslerin doğru bir şekilde tespit edilebileceğini öne sürmüşlerdir.

Gai, Chen ve Yuan (2021), 3000x4000 piksel Sony DSC HX400 ve Huawei cep telefonu kameraları ile Çin'de bulunan Dalian bölgesinden 2016-2019 yılları arasında güneşli günlerde çekilmiş kiraz resimlerini kullanarak bir kiraz tespit modeli oluşturmuşlardır. Elde etmiş oldukları veri seti ile YOLOv3, YOLOv3-Dense ve YOLOv4 gibi farklı algoritmaları eğiterek bu algoritmaların performanslarını karşılaştırmışlardır. Deneysel çalışmalarının sonuçlarına göre YOLOv4'ün doğruluk ve eğitim hızının diğer iki algortmadan daha iyi olduğunu ve YOLOv4'ün daha çok kiraz tespit edebildiğini belirtmişlerdir. Yapmış oldukları bu çalışmada kiraz meyvesinin tespit doğruluğunu iyileştirmek için YOLOv4 ve ağ seviyesini derinleştiren DenseNet ağ modellerinin kombinasyonundan oluşan bir model önermişlerdir. Önermiş oldukları YOLOv4-Dense modelinin olgun, yarı olgun ve olgunlaşmamış meyveleri doğru bir şekilde tespit ettiğini ve 0,947'lik bir F1 skoru elde ettiklerini belirtmişlerdir.

Li ve arkadaşları (2021), longan (ejderha gözü) meyvesinin insansız hava araçları (İHA) ile toplanmasını sağlamak amacıyla dallar üzerindeki uygun toplama noktalarını hızlı ve doğru bir şekilde tespit edebilmek ve konumlandırmak amacıyla derin öğrenme tabanlı bir tespit modeli önermişlerdir. Eğitim ve test için gerekli veri setini oluşturmak amacıyla İHA'yı renkli ve kızılötesi kameradan oluşan Intel Realsense D455 kamera ile donatarak Temmuz 2020 döneminde Guangzhou'daki Guangdong Tarım Bilimleri Akademisi'nin Longan Germplasm Kaynak Fidanlığından bulutlu ve güneşli günlerde çeşitli görüntüler almışlardır. YOLOv4'ün İHA bellek işgali, hesaplama ve algılama süresinin azaltılması gibi performanslarını iyileştirmek için özellik çıkarma algoritması olarak light-MobileNet kullanılmıştır. Hesaplama ve parametre sayısını daha da azaltabilmek için MobileNet-SSD ve MobileNet-YOLOv4 modellerinin performansları karşılaştırılmıştır. Test seti klasik piramit ağı (FPN), YOLOv3 ve YOLOv4 ve önerilen model üzerinde denendiğinde önerilen MobileNet-YOLOv4 algoritmasının modelin hesaplamasını, parametrelerini ve tespit süresini azalttığını öne sürmüşlerdir.

Junos, Mohd Khairuddin, Thannirmalai ve Dahari (2021), bir insansız hava aracına (Tello İHA) takılan CMOS 5Mp kamera ve 12 MP mobil kamera ile Malezya Selangor'da bulunan

Sime Darby palmye yağı tarlasından hasat mevsiminde toplamış oldukları veri setini kullanarak bir yağ palmyesi meyvesi tespit modeli önermişlerdir. Veri seti, yere düşmüş (serbest) yağ palmyesinin güneşli ve bulutlu havalarda, tek, yoğun örtüşme ve dağınık koşullarda çekilmiş görüntülerinden oluşturulmuştur. Junos ve diğerleri meyve tespiti için öznelik çıkarıcı olarak yoğun bağlantılı sinir ağı (DenseNet), aktivasyon fonksiyonu olarak Swish fonksiyonu ve çok ölçekli hedef tespiti gibi bileşenlerden oluşan YOLOv3-tiny modelini önermişlerdir. Önermiş oldukları bu yöntem ile 0,98 F1 skoru ve algılama süresi bakımından YOLOv3 modelinden daha hızlı bir performans elde ettiklerini belirtmişlerdir.

Hasat robotlarının meyveleri hasat edebilmesi için meyvelerin varlığını tespit etmenin yanı sıra meyvenin robot kola olan uzaklığının da bilinmesi gerekmektedir.

Hemming, Bac, Van Tuijl, Barth, Bontsema, Pekkeriet ve Van Henten (2014), tatlı biberlerin hasat robotunda biberlerin koordinatlarını tespit edebilmek için iki adet 5 megapiksel renkli kamera ve ToF (Uçuş Süresi, Time of Flight) kamerası kullanmışlardır. Renkli kameralardan elde edilen görüntü, ToF kamerasının görüntüsü ile piksel olarak eşleştirilmiştir. Böylece renkli kameradan alınan renk verilerine karşılık gelen 3B ölçümleri elde edilmiştir.

Fernández, Salinas, Montes ve Sarria (2014), doğal senaryolarda farklı bitki türlerinden meyveleri (üzüm ve kırmızı elmada test edilmiş) tespit etmek için bir AVT Prosilica C2450 yüksek çözünürlüklü CCD renkli kamera, bir multispektral görüntüleme sistemi ve bir Mesa SwissRanger SR-400011 ToF 3D kameradan oluşan çok duyu bir sistem önermişlerdir. Önermiş oldukları sistemde ToF 3D kamera ile hedeflerin doğru mesafe ve yoğunluk görüntülerinin eşzamanlı olarak hızlı bir şekilde alınmasını sağlayarak meyvelerin koordinat uzayında lokalizasyonu mümkün hale gelmiştir. Peebles, Barnett, Duke ve Lim (2020), kuşkonmaz bitkisinin hasadı için RGB kamera (Basler ACE) ile birlikte ToF (Microsoft Kinect V2) kamera kullanmışlardır.

3. MATERYAL VE YÖNTEM

Bu çalışmada, YOLO (You Only Look One) derin öğrenme algoritması ve evrişimsel sinir ağları (Convolutional Neural Network-CNN) algoritması kullanılarak gerçek ortamda farklı zaman dilimlerinde ve doğal ışık altında zeytin meyvesinin tespiti gerçekleştirilmiştir.

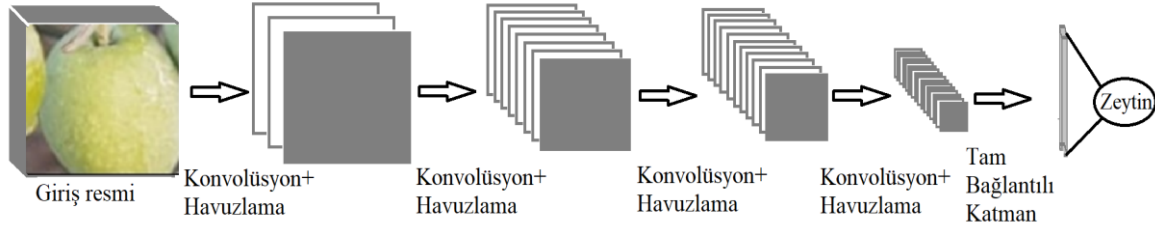
3.1. Veri Seti

Veri setini oluşturmak üzere 2020 yılının Ekim ayında Hatay ilinin Defne ilçesinde bulunan zeytin bahçesinde 12 MP Optik görüntü sabitleyiciye (OIS) sahip kamera ile 1,3 MP CMOS fotoğraf sensörüne sahip IP kamera kullanılarak 30s - 7dk arasındaki farklı sürelerle çekilmiş 18 adet video görüntüsü kaydedilmiştir. Daha sonra bu video görüntüleri OpenCV kütüphanesi kullanılarak resimlere dönüştürülmüş ve toplamda 13170 adet resim elde edilmiştir. Güvenlik kamerası görüntülerinden elde edilen resimler 1280x960 iken cep telefonu kamerasından elde edilen resimler ise 4032x2268 piksel çözünürlüğe sahiptir. Elde edilen resimler arasından, gün ışığı, güneş ışığı, yapay ışık (flaş) gibi farklı ortam özelliklerine sahip ve içerisinde yeşil zeytin meyvesi bulunan toplam 1143 tanesi çalışmalarda kullanılmak üzere seçilmiştir.

3.2. Evrişimsel Sinir Ağları

Evrişimsel Sinir Ağı, derin öğrenme algoritmaları arasında en popüler olan çok katmanlı bir yapay sinir ağıdır. Ağ, ismini konvolüsyon matrisleri arasındaki matematiksel doğrusal işlemlerden almaktadır. Bir CNN ağında doğrusal olmayan katman, havuz katmanı, tam bağlantılı katman ve evrişim (konvolüsyon) katmanı gibi katmanlar bulunmaktadır. Evrişimli ve tam bağlantılı katmanların parametreleri bulunurken havuzlama ve doğrusal olmayan katmanlarda parametre yoktur. CNN yapısında “soyut özellikler, girdi daha derin katmanlara yayıldıkça elde edilmektedir. Yani, nesne sınıflandırmasında ilk katmanlarda kenarlar, ikinci katmanlarda daha basit şekiller ve sonrasında da daha üst seviyede özellikler algılanabilmektedir. Ancak nesne tespitinde nesnelere görüntülerdeki konumlarından bağımsız olarak tespit edilmektedir. CNN algoritmasında görüntünün tamamı yerine resimde yerel bölgeler aranmaktadır. Yani bir sonraki katmandaki gizli nöronlar, yalnızca önceki katmanın karşılık gelen kısmından giriş almaktadır. Ses gibi sıralı verilerde tek

boyutlu, resimlerde iki boyutlu, video veya 3D görüntüler için ise üç boyutlu bir evrişim kullanılabilir.



Şekil 3.1. CNN modelinin yapısı

3.2.1. Evrişimsel sinir ağının elemanları

Konvolüsyon:

32x32x3 genişlik, yükseklik ve RGB kanal derinliğine sahip bir veri seti için ağın ham pikselleri girdi olarak aldığı varsayılırsa, giriş katmanını sadece bir nörona bağlamak için $32 \times 32 \times 3$ ağırlıklı bağlantıların olması gerekir. Gizli katmana bir nöron daha eklenirse $32 \times 32 \times 3$ ağırlıklı bir bağlantıya daha ihtiyaç olacağından girişi yalnızca iki düğüme bağlayabilmek için 6000'den fazla ağırlık parametresi gerekecektir. Bir görüntü sınıflandırma uygulamasının iyi sonuç verebilmesi için iki nöronun yeterli olmayacağı düşünülürse, uygulamayı daha verimli hale getirmek için giriş görüntüsü bir sonraki katmanda bulunan nöronlara tamamen aynı yükseklik ve genişlik değerleriyle bağlanabilir. Örneğin, tam bağlantı yerine 5x5 lik bir nörona bağlanırsa ve sonraki katmanda 32x32 nörona sahip olmak isteniyorsa 3.145.728 yerine 76.800 ($5 \times 5 \times 3 \times 32 \times 32$) bağlantı elde edilir. Bağlantı boyutu büyük ölçüde azalmış olsa da çözülmesi gereken çok fazla parametre gerektiğinden daha da sadeleştirmek amacıyla bir sonraki katmanın tüm nöronları için yerel bağlantı ağırlıkları sabit tutulmaktadır. Dolayısıyla bir sonraki katmandaki komşu nöronlar, bir önceki katmanın yerel bölgesine tamamen aynı ağırlıkta bağlanır. Böylece bir sonraki katmanda 32x32x3 nöronu 32x32'ye bağlamak için ağırlık sayısı $5 \times 5 \times 3 = 75$ 'e düşmüş olur. Verilen örnekten anlaşılacağı gibi bu yöntem ile hem bağlantı sayısı 3 milyondan 75 bağlantıya düşmüş hem de yerel bağlantılar için ağırlıkları sabitlemek, giriş nöronlarında $5 \times 5 \times 3$ 'lük bir pencereyi kaydırmaya ve üretilen çıktıyı karşılık gelen yere eşlemeye denk olmuştur. Özellikleri, görüntüdeki konumlarından bağımsız olarak algılama ve tanıma olanağı sunduğundan konvolüsyon olarak adlandırılmaktadır.

Kaydırma Adımı (Stride):

Bölgelerin, bir sonraki katmanın düğümünün komşularıyla örtüşmesi gibi yan etkilerin kontrol altına alınması ve CNN parametrelerinin daha da azaltılması adım kontrolü ile yapılmaktadır. Adım kaydırma doğrudan çıkış boyutunu etkileyen bir parametredir.

Doldurma (Padding):

Konvolüsyon adımının dezavantajlarından biri, sadece filtre kaydıgında yakalandıkları için, görüntünün sınırında bulunan bilgilerin kaybolmasıdır. Bu durumu önlemek için sıfır doldurma (piksel ekleme) işlemi yapılmaktadır. Sıfır doldurma işleminin diğer bir avantajı da çıkış boyutunun yönetilebilir olmasını sağlamaktır. Ayrıca, ağ çıktısının boyutunun derinlikle küçülmesini önleyerek herhangi bir sayıda derin evrışimsel ağ elde edebilmeyi sağlamaktadır.

Doğrusal Olmayan Katman (Non-linear Layer):

Konvülyasyondan sonra gelen doğrusal olmayan katman üretilen çıkışı doyum noktasına getirmek veya sınırlamak için uygulanmaktadır. Bu amaçla kullanılan aktivasyon fonksiyonlarından bazıları ReLU (Rectified Linear Unit), tanh ve Sigmoid'tir. Sigmoid ve tanh gibi doymuş fonksiyonlar geri yayımda problemlere neden olduğu gibi, sinir ağı daha derin tasarlandığında "kaybolan gradyan" olarak adlandırılan gradyan sinyali de kaybolmaya başlamaktadır. Ayrıca Sigmoid ve tanh, eğimden her zaman sıfır olmayan sonuçlara sahip olduklarından eğitim için uygun değildirler. Ancak, ReLU'nun hem fonksiyon hem de gradyan olarak daha basit tanımlara sahip olması, türevlenebilir olmasa da pozitif giriş için sabit bir eğimi olması ve gradyandaki sıfırın, tam bir sıfır elde edilmesine yol açması sebebiyle son zamanlarda Sigmoid ve tanh fonksiyonlarından daha çok kullanılmaktadır. (Albawi, Mohammed ve Al-Zawi, 2017).

Havuzlama (Pooling):

Görüntü işleme alanındaki çözünürlüğün azaltılmasına benzeyen havuzlama katmanı, ileri katmanlar için karmaşıklığı azaltmak amacıyla alt-örnekleme yapar ve filtre sayısını

etkilemez. En yaygın kullanılan havuzlama, genellikle 2x2 boyutunda kullanılan maksimum havuzlamadır. Maksimum havuzlama görüntüyü alt-bölge dikdörtgenlerine böler ve yalnızca o alt-bölgenin içinin maksimum değerini döndürür. Ancak alt-örnekleme bilginin konumunu korumadığından, mekansal bilgidен ziyade bilginin varlığının önemli olduğu durumlarda uygulanmalıdır. Havuzlama, verimliliği arttırmak için, eşit olmayan filtreler ve adımlarla kullanılabilir.

Tam Bağlantılı Katman (Fully-Connected Layer):

Bu katmanın yapısı, nöronların klasik bir sinir ağına benzer. Yani her düğüm hem önceki hem de sonraki her düğüme doğrudan bağlıdır. Havuzlama katmanın son çerçevelerindeki düğümlerin her biri, tam bağlı katmandan ilk katmana vektör olarak bağlanır. Eğitim örneklerinde karmaşık hesaplama gerektiren birçok parametre içermesi, bu katmanın en önemli dezavantajıdır.

3.2.2. Evrişimsel sinir ağları ile zeytin meyvesinin tespiti

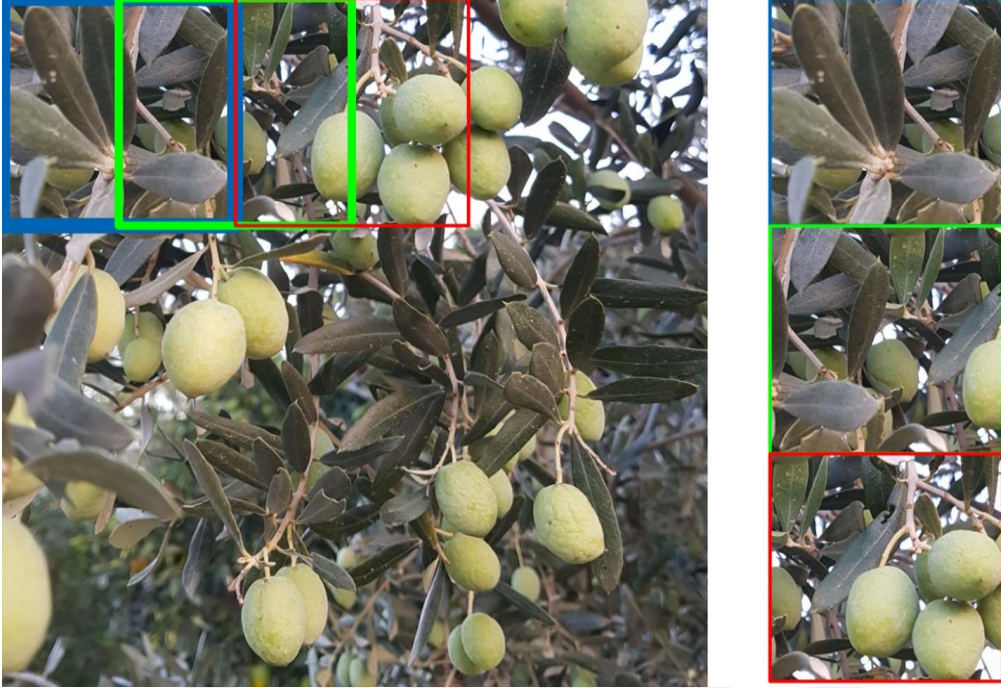
Ağ için gerekli veri setinin oluşturulması:

Cep telefonu ve güvenlik kamerası video görüntülerinden elde edilen 1143 adet resme Resim 3.1’de verilen örnekte olduğu gibi kayan pencere yöntemi uygulanmış ve resimler 300x300 piksel boyutunda kırılmıştır.

Kırpma işlemi sonucunda yaklaşık 33000 resim elde edilmiş ve aralarından sadece zeytin ya da sadece yaprak-dal içeren resimler seçilmiştir. Resimler zeytin (0) ve yapraklar (1) olarak iki sınıfa ayrılmıştır. Veri setinde toplam 3961 resim bulunmaktadır. Sınıflara ait veri dağılımları Çizelge 3.1’de verilmiştir. Resim 3.2’de ise veri setinde kullanılan resimlerden örnekler bulunmaktadır.

Çizelge 3.1. CNN modeline ait veri seti dağılımı

Zeytin (0)	Yaprak (1)
2024	1937



Resim 3.1. Kayan pencere yöntemi (150px kaydırılmış, 300x300px pencere)



Resim 3.2. CNN modeline ait veri setinde bulunan resim örnekler

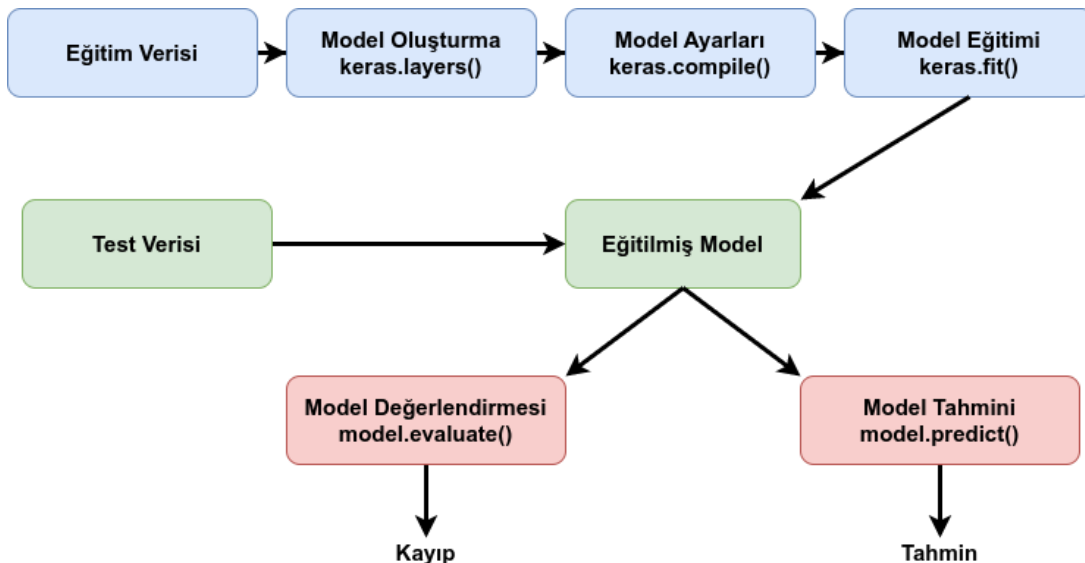
CNN Modelinin Eğitimi:

Yapay zekâ modelinin eğitimi; python yazılım dili ile OpenCV, Keras ve Scikit-Learn kütüphaneleri kullanılarak gerçekleştirilmiştir.

Scikit-Learn kütüphanesi; veri setini test, train ve validation olarak bölmek için kullanılmıştır. Testler 8GB RAM barındıran Intel Core i5 işlemcili bir sistem üzerinde gerçekleştirilmiştir. Eğitim ve test işlemleri için bir python dağıtım platformu olan Anaconda Navigator - Spyder uygulaması kullanılmıştır.

Keras kütüphanesi, python dili ile yazılmış açık kaynak kodlu bir derin öğrenme kütüphanesi olup TensorFlow ve Theano'ya üst katman olarak geliştirilmiştir. Modelleri eğitmek ve değerlendirmek bakımından basit bir iş akışı sağlayan kerasın akış diyagramı Şekil 3.2'de verilmiştir (Pashine, Dixit ve Kushwah, 2021). Keras kütüphanesinde, Conv2D, Dense, MaxPooling2D, Flatten, Dropout, BatchNormalization gibi katmanlar bulunmaktadır.

- Conv2D: Evrişim ağını temsil etmektedir.
- Flatten: Veri matrisini sınıflandırıcıya vermeden önce vektöre dönüştürerek düzleştirme işlemini yapar.
- Dropout: Ağın ezberlemesini önlemek için özneliklerin seyreltilmesini sağlar.
- Batch normalization: Evrişimsel sinir ağını daha düzenli hale getirir. Ayrıca sinir ağının eğitimi sırasında yok olma gradyanına direnç göstererek eğitim süresini azaltıp modelin daha iyi performans ile çalışmasını sağlayabilir.



Şekil 3.2. Keras akış diyagramı

Sınıflandırma işleminden önce resimlere piramit gösterimi, kayan pencere ve maksimum olmayan bastırma işlemleri uygulanmıştır.

- Maksimum Olmayan Bastırma (Non-Maximum Suppression):

Resim üzerinde bulunan sınırlayıcı kutular arasında bir birlik üzerinden kesişme (Intersection over Union) değeri Eş. 3.1 ile hesaplanır ve bir eşik değeri belirlenir. Bu eşik değerin altında kalan kutular elenerek her bir nesne için tek bir sınırlayıcı kutu elde edilir. Böylece sınıflandırma işlemi gerçekleştirilmiş olur.

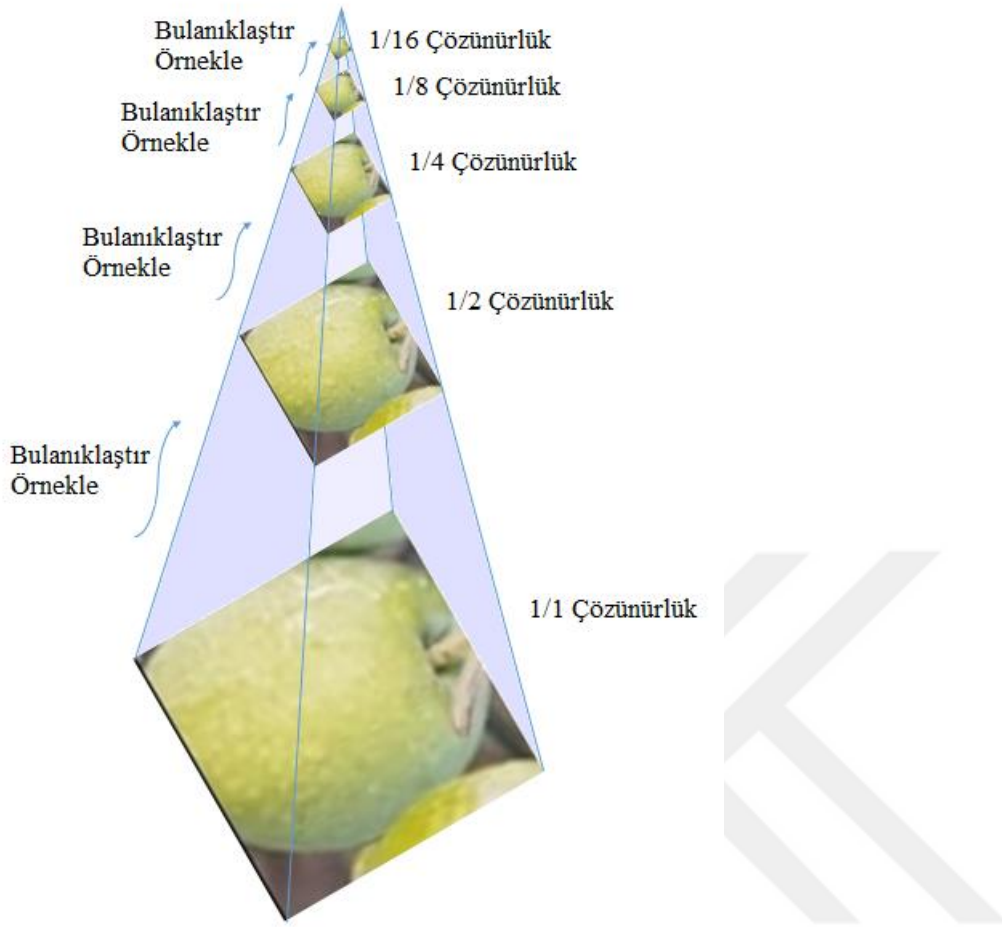
$$IoU = \frac{\text{Kesişim Alanı}}{\text{Birleşim Alanı}} \quad (3.1)$$


- Kayan pencere:

Sabit genişlikte ve yükseklikteki dikdörtgen bir bölgenin görüntü üzerinde kaymasına dayanır. Bu pencerelerin her biri için, pencere bölgesi alınır ve pencere içerisinde ilgilenilen nesnenin olup olmadığını belirlemek için bir görüntü sınıflandırıcı uygulanır.

- Piramit gösterimi:

Bir görüntünün farklı ölçeklerdeki görüntülerden nesnelerin bulunmasını sağlayan çok ölçekli temsildir. Tespit edilmek istenen nesnenin veya resmin boyutunun farklı olduğu durumlarda tanıma işlemi zorlaşır. Örneğin, bir nesneyi çok küçük çözünürlüklü bir resimden tespit etmek, büyük çözünürlüklü bir resimden tespit etmekten çok daha zordur. Piramidin en altında orijinal boyuttaki orijinal görüntü bulunur. Sonraki her katmanda görüntü belirli bir ölçekte yeniden boyutlandırılır ve isteğe bağlı olarak düzleştirilir.



