



**T.C. İSTANBUL T CARET
 NİVERSİTESİ**

FEN B LİMLERİ ENSTİT S 

KAMERA KONTROLL  AKILLI MAMA KABİ TASARIMI

Hatice Selenay  AKICI

Danışman

Dr.  ğr.  yesi Metin TURAN

**Y KSEK LİSANS TEZİ
B LGİSAYAR M HENDİSLİĐİ ANAB LİM DALI
İSTANBUL - 2022**

KABUL VE ONAY SAYFASI

Hatice Selenay AKICI tarafından hazırlanan "**Kamera Kontrollü Akıllı Mama Kabı Tasarımı**" adlı tez çalışması 28/06/2022 tarihinde aşağıdaki jüri üyeleri önünde başarı ile savunularak, İstanbul Ticaret Üniversitesi Fen Bilimleri Enstitüsü **Bilgisayar Mühendisliği Anabilim Dalı**'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Danışman

Dr. Öğr. Üyesi Metin TURAN
İstanbul Ticaret Üniversitesi

Jüri Üyesi

Dr. Öğr. Üyesi Ali AKMAN
İstanbul Ticaret Üniversitesi

Jüri Üyesi

Dr.Öğr.Üyesi Zeynep TURGUT AKGÜN
İstanbul Medeniyet Üniversitesi

Onay Tarihi : 08.07.2022

İstanbul Ticaret Üniversitesi, Fen Bilimleri Enstitüsünün 08.07.2022 tarih ve 2022/351 numaralı Yönetim Kurulu Kararının 2. maddesi gereğince, ders yüklerini ve tez yükümlülüğünü yerine getirdiği belirlenen “Hatice Selenay AKICI” adlı öğrencinin mezun olmasına oy birliği ile karar verilmiştir.

Prof. Dr. Necip ŞİMŞEK
Enstitü Müdürü

AKADEMİK VE ETİK KURALLARA UYGUNLUK BEYANI

İstanbul Ticaret Üniversitesi, Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmada,

- tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- ve bu tezin herhangi bir bölümünü bu üniversitede veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

Hatice Selenay ÇAKICI

İÇİNDEKİLER

	Sayfa
İÇİNDEKİLER	i
ÖZET.....	ii
ABSTRACT	iii
TEŞEKKÜR.....	iv
ŞEKİLLER DİZİNİ.....	vi
ÇİZELGELER DİZİNİ	vii
1. GİRİŞ	1
1.1. Akıllı Uygulama Nedir?	3
1.2. Akıllı Uygulama Avantajları	4
1.2.1. Akıllı uygulamalar ile sosyal sorumluluk.....	4
1.2.2. Akıllı uygulama ile güvenilir çözümler.....	5
1.2.3. Akıllı uygulama ile düşük maliyet.....	6
1.2.4. Akıllı uygulamala ile yaşamı kolaylaştırma	6
2. LİTERATÜR ÖZETİ.....	7
2.1. Nesne Algılama ve İzleme Sistemleri	7
2.2. Nesne Takibi İçin Kullanılabilecek Araçlar	11
2.2.1. Arduino kart.....	11
2.2.1.1.Arduino programlama ortamı	12
2.2.2.Sensörler.	12
2.2.2.1.RFID sensörü	13
2.2.2.2.Nem ve sıcaklık sensörü	13
2.2.2.4 Manyetik (Reed) Anahtar sensörü	14
3. UYGULAMA	16
3.1. Çalışmada Tercih Edilen Donanım Parçaları	16
3.1.1.Arduino UNO R3 USB Chip CH340(ATmea328 mikro denetleyici)	16
3.1.2.MG996R servo motor	17
3.1.1. TCS230/TCS3200 renk seönsörü	18
3.2. Kamera Kontrollü Akıllı Mama Kabı Çalışma Prensibi ve Tasarımı	20
3.3. Programlar ve Kütüphaneler	25
3.3.1. Python system yazılımı.....	26
3.3.2. TensorFlow	26
3.3.3. OpenCV /Açık kaynak Bilgisayarla Görme).....	27
3.3.4. Görüntü işleme algoritması	27
3.3.5. Aksiyonların e-posta ile raporlanması	29
3.3.6. Arduino Yazılımı	32
4. ARAŞTIRMA BULGULARI VE TARTIŞMA	32
5. SONUÇ VE ÖNERİLER	34
KAYNAKLAR	36
EKLER.....	40
EK A. Smart_food_container.py	41
EK B. Sensor_with_arduino.ino.....	44
EK C. Ürünün Gerçek Boyutlu Teknik Çizimleri	45
ÖZGEÇMİŞ	47

ÖZET

Yüksek Lisans Tezi

KAMERA KONTROLLÜ AKILLI MAMA KABI TASARIMI

Hatice Selenay ÇAKICI

İstanbul Ticaret Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Dr. Öğr. Üyesi Metin TURAN
2022, 47 sayfa

Günümüzde sokak hayvanları, çeşitli sebeplerden beslendiği kaynaklar üzerinden zehre maruz kalabilmekte veya mevsimsel hava koşullarının olağandışı değişiklikleri nedeniyle besinleri muhafaza edilememektedir. Ayrıca evcil hayvan sahibi bireyler tatile/işe gittiklerinde mamaları bayatlayabilir ve tükenebilir. Her iki durumda da insan müdahalesi veya yardımı olmadan çalışmasını sağlayan kapalı bir sistem sayesinde evcil hayvanların ihtiyaç duyduğu kadar yiyecek sağlanması gerekmektedir.

Bu tez ile, gerekli donanım sağlanarak hayvanı tespit eden ve bu tespit doğrultusunda otomatik olarak yem veren kapalı bir sistem oluşturulması amaçlanmıştır. Bu öğrenme yoluyla kendi işleyişini sağlayan bir sistem sayesinde, hayvanlara doğal yollardan beslenemediği bir ortamda, ihtiyaçları kadar mama sağlayabilmek amaçlanmıştır. Bu sayede, olumlu veya olumