



## Deep learning-based autonomous harvesting robot design for kumquat fruit

Taner Gündüz\*<sup>ID</sup>, Mehmet Dersuneli<sup>ID</sup>, Yakup Kutlu<sup>ID</sup>

Department of Computer Engineering, Faculty of Engineering, İskenderun Technical University, 31200, Iskenderun, Hatay, Türkiye

### Highlights:

- Leaf problem solution in collection point detection with deep learning
- Use of MLP algorithm for inverse kinematics calculations
- Autonomous harvesting from tree with mobile structure

### Keywords:

- Autonomous systems
- End effectors
- Machine learning
- Multilayer perceptron

### Article Info:

Research Article  
Received: 03.11.2022  
Accepted: 09.11.2023

### DOI:

10.17341/gazimmfd.1199140

### Correspondence:

Author: Taner Gündüz  
e-mail:  
tanergunduz44@gmail.com  
phone: +90 505 789 9956

### Graphical/Tabular Abstract

In this study, an autonomous harvesting robot has been developed as a solution proposal to the workforce problem in the field of agriculture. The autonomous harvest robot consists of mechanical, electronic and artificial intelligence modules and harvests kumquat fruit with a vacuum gripper model (Figure A). YOLOv7 architecture and image processing libraries were used as the deep learning model.

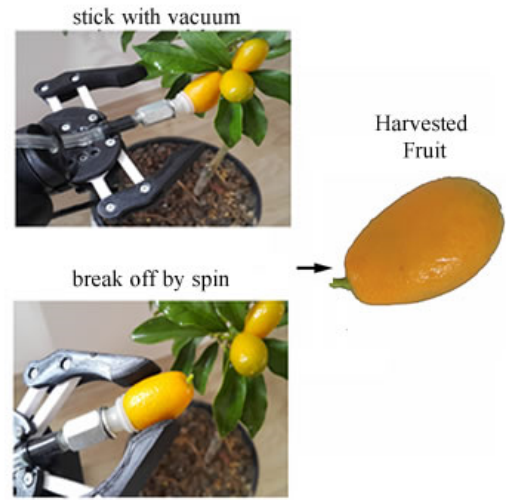
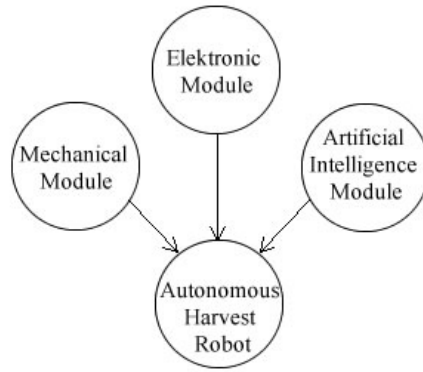


Figure A. System structure and fruit harvest

### Purpose:

The main purpose of this study is to provide an autonomous solution supported by artificial intelligence for harvesting processes where the labor problem is felt most in agricultural areas.

### Theory and Methods:

Autonomous harvesting robot system consists of three main modules. In the first module, a 6-axis robot arm and a vacuum gripping that can perform the harvesting process is designed. In the second module, the YOLOv7 deep learning algorithm was customized according to the study. Position determination was made with using RGB-D camera. The collection point was determined using image processing algorithms. Then, inverse kinematics calculations were made with the MLP algorithm. In the last module, the system was completed by creating a mobile platform.

### Results:

In the test phase, the data obtained from the natural environment images without any changes were used. Object detection performance with the YOLOv7 algorithm has been determined as 93%. For harvest performance, two trees with a total of 120 kumquat fruits were used. Picking performance was determined as 75% for a full round of fruit detection, picking and placing in the basket. At this point, the leaf and sticky fruit problem mentioned in the literature has been solved by determining the collection point.

### Conclusion:

The developed system includes new approaches for autonomous harvesting robots. Although it gives more successful results than the studies in the literature, it is necessary to fix some position detection errors in mobile motion systems and to be applied in real fields.



## Kamkat meyvesi için derin öğrenmeye dayalı otonom hasat robotu tasarımı

Taner Gündüz\*<sup>ID</sup>, Mehmet Dersuneli<sup>ID</sup>, Yakup Kutlu<sup>ID</sup>

İskenderun Teknik Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, 31200, İskenderun, Hatay, Türkiye

### ÖNEÇIKANLAR

- Derin öğrenme ile toplama noktası tespitinde yaprak problemi çözümü
- Ters Kinematik hesaplamaları için MLP algoritması kullanımı
- Mobil yapı ile ağaç üzerinden otonom hasat

### Makale Bilgileri

Araştırma Makalesi

Geliş: 03.11.2022

Kabul: 09.11.2023

### DOI:

10.17341/gazimmfd.1199140

### Anahtar Kelimeler:

Otonom sistemler,  
son efektörler,  
makine öğrenmesi,  
çok katmanlı algılayıcılar

### ÖZ

Otonom robotlar, dünya nüfus artışı karşısında azalan tarımsal üretim alanlarına ve tarımsal işgücü ihtiyacına çözüm olarak ortaya çıkmaktadır. Dünya genelinde insan hatalarından ve çalışma sürelerinden bağımsız bir yöntem olarak otonom hasat robotları üzerinde çalışmalar yapılmaktadır. Bu çalışmada, mobil bir platform üzerinde 6 eksenli bir robotik kol tasarlanmıştır. Derin öğrenme algoritmaları ile kamkat meyve tespiti yapılmış ve özel tasarlanmış bir vakum tutucu ile entegre bir görüntü işleme algoritması oluşturulmuştur. Ayrıca literatürde hasat performansını düşüren yaprak sorunu ele alınmış ve çözüm önerilmiştir. Nesne tespiti sonrası geliştirilen algoritma ile yaprak veya herhangi bir engele takılmadan hasat gerçekleştirilmiştir. Denavit-Hartenberg (D-H) yöntemi kullanılarak elde edilen veri setinin ters kinematik hesaplamaları için yapay sinir ağı tabanlı model oluşturularak robot hareketleri hesaplanmıştır. Nesne tespit başarıları %93 olup, saksılı kamkat ağaçlarında yapılan testler sonucunda %75 hasat başarıları elde edilmiştir.

## Deep learning-based autonomous harvesting robot design for kumquat fruit

### HIGHLIGHTS

- Leaf problem solution in collection point detection with deep learning
- Use of MLP algorithm for inverse kinematics calculations
- Autonomous harvesting from tree with mobile structure

### Article Info

Research Article

Received: 03.11.2022

Accepted: 09.11.2023

### DOI:

10.17341/gazimmfd.1199140

### Keywords:

Autonomous systems,  
end effectors,  
machine learning,  
multilayer perceptrons

### ABSTRACT

Autonomous robots emerge as a solution to the decreasing agricultural production areas and agricultural workforce needs in the face of world population growth. All over the world, studies are carried out on autonomous harvesting robots as a method independent of human errors and working times. In this study, a 6-axis robotic arm is designed on a mobile platform. Kumquat fruit detection was made with deep learning algorithms, and an integrated image processing algorithm was created with a specially designed vacuum gripper. In addition, the leaf problem that reduces the harvest performance has been discussed in the literature and a solution has been proposed. With the algorithm developed after object detection, harvesting was carried out without getting stuck on leaves or any obstacles. For the inverse kinematics calculations of the data set obtained by using the Denavit-Hartenberg (D-H) method, an artificial nerve-based model was created and robot movements were calculated. Object detection success was 93% and 75% harvest success was achieved as a result of the tests performed on potted kumquat trees.

## 1. Giriş (Introduction)

Birleşmiş Milletler Gıda ve Tarım Örgütü raporlarında tarım sektörünün önümüzdeki yıllarda hiç olmadığı kadar önemli bir hale geleceğinden bahsedilmiştir. Bu örgütün geçmiş yıllardaki raporlarında, 2050 yılında artan dünya nüfusunun gıda ihtiyacının karşılanması için 2006 yılı itibarıyla %70 daha fazla gıda üretilmesi gerektiği belirtilmiştir [1]. Bu talebi karşılamak için çiftçiler ve tarım şirketlerinin, Endüstri 4.0 çözümlerine yönelmelerinin gerektiği belirtilmiştir [2]. Bu kapsamda; Tarım sektöründe tohum ekimi, ilaçlama, zararlı bitki temizliği, toprak analizi ve ürün hasadı gibi doğrudan tarlada etkili olacak alanlarda tüm dünyada yapay zekâ uygulamalarına dayalı otonom çözümler üzerine çalışmalar yapılmaya başlanmıştır [3]. Özellikle işgücü açığının en üst düzeyde olduğu tarımsal ürün hasadı konusu üretim miktarını doğrudan etkilediğinden, bu kapsamda otonom hasat robotları üzerine yapılan çalışmalar birçok farklı ürün için önemli bir konu haline gelmiştir. Literatürde otonom hasat robotları ile ilgili çalışmalara bakıldığında; 2008 yılında, kiraz hasadı için 4 eksenli robotik kol tasarlanmıştır. Görü sistemi için 3 boyutlu görüntü sensörü tasarlanmış ve kızılötesi lazer diyotları kullanılmıştır. Sensörden alınan görüntülerden meyvelerin ve engellerin yerleri belirlenmiş, ancak kirazların çoğunun yaprakların arkasında olması ve ışık yansımaları nedeniyle nesne algılamada yeterli performans elde edilememiştir. Bu nedenle verimli deneylerin yapılamadığı bildirilmiştir [4]. 2018 yılında, laboratuvar ortamında yapay ağaçlar ve meyveler üzerinde yapılan bir çalışmada 6 eksenli robot kola sahip bir hasat sistemi geliştirilmiştir. Ağaç tespiti Zed stereo kamera ile meyve tespiti ise ikinci bir web kamera kullanılarak Ton Doygunluk Değeri (HSV) renk uzayı ve morfolojik işlemler ile yapılarak hasat işleminin %80 başarı ile gerçekleştiği belirtilmiştir [5]. 2018 yılında, kiraz domateslerini sera ortamında toplamak için ray üzerinde hareket edebilen robotik bir sistem geliştirilmiştir. Kırmızı (R) ve Yeşil (G) renk parametreleri ile nesne tespiti ve sınıflandırması sonucu %83 toplama başarısı sağlandığı ve 8 saniyede bir domatesin toplandığı bildirilmiştir [6]. 2018 yılında yapılan bir başka çalışmada, üst üste binen salkımların hasadı için kesim noktasının belirlenmesi konusu ele alınmış, HSV renk uzayı ve K-ortalama kümeleme yöntemleri kullanılmıştır. Kesişim üzüm salkımı parçaları için Canny kenar dedektörü kullanılarak geometrik bir hesaplama yöntemi oluşturulmuş ve sap kısımlarda %81,6 doğrulukla kesme noktaları işaretlendiği belirtilmiştir [7]. 2019 yılında, mobil platform üzerinde robot kol ile çilek toplayan robotik bir sistem geliştirilmiştir. Çalışmada Kırmızı Yeşil Mavi (RGB) formatındaki görüntülerden kırmızı olanlar filtrelenerek, çilekler bulunması sağlanmıştır. Sonuç olarak robotun sera ortamında izole edilen çilekleri %96,8, izole olmayan çilekleri ise %53,6 başarı ile topladığı belirtilmiştir [8]. 2019 yılında yapılan bir başka çalışmada, Evrişimli Sinir Ağı (CNN) tabanlı kivi toplamak için hasat robotu geliştirilmiştir. Dört stereo kamera ile çalışan sistemde, meyvelerin tespit performansını artırmak için tüm görüntüler eşleştirilmiştir. Gerçek alanda toplama başarısı %51, kayıp %24,6 ve kalan %24,4 olarak bildirilmiştir [9]. 2019 yılında, seradan elma toplamak için robotik bir sistem geliştirilmiştir. Nesne tespiti için CNN tabanlı Tek Çekim Çoklu Kutu Dedektörü (SSD) yöntemi kullanılmış ve meyvelerin 3 boyutlu konumu için stereo kamera kullanılarak ters kinematik hesaplamaları yapılmıştır. Meyve tespit başarısı %90 olarak belirtilmiş olup bir meyve için toplama süresi 16 saniye olarak ölçülmüştür [10]. 2020 yılında otonom tatlı biber hasat robotunun geliştirilmesi üzerine yapılan çalışmada, RGB Derinlik (RGB-D) kamera ile biberler ve saplarının nesne tespiti için semantik bir segmentasyon algoritması kullanılarak ayrı renklerde işaretlenmiştir. Kesim noktaları görüntü işleme yöntemleri kullanılarak belirlenmiş ve test prosedürleri nedeniyle robot toplama için özel olarak hazırlanan serada %61 başarı ile toplandığı raporlanmıştır. Bir ürün için toplama süresi 24 saniye olarak belirtilmiştir [11]. Diğer bir

çalışmada ise 3 eksenli robot kol kullanılarak elma hasadı yapılmıştır. Bu çalışmada nesne tespiti için Maske Bölgesi Tabanlı Evrişimli Sinir Ağları (Mask-R-CNN) kullanılarak saha performansı %64,06 olarak belirlenmiştir [12]. 2021 yılında, üzüm salkımı hasat robotu için nesne algılama algoritması tasarlanmıştır. Nesne tespiti için Mask R-CNN, segmentasyon maske bulutu ve üzüm silindirik modelini oluşturmak için Rastgele Örnek Konsensüs (RANSAC) algoritması kullanılmıştır. Bu model sayesinde meyve tespitindeki performansın artırılacağı ve toplama sırasındaki hataların azalacağı ön görülmüş, testler sonucunda %89,53 başarı elde edildiği belirtilmiştir [13]. 2021 yılında, domates hasat sistemi için 6 eksenli robotik kol üzerine vantuzlu ve makaslı özel bir kol tasarımı yapılmıştır. Nesne algılama için Sadece Bir Kez Bak (YOLOv3) kullanılmış ve test ortamında nesne algılama başarısı %90 olarak elde edilmiştir [14]. 2022 yılında, Mask R-CNN algoritması ve RGB-D kamera kullanılarak armut ve elma meyvelerinin otonom hasadı için bilgisayarla görme çalışılmıştır. Meyveler için 2 boyutlu görüntüler kullanılarak nesne tespiti yapılmıştır. Armut için alt bölgeler belirlenerek CNN tabanlı bir algoritma ile %87 doğrulukla olgunluk ayrımı yapılmıştır. Dış ortamda 3 boyutlu uzayda elde edilen görüntülerin eksik kısımlarının 2 boyutlu algılama ile tamamlanarak performansın artırıldığından bahsedilmiştir [15]. Literatürdeki otonom hasat robotları incelendiğinde yaprak, dal, yapışık meyve, üst üste gelen meyve, önu kapalı meyve ve yetersiz aydınlatma altında bir meyvenin hasat uygulamalarında performansın düştüğü bildirilmiştir [16]. Bu nedenle genel olarak tarla/sera ortamının toplanacak ürüne göre düzenlenmesinin otonom hasat performansını artırdığı belirtilmektedir. Fakat bu durum otonom hasat uygulanabilir ürün sayısını sınırlı hale getirmektedir.

Çalışmamızda ülkemizde yaygın olarak Akdeniz Bölgesi'nde yetiştirilen kamkat meyvesi üzerinde otonom hasat robotu uygulaması yapılmıştır. Bu kapsamda meyveleri mobil platform üzerinden toplamak için 6 eksenli robotik kol tasarlanmıştır. Derin öğrenme ağlarında kamkat meyvesi için özel olarak eğitilmiş yapı ile nesne tespiti sağlanmıştır. Derinlik kamerası ve görüntü işleme teknikleri [17] kullanılarak robot tutma ucu için toplama noktası belirlenmiştir. Yapay sinir ağı tabanlı ters kinematik model kullanılarak robot kol hareket ettirilerek toplama işlemi gerçekleştirilmiştir. Ağaçta olgun kamkat meyvelerinin tespiti tamamlanana kadar mobil platform ağacın etrafında hareket ettirilecektir. Sonuç olarak otonom bir hasat robotu tasarlanmış ve tarla uygulamalarında yaprak, dal, yapışık meyve, üst üste gelen meyve, önu kapalı meyveden kaynaklanan hasat problemlerine çözüm sunulmaktadır.

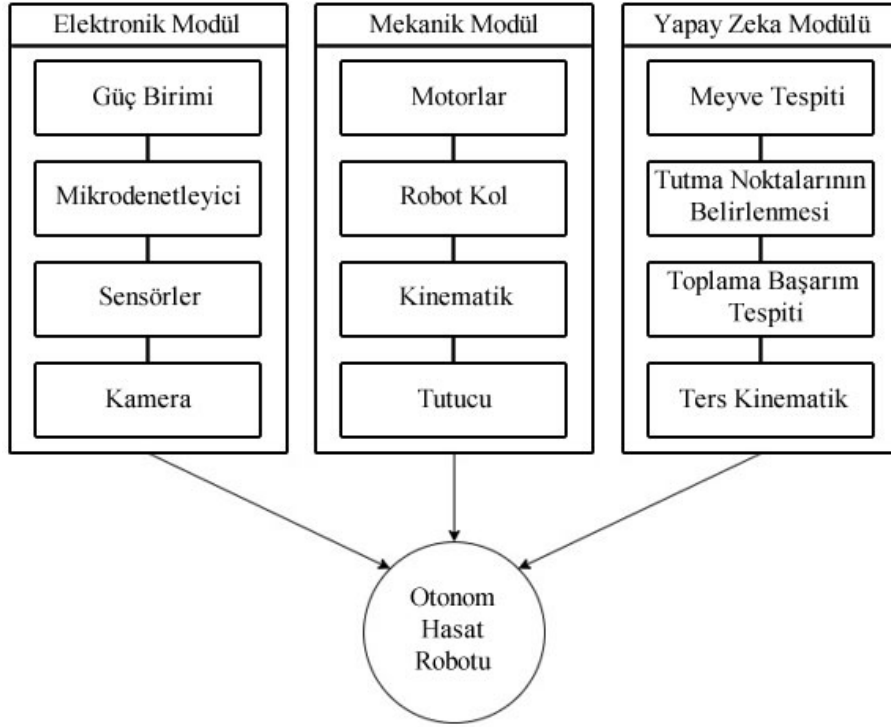
## 2. Teorik Metot (Theoretical Method)

Otonom hasat robotu projesi, Şekil 1'de gösterildiği gibi yapay zekâ, mekanik ve elektronik modüller olmak üzere üç temel modülden oluşmaktadır. Bu modüllerin detayları alt bölümlerde açıklanmaktadır.

### 2.1. Veri Kümesini Oluşturma (Creating the Dataset)

#### 2.1.1. Kamkat Meyve Veri Seti (Kumquat Fruit Dataset)

Kamkat meyve tespiti için Şekil 2'de görüldüğü gibi farklı açı, uzaklık ve sayıda meyve içeren kamkat fidanlarından 220 adet resim çekilerek veri seti oluşturulmuştur. Bu görüntüler üzerinde gürültü (%30 tuz biber gürültüsü) ve bulanıklaştırma (%10) yöntemleri kullanılarak görüntü sayısı 312 ädete çıkarılmıştır. Böylece hem veri artırımı sağlanmış hem de gerçek tarım alanlarında görüntüde oluşabilecek olumsuz koşulların nesne algılama performansına etkisi indirgenmiş olacaktır [18]. Resimlerde 2962 adet kamkat meyvesi etiketlenerek derin öğrenme algoritması için gerekli veri seti oluşturulmuştur.



Şekil 1. Otonom hasat robotunun genel yapısı (General structure of autonomous harvesting robot)



Şekil 2. Veri seti örnek görüntüleri (Dataset sample images)

### 2.1.2. Robot Kol Açılı Veri Seti (Robot Arm Angle Dataset)

BCN3D robotunun ters kinematik hesaplamasında dönüşüm matrisleri kullanılarak açı bilgilerinden oluşan bir veri seti elde edilmiştir. Dönüşüm matrisini hesaplamak için D-H notasyonu uygulanır. Bunun için eklemler (Q1...Q6), koordinat eksenleri (Xi, Yi, Zi), eklemler arası mesafe ve robot kol üzerindeki dönüş yönleri tanımlanır. BCN3D robot koluna ilişkin bu bilgiler Şekil 3' te gösterilmektedir.

Robot kolu ters kinematik hesaplama algoritması kapsamında 6 eksenli robot kol için eklemlerin üst ve alt hareket limitleri (Tablo 1) kullanılarak rastgele 15000 sıra eklemler açısı verisi ve BCN3D robotu için D-H parametreleri tablosu oluşturulmuştur (Tablo 2).

Tablo 2'deki D-H verilerine göre her bir eklemler için dönüşüm matrisleri hesaplanmış ve bunların ters çarpımı ile ileri kinematik uç efektörün konumunu(xyz) veren Eş. 1'deki Dönüşüm Matrisi(DM) ortaya çıkmıştır.

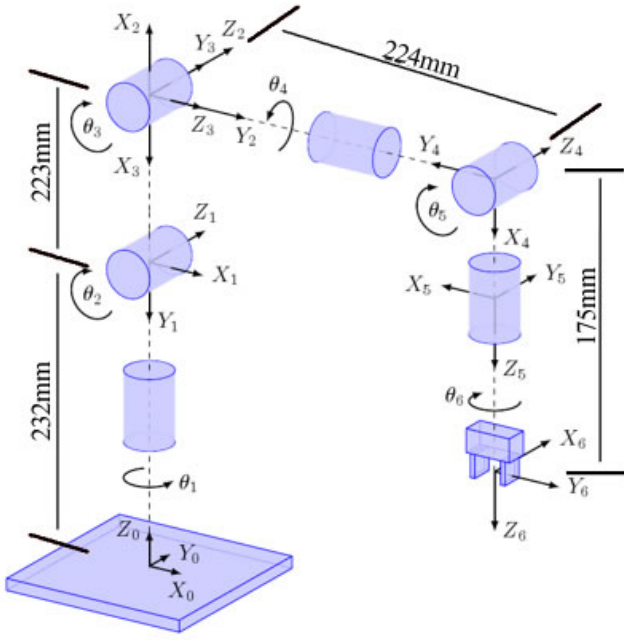
Tablo 1. BCN3D robot kol eklemleri alt ve üst hareket sınır dereceleri

(BCN3D robot arm joints lower and upper motion limit degrees)

Eklemler	Alt Limit(°)	Üst Limit(°)	Aralık(°)
1	-90	90	180
2	-100	100	200
3	-120	120	240
4	-180	180	360
5	-80	80	160
6	0	360	360

$$DM = \begin{bmatrix} 1 & 0 & 0 & 224 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 280 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

Rastgele oluşturulmuş açı verileri kullanılarak dönüşüm matrisine göre uç efektörün x, y ve z koordinatları hesaplanarak veri seti tamamlanmıştır.



**Şekil 3.** 6 eksenli robot kolün kinematik zinciri (Kinematic chain of 6-axis robotic arm) [19].

**Tablo 2.** Robot kol için D-H parametreleri (D-H parameters for the robot arm)

Joint	d(mm)	r(mm)	$\alpha$ (rad)	$\theta$ (rad)
J0	232	0	$-\pi/2$	0
J1	0	223	0	$-\pi/2$
J2	0	0	$\pi/2$	$\pi$
J3	224	0	$-\pi/2$	0
J4	0	0	$\pi/2$	$\pi/2$
J5	175	0	0	0

d: örtüşen bağlar arasında eklem kayması

r: iki eksen arasındaki eklem uzunluğu

$\alpha$ : iki bitişik eksen arasındaki eksen açısı

$\theta$ : iki bitişik uzuv arasındaki eklem açısı

## 2.2. Derin Öğrenme Yöntemi (Deep Learning Method)

Çalışmada, derin öğrenme yöntemleri içerisinde gerçek zamanlı nesne tespitinde hız ve doğruluk açısından başarılı bir performans sergileyen YOLO (Sadece Bir Kez Bak) algoritmasının YOLOv4, YOLOv5, YOLOv6 ve YOLOv7 mimarileri kullanılmıştır. YOLO algoritması nesne tespitini tek bir regresyon problemi olarak değerlendirir. Giriş görüntüsünü  $N \times N$ 'lik ızgaralara bölerek kullanır ve hızlı sonuçlar verir.

YOLOv4 algoritması 106 katmandan oluşmaktadır. Yapısında CNN [20], residual bloklar, skip connections, up-sampling, Leaky Relu, intersection over union, non-maximum suppression gibi yöntemler kullanılmaktadır. Özellik çıkarma katmanlarından önce  $416 \times 416$ ,  $618 \times 618$  vb. boyutlandırma yapıp daha sonra özellik piramidi ağ yöntemi ile öznetelik tespiti yapılmaktadır. Tespit işleminden sonra, nesnenin merkez noktasını, en boy uzunluğunu ve sınıfların tahmin oranlarını içeren tahmin dizileri ( $[t_x \ t_y \ t_w \ t_h \ P_o \ P_1 \ P_2 \dots P_c]$ ) oluşturulmaktadır. Nesne tespiti, en yüksek güven endeksi değerleri üzerinden gerçekleştirilmektedir [21].

Diğer model YOLOv5, genel mimarisi üç kısımdan oluşmaktadır. İlk kısım Omurga ağı, çok sayıda evrişim ve havuzlama katmanıyla giriş

görüntülerinden farklı boyutlarda özellik haritaları çıkaran bir yapısıdır. Boyun ağı, omurgadan gelen özellik haritalarını birleştiren ikinci kısımdır. Üçüncü kısım çıkış bölümünde ise nesne algılama ve sınıflandırma gerçekleştirilir. YOLOv5 mimarisinde odaklanma modülü görüntüleri dilimlere ayırıp-birleştirerek alt örnekleme sırasında özellikleri daha iyi çıkartabilmesini sağlamaktadır [22].

Diğer bir model YOLOv6 mimarisi, YOLO omurgasını ve boynunu, donanım düşünülerek yeniden tasarlayarak EfficientRep Omurgası ve Rep-PAN boyun yapısını sunmaktadır [23]. Son model YOLOv7 mimarisi, genişletilmiş verimli katman toplama ağı (E-ELAN) yapısını kullanmaktadır. Bu yapı karşılaştırma, genişletme ve birleştirme kardinalitesi kullanarak gradyan yollarını kaybetmeden model öğrenme yeteneğini geliştirmektedir [24].

## 2.3. Sınıflandırma Modeli için Performans Ölçümleri (Performance Measurements for Classification Model)

Sınıflandırma algoritmalarının performans değerlendirilmesi için farklı metrikler kullanılabilir. Bu çalışmada sınıflandırma ile ilgili performans kriterleri duyarlılık (Eş. 2), özgüllük (Eş. 3), F1 puanı (Eş. 4) ve doğruluk (Eş. 5) olarak belirlediğimiz metrikler kullanılacaktır [25]. Bu hesaplamalar belirlenirken Tablo 3'te gösterilen karışıklık matrisi oluşturulur ve hesaplamalar bu tablo bilgilerine göre yapılmaktadır. Karışıklık matrisi belirtilen doğru pozitifler (TP), sınıflandırıcı tarafından pozitif sınıfa ait verilerden kaç tanesinin doğru şekilde sınıflandırıldığını temsil etmekte; doğru negatifler (TN), sınıflandırıcı tarafından negatif sınıfa ait verilerden kaç tanesinin doğru şekilde sınıflandırıldığını temsil etmektedir. Diğer taraftan yanlış pozitifler (FP), gerçekte pozitif sınıfa ait olan bir verinin sınıflandırma sonucunda negatif sınıf olarak etiketlenmesi; yanlış negatifler (FN), gerçekte negatif sınıfa ait olan bir verinin sınıflandırma sonucunda pozitif sınıf olarak etiketlenmesidir.

**Tablo 3.** Sınıflandırma sonucu karışıklık matrisi (Classification result confusion matrix)

	Tahmin Edilen Sınıf	
	Pozitif	Negatif
Gerçek Sınıf	TP	FN
	FP	TN

$$\text{Duyarlılık} = \frac{TP}{TP+FN} \quad (2)$$

$$\text{Özlülük} = \frac{TN}{TN+FP} \quad (3)$$

$$F1 \text{ Puanı} = 2 \times \frac{\text{duyarlılık} \times \text{özlülük}}{\text{duyarlılık} + \text{özlülük}} \quad (4)$$

$$\text{Doğruluk} = \frac{\sum TP + \sum TN}{\sum (TP+TN+FP+FN)} \times 100\% \quad (5)$$

## 2.4. Robot Kolu için ML Tabanlı Ters Kinematik (ML Based Inverse Kinematic for Robot Arm)

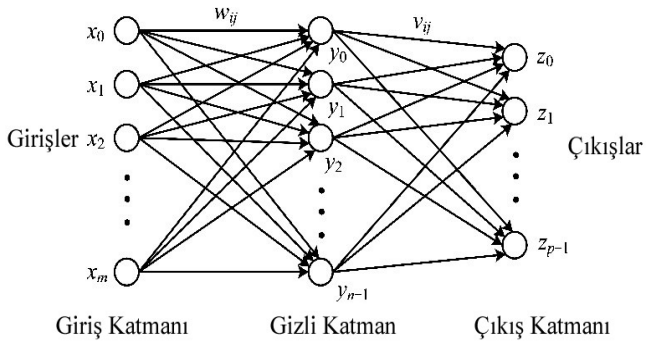
Robot kolun uç efektörünün meyveye yönelmesi için meyvenin x, y ve z konum bilgileri kullanılarak hareket planlaması ve kinematik hesaplamaları yapılmalıdır. Bu hesaplama ile 6 eksenli bir robotik kolda her meyve için  $2^6=64$  farklı kinematik çözüm ortaya çıkacaktır. Ters kinematik hesaplamaları için farklı hesaplama yolları vardır [26, 27]. Fakat bu yöntemlerin karmaşık hesaplama formülleri, yüksek hesaplama maliyetleri ve eklem sayısı arttıkça çözüm bulunamadığı durumlar gibi sorunları vardır. Bu sebeple makine öğrenmesi destekli ters kinematik hesaplama yöntemi tercih edilmiştir. Bu amaçla tahminlerde farklı türde yapay sinir ağı modelleri kullanılmıştır. Ters kinematik hesaplamada analiz için Radial Basis Function (RBF),



Extreme Learning Machine (ELM) ve Multi-Layer Perceptron (MLP) modelleri tercih edilmiştir. Tek bir gizli katmana sahip tam bağlantılı ağ yapısı Şekil 4'te gösterilmiştir [28]. RBF, çok sayıda bağımsız değişkenli problemler için kullanılan ağ yapısı gerektirmeyen bir modeldir [29]. ELM, eğitim sırasında tüm ağırlıkların yenilenmesine ihtiyaç duymayan bu sebepten dolayı eğitim hızı yüksek bir model olarak karşımıza çıkmaktadır [30].

MLP, bir veya daha fazla gizli katmana sahip ileri beslemeli bir yapay sinir ağıdır [31]. MLP, doğrusal olmayan karmaşık işlevi gerçekleştirebilir. Burada giriş olarak meyvelerin konumları (x, y, z) verilmiş, çıkışta ise eklem açıları (Q<sub>1</sub>, Q<sub>2</sub>, Q<sub>3</sub>, Q<sub>4</sub>, Q<sub>5</sub>, Q<sub>6</sub>) istenmiştir. Tüm veriler Eş. 6'daki normalizasyon formülü kullanılarak 0-1 arası normalize edilmiştir.

$$X_{\text{Norm}} = \frac{(X - X_{\text{min}})}{(X_{\text{max}} - X_{\text{min}})} \quad (6)$$



Şekil 4. Bir gizli katmana sahip MLP grafiği (Graph of a MLP with one hidden layer)

(Xi: giriş nöronu, Yi: gizli nöron, Zi: çıkış nöronu, Wij-Vij: Ağırlıklar)

Bu algoritmalar daha iyi hata oranı ve daha iyi tepki süresi elde etmek için incelenmiştir. Bu çalışmada sunulan algoritmaların performansını hesaplamak için Eş. 7'deki MAE formülü kullanılmıştır [32].

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

n = hata sayısı,  
 $\Sigma$  = toplama sembolü,  
 $|y_i - \hat{y}_i|$  = mutlak hatalar. (7)

### 3. Deneysel Metot (Experimental Method)

#### 3.1. Meyve Tespit Modülü (Fruit Detection Module)

Çalışmanın meyve algılama modülünde kullanılacak derin öğrenme algoritmasının yapılandırılması ve nesne algılama süreçlerinin açıklandığı kısımdır.

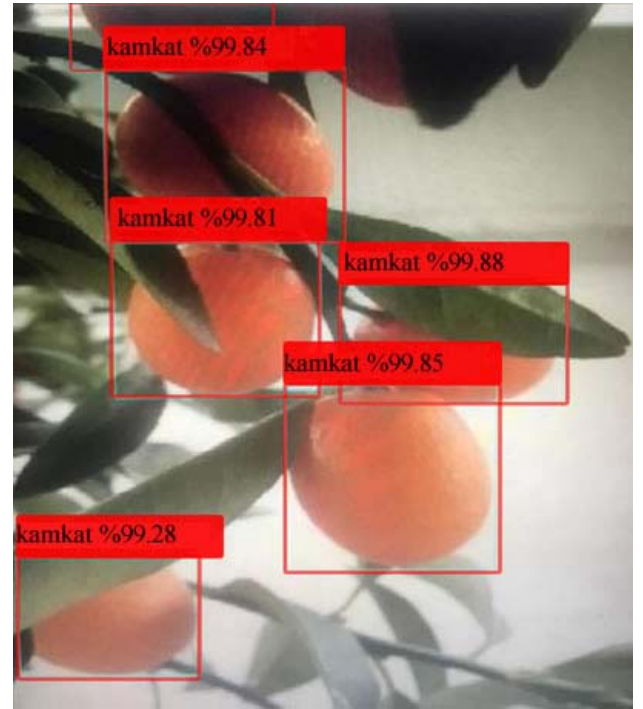
##### 3.1.1. Derin öğrenme algoritması (Deep learning algorithm)

Derin öğrenme algoritmalarında veri seti %80 eğitim verisi ve %20 test verisi olarak ayarlanmıştır. Kullanılacak modellerin mimarileri değişse de benzer adımlarla eğitim işlemleri gerçekleştirilmektedir. Bu kapsamda ilk olarak etiketli veri seti Google Drive'a yüklenmiştir. Eğitimler Google Colab üzerinden yapılmıştır. Yapılandırma işlemlerinde öncelikle Google grafik işleme birimi aktif edilerek drive üzerinden veri seti bu ortama aktarılmıştır. Sonraki adımda YOLOv4 için darknet kütüphanesi diğer versiyonlar için ise PyTorch kütüphanesi yüklenmiştir.

Bu adımlar tamamlandıktan sonra kamkat veri seti eğitim işlemleri için gerekli parametre ayarlamaları yapılmıştır. Parametre ayarlarında ilk olarak GPU, CuDNN (GPU ile Hızlandırılmış Derin Sinir Ağı Kütüphanesi) ve OpenCV aktif edilmiştir. Daha sonra her bir eğitim adımında kaç görüntünün kullanılacağını belirten batch size ve her bir batch işleminin kaç parça haline yapılacağını gösteren subdivision değerleri ayarlanmıştır. Bizim modelimizde batch değeri 64, subdivision değeri 16 olarak alınmıştır. Öğrenme oranı olarak belirlenen learning rate değerini 0,001 olarak kullanılmıştır. Eğitimin kaç adımda gerçekleşeceğini gösteren max\_batches parametresi için sınıf sayısı\*2000 formülasyonu kullanılmıştır. Ağımızdaki filtre sayıları için ise (sınıf sayısı + 5) \* 3 formülasyonu kullanılarak tek sınıfımız için 18 filtre kullanılmıştır. Model mimarileri bu şekilde ayarlanarak eğitim işlemine hazır duruma getirilmiştir. Eğitim işlemlerinin daha hızlı şekilde daha başarılı sonuçlara ulaşması için transfer learnin yöntemi kullanılmıştır. Bu kapsamda eğitime daha önceden COCO veri setinde eğitilmiş YOLO ağırlık dosyaları üzerinden başlanmıştır. Her bir mimari işlemi için eğitim işlemleri gerçekleştirilip test sonuçları ve ağırlık dosyaları elde edilmiştir.

##### 3.1.2. Meyve tespiti (Fruit detection)

Eğitim sonucunda elde edilen ağırlık dosyası kullanılarak Python'da OpenCV Kütüphanesi [33] ile bir nesne algılama algoritması oluşturulmuştur. Bu algoritma ile belirlediğimiz kamkat meyvelerinin çevrelerine tespit kutuları çizdirilmiştir. Bu kutucuklar çizilirken nesne algılama algoritmamızda elde ettiğimiz güven indeksleri kullanılarak %90'ın altındakiler elenmiştir. Şekil 5'te nesne tespiti sonrası güven endeksine göre geriye kalan kamkatları gösteren örnek çıktı gösterilmiştir.



Şekil 5. Kamkat nesne algılama (Kumquat object detection)

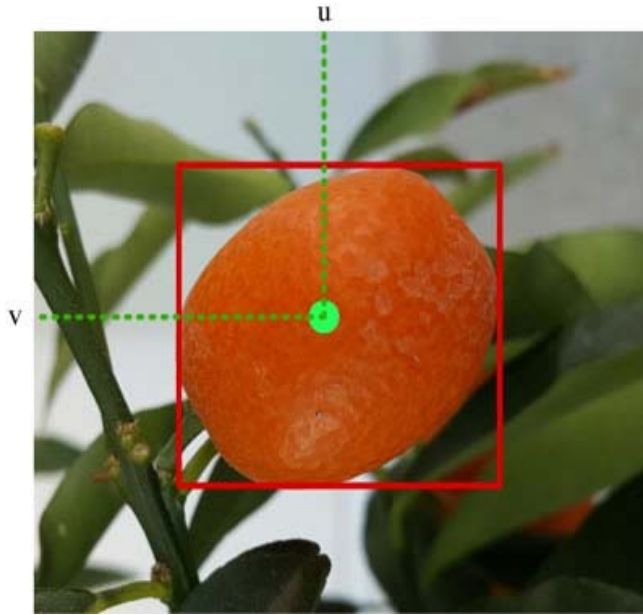
##### 3.1.3. Robot tutacağı için toplama noktası ve derinlik bilgilerinin belirlenmesi

(Determining the handle point and depth information for the robot handle)

Robot kolun cismi tespit ettikten sonra meyveleri hangi noktadan ve yapraklara zarar vermeden nasıl toplayabileceği konusunda

çalışmalar yapılmıştır. Bu kapsamda ilk aşamada çerçeveye alınan alan içerisinde kamkat bulunma oranları belirlenmiştir. Bu orana göre hangi alan içinde daha az yaprak var veya hangi meyve daha belirgin durumda şeklinde bir sınıflandırma bilgisi elde edilmektedir. Temel olarak geliştirilen algoritma ile hangi meyvelerin daha kolay hasat edilebileceği belirlenmiştir. İkinci aşamada robot kolun vakumlu kavrama ucunun meyvelere hangi noktadan yapışacağını belirlemek için çerçeve alan içerisindeki turuncu piksellerin en yoğun olduğu bölgeyi işaretleyen yapı algoritmaya dâhil edilmiştir. Bulunan nokta meyve üzerinde yaprağın olmadığı ve meyvenin en belirgin kısmı olduğu için vakum sisteminin meyveye yapışabileceği en ideal toplama noktasıdır (Şekil 6). Toplama noktalarının koordinat bilgileri yapılacak kinematik hesaplamalarda kullanılarak hasat işlemi gerçekleştirilmektedir.

Gerçek zamanlı nesne tespiti ve koordinat bilgilerini elde etmek için iki görüntü sensörlü, kızılötesi projektörlü ve RGB sensörlü Realsense D415 kamera kullanılmıştır. Bu kamera pythonda “pyrealsense2” kütüphanesi ile kullanılmıştır. Derinlik bilgisi, kamerada aynı yatay ekseninde bulunan ve mesafesi bilinen iki görüntü sensöründen (insan gözü dizisi gibi) gelen görüntülerin karşılaştırılmasından oluşmaktadır. Bu kameranın derinlik bilgisi ölçümlerinde 2 metre mesafede %2’den küçük hata oranı belirtilmiştir. Görüntüler RGB sensörü ile mesafelerine göre renklendirilerek RGB-D olarak adlandırılan derinlik renk haritası oluşturulmuş ve daha önce belirlediğimiz toplama noktasına karşılık gelen kısımlardaki derinlik bilgisi elde edilmiştir. Eş. 8’de gösterilen denklemler kullanılarak mevcut derinlik bilgisi ile toplama noktalarının gerçek düzlemdeki x ve y koordinatları belirlenmiştir.

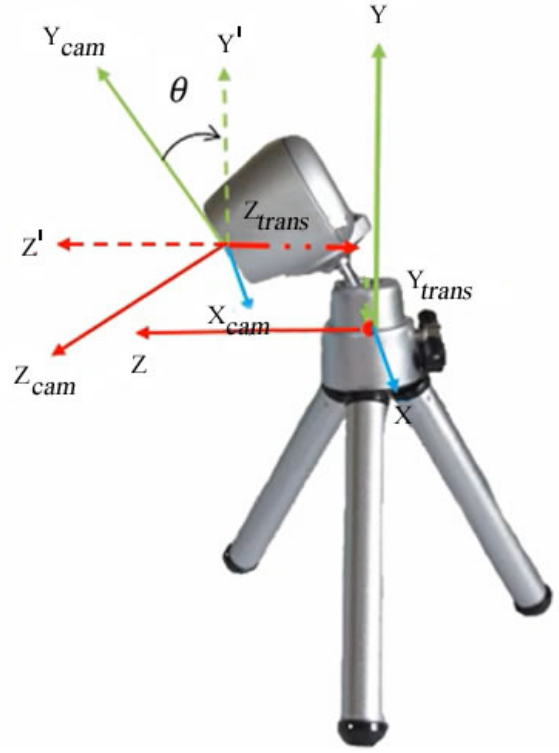


Şekil 6. Kamkat üzerindeki tutma noktası konumu (Handle point location on kumquat)

$$X=Z \frac{u-c_x}{f_x} \quad Y=Z \frac{v-c_y}{f_y} \quad (8)$$

Elde edilen bilgiler Eş. 9 kullanılarak Şekil 7’de gösterildiği gibi kamera konumlandırılmasına göre kamkat meyvesinin gerçek zamanlı toplama noktası koordinatlarına dönüştürülmüştür.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} X_{cam} \\ Y_{cam} \\ Z_{cam} \end{bmatrix} + \begin{bmatrix} 0 \\ Y_{trans} \\ Z_{trans} \end{bmatrix} \quad (9)$$



Şekil 7. D415 kamera konumlandırılması (D415 camera positioning)

### 3.1.4. Toplama için doğrulama (Validation for picking)

Literatürdeki birçok otonom hasat robotu çalışmasında, yaprakların arkasında daha az görünen meyvelerin nesne algılama performansını düşürerek toplama performansını olumsuz yönde etkilediği belirtilmektedir [7-10]. Geliştirilen vakum tutma ucu için en uygun toplama noktasının belirlenmesi yaklaşımı ile bu problem çözülmüştür. Ayrıca algılama ve doğrulama algoritması olarak adlandırılan toplama için yeni bir algoritma geliştirilmiştir. Geliştirilen algoritmada nesne tespiti yapıldıktan sonra çizilen çerçeve kutu içindeki alanda uygunlaşmış kamkat meyvesinin toplanıp toplanamayacağını belirlemek gerekmektedir. Bu nedenle, bu alan içindeki kamkatın kapladığı alanın oranı dikkate alınmıştır. Çerçeve kutusu içerisindeki kamkatın alanı ne kadar büyük olursa yaprak veya dal gibi harici engellerin hasat işlemini olumsuz etkileme ihtimali o kadar düşük olacaktır. Bu kapsamda çerçeve alan içinde bulunan kamkatın oranı %50 ve üzeri ise tespit yapılan açıdan toplamaya uygun olarak kaydedilmiştir. Elde edilen uygun koleksiyon setinde toplama noktasının belirlenmesi için bu alanda turuncu piksel yoğunluğunun en yüksek olduğu yer işaretlenmiştir. Bir açıdan bu işlemle işaretlenen tüm kamkatlar toplandığında ağacın etrafında tam tur tamamlayana kadar döndürülerek aynı işlemler tekrarlanmıştır. 36 derecelik açılarla yapılan 10 tekrarlı hasat işlemi ile toplama noktası işaretlenen tüm kamkatlar hasat edilmeye çalışılmıştır.

Çalışmada çerçeve içerisinde kamkat bulunma oranı %50 ve üzeri olan kamkatlar için toplama noktası işaretlenerek hasat işlemi yapılmıştır. Bu oran belirlenirken %25, %35, %50 ve %60 değerleri için hasat test işlemleri yapılmıştır. Bu kapsamda farklı açılardan tespit edilen %25 ile %35 arasında kamkat bulunma oranı olan çerçevelerde hasat başarımı 25 meyve için 5 başarılı toplama olmuştur. %35 ile %50 arasında kamkat bulunma oranına sahip çerçevelerde hasat başarımı 25 meyve için 8 başarılı toplama gerçekleşmiştir. %50 ile %60 arasında kamkat bulunma oranına sahip çerçevelerde hasat başarımı 25 meyve için 13 başarılı toplama

olmuştur. %60 ve üzeri kamkat bulunma oranına sahip çerçevelerden alınan 25 meyve örneğinde başarılı hasat 19 adet olmuştur. Fakat %60 bulunma oranı altındaki kamkatlarda hasat işlemi yapılmadığında geriye fazla miktarda hasat edilmemiş kamkat kalmaktadır. Sonuç olarak tek açıdan nesne tespiti yapıldığında %50 ve üzeri kamkat bulunma oranına sahip çerçevelerdeki kamkatların hasat işleminin yapılmasının en ideal referans değeri olduğu tespit edilmiştir. Bir açıdan yapılan tespitite %50 referans değerinin altında kalan kamkatların da diğer açılardan yapılan nesne tespitinde %50 değerinin üzerinde sonuçlar vererek farklı açılarda hasat edilen kamkatlar arasında yer aldığı gözlemlenmiştir.

Şekil 8’de kamkat bulunma oranı %25 ve üzeri olan çerçevelerde toplama noktasının işaretlendiği bir örnek yer almaktadır. Bu örnekte yer alan %25,66, %30,38, %38,51 ve %29,13 kamkat bulunma oranları olan çerçevelerde yer alan kamkatların robotik sistemin tutma ucunun çapı ve bu tutuma ucunun yapışacağı alanın yetersiz olması sebepleri ile hasat başarımının çok düşük kalacağı görülmektedir. Bu kamkatların hasadının %50 ve üzeri bulunma oranı ile tespit edildikleri açılardan yapılması daha başarılı sonuçlar vermektedir.

### 3.2. Mekanik Modül (Mechanical Module)

Mekanik kısım üç başlık altında incelenmiştir. Mekanik sistemin genel yapısı Şekil 9’da gösterilmiştir.

#### 3.2.1. Robot kol (Robot arm)

Kamkatların küçük ve birbirine yakın olduğu düşünülerek mevcut robot kol tasarımları incelenmiş ve toplama işlemi yapabilen 6 eksenli robot kol tasarımı seçilmiştir (Şekil 10). Açık kaynak olarak sunulan robot kolun dokümanları GitHub üzerinde paylaşılmaktadır [34]. Robot kol tasarımını amacımıza göre modifiye ederek kamkat meyvesi toplamaya elverişli hale getirilmiş ve uygun algoritmalarla karşılaşılan problemlerin çözülmesi sağlanmıştır.

Robot kolunun tutucu kısmında bir adet servo motor bulunmaktadır. Diğer mafsallarda hassasiyet sağlayacak farklı güç ve derecelerde 6

kademeli motora sahiptir. Parçalar, PLA (Polilaktik Asit) + filament kullanılarak 0,2 mm katman yüksekliği ve %25 dolum yoğunluğu ile 3D yazıcı ile basılmıştır. Step motorlarda dezavantaj gibi görünen robot kolun güç sorunu, dişli kasnak ve kayış sistemi yardımıyla minimize edilmiştir.

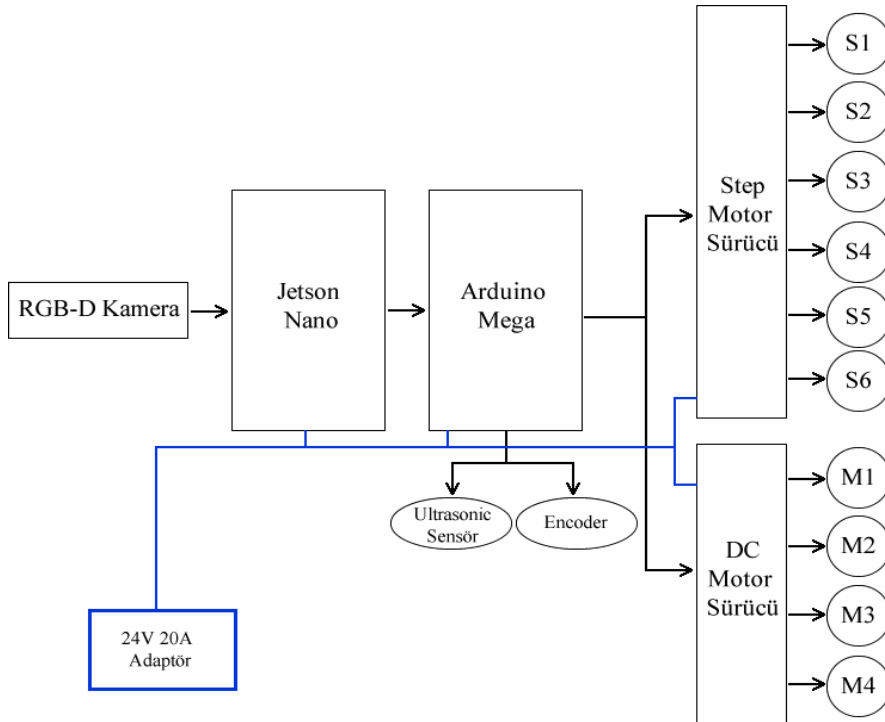
#### 3.2.1. Tutucu modeli (Gripper model)

Kamkat meyvelerini tutup çekerek toplamak zordur ve güç gerektirir. Bunun için 2 parmaklı ve 3 parmaklı farklı kısıp modelleri denenmiştir. Testler sonucunda dalların gerilmesinden dolayı direkt çekmenin toplama işlemi başarılı bir şekilde gerçekleştirmediği gözlemlenmiştir. Bunun yerine döndürerek koparma işlemi değerlendirilmiştir. Bu işlem klasik tutucu ile yapılırken meyve sert tutulduğunda meyve yüzeyinin zarar gördüğü, yumuşak tutulduğunda ise döndürme işleminin yapılmadığı görülmüştür. Bu sebeple tutucunun hem sert bir tutuş sağlaması hem de meyveye zarar vermemesi gerekmektedir. Bu noktada vakum uç kullanımına karar verilmiş ve her iki özelliği de sağlayarak meyvelerin zarar görmeyen toplanmasını sağlamıştır. Robot kol tasarımında, görüntü işleme algoritmamızı üzerine kurguladığımız tutucu kısmı tamamen bu çalışmaya özel vakum ve kavrayıcı parmaklardan oluşacak şekilde tasarlanmıştır. Vakum tutucu tasarım modeli Şekil 11’de gösterilmektedir.

Vakum ucu, algoritmada işaretlenen noktada meyveye yapışır, meyveyi biraz kendine doğru çeker ve meyveyi kavramak için tutucu kapatılır. Bu işlemden sonra kısıp uç 2 tur dönerek meyvenin dal ile olan bağı zayıflatır ve ardından meyve çekilerek çıkarılır (Şekil 12). Meyvenin koparılması işlemi sonunda gövdenin bir kısmının meyvede kalması bir avantajdır ve ayrıca tazelik süresini de uzatır.

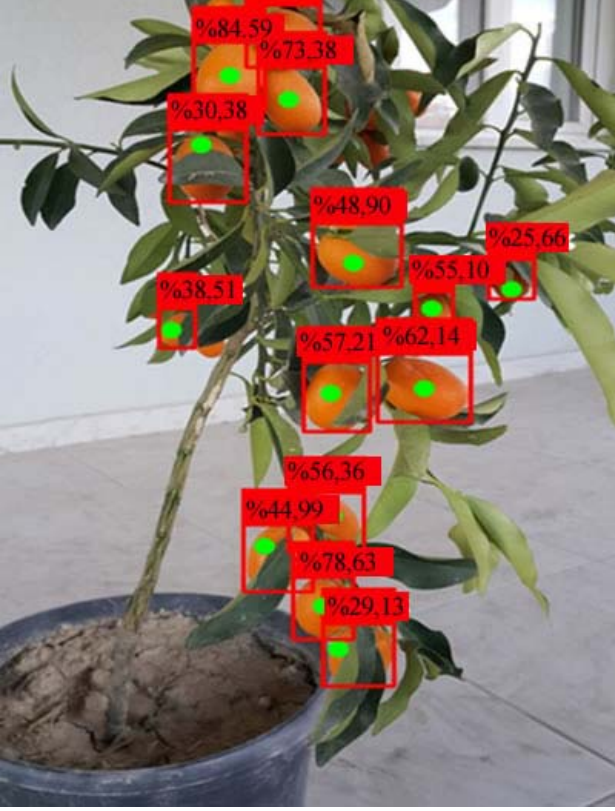
#### 3.2.2. Mobil platform (Mobile platform)

Üzerindeki robotik sistemi taşıyarak kamkat ağacına yönelmesini sağlamak için 6 tekerlekli mobil platform oluşturulmuştur. Robot kol ve toplanan meyvelerin ağırlığı hesaplandığında 6 adet kara tekerleğine ihtiyaç duyulmuştur. Hareket sistemi için saha





koşullarında yeterli kabiliyeti sağlayan yüksek torklu 12 V-76 rpm Enkoderli (Adım sayan) DC Motor kullanılmıştır. Enkoder kat edilen yolu hesaplamaktadır.



**Şekil 8.** Alan içindeki kamkat bulunma oranının belirlenmesi ve toplama noktalarının işaretlenmesi  
(Determining the occurrence rate of kumquat in the area and marking collection points)

### 3.2.3. BCN3D robot kolu için ML tabanlı ters kinematik (ML based inverse kinematic for BCN3D robot arm)

Bu bölümde BCN3D robotunun dönüşüm matrisi kullanılarak açılımlardan oluşan 15000 satırlık veri seti ile farklı yapay sinir ağı algoritmalarında eğitimler yapılmış ve verilen meyve koordinatlarından uygun eklem açıları makine öğrenmesi tabanlı hesaplanmıştır. Tablo 7'de detayları verilen algoritma sonuçlarına göre en iyi model MLP algoritması olmuş ve yapılan eğitim sonucunda her bir eklem için ortalama mutlak hata (MAE) değerinin ilk 5 eklemden %0,1'den az, uç efektör eklemden ise hatanın %0,1'den büyük olduğu tespit edilmiştir.

### 3.3. Elektronik Modül (Electronic Module)

Elektronik kısımda kontrolörler, motor sürücüler, sensörler ve enerji kaynakları yer almaktadır.

#### 3.3.1. Kontrolörler ve motor sürücüler (Controllers and motor drives)

Robot üzerindeki tüm motorları, motor sürücülerini ve sensörleri yönetmek için gereken 23 adet giriş-çıkış pini Arduino Mega 2560 geliştirme kartında yer almaktadır. Step motorların enerji beslemesi ve hız kontrolü için kanal başına 3A akım verebilen TB6560 motor sürücü kartı kullanılmaktadır. Ramps V1.4 kontrol kartı, motor sürücülerinin kontrolü için kullanılmıştır. Derin öğrenme algoritmalarını mobil çalıştırmak için 4GB GPU kapasiteli jetson nano kullanılmıştır.



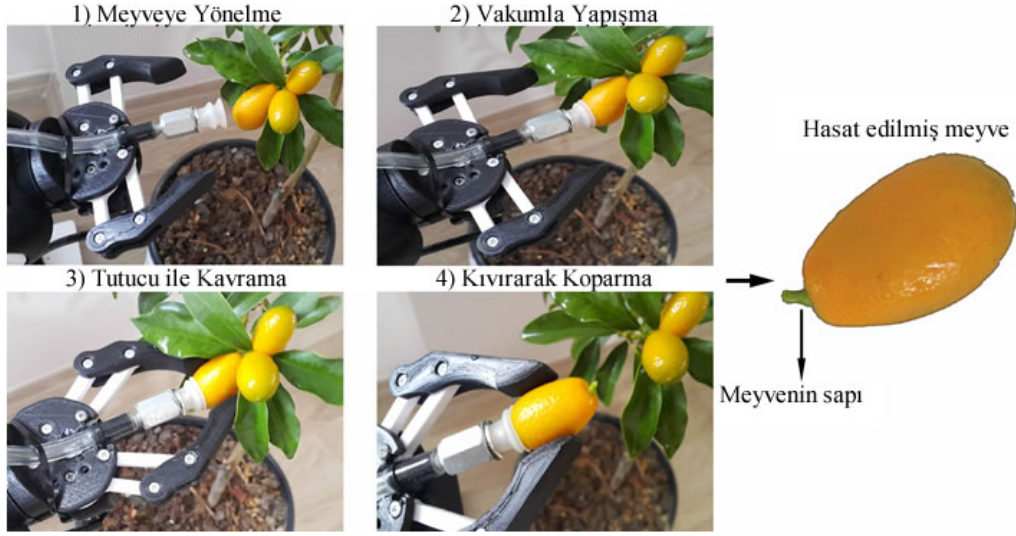
**Şekil 10.** Robot kol tasarımı (Robot arm design)

#### 3.3.2. Sensörler (Sensors)

Robotların ihtiyaç duydukları bilgileri alabilmeleri için sensörler kullanılmaktadır. Literatürde birçok otonom meyve hasadında karşılaşılan RGB-D kamera sistemi kullanılmıştır [4, 5, 6]. Kamera, robot kolunun arka diyagonalindeki tripod üzerinde sabit bir şekilde konumlandırılmıştır (Şekil 13).



**Şekil 11.** Vakum tutucu modeli (Vacuum gripper model)



Şekil 12. Meyve toplama adımları (Fruit picking steps)



Şekil 13. RGB-D kamera konumu (RGB-D camera location)

### 3.3.3. Enerji kaynağı (Energy supply)

Robot kolunun ihtiyaç duyduğu akım değeri 12V/0,3A olarak belirlenmiştir. Jetson nano kartının ihtiyaç duyduğu enerji kaynağı 5V/4A'dır. Arduino mega ve motor sürücülerinin ihtiyaç duyduğu enerji miktarı 24V/15A olarak hesaplanmıştır. Sonuç olarak 24V/20A güç kaynağı kullanılmıştır.

## 4. Sonuçlar ve Tartışmalar (Results and Discussions)

### 4.1. Yolo Algoritması Eğitim Sonuçları (Yolo Algorithm Training Results)

YOLOv4 modeli ile yapılan eğitim sonucunda elde edilen kayıp – ortalama keskinlik grafiği Şekil 14'te gösterilmektedir. İterasyon

adımları ilerledikçe kayıp miktarının azaldığı başarımın yükseldiği görülmektedir.

YOLOv5 modeli eğitimi sonucunda elde edilen başarımların grafikleri Şekil 15'te gösterilmektedir. Eğitim sonucunda recall (hatırlama) değeri 0,87, precision (keskinlik) değeri 0,97, F1 puanı 0,92 ve Map değeri 0,93 olarak elde edilmiştir.

YOLOv6 modeli eğitimi sonucunda elde edilen başarımların grafikleri Şekil 16'da gösterilmektedir. Eğitim sonucunda recall (hatırlama) değeri 0,60, precision (keskinlik) değeri 0,72, F1 puanı 0,81 ve Map değeri 0,89 olarak elde edilmiştir.

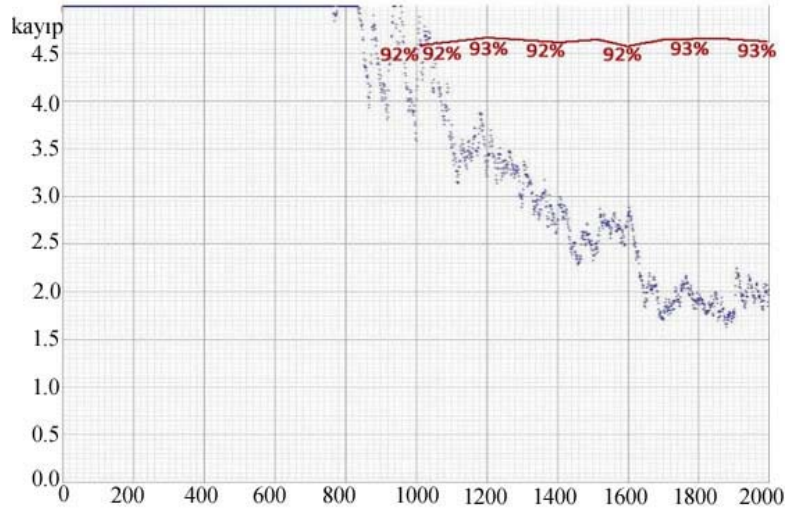
YOLOv7 modeli eğitimi sonucunda elde edilen başarımların grafikleri Şekil 17'de gösterilmektedir. Eğitim sonucunda recall (hatırlama) değeri 0,84, precision (keskinlik) değeri 0,95, F1 puanı 0,90 ve Map değeri 0,93 olarak elde edilmiştir.

YOLO mimarileri ile yapılan eğitim işlemleri sonucunda elde edilen sonuçlar Tablo 4'te gösterilmiştir. Sonuçlar incelendiğinde nesne tespit başarımında YOLOv5 ve YOLOv7 algoritmalarında yakın sonuçlar elde edilmiştir. Bu noktada FPS oranları incelendiğinde en iyi sonuçların YOLOv7 ile elde edildiği görülmüştür. Sonuç olarak otonom hasat robotunun gerçek zamanlı nesne tespiti işlemleri için YOLOv7 mimarisi kullanılmıştır.

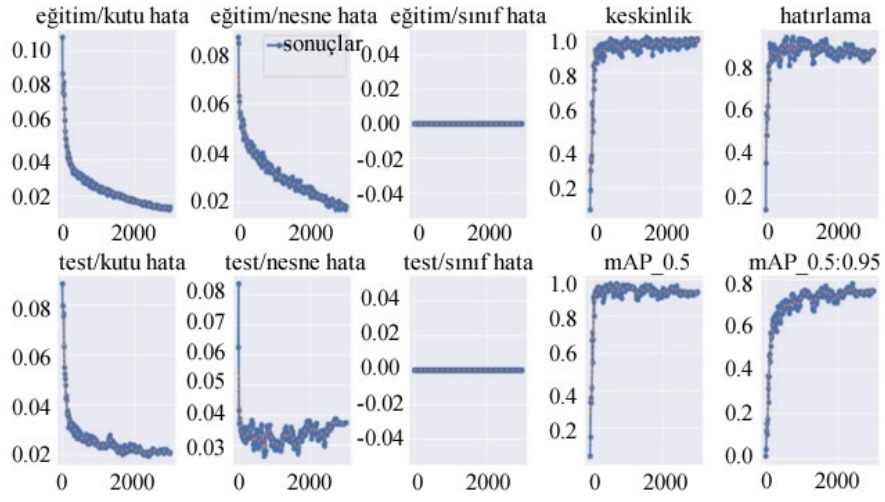
### 4.2. Sınıflandırma ve Performans Ölçümleri (Classification and Performance Measurements)

Test işlemi sonucunda en başarılı sonuçları veren YOLOv7 mimarisinin karmaşıklık matrisi Şekil 18'de verilmiştir.

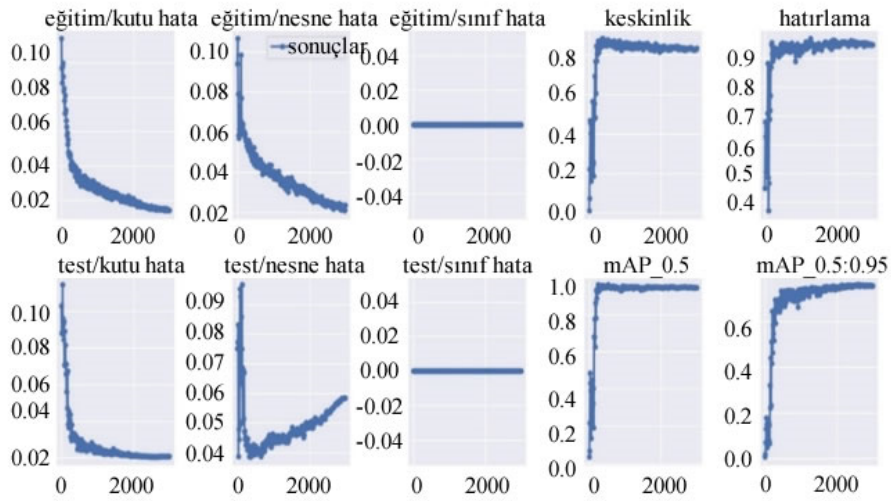
Test işlemleri sonucunda YOLOv7 mimarisi ile kamkat meyvesi tespitinde genel başarımlar %93 olarak ölçülmüştür. Literatürdeki çalışmalarda kamkat meyvesi özelinde bir çalışma bulunmadığı için diğer meyveler ile yapılan çalışmalarda değiştirilmemiş (yapraklar seyrletilmemiş veya yapışkan meyve toplanmamış) test ortamlarında yapılan nesne algılama performansları incelenmiştir. En başarılı olanları SSD ile %90 doğrulukta elma tespiti [10] ve YOLOv3 ile %90 doğrulukta domates tespiti [14]. Bu referanslar ile kamkat meyve hasat robotunda kullanılmak üzere %93 tespit doğruluğu yeterli görülmüştür.



Şekil 14. YOLOv4 kayıp – ortalama keskinlik grafiği (YOLOv4 loss – average precision chart)

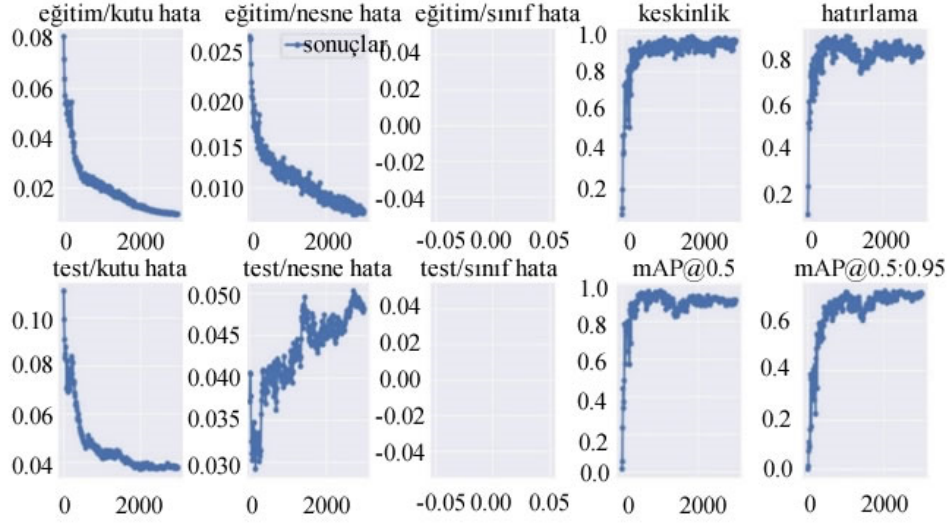


Şekil 15. YOLOv5 sonuç grafikleri (YOLOv5 results graphs)



Şekil 16. YOLOv6 sonuç grafikleri (YOLOv6 results graphs)

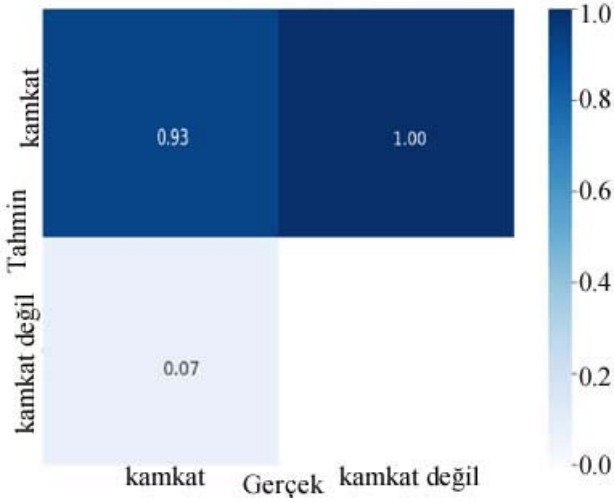




Şekil 17. YOLOv7 sonuç grafikleri (YOLOv7 results graphs)

Tablo 4. Modellerin FPS ve genel doğruluk karşılaştırma tablosu (FPS and overall accuracy comparison chart of models)

Modeller	Hatırlama (Recall)	Keskinlik (Precision)	F1 Puanı	Map50	Başarım Sonucu(%)	FPS(Saniyedeki Kare Sayısı)
YOLOv4	0,91	0,74	0,82	0,90	90	25
YOLOv5	0,87	0,97	0,92	0,93	93	29
YOLOv6	0,60	0,72	0,81	0,86	86	33
YOLOv7	0,84	0,95	0,90	0,93	93	35



Şekil 18. Karmaşıklık matrisi (Confusion matrix)

Meyve pozisyonlarına göre eklem açılarının ortalama MAE/Range değerleri için, makine öğrenme yöntemlerinden RBF, ELM, MLP yapıları test edilmiş ve sonuçları karşılaştırılmıştır (Tablo 5). Sonuç olarak MLP modelinin en iyi sonucu verdiği görülmüştür. Burada doğru hasat için eklemlerin doğru hareketi önemli olduğundan MLP modeli tercih edilmiştir.

Tasarlanan MLP algoritmalarının test hatası sonuçları incelenmiştir. En başarılı sonuca MLP algoritması ile 5 gizli katman ve her katmanda 120 nöron bulunan model ile ulaşılmıştır. Ortalama hata değeri tüm eklem toplamında %1'den az bulunmuştur. Bu noktada her bir eklem için ayrı ayrı bakıldığında en yüksek hata değeri son eklemden meydana gelmiştir (Tablo 6).

1890

Tablo 5. Eklem açılarının ortalama MAE/Range karşılaştırma tablosu (Average MAE/Range comparison chart of joint angles)

Makine Öğrenme Yöntemi	Ortalama MAE/Range(%)
RBF	0,182
ELM	0,158
MLP	0,116

Tablo 6. Eklemlerdeki hareket aralığındaki test hata değerleri (Test error values in the range of motion in the joints)

Joint	MAE/Range (%)
q1	0,03225
q2	0,02884
q3	0,04126
q4	0,05248
q5	0,01912
q6	0,52474

Literatürdeki benzer çalışmalarda olduğu gibi hata değerlerinde de başarılı sonuçlar elde edilmiştir. Bu amaçla yapılan bir çalışmada, NN algoritmaları kullanılarak 6 eksenli bir robotun eklem açıları için ters kinematik çözümü gerçekleştirilmiştir [35]. Önemli olan, çalışmamızda son efektörde vakum ucu olduğu için son eksen hata değerinin robot kolun meyve toplama performansını etkilememesidir.

#### 4.3. Otonom Hasat Testi Sonuçları (Autonomous Harvest Test Results)

Literatürdeki benzer meyve hasadı çalışmaları incelendiğinde, test aşamalarında modifiye ağaçlar kullanılarak (yaprakları seyreltilerek veya yapışkan meyveler kopararak) performansın yüksek olduğu ancak tarla koşullarında bu performansın sağlanamadığı belirtilmektedir. Çalışmada otonom hasat test işlemleri için 2 adet kamkat ağacı kullanılmıştır. Bu kapsamda ağaçlarda herhangi bir



değişiklik yapılmadığı için test işlemi doğal halde gerçekleştirilmiştir. Ağaç üzerinde robotik sistem açısından tespit edilen meyveler toplanır ve ardından ağaçta tespit edilen tüm meyveler toplanıncaya kadar tam bir tur yapılır. Sonuçlara bakıldığında 120 kamkat meyvesi hasat işlemi için meyve tespiti, toplama noktasının belirlenmesi, vakumlu kavrama ucu ile toplama ve sepete konulması işlemleri tam bir devir hasat performansı olarak nitelendirilerek %75 başarımla sağlanmıştır.

Literatürde farklı meyve türleri için yapılan çalışmaların sonuçları incelendiğinde doğal test ortamında performansın çok düşük olduğu bildirilmiştir. En önemli sorun, doğal görünümde toplanacak meyvenin yapraklarla kaplı olmasıdır [36]. Bu duruma örnek olarak çilekler üzerinde yapılan çalışmada test için modifiye meyvelerdeki performans %96, doğal ortamdaki başarı ise %53,6 olarak bulunmuştur [8]. Bir başka çalışmada elma hasadı için yapılmış, nesne tespiti için Mask-R-CNN mimarisi kullanılmış ve %90,5 nesne tespiti başarımla sağlanmıştır. Deneysel ortamda yapılan hasat testlerinde ise %64,06 oranında bir başarımla sağlanmıştır [37]. Çalışmamızda yaprak sorununa çözüm getiren bir yaklaşım ortaya konmuştur. Değiştirilmemiş durumda, nesne algılama performansı %93'te ve hasat performansı %75'te test edildi. Ayrıca çalışmamız sera ortamında değil, doğal saksı ağacı üzerinde gerçekleştirilmiştir. Otonom hasat robotlarının sadece sera ortamında değil normal tarım arazilerinde de kullanımına referans olması açısından önemlidir.

#### 4.4. Öneriler ve tartışma (Suggestions and Discussion)

Bu çalışmada, gerçek tarla koşullarına en yakın ortamda kamkat meyvesinin otonom hasadı test edilmiştir. Meyve ağacında herhangi bir değişiklik yapılmamıştır. Bu noktada otonom hasat robotunun hasat performansının %75'i sağlanmış olsa da gerçek tarla ortamında ticari bir ürün olarak kullanılması hala yeterli değildir.

Çalışmada genel toplama performansının %75 seviyesinde kalmasının sebepleri incelenmiştir. İlk olarak derinlik bilgisi ölçümlerinde karşılaşılan %1 - %1,4 hata oranları sebebiyle vakumlu tutma ucunun meyveye tam yapışmadığı durumlar gerçekleşmiştir. Sap ucuna mesafe sensörü eklenerek ve yapıştırmadan önce ince bir mesafe ayarı yapılarak bu durumun çözülebileceği düşünülmektedir. Karşılaşılan ikinci durum meyve toplandıktan sonra sepete konulurken düşmesidir. Kompresörün kademeli olarak çalıştırılması yani meyvenin dikey eksenenden yatay eksene getirilmesi ve sepete yaklaşırken daha yüksek kademelerde çalıştırılmasıyla bu sorunun çözülebileceği düşünülmektedir.

## 5. Simgeler (Symbols)

### 5.1. Kısaltmalar (Abbreviations)

HSV	: Hue Saturation Value
CNN	: Convolutional Neural Network
SSD	: Single Shot Multibox Detector
RANSAC	: Random Sample Consensus
YOLO	: You Only Look One
MLP	: Multi Layer Perceptron
MAE	: Mean Absolute Error

## 6. Sonuç (Conclusions)

Çalışma sonucunda %93 nesne algılama başarımla ve %75 genel hasat başarımla elde edilmiştir. Bu kapsamda, öncelikle hasat edilecek meyvenin cinsine göre tasarlanan tutucu için özel bir algoritma geliştirilmiştir. Bu sayede kamkatın toplanması için en uygun toplama noktası belirlenmiştir. Ek olarak bu nokta için x, y ve z koordinat bilgileri elde edilmiştir. Bu bölümde hız ve doğruluk dengesi açısından en başarılı sonuçları veren YOLOv7 algoritması ilk

aşamada meyveye özel geliştirmeler ve nesne tespiti eğitimleri yapılarak sınır kutularının çizilmesi için kullanılmıştır. Çalışmada ayrıca bir mobil robot kol tasarımı gerçekleştirilmiştir. Robot kolu ters kinematik hesaplamaları yapılırken sınır ağı modelleri incelenmiş ve farklı yaklaşımlara göre hassasiyet açısından MLP tercih edilmiştir. Robot koldaki önemli özgünlüklerden birisi de kamkat meyvesi için özel olarak tasarlanmış ve tüm algoritma tasarımı özel yapısına göre planlanmış olan vakumlu tutucudur. Geliştirilen bu yaklaşımlar test edilirken sahada olduğu gibi hiçbir şekilde değiştirilmemiş kamkat ağaçları üzerinde gerçekleştirilmiştir. Bu noktada literatürde yer alan birçok çalışmada meyvelerin üzerini örten yapraklar bir serada veya deneme ortamında çıkarıldıktan sonra testlerin yapıldığı görülmektedir. Bu kapsamda literatürdeki hasat robotları yaklaşımlarının dışında yeni yaklaşımlar da literatüre kazandırılmıştır. Benzer yaklaşımların sadece kamkat meyvesi için değil birçok ürünün otonom hasadı ile ilgili sorunları çözebileceği öngörülmektedir.

## Teşekkür (Acknowledgement)

Bu araştırma, 2021 TF-01 numaralı hibe kapsamında İskenderun Teknik Üniversitesi Bilimsel Araştırma Projeleri tarafından desteklenmiştir.

## Kaynaklar (References)

- Odegard I. Y. R., Van der Voet E., The future of food Scenarios and the effect on natural resource use in agriculture in 2050, *Ecological Economics*, 97, 51-59, 2014.
- Iqbal J., Islam R. U., Abbas S. Z., Khan A. A., Ajwad S. A., Automatizacija industrijskih poslova kroz mehatroničke sustave pregled robotike iz industrijske perspective, *Tehnički vjesnik*, 23 (3), 917-924, 2016.
- Hassan M. U., Ullah M., Iqbal J., Towards autonomy in agriculture: Design and prototyping of a robotic vehicle with seed selector, 2nd International Conference on Robotics and Artificial Intelligence ICRAI, 37-44, IEEE, 2016.
- Tanigaki K., Fujiura T., Akase A., Imagawa J., Cherry-harvesting robot, *Computers and electronics in agriculture*, 63 (1), 65-72, 2008.
- Almendral K. A. M., Babaran R. M. G., Carzon B. J. C., Cu K. P. K., Lalanto J. M., Abad A. C., Autonomous fruit harvester with machine vision. *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, 10 (1-6), 79-86, 2018.
- Feng Q., Zou W., Fan P., Zhang C., Wang X., Design and test of robotic harvesting system for cherry tomato, *International Journal of Agricultural and Biological Engineering*, 11 (1), 96-100, 2018.
- Luo L., Tang Y., Lu Q., Chen X., Zhang P., A vision methodology for harvesting robot to detect cutting points on peduncles of double overlapping grape clusters in a vineyard, *Computers in industry*, 99, 130-139, 2018.
- Xiong Y., Peng C., Grimstad L., From P. J., Isler V., Development and field evaluation of a strawberry harvesting robot with a cable-driven gripper, *Computers and electronics in agriculture*, 157, 392-402, 2019.
- Williams H. A., Jones M. H., Nejati M., Seabright M. J., Bell J., Penhall N. D., MacDonald B. A., Robotic kiwifruit harvesting using machine vision, convolutional neural networks and robotic arms, *Biosystems Engineering*, 181, 140-156, 2019.
- Onishi Y., Yoshida T., Kurita H., Fukao T., Arihara H., Iwai A., An automated fruit harvesting robot by using deep learning, *Robomech Journal*, 6 (1), 1-8, 2019.
- Arad B., Balendonck J., Barth R., Ben Shahar O., Edan Y., Hellström T., van Tuijl B., Development of a sweet pepper harvesting robot, *Journal of Field Robotics*, 37 (6), 1027-1039, 2020.
- Zhang K., Lammers K., Chu P., Li Z., Lu R., System design and control of an apple harvesting robot, *Mechatronics*, 79, 102644, 2021.
- Yin W., Wen H., Ning Z., Ye J., Dong Z., Fruit Detection and Pose Estimation for Grape Cluster Harvesting Robot Using Binocular Imagery Based on Deep Neural Networks, *Frontiers in Robotics and AI*, 8, 2021.
- Jun J., Kim J., Seol J., Kim J., Son H. I., Towards an Efficient Tomato Harvesting Robot: 3D Perception, Manipulation, and End-Effector, *IEEE Access*, 9, 17631-17640, 2021.

15. Yoshida T., Kawahara T., Fukao T., Fruit Recognition Method for a Harvesting Robot with RGB-D Cameras, 2022.
16. Wan H., Fan Z., Yu X., Kang M., Wang P., A real-time branch detection and reconstruction mechanism for harvesting robot via convolutional neural network and image segmentation, *Computers and Electronics in Agriculture*, 192, 106609, 2022.
17. Dersuneli M., Gündüz T., Kutlu Y., Bul-Tak Oyuncağı Şekillerinin Klasik Görüntü İşleme ve Derin Öğrenme Yöntemleri ile Tespiti, *Bitlis Eren Üniversitesi Fen Bilimleri Dergisi*, 10 (4), 1290-1303, 2021.
18. Cekic I., Cavdar K., Detection of the cracks in metal sheets using convolutional neural network (CNN), *Journal of the Faculty of Engineering and Architecture of Gazi University*, 38 (1), 153-162, 2023.
19. Dumitrache A. Robot kinematics diagram. <http://alexdu.github.io/sketch-lib>. Erişim tarihi Temmuz 12, 2023.
20. Camgözlü Y., Kutlu Y., Analysis of pooling effect on CNN using leaf database, *Natural and Engineering Sciences*, 4 (3), 115-121, 2019.
21. Bochkovskiy A., Wang C. Y., Liao H. Y. M., Yolov4: Optimal speed and accuracy of object detection, *arXiv preprint arXiv:2004.10934*, 2020.
22. Zhu L., Geng X., Li Z., Liu C., Improving yolov5 with attention mechanism for detecting boulders from planetary images. *Remote Sensing*, 13 (18), 3776, 2021.
23. Li C., Li L., Jiang H., Weng K., Geng Y., Li L., Ke Z., Li Q., Cheng M., Nie M., Li Y., Zhang B., Liang Y., Zhou L., Xu X., Chu X., Wei X., Wei X., YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications, *arXiv preprint arXiv:2209.02976*, 2022.
24. Munawar M. R., YOLOv7 Architecture Explanation. <https://www.cameralyze.co/blog/yolov7-architecture-explanation>. Erişim tarihi Ocak 11, 2023.
25. Yayik A., Kutlu Y., Diagnosis of congestive heart failure using poincare map plot, 20th Signal Processing and Communications Applications Conference (SIU), Muğla, 1-4, 2012.
26. Ertugrul S., Kaya O., Turkmen D., Eraslan H., Taglioglu G.B., Gulec M.O., Humanoid robot arm design, simulation, kinesthetic learning, impedance control and suggestions, *Journal of the Faculty of Engineering And Architecture of Gazi University*, 37 (2), 1139-1154, 2022.
27. Kutlu Y., Alanoglu Z., Gökçen A., Yeniad M., Raspberry pi based intelligent robot that recognizes and places puzzle objects, *arXiv preprint arXiv:2101.12584*, 2021.
28. Shao C., A quantum model for multilayer perceptron, *arXiv preprint arXiv:1808.10561*, 2018.
29. Altınkaynak A., Ağsız Yöntem Uygulamaları için Trigonometri Tabanlı Radyal Özelliğe Sahip Yeni Bir Temel Fonksiyon, *International Journal of Advances in Engineering and Pure Sciences* 32.1, 96-110, 2020.
30. Huang G.B., Qin-Yu Z., Chee-Kheong S., Extreme learning machine: theory and applications, *Neurocomputing* 70.1-3, 489-501, 2006.
31. Haykin, Simon (1998). *Neural Networks: A Comprehensive Foundation* (2 ed.). Prentice Hall. ISBN 0-13-273350-1.
32. Camero A., Toutouh J., Alba E., A specialized evolutionary strategy using mean absolute error random sampling to design recurrent neural networks, *arXiv:1909.02425*, 1-10, 2019.
33. Bradski G., Kaehler A., *Learning OpenCV: Computer vision with the OpenCV library*, O'Reilly Media, Sebastopol, 1-580, 2008.
34. Marcmateo, BCN3D Technologies. <https://github.com/BCN3D/BCN3D-Moveo>, 2018, Erişim tarihi: 31.04.2021.
35. Šegota S. B., Andelić N., Mrzljak V., Lorencin I., Kuric I., Car Z., Utilization of multilayer perceptron for determining the inverse kinematics of an industrial robotic manipulator, *International Journal of Advanced Robotic Systems*, 18 (4), 2021.
36. Zhao Y., Gong L., Huang Y., Liu C., A review of key techniques of vision-based control for harvesting robot, *Computers and Electronics in Agriculture*, 127, 311-323, 2016.
37. Zhang K., Lammers K., Chu P., Li Z., Lu R. System Design and Control of An Apple Harvesting Robot. *Mechatronics*, 79, 102644, 2021.