Şekil 3.3. Piramit gösterimi

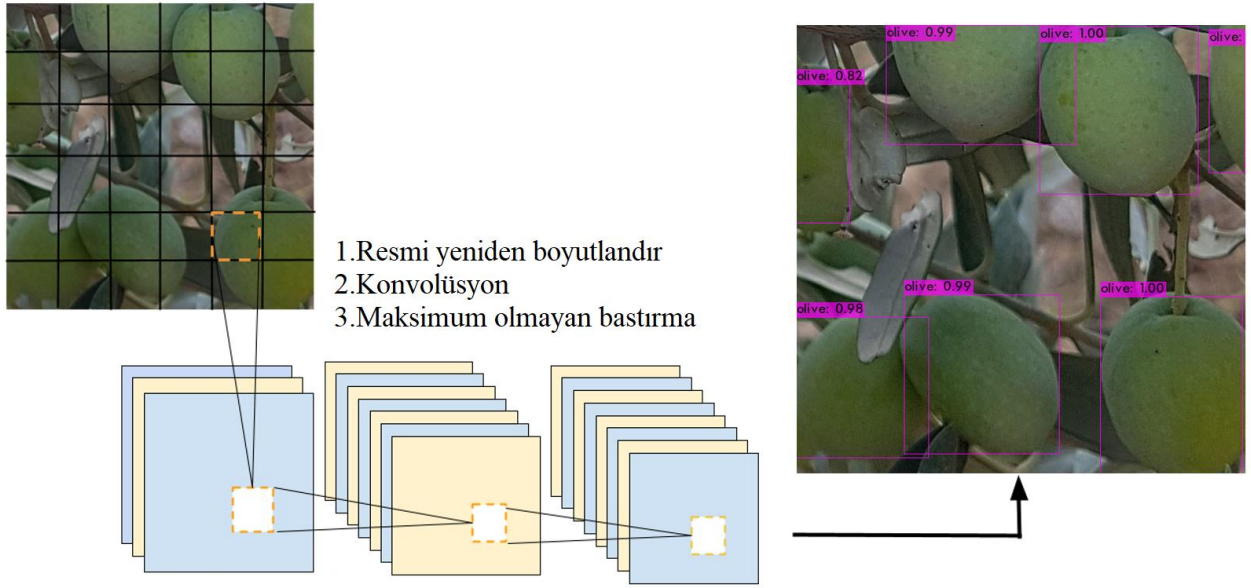
3.3. YOLO Mimarisi

YOLO (You Only Look One) mimarisi, nesnelerin algılanması amacıyla Redmon, Divvala, Girshick ve Farhadi (2016) tarafından önerilmiştir. Nesne algılama üzerine yapılmış önceki çalışmalarda algılamanın gerçekleşmesi için sınıflandırıcılar yeniden kullanılmaktadır. Bu algılama sistemlerinde nesneyi algılamak için bir sınıflandırıcı alınır ve sınıflandırıcı bir test görüntüsünde çeşitli konumlarda ve ölçeklerde değerlendirilir. Bazı mimarilerde sınıflandırıcı, tüm görüntü üzerinde eşit aralıklı konumlarda çalışan kayan pencere yaklaşımını kullanırken, R-CNN gibi yapılarda önce bir görüntüde potansiyel sınırlayıcı kutular oluşturulup sonrasında önerilen bu kutular üzerinde bir sınıflandırıcı çalıştırmak için bölge önerme yaklaşımı kullanılır. Sınıflandırma tamamlandıktan sonra sınırlayıcı kutuları iyileştirip birden fazla algılamayı ortadan kaldırmak ve görüntüdeki diğer nesnelere göre kutuları yeniden puanlamak için son-işleme kullanılır. Bu işlemler satırları yavaşlatır ve her bir bileşenin ayrı ayrı eğitilmesi gerektiği için optimize edilmesi zordur. Ayrıca nesnelerin görüntüdeki koordinatlarının belirlenmesi sınıflandırmadan daha karmaşık bir problemdir.

YOLO algoritması, görüntünün tamamına tek bir evrişimli sinir ağı uyguladıktan sonra, görüntüyü bölgelere ayırarak aynı anda birden fazla sınırlayıcı kutu ve bu kutulara ait sınıf olasılıklarını tahmin eder.

YOLO'da eğitim tam görüntüler üzerinden gerçekleşmekte ve algılama performansı doğrudan optimize edilmektedir. YOLO modelinin bazı avantajları aşağıda belirtildiği gibidir:

- YOLO son derece hızlıdır ve karmaşık bir veri hattına ihtiyaç duymaz.
- Test aşamasında algılama tahminleri için, sinir ağı yeni bir görüntü üzerinde çalıştırılır. Ayrıca yüksek ortalama hassasiyeti korurken uçtan uca eğitim ve gerçek zamanlı başarılarla olanak sağlar.
- YOLO, her bir sınırlayıcı kutuyu tahmin etmek için görüntünün tamamındaki özellikleri kullanır. Aynı zamanda bölge önerisi tabanlı ve kayan pencere gibi yöntemlerin aksine bir görüntü için tüm sınıflardaki tüm sınırlayıcı kutuları aynı anda tahmin eder. Bu da, ağı tahminlerde bulunurken tam görüntü ve görüntüdeki tüm nesnelere evrensel olarak akıl yürüttüğü anlamına gelir.
- Daha geniş bağlamı göremediği için bir görüntüdeki arka plan yamalarını nesnelere karıştıran yöntemlere kıyasla daha az arka plan hatası yapar.
- YOLO, nesnelere genelleştirilebilir temsillerini öğrenir ve kendisi de genelleştirilebilir olduğundan çökme olasılığı düşüktür.
- Doğal görüntüler üzerinde eğitilip sanat eserleri üzerinde test edildiğinde de diğer algılama yöntemlerinden (R-CNN gibi) daha iyi sonuçlar vermektedir (Redmon ve diğerleri, 2016).



Şekil 3.4. YOLO zeytin meyvesi tespit sistemi

YOLO mimarisinde giriş görüntüsü bir $S \times S$ ızgarasına bölünür. Bir nesnenin merkezi herhangi bir ızgara hücreğine düştüğünde, ızgara hücresi o nesneyi algılamakla yükümlüdür. Iızgara hücrelerinin her biri sınırlayıcı kutuları ve bu kutulara ilişkin güven puanlarını tahmin eder. Tahmin edilen güven puanları ağırlık, kutunun bir nesne içerdiğinden ne kadar emin olduğunu ve tahmininin doğruluk değerini yansıtır. Yani, hücrede herhangi bir nesne yoksa o hücrenin güven puanları sıfır olmalıdır.

Her sınırlayıcı kutu x , y , w , h ve güven parametreleri olmak üzere beş tahminden oluşur. x ve y koordinatları ızgara hücrelerinin sınırlarına göre kutunun merkezini, güven parametresi ise tahmin edilen kutu ile herhangi bir temel doğruluk kutusu arasındaki IOU (Intersection Over Union) değerini ifade eder. Genişlik (w) ve yükseklik (h) ise görüntünün tamamına göre tahmin edilir.

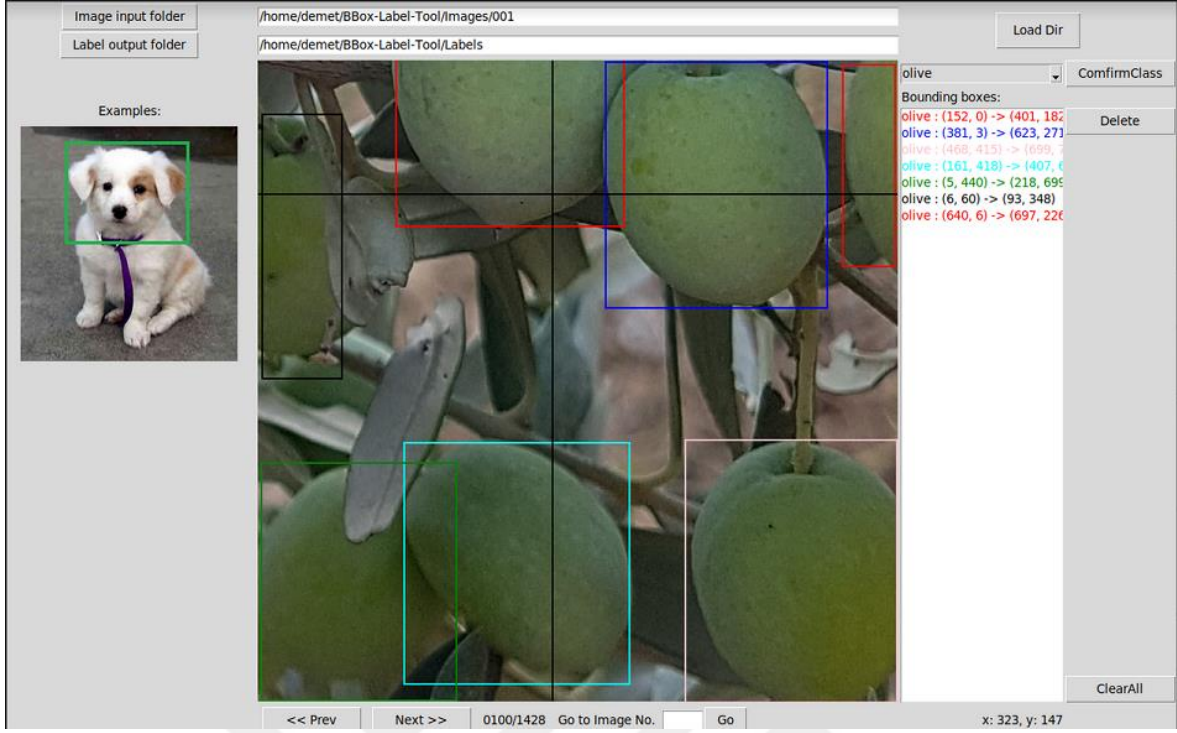
3.3.1. YOLO model eğitimi için meyve etiketlemesi

Veri setinde bulunan resimlerin çözünürlüklerinin birbirinden farklı olması nedeniyle, resimlere kayan pencere yöntemi (Resim 3.3) uygulanmış ve resimler 700×700 piksel boyutunda kırılmıştır. YOLO modellerinin eğitimi gerçekleştirilmek amacıyla 1413 adet resim üzerinde 8191 adet zeytin etiketlenmiştir. Buna göre veri setindeki resim başına düşen ortalama etiketlenmiş zeytin meyvesi sayısı 6 olarak hesaplanır.

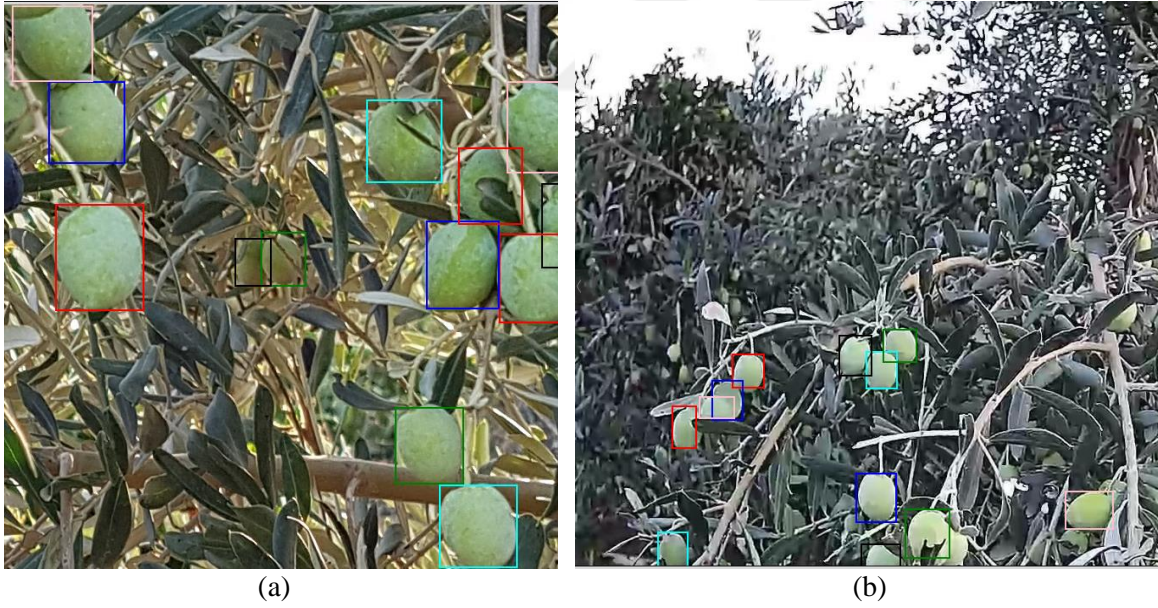


Resim 3.3. Kayan pencere yöntemi ile resimlerin kırılması

Veri seti oluşturulduktan sonra BBox-Label-Tool (Shi Qiu, 2014) etiketleme aracı kullanılarak resim üzerindeki zeytinler etiketlenmiş (Resim 3.4 ve Resim 3.5) ve her bir zeytinin resim dosyasındaki koordinatları 0-1 aralığına normalize edilerek bir metin dosyasına kaydedilmiştir. Elde edilen koordinat dosyalarının rastgele seçilen %10'luk bir kısmı test seti olarak ayrıldıktan sonra YOLO algoritmasının YOLOv3-tiny, YOLOv4-tiny-custom versiyonlarında eğitim gerçekleştirilmiştir.



Resim 3.4. BBox-Label-Tool etiketleme aracı arayüzü



Resim 3.5. BBox-Label-Tool ile etiketlenmiş görüntüler

3.3.2. YOLO model eğitimleri

YOLO eğitimi gerçekleştirilmeden önce algoritmada kullanılan ve Çizelge 3.2’te verilmiş olan parametreler tespit edilmek istenen sınıf sayısına göre düzenlenmiştir. Algoritmada güncellenmesine ihtiyaç duyulan parametreler:

- **Batch:** Modelin öğrenme aşamasında aynı anda kaç veriyi işleyeceğini belirleyen parametredir. Her iterasyonda batch değeri olarak belirlenen sayıda veri, grup halinde işlenir. Batch değerinin GPU belleğine sığması gerektiğinden 2’nin katı şekilde seçilmesi gerekmektedir.
- **Subdivision:** Subdivision parametresi batch parametresini mini-batch parçalarına böler. Batch= 64 ve Subdivision=16 olarak seçildiğinde mini-batch=64/16= 4 olur. Yani 4 adet görüntü işlenmek üzere GPU ya gönderilir. Bu işlem 16 defa tekrarlandıktan sonra bir iterasyon tamamlanmış olur ve sonraki iterasyona geçilir.
- **Max_batches:** Değeri Eş. 3.2 ile bulunur. Ancak, hesaplanan değer (max_batches) “eğitimde kullanılacak resim sayısı” ndan veya “6000” den küçük olursa en az “6000” seçilmelidir.

$$max_batches = classes * 2000 \quad (3.2)$$

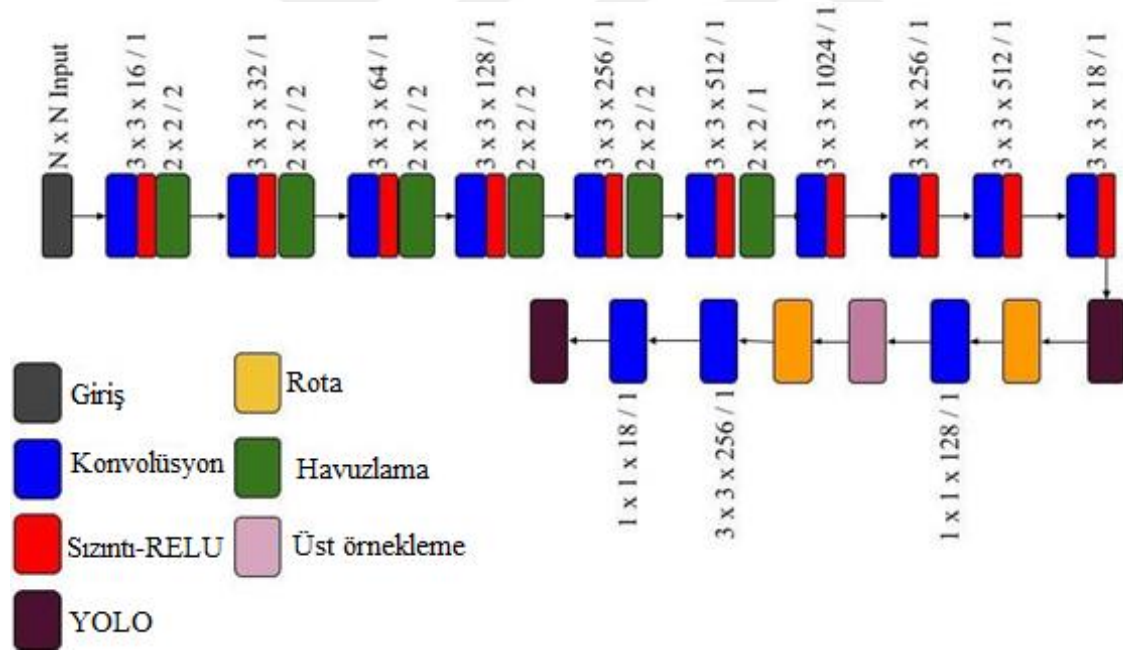
- **Steps:** Alt değer max_batches değerinin 80%’i üst değer ise 90%’i olacak şekilde hesaplanır.
- **Classes:** Sınıf sayısı her [yolo] katmanında bulunur ve tespit edilmek istenen sınıf sayısına göre güncellenir. Yapmış olduğumuz çalışmada sadece zeytin meyvesi tespiti yaptığımız için sınıf sayımız 1’dir.
- **Filters:** Her [yolo] katmanından bir önceki [convolutional] katmanında yer alan Eş. 3.3’e göre güncellenir.

$$filters = (classes + 5) * 3 \quad (3.3)$$

Çizelge 3.2. Güncellenen YOLO parametreleri

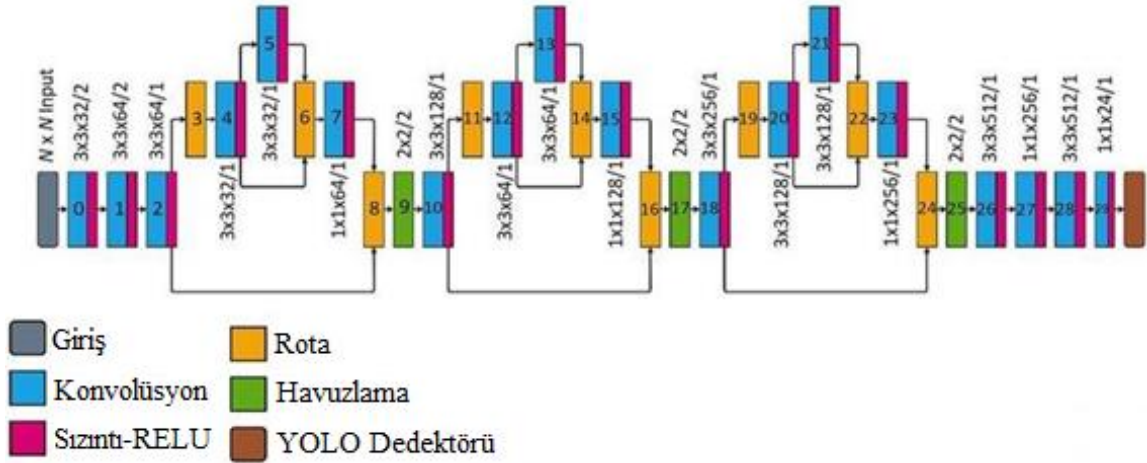
Parametre	Değeri
Batch	64
Subdivision	16
Max_batches	15000
Steps	12000, 13500
Classes	1
Filters	18

YOLO derin öğrenme modellerinde özellik çıkarıcı olarak Darknet ağ mimarisi kullanılmaktadır.



Şekil 3.5 YOLOv3-tiny mimarisi

Çalışmada kullanılan YOLOv3-tiny mimarisi Şekil 3.5, YOLOv4-tiny mimarisi ise Şekil 3.6 da verilmiştir (Montalbo, 2020).



Şekil 3.6 YOLOv4-tiny mimarisi

3.4. Kullanılan Nesne Tespit Modellerinin Performans Ölçme Parametreleri

Nesne tespit modellerinin eğitim ve test aşamaları gerçekleştirildikten sonra performanslarını karşılaştırmak için öncelikle modelin kesinlik (precision), doğruluk (accuracy) ve hassasiyet (recall) değerleri hesaplanır. Kesinlik değeri, tespit edilen toplam meyvelerden doğru tespit edilen meyve sayısını verir. Doğruluk değeri, modelin doğru tespit edilen meyve sayısının toplam veri kümesine oranıdır. Hassasiyet değeri, resimlerde bulunan gerçek meyvelerin toplamından doğru tespit edilen meyve sayısını verir. Kesinlik ve hassasiyet değerlerinin harmonik ortalaması ise modelin doğruluk ölçüsü olan F1 skoru değerini verir (Eş. 3.6).

Doğru Pozitifler (True Positives-TP): Model tarafından doğru sınıflandırılan meyve sayısıdır.

Yanlış Pozitifler (False Positives-FP): Model tarafından yanlış sınıflandırılan meyve sayısıdır.

Yanlış Negatifler (False Negatives-FN): Model tarafından sınıflandırılmayan meyve sayısıdır.

$$\text{Hassasiyet} = \frac{TP}{TP + FN} \quad (3.4)$$

$$Kesinlik = \frac{TP}{TP + FP} \quad (3.5)$$

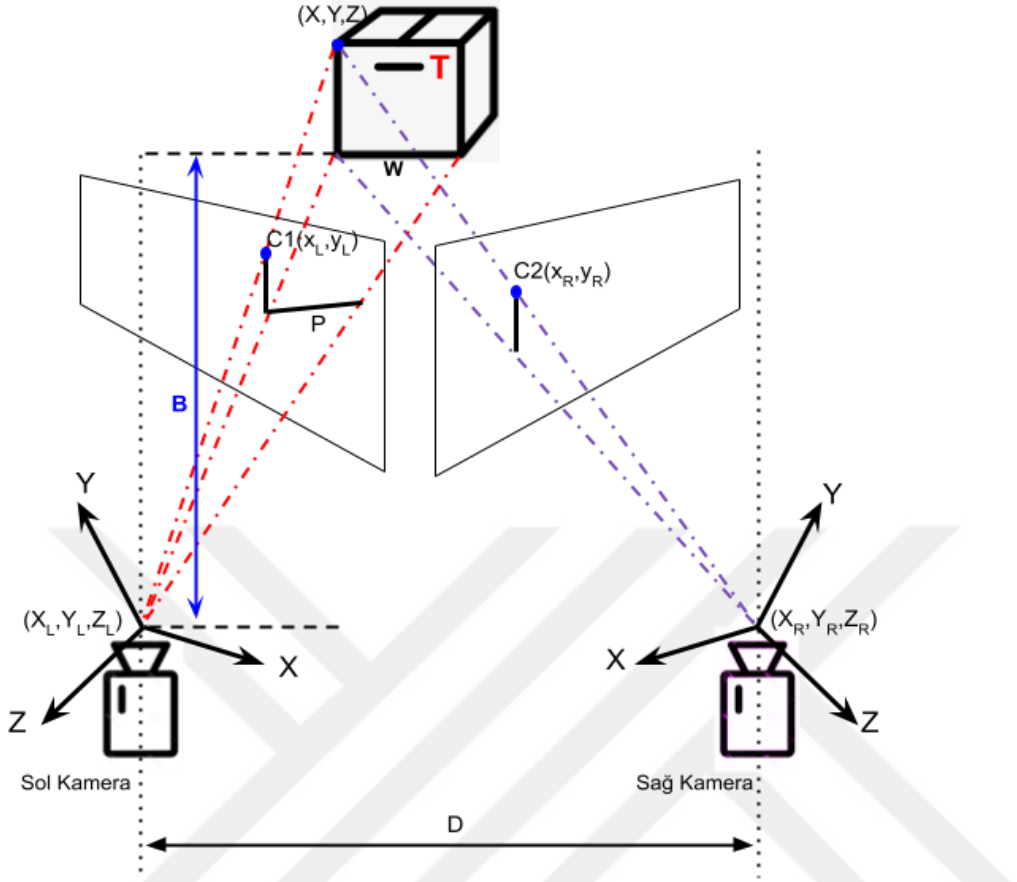
$$F1 = 2 * Precision * \frac{Recall}{Precision + Recall} \quad (3.6)$$

3.5. Zeytin Meyvesinin Uzaklığının Tespiti

Meyvelerin hasat edilmesinde ilk aşama meyvenin tespit edilerek konumunun belirlenmesi, ikinci aşama ise, robot kolun meyveye yönelerek hasat işlemini gerçekleştirmesidir. Hasat işleminin gerçekleştirilmesi için meyvenin koordinatlarının yanı sıra derinlik bilgisinin de doğru olarak hesaplanması gerekmektedir. Literatürdeki benzer çalışmalar dikkate alındığında derinlik bilgisinin tespitine yönelik farklı uygulamaların gerçekleştirilmiş olduğu görülmektedir. Örneğin, kızılötesi (IR) ışık yayan bir kaynaktan gönderilen ışığın nesnelere çarpıp yansması için geçen sürenin hesaplanmasına dayanan ToF (Uçuş Süresi, Time of Flight) kamera kullanımı (Fernández, Salinas, Montes ve Sarria, 2014; Xu, Jia, Sun, Liu ve Cui, 2020; Arad ve diğerleri, 2020), birbirine mesafeli olarak yerleştirilen iki kameradan alınan görüntülerin açılış farkları üzerinden derinlik hesaplanmasına dayanan uygulamalar (Onishi ve arkadaşları, 2019) yaygın şekilde kullanılmaktadır.

Çalışmanın bu bölümünde bir objenin uzaklığının farklı kameralarla tespiti konusu incelenmiştir:

- ToF (Uçuş Süresi, Time of Flight) kameralar, lazer gibi bir ışık kaynağı kullanarak yüksek kare hızında 3 boyutlu koordinat ve genlik görüntüleme ile aynı anda her piksel için yoğunluk verileri sağlayan bir sensör türüdür (Foix, Alenya ve Torras, 2011). Bu tür sistemlerde ışık kaynağının sistem tarafından sağlanıyor olması düşük ortam aydınlatmalarında bile derinlik bilgisinin doğru olarak elde edilmesine imkân sağlamaktadır.
- Literatürde kullanılan diğer bir yöntem ise stereo kameralar ile derinlik tespitinin triangulation (üçgenleme) (Mattocchia, 2011) ile gerçekleştirilmesidir. Bu yönteme ilişkin düzenek Şekil 3.7 ile gösterilmiştir.



Şekil 3.7. Stereo kamera ile derinlik tespit düzeneği

Şekil 3.7 ile verilen T nesnesinin derinlik (Z) bilgisi Eş. 3.7 ile hesaplanmaktadır.

$$Z = F * \frac{D}{X_L - X_R} \quad (3.7)$$

Burada;

D = Birbirine paralel olan sağ ve sol kameraların aralarındaki uzaklık,

X_L = T nesnesinin sol kamera ile çekilmiş resim üzerindeki x koordinatı (XL),

X_R = T nesnesinin sağ kamera ile çekilmiş resim üzerindeki x koordinatı (XR),

F= Kameraların odak uzaklığıdır (Kameraların aynı özellikte olduğu varsayılmıştır).

Kameranın odak uzaklığı (F) tasarımsal bir parametredir. Bu nedenle kameraların odak uzaklıklarının hesaplanabilmesi için bir kameradan kontrollü olarak elde edilen görüntü yeterli olmaktadır. Parametrenin elde edilebilmesi için Eş. 3.8 kullanılmaktadır. T nesnesinin

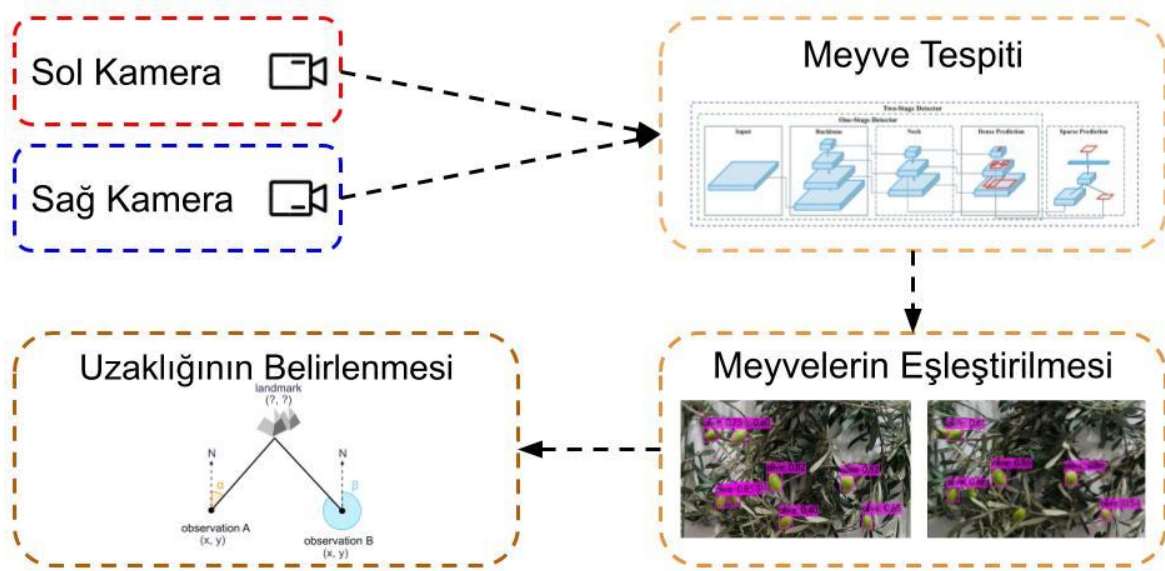
gerçek genişlik değeri (W) ve kamera tarafından elde edilen görüntüdeki genişlik değeri (P) ve nesnenin kameraya olan uzaklığı (B) dikkate alınır.

$$F = P * \frac{B}{W} \quad (3.8)$$

W = Referans olarak alınan nesnenin genişliği (cm),

P = Nesnenin resim üzerindeki genişliği (piksel),

B = Nesnenin kameraya olan uzaklığı (cm).



Şekil 3.8. Stereo kamera ile zeytin meyvesinin uzaklık tespitine ait blok diyagramı

Şekil 3.8 ile verilen düzeneğin kurulabilmesi için eş özelliklerde iki kameraya ihtiyaç vardır. Bu nedenle çalışmada Çizelge 3.3 ile verilen teknik özelliklere sahip kameralardan yararlanılmıştır. Eş. 3.7 ile istenen D parametresi kameralar arasındaki mesafeyi temsil etmektedir. Düzenek için kameralar arası mesafe 12,5 cm olarak seçilmiştir.

Çizelge 3.3. Samsung S20 FE kamerasının teknik özellikleri

Özellik Adı	Değeri
Sensör	Samsung S5K2LD
Renkli Kamera	3840x2160 @60 fps 1920x1080 @240 fps 1280x720 @960 fps
Perspektif	120 derece

Stereo kameralardan elde edilen görüntülerdeki zeytin meyvesi, meyve tanıma algoritmasından geçirilerek merkez koordinatları ve genişlik-yükseklik değerleri bulunur. Stereo kameraların düzenekte (Resim 3.6) dik bir açı ile yerleştirilmiş olması ve aralarındaki mesafenin sabit olması sebebiyle elde edilen görüş açısından kaynaklı olarak görüntüler arasında yatay düzlemde bir kayma meydana gelecektir (yaklaşık 750 piksel). Görüntülerin düşey düzlemdeki sapması perspektiften dolayı daha sınırlıdır (± 50 piksel). Zeytin meyvelerinin derinlik hesaplamasının yapılabilmesi için her iki görüntüdeki meyvelerin eşleştirilmesi gerekmektedir. Düşey düzlem sabit tutularak yapılacak olan yatay düzlem taramasında elde edilen meyvelere ilişkin yatay düzlemdeki sapma ve genişlik/yükseklik oranları dikkate alınarak meyveler eşleştirilebilmektedir.



Resim 3.6. Stereo kamera düzeni

Çizelge 3.4. Microsoft Kinect v1 kamerasının teknik özellikleri

Özellik Adı	Değeri
Renkli Kamera	640 x 480 @30 fps
Derinlik Kamerası	320 x 240
Maksimum Ölçüm Mesafesi	~4.5 metre
Minimum Ölçüm Mesafesi	40 cm
Eğim Motoru	Var
Yatay Görüş Alanı	57 derece
Dikey Görüş Alanı	43 derece

Çalışmada ayrıca ToF kamera uygulaması ile de derinlik tespiti yapılması amaçlanmıştır. Bu nedenle envantere yer alan Microsoft Kinect v1 kamerası kullanılmıştır. Kamera ile ilişkin teknik özellikler Çizelge 3.4’te verilmiştir.



4. ARAŞTIRMA BULGULARI

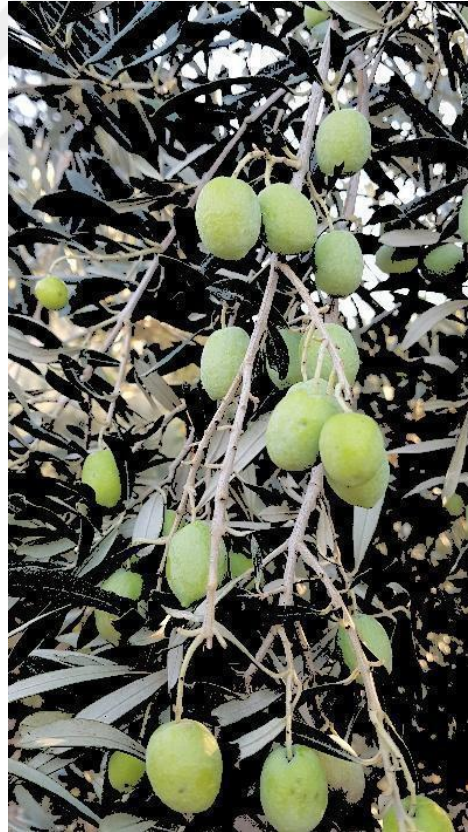
Çalışmanın bu bölümünde elde edilen ve etiketlenen veri setine ilişkin olarak derin öğrenme algoritmaları ile sınıflandırma çalışmaları yapılmış ve başarımları incelenmiştir.

4.1. CNN Modeli İle Zeytin Meyvesi Tespiti

Evrişimsel sinir ağları ile gerçekleştirilen sınıflandırma çalışmasının ilk aşaması olarak resimlere, yeşil renk dışındaki diğer renkleri maskeleyerek için bant geçiren filtre uygulanmıştır. Bu filtreleme işlemi sonucunda arka plan kaynaklı gürültülerin azaltılması sağlanmıştır (Resim 4.1 (b)). Veri setinin %70'i eğitim, %30'u test seti olacak şekilde ayrılmıştır. Eğitim sırasındaki başarımın test edilebilmesi için kullanılan veri setinin çapraz doğrulama (cross-validation) değeri 0.2 olarak seçilmiştir.



(a)

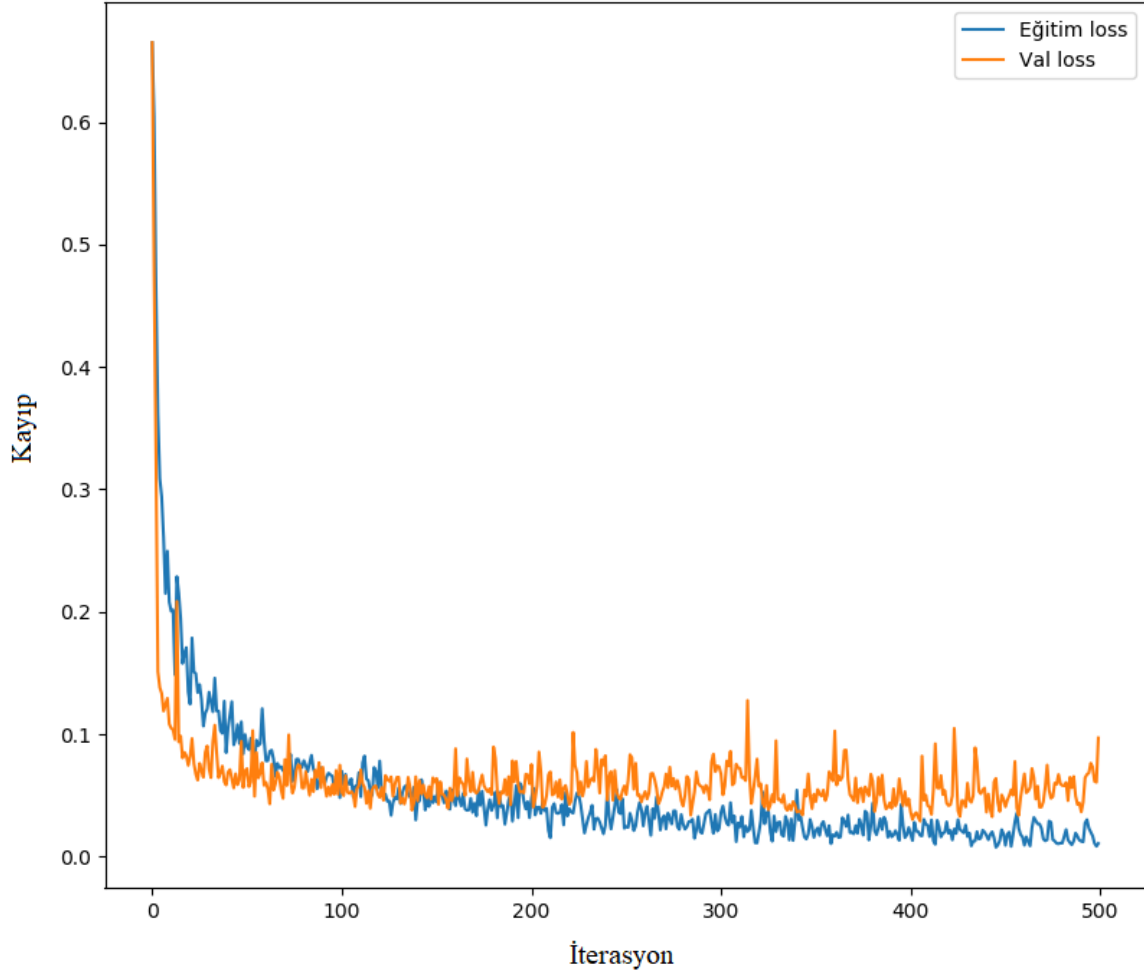


(b)

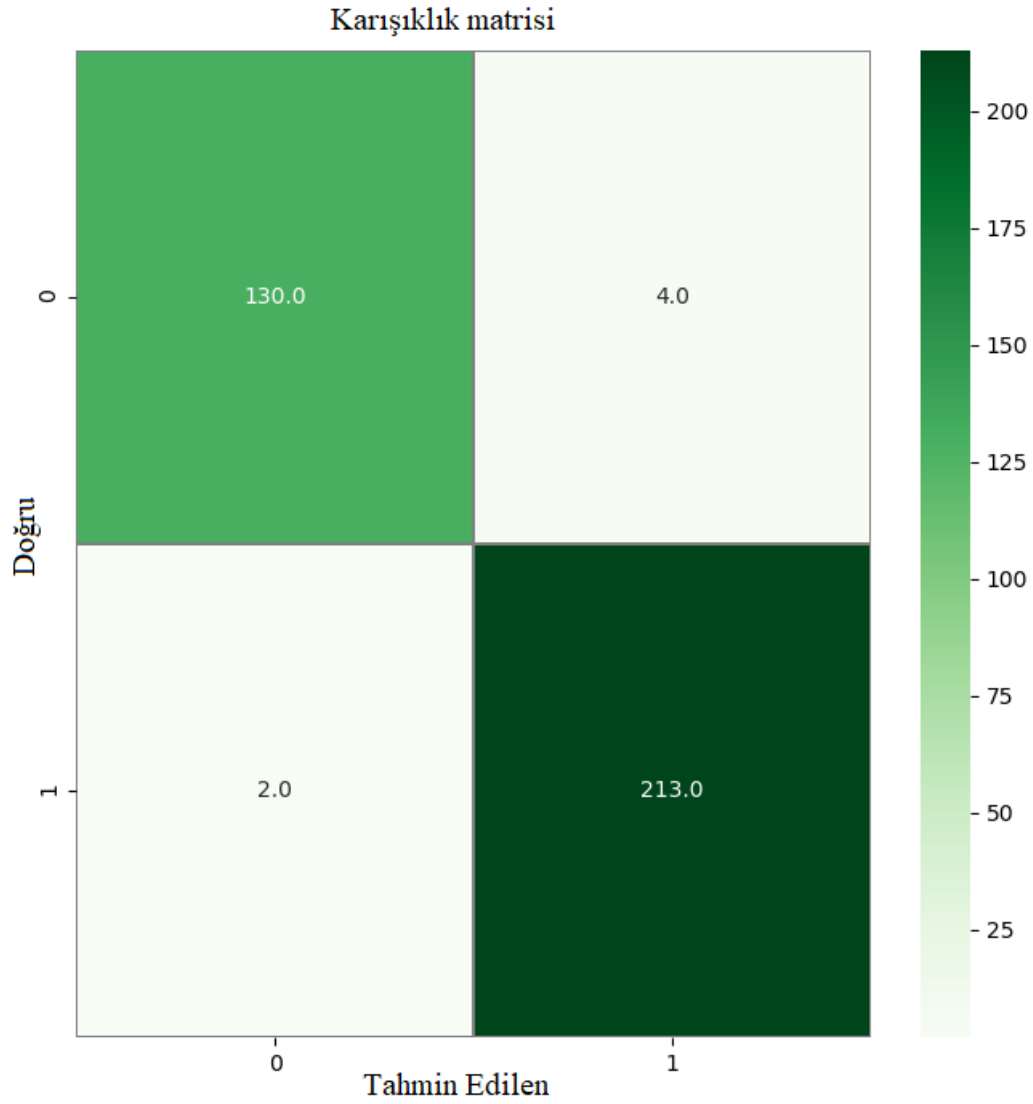
Resim 4.1. a) Orjinal ve b) renk filtresi uygulanmış görüntüler

Yapılan çalışmada 500 iterasyon sonucunda hata değerinin literatürde yer alan çalışmaların

kabul edilebilir deęerine ulaşması nedeniyle eğitim bu iterasyon deęeri ile sınırlandırılmıştır. Eğitim sonucunda elde edilen öğrenme eğrisi ve karışıklık matrisi sırasıyla Şekil 4.1. ve 4.2.'de verilmiştir.

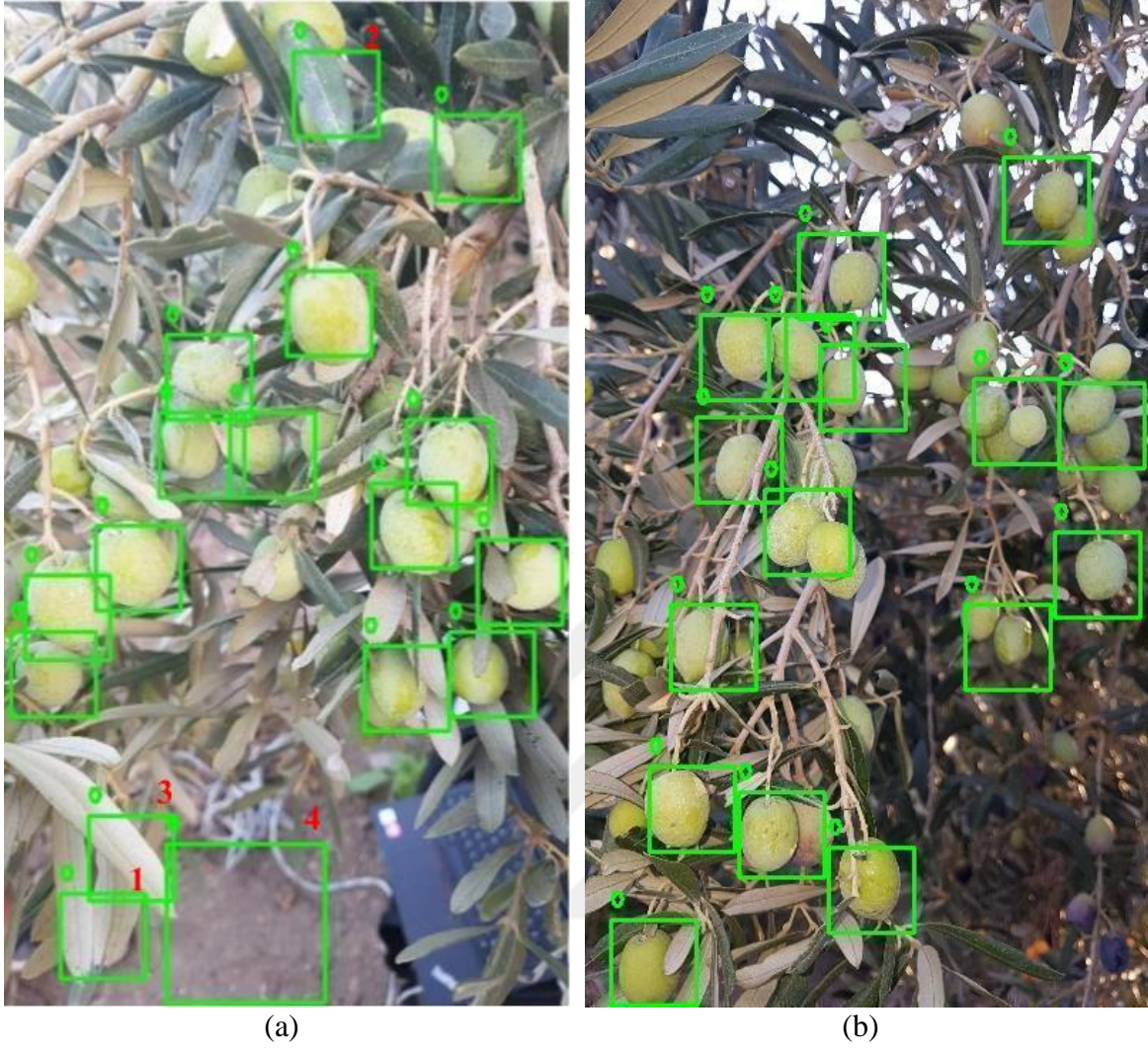


Şekil 4.1. CNN modelinin eğitimine ait öğrenme eğrisi



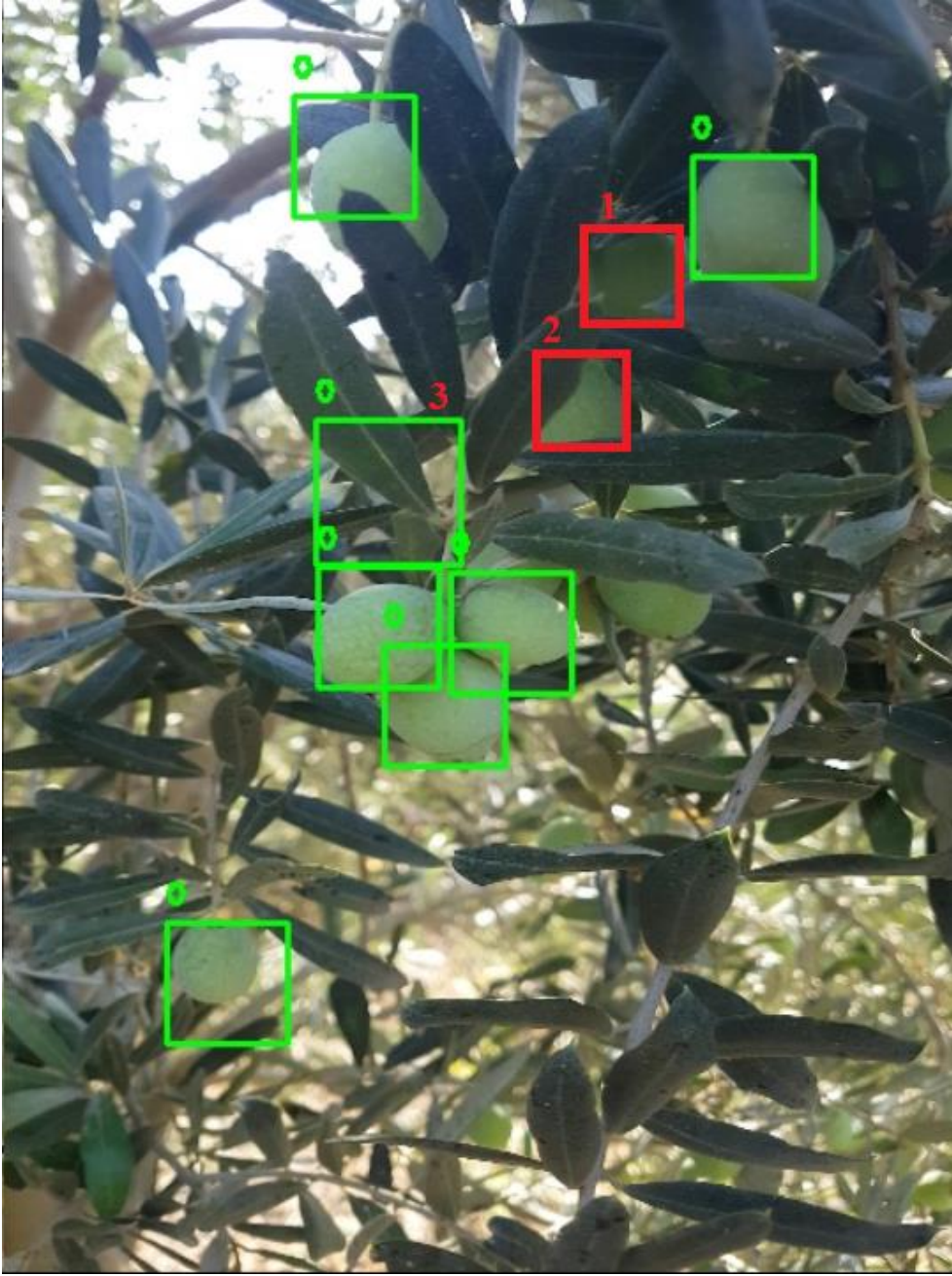
Şekil 4.2. CNN modelinin eğitimine ait karışıklık matrisi

Şekil 4.2’de verilen karışıklık matrisi eğitim öncesinde rastgele seçilen ve eğitimde kullanılmayan 349 resim test amaçlı kullanılmasıyla elde edilmiştir. Bu resimlerden 0 (zeytin) olarak belirtilen sınıfta 4 adet, 1 (yaprak) olarak belirtilen sınıfta ise 2 adet resim hatalı sınıflandırılmıştır. CNN modeli tam resimler üzerinde test edildiğinde ise Resim 4.2.’deki gibi sonuçlar elde edilmiştir.



Resim 4.2. CNN ile eğitilmiş modelin test sonuçları

CNN modelinin başarı oranını hesaplamak amacıyla toplamda 269 adet zeytin meyvesi içeren bir grup resim kullanılmıştır. Bu resimlerden bazıları Resim 4.2. ve Resim 4.3. ile verilmiştir. Resim 4.2.a'da 1, 2 ve 3 ile işaretlenmiş sınırlayıcı kutularda model yaprakları zeytin olarak sınıflandırmıştır. Model, 4 ile işaretlenmiş sınırlayıcı kutuda ise boşluğu zeytin olarak sınıflandırmıştır. Resim 4.2.b' de ise aynı sınırlayıcı kutuda birden fazla zeytin meyvesinin olduğu hatalı sınıflandırma durumları mevcuttur.



Resim 4.3. CNN modelinin test sonuçları

Resim 4.3.'te 3 ile işaretlenmiş sınırlayıcı kutuda yapraklar zeytin olarak sınıflandırılmıştır. Dolayısıyla “yanlış pozitif” (TP) olarak, 1 ve 2 ile işaretlenmiş zeytin meyveleri gibi, küçük bir kısmı görünen ancak model tarafından tespit edilemeyen meyveler ise “yanlış negatif” (FN) olarak değerlendirilmiştir. Bu veriler dikkate alındığında CNN modeli için Çizelge 4.1’de verilen karışıklık matrisi elde edilmiştir.

Çizelge 4.1 CNN modeline ilişkin test sonuçlarının karışıklık matrisi

<i>TP (Doğru Pozitifler)</i>	<i>FN (Yanlış Negatifler)</i>
117	152
<i>FP (Yanlış Pozitifler)</i>	<i>TN (Doğru Negatifler)</i>
29	0

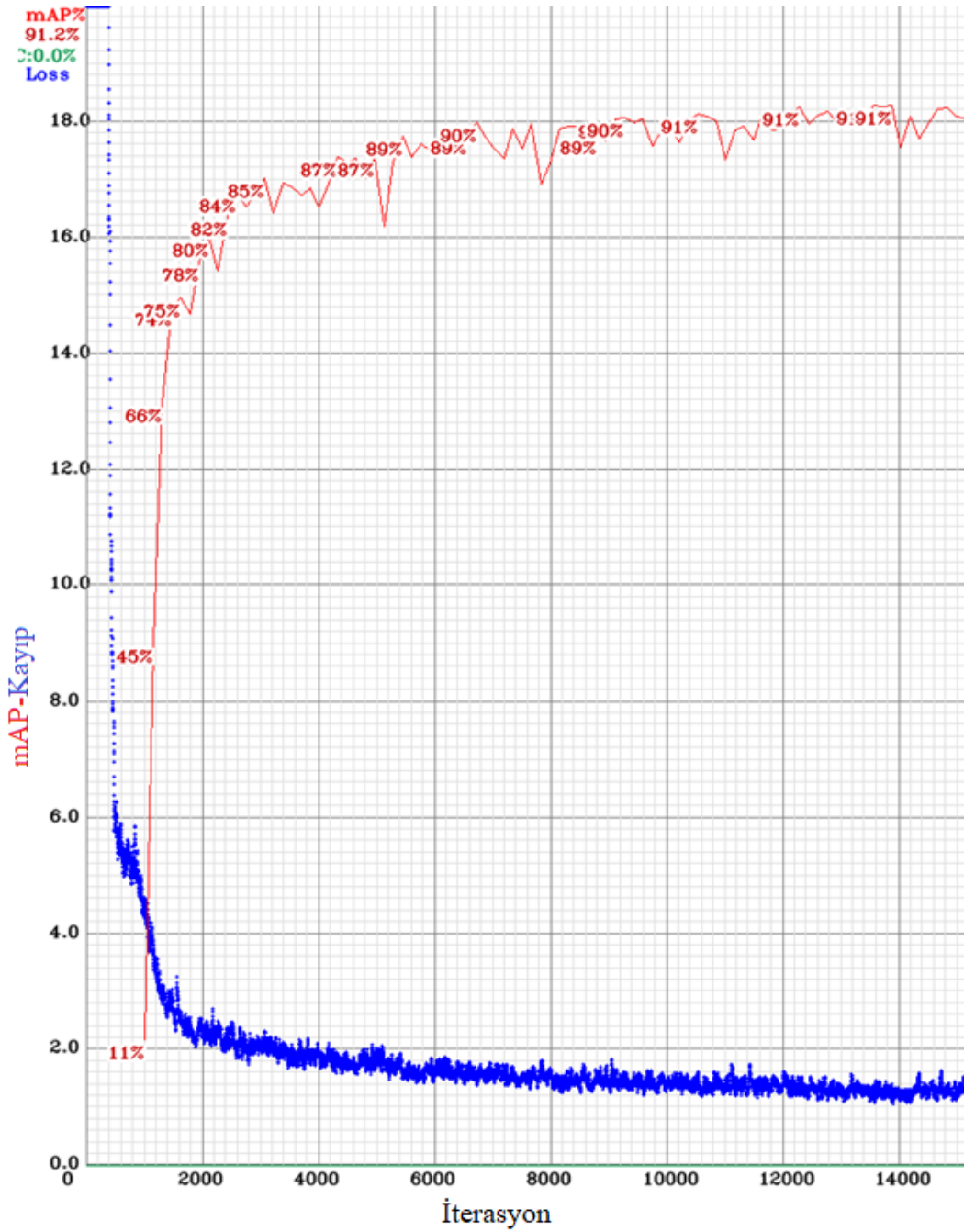
CNN modeline ilişkin karışıklık matrisi incelendiğinde modelin tespit edemediği zeytin sayısı tespit etmiş olduğu zeytin sayısından fazladır. Çizelge 4.1’ deki verilere göre modelin F1 skoru 0.56 olarak hesaplanmıştır.

4.2. YOLOv3-tiny ve YOLOv4-tiny-custom Model Sonuçları

Eğitim ve test aşamaları Google Colab ortamında gerçekleştirilmiştir. Google Colab, NVIDIA T4 modele sahip 320 Tensor Core ve 2560 CUDA Core özellikli ekran kartlarını araştırmacılara sunmaktadır. Çalışma esnasında Google Colab ortamı tarafından sağlanan 40 çekirdek kullanılmıştır.

4.2.1. YOLOv3-tiny

YOLOv3-tiny modelinin eğitiminden sonra elde edilen ortalama kayıp değeri (loss) ve ortalama kesinlik (mean Average Precision- mAP) eğrileri Şekil 4.3.’ te verilmiştir. Ortalama kesinlik değeri, yerel-gerçek sınırlayıcı kutu ile tespit edilen kutunun karşılaştırılması sonucunda elde edilir. Kesinlik değeri ne kadar yüksek olursa modelin tespitleri o kadar doğru olur.



Şekil 4.3. YOLOv3-tiny ortalama kayıp ve kesinlik eğrisi

Ortalama kayıp değeri eğrisine bakıldığında, hata değeri yaklaşık 14000 iterasyon sonucunda 1.20 - 2 değerleri aralığında kalmıştır. Ortalama kesinlik değeri ise %90 seviyelerine ulaşmıştır.



Resim 4.4 Yolov3-tiny sonuçları-1



Resim 4.5 Yolov3-tiny sonuçları-2

Resim 4.4 ve Resim 4.5 ile verilen görüntülerde YOLOv3 modelinin tespit edemediği zeytin sayısı tespit ettiği sayıdan çok daha fazladır. Ancak yanlış sınıflandırılmış zeytin (FP) bulunmamaktadır.



Resim 4.6 Yolov3-tiny sonuçları-3



Resim 4.7 Yolov3-tiny sonuçları-4

Resim 4.7 ile verilen görselde model, zeytinlerin büyük çoğunluğunu tespit etmiştir. Resim 4.6 ve Resim 4.7’de de yanlış tespit edilen zeytin bulunmamaktadır. Toplamda 269 adet zeytin meyvesi içeren bir grup resimden elde edilen sonuçlara ilişkin karışıklık matrisi Çizelge 4.2’ de verilmiştir. Çizelgede bulunan verilere göre YOLOv3-tiny modelinin F1 skoru 0.57 olarak hesaplanmıştır.

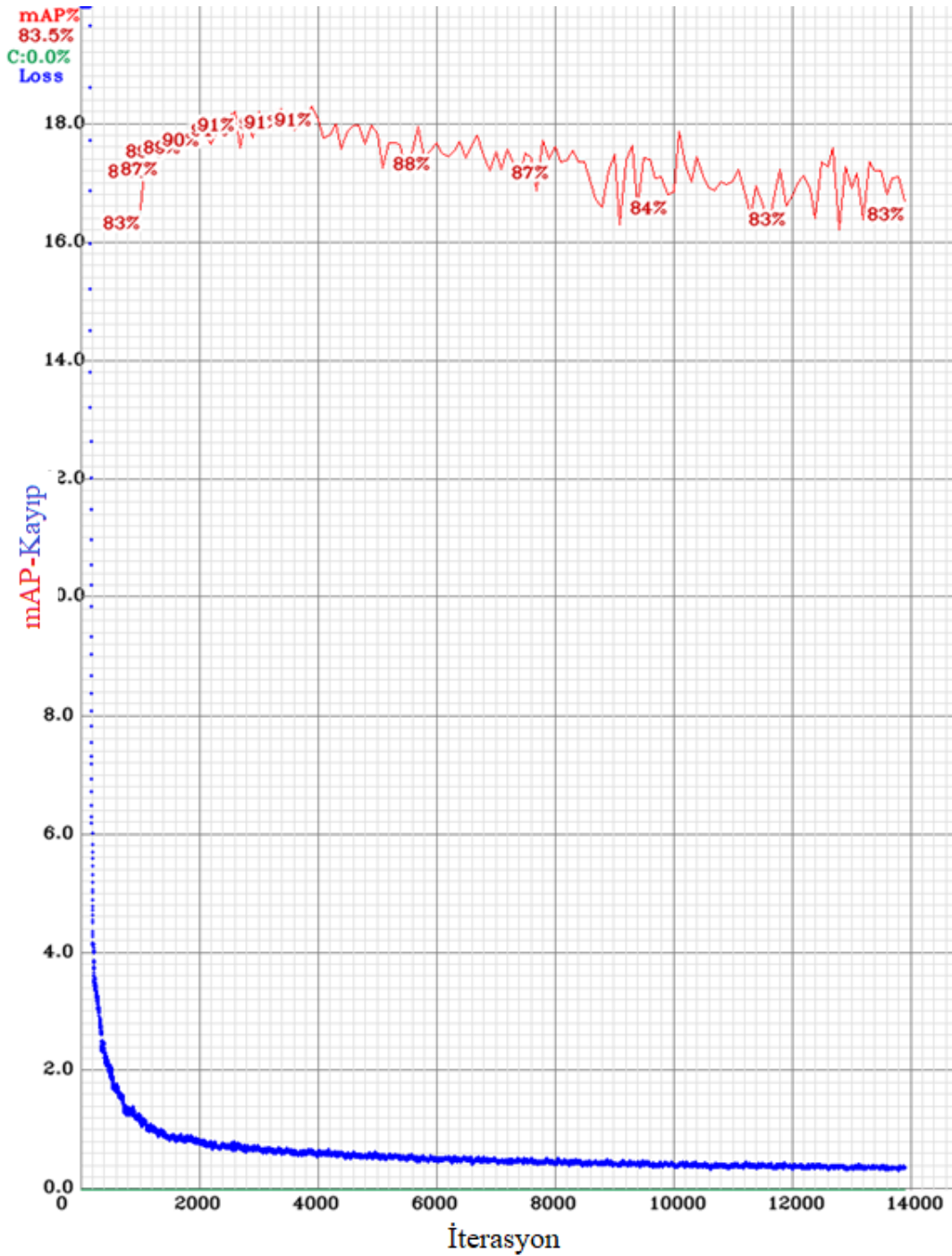
Çizelge 4.2 YOLOv3-tiny modeline ilişkin test sonuçlarının karışıklık matrisi

<i>TP (Doğru Pozitifler)</i>	<i>FN (Yanlış Negatifler)</i>
108	161
<i>FP (Yanlış Pozitifler)</i>	<i>TN (Doğru Negatifler)</i>
0	0

4.2.2. YOLOv4-tiny-custom

YOLOv4 ile yapılan eğitime ilişkin ortalama kayıp değeri (loss) ve ortalama kesinlik (mean Average Precision- mAP) eğrileri Şekil 4.4’ te verilmiştir.

Şekil 4.4’ te verilen eğriye göre ortalama kayıp değeri 14000 iterasyondan önce 0,4 değerinin altına düşmüştür. Ortalama kesinlik değeri ise 0-4000 iterasyon arasında %90 seviyelerine ulaşmışken, daha sonraki iterasyonlarda % 80 seviyelerine düşmüştür.



Şekil 4.4. YOLOv4-tiny-custom ortalama kayıp ve kesinlik eğrisi

Modelin test aşamasında toplamda 269 adet zeytin meyvesi içeren bir grup resim kullanılmıştır. Resim 4.8’de 1 ve 2 ile işaretlenmiş zeytin meyveleri gibi küçük bir kısmı görünen ancak model tarafından tespit edilemeyen meyveler “yanlış negatif” (FN) olarak değerlendirilmiştir.



Resim 4.8 Yolov4-tiny-custom sonuçları-1



Resim 4.9 Yolov4-tiny-custom sonuçları-2



Resim 4.10 Yolov4-tiny-custom sonuçları-3

Resim 4.9 ve Resim 4.10' de modelin tespit edebildiği zeytin sayısı resim üzerinde etiketlenebilen resim sayısına neredeyse eşittir. Modelin bu resimler için başarımı çok yüksek olmuştur.

meyvesinin çok küçük ve yapraklarla neredeyse aynı boy ve renkte olduğu dikkate alındığında YOLOv4-tiny-custom diğer derin öğrenme modellerine göre oldukça başarılı sonuçlar vermiştir.

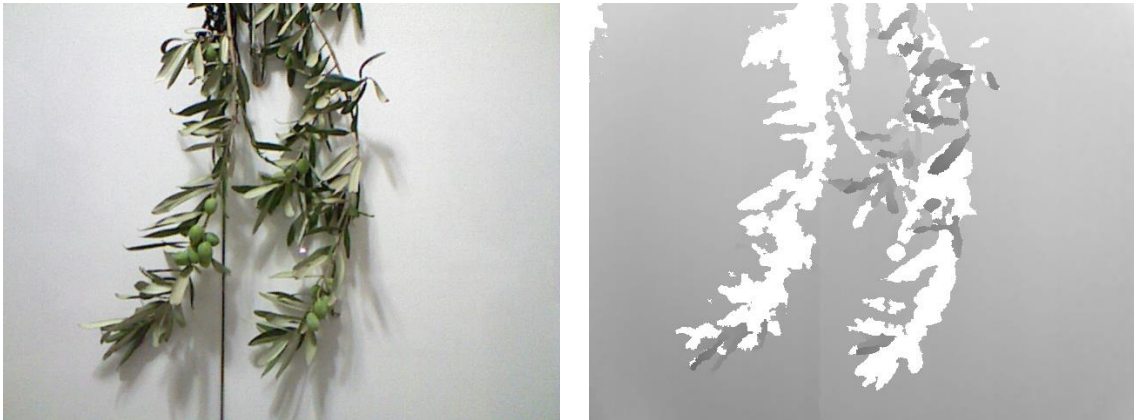
Çizelge 4.4 Derin öğrenme modellerine ilişkin karşılaştırmalı sonuçlar

	<i>Kesinlik</i>	<i>Hassasiyet</i>	<i>F1 skoru</i>	<i>Doğruluk</i>
<i>CNN</i>	<i>0,8</i>	<i>0,43</i>	<i>0,56</i>	<i>0,39</i>
<i>YOLOv3-tiny</i>	<i>1</i>	<i>0,4</i>	<i>0,57</i>	<i>0,40</i>
<i>YOLOv4-tiny-custom</i>	<i>0,97</i>	<i>0,81</i>	<i>0,88</i>	<i>0,79</i>

4.3. Zeytin Meyvesinin Uzaklık Tespitine Yönelik Ön Çalışma Sonuçları

Zeytin meyvesinin hasat edilebilmesi için robot kolun erişim mesafesinin doğru olarak hesaplanması önemlidir.

Microsoft Kinect v1 ile yapılan çalışmalara ilişkin kontrollü ortam deneylerine ilişkin elde edilen sonuçlar Şekil 4.12 ve Şekil 4.13 ile gösterilmektedir.



(a)

(b)

Resim 4.12 Kinect v1 kullanılarak yakın çekim ile elde edilen a) RGB görüntü b) derinlik görüntüsü



Resim 4.13 Kinect v1 kullanılarak uzak çekim ile elde edilen a) RGB görüntü b) derinlik görüntüsü

Kinect v1 ile yapılan çalışmalarda cihazın mesafe tespiti yapılabilmesi için gerekli olan mesafenin 40 cm den daha büyük olması gerekmektedir. Ancak, 40 cm den daha uzak mesafelerden elde edilen RGB resimlerden YOLO algoritması ile zeytin meyvesinin tespiti mümkün olmamıştır. Şekil 4.12 ile verilen görseller 40 cm daha az mesafeden yapılan kayıtlara aittir. Bu mesafeden derinlik bilgisi okunamamıştır. Şekil 4.13 ile verilen görseller 40 cm üzerindeki bir mesafeden elde edilmiştir. Bu verilerde derinlik bilgisi okunmuş fakat ağaç üzerindeki zeytin meyveleri model tarafından sınıflandırılmamıştır.

Üçgenleme (Mattocchia, 2011) yönteminin kullanıldığı düzenekte Samsung S20FE kamera ile çalışmalar gerçekleştirilmiştir.



Resim 4.14 Samsung S20 FE ile sol taraftan yapılan çekimden elde edilen görüntü



Resim 4.15 Samsung S20 FE ile sađ taraftan yapılan çekimden elde edilen görüntü



Resim 4.16 Sol kamera çekimi ile elde edilen görüntüden YOLOv4 modeli ile zeytin meyvesinin tespiti



Resim 4.17 Sađ kamera çekimi ile elde edilen görüntüden YOLOv4 modeli ile zeytin meyvesinin tespiti

Resim 4.14 ve Resim 4.15 ile verilen, zeytin düzeneğinin sol ve sađ çekimlerinden elde edilen görüntülerden YOLOv4-tiny-custom modeli ile zeytin meyvelerinin (Resim 4.16 ve Resim 4.17) ve koordinatlarının tespiti gerçekleştirilmiştir.

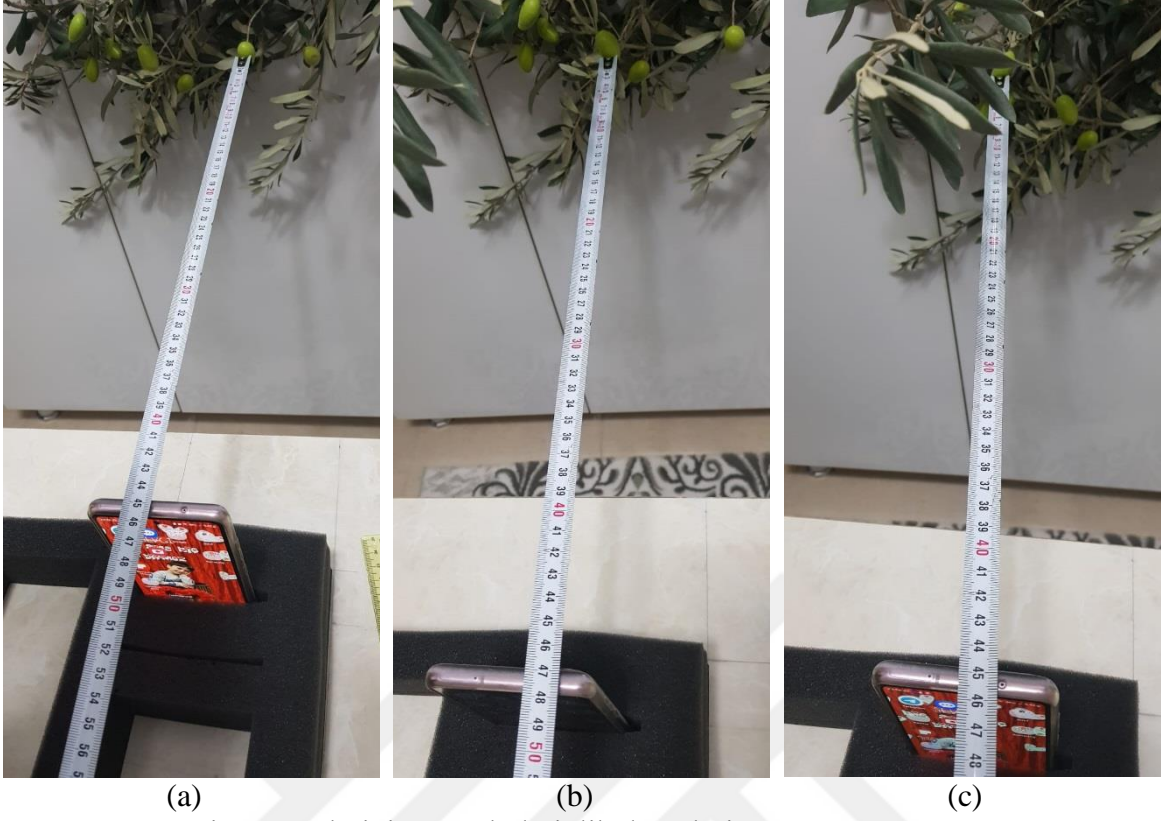
Zeytin meyveleri ve koordinatlar tespit edildikten sonra her iki kameradan elde edilen görüntüler bir referans noktasına göre çakıştırılmış ve odak uzaklığı hesaplanmıştır. Resim 4.16 ve Resim 4.17’de R ile işaretli zeytin referans olarak alınmış ve odak uzaklığı hesabı bu zeytine göre yapılmıştır. R zeytinin eni (W) ve kameraya olan uzaklığı (D) ölçülmüştür. Daha sonra kameradan elde edilen görüntülerden R zeytininin resim üzerindeki boyutu piksel biriminden hesaplanmış ve Eş. 3.8 ile verilen denklem çözümlenerek odak uzaklığı 3360 piksel olarak bulunmuştur. Bu değer, gerçek mesafeler hesaplanmadan önce kalibrasyon amacıyla bir kere hesaplanmıştır. Daha sonra YOLO algoritmasının sağladığı, tespit edilen zeytinlerin merkez koordinatları kullanılarak zeytinlerin (1, 2 ve 3 numaralı) kameraya olan gerçek uzaklığı Eş. 3.7 ile bulunmuştur. Örneğin, YOLO algoritmasında 1 numaralı zeytinin merkezinin sağ kamera ile elde edilen görüntüsü için X koordinatı (XR) 2222 piksel, sol kamera ile elde edilen görüntüsü için de X koordinatı (XL) 1312 piksel olarak hesaplanmıştır. Odak uzaklığı 3360 piksel ve kameralar arasındaki mesafe 12,5 cm’dir. Buna göre 1 numaralı zeytinin gerçek uzaklığı;

$$Z = 3360 * \frac{12,5}{2222 - 1312} = 46,1 \text{ cm} \quad (4.1)$$

Elde edilen sonuçlar Çizelge 4.5 ile gerçek mesafeler ise Resim 4.18 ile verilmiştir.

Çizelge 4.5. Zeytin meyvesinin derinlik tespitine ilişkin sonuçlar

Zeytin Meyvesi	XL (piksel)	XR (piksel)	Hesaplanan Derinlik (Z)	Gerçek Derinlik	Hata Oranı (%)
Resim 4.18.a	2222	1312	46,1 cm	45 cm	2,44
Resim 4.18.b	1538	658	47,7 cm	47 cm	1,49
Resim 4.18.c	1208	249	43,8 cm	45 cm	2,66



Resim 4.18. Zeytin meyvelerinin gerçek derinlik deęerleri

5. SONUÇLAR VE ÖNERİLER

Zeytin meyvesinin hasadı geçmişten bu güne kadar sırikla vurarak, elle veya ağaç dallarını sarsarak yapılmaktadır.

Bu çalışmada amaç, doğal ortamda bulunan zeytin ağacı üzerinde zeytin meyvesini ve meyvenin üç boyutlu konumunu tespit etmektir. Çalışmada kullanılan veri seti 2020 yılının Ekim ayında Hatay ilinin Defne ilçesinde bulunan zeytin bahçesinde 12 MP Optik görüntü sabitleyiciye (OIS) sahip kamera ile 1,3 MP CMOS fotoğraf sensörüne sahip IP kamera kullanılarak gün ışığı, güneş ışığı, yapay ışık (flaş) gibi farklı aydınlatma koşullarında elde edilmiştir. Zeytin meyvesinin tespit edilmesi için CNN, YOLOv3-tiny ve YOLOv4-tiny-custom gibi derin öğrenme modelleri kullanılarak eğitim gerçekleştirilmiş ve modellerin performansları test edilmiştir.

Çalışmaya konu olan veri setinin CNN modeli ve ön işleme yöntemleriyle birlikte kullanılması sonucunda elde edilen modelin başarı oranı 0.56 olarak bulunmuştur. Bu değer YOLOv3-tiny ile elde edilen tahmin sonucuna çok yakındır (0.57). YOLOv4-tiny ile oluşturulan modelin tahmin başarısı ise 0.88 bulunmuştur. Bu model ağaç üzerinde yer alan zeytin meyvesi için diğer modellerden daha iyi bir öğrenme gerçekleştirmiştir.

Üçgenleme metodu ile x ve y si bilinen zeytin meyvesinin derinliği de %98 başarı ile ölçülmüştür.

Elde edilen ve kullanılan veri seti, belirli bir türe ve renge özgü zeytin meyvesi ile oluşturulmuştur. Farklı türe ve renge sahip zeytin meyveleri ile modelin diğer türler üzerindeki başarısı artırılabilir. Yapılan çalışma kapsamında zeytin meyvesinin derinlik bilgisi de elde edildiği için çalışma zeytin meyvesi hasadı için gerekli altyapıyı sağlamaktadır. Çalışmanın sonraki adımı olarak robot kol ile toplama sürecinin gerçekleştirilmesi planlanmaktadır.

KAYNAKLAR

- Afonso, M., Fonteijn, H., Fiorentin, F. S., Lensink, D., Mooij, M., Faber, N. ve Wehrens, R. (2020). Tomato fruit detection and counting in greenhouses using deep learning. *Frontiers in plant science*, 1759.
- Arad, B., Balendonck, J., Barth, R., Ben-Shahar, O., Edan, Y., Hellström, T. ve van Tuijl, B. (2020). Development of a sweet pepper harvesting robot. *Journal of Field Robotics*, 37(6), 1027-1039.
- Aygün, İ., Urkan, E., Alayunt, F. N., Yalçın, H. ve Tekin, A. B. (2019). İzmir ilinde zeytin hasadında kullanılan yerli ve ithal çırpıcı tip makinaların hasat performanslarının değerlendirilmesi. *Bursa Uludağ Üniversitesi Ziraat Fakültesi Dergisi*, 33(2), 265-271.
- Bulanon, D. M., Burks, T. F. ve Alchanatis, V. (2009). Image fusion of visible and thermal images for fruit detection. *Biosystems engineering*, 103(1), 12-22.
- Feng, J., Liu, G., Wang, S., Zeng, L. ve Ren, W. (2012). A novel 3D laser vision system for robotic apple harvesting. In 2012 Dallas, Texas, July 29-August 1, 2012 (p. 1). American Society of Agricultural and Biological Engineers.
- Fernández, R., Salinas, C., Montes, H. ve Sarria, J. (2014). Multisensory system for fruit harvesting robots. Experimental testing in natural scenarios and with different kinds of crops. *Sensors*, 14(12), 23885-23904.
- Foix, S., Alenya, G. ve Torras, C. (2011). Lock-in time-of-flight (ToF) cameras: A survey. *IEEE Sensors Journal*, 11(9), 1917-1926.
- Gai, R., Chen, N. ve Yuan, H. (2021). A detection algorithm for cherry fruits based on the improved YOLO-v4 model. *Neural Computing and Applications*, 1-12.
- Gatica, G., Best, S., Ceroni, J. ve Lefranc, G. (2013). Olive fruits recognition using neural networks. *Procedia Computer Science*, 17, 412-419.
- Ge, Y., Xiong, Y., Tenorio, G. L. ve From, P. J. (2019). Fruit localization and environment perception for strawberry harvesting robots. *IEEE Access*, 7, 147642-147652.
- Häni, N., Roy, P. ve Isler, V. (2020). A comparative study of fruit detection and counting methods for yield mapping in apple orchards. *Journal of Field Robotics*, 37(2), 263-282
- Hannan, M. W., Burks, T. F. ve Bulanon, D. M. (2007). A real-time machine vision algorithm for robotic citrus harvesting. In 2007 ASAE Annual Meeting (p. 1). American Society of Agricultural and Biological Engineers.
- Hemming, J., Bac, C. W., van Tuijl, B. A., Barth, R., Bontsema, J., Pekkeriet, E. J. ve Van Henten, E. (2014). A robot for harvesting sweet-pepper in greenhouses.
- Işık, E. ve Ünal, H. (2003). Mekanik titreşimli zeytin hasat makinasının performans değerlerinin belirlenmesi. *Uludağ Üniversitesi Ziraat Fakültesi Dergisi*, 17(2), 103-110.

- Junos, M. H., Mohd Khairuddin, A. S., Thannirmalai, S. ve Dahari, M. (2021). Automatic detection of oil palm fruits from UAV images using an improved YOLO model. *The Visual Computer*, 1-15.
- Jia, W., Mou, S., Wang, J., Liu, X., Zheng, Y., Lian, J. ve Zhao, D. (2020). Fruit recognition based on pulse coupled neural network and genetic Elman algorithm application in apple harvesting robot. *International Journal of Advanced Robotic Systems*, 17(1), 1729881419897473.
- Jimenez, A. R., Ceres, R. ve Pons, J. L. (2000). A survey of computer vision methods for locating fruit on trees. *Transactions of the ASAE*, 43(6), 1911.
- Kang, H. ve Chen, C. (2019). Fruit detection and segmentation for apple harvesting using visual sensor in orchards. *Sensors*, 19(20), 4599.
- Koirala, A., Walsh, K. B., Wang, Z. ve McCarthy, C. (2019). Deep learning for real-time fruit detection and orchard fruit load estimation: Benchmarking of ‘MangoYOLO’. *Precision Agriculture*, 20(6), 1107-1135.
- Kuznetsova, A., Maleva, T. ve Soloviev, V. (2020). Using YOLOv3 algorithm with pre-and post-processing for apple detection in fruit-harvesting robot. *Agronomy*, 10(7), 1016.
- Lehnert, C., English, A., McCool, C., Tow, A. W. ve Perez, T. (2017). Autonomous sweet pepper harvesting for protected cropping systems. *IEEE Robotics and Automation Letters*, 2(2), 872-879.
- Li, B., Wang, M. ve Wang, N. (2010). Development of a real-time fruit recognition system for pineapple harvesting robots. In 2010 Pittsburgh, Pennsylvania, June 20-June 23, 2010 (p. 1). American Society of Agricultural and Biological Engineers.
- Li, P., Lee, S. H. ve Hsu, H. Y. (2011). Review on fruit harvesting method for potential use of automatic fruit harvesting systems. *Procedia Engineering*, 23, 351-366.
- Li, D., Sun, X., Elkhouchlaa, H., Jia, Y., Yao, Z., Lin, P. ve Lu, H. (2021). Fast detection and location of longan fruits using UAV images. *Computers and Electronics in Agriculture*, 190, 106465.
- Montalbo, F. J. P. (2020). A computer-aided diagnosis of brain tumors using a fine-tuned YOLO-based model with transfer learning. *KSII Transactions on Internet and Information Systems (TIIS)*, 14(12), 4816-4834.
- Mu, L., Liu, Y., Cui, Y., Liu, H., Chen, L., Fu, L. ve Gejima, Y. (2017). Design of end-effector for kiwifruit harvesting robot experiment. In 2017 ASABE Annual International Meeting (p. 1). American Society of Agricultural and Biological Engineers.
- Onishi, Y., Yoshida, T., Kurita, H., Fukao, T., Arihara, H. ve Iwai, A. (2019). An automated fruit harvesting robot by using deep learning. *Robomech Journal*, 6(1), 1-8.
- Pashine, S., Dixit, R. ve Kushwah, R. (2021). Handwritten Digit Recognition using Machine and Deep Learning Algorithms. *arXiv preprint arXiv:2106.12614*.
- Peebles, M., Barnett, J. J., Duke, M. ve Lim, S. H. (2020, August). Robotic Harvesting of Asparagus using Machine Learning and Time-of-Flight Imaging—Overview of

- Development and Field Trials. In 2020 IEEE 16th International Conference on Automation Science and Engineering (CASE) (pp. 1361-1366). IEEE.
- Puttemans, S., Vanbrabant, Y., Tits, L. ve Goedemé, T. (2016, December). Automated visual fruit detection for harvest estimation and robotic harvesting. In 2016 sixth international conference on image processing theory, tools and applications (IPTA) (pp. 1-6). IEEE.
- Redmon, J., Divvala, S., Girshick, R. ve Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).
- Scarfe, A. J., Flemmer, R. C., Bakker, H. H. ve Flemmer, C. L. (2009). Development of an autonomous kiwifruit picking robot. In 2009 4th International Conference on Autonomous Robots and Agents (pp. 380-384). IEEE.
- Schertz, C. E. ve Brown, G. K. (1968). Basic considerations in mechanizing citrus harvest. Transactions of the ASAE, 11(3), 343-0346.
- Seçer, A. ve Ekmeksiz, F. (2012). Doğu Akdeniz Bölgesi'nde zeytin ve zeytinyağı üretimi, pazarlaması ve bölgede zeytinciliği geliştirme olanakları. Tarımsal Ekonomi ve Politika Geliştirme Enstitüsü Yayın 206.
- Sumner, H. R. ve Hedden, S. L. (1975). Harvesting oranges with a full-powered positioning limb shaker. In Proceedings of the Florida State Horticultural Society (Vol. 88, pp. 117-119).
- Tanigaki, K., Fujiura, T., Akase, A. ve Imagawa, J. (2008). Cherry-harvesting robot. Computers and electronics in agriculture, 63(1), 65-72.
- Xiong, Y., Ge, Y., Grimstad, L. ve From, P. J. (2020). An autonomous strawberry-harvesting robot: Design, development, integration, and field evaluation. Journal of Field Robotics, 37(2), 202-224.
- Xu, Z. F., Jia, R. S., Sun, H. M., Liu, Q. M. ve Cui, Z. (2020). Light-YOLOv3: fast method for detecting green mangoes in complex scenes using picking robots. Applied Intelligence, 50(12), 4670-4687.
- Xu, Z. F., Jia, R. S., Liu, Y. B., Zhao, C. Y. ve Sun, H. M. (2020). Fast method of detecting tomatoes in a complex scene for picking robots. IEEE Access, 8, 55289-55299
- Wang, C., Lee, W. S., Zou, X., Choi, D., Gan, H. ve Diamond, J. (2018). Detection and counting of immature green citrus fruit based on the local binary patterns (LBP) feature using illumination-normalized images. Precision Agriculture, 19(6), 1062-1083.
- Wang, C., Luo, T., Zhao, L., Tang, Y. ve Zou, X. (2019). Window zooming-based localization algorithm of fruit and vegetable for harvesting robot. IEEE Access, 7, 103639-103649.
- Yasukawa, S., Li, B., Sonoda, T. ve Ishii, K. (2017, January). Development of a tomato harvesting robot. In 2017 International Conference on Artificial Life and Robotics (ICAROB), Miyazaki (pp. 408-411).
- Mattocchia, S. (2011). Stereo vision: Algorithms and applications. University of Bologna, 22.

DİZİN

B

BBox-Label-Tool · 23, 24

C

CNN · 7, 8, 9, 10, 11, 13, 14, 17,
20, 21, 31, 32, 33, 34, 35, 36,
44, 45, 52

Cuda · 38

D

Derinlik · 6, 28, 29, 31, 32, 47,
48, 53, 54, 55

Derin öğrenme · 7, 8, 11, 13, 18,
26, 33, 46, 47, 55

E

Eğitim · 7, 9, 10, 11, 15, 16, 18,
21, 22, 23, 25, 27, 33, 34, 35,
38, 42, 55

Evrişimsel sinir ağı · 7, 9, 13, 14,
18

I

IOU · 22

K

Karışıklık matrisi · 34, 35, 37,
38, 42, 46

Kayan pencere · 16, 17, 19, 20,
21, 22, 23

Keras · 18

Kinect · 4, 12, 31, 32, 47, 48

Konvolüsyon · 13, 14, 15

M

Maksimum olmayan bastırma ·
19

R

RGB · 6, 8, 12, 14, 47, 48

S

Stereo kamera · 7, 29, 30, 31

T

Test sonuçları · 36, 37, 38, 42, 46

ToF · 12, 28, 32

Triangulation · 28

U

Uzaklık · 30, 47

Ü

Üçgenleme · 48, 55

V

Veri seti · 5, 6, 7, 8, 9, 10, 11, 12,
13, 14, 16, 17, 18, 22, 23, 33,
46, 55

Y

Yaprak · 1, 2, 3, 4, 6, 9, 16, 35,
36, 37, 47

Yolov3 · 8, 9, 10, 11, 12, 23, 26,
38, 39, 40, 41, 42, 46, 47, 55

Yolov4 · 11, 23, 26, 27, 38, 42,
43, 44, 45, 46, 47, 51, 52, 55

Z

Zeytin meyvesi · 1, 2, 13, 16, 22,
25, 28, 30, 31, 33, 36, 42, 43,
46, 47, 48, 51, 52, 53, 55



TEKNOVERSITE



teknoversite **AYRICALIĞINDASINIZ**

İSTE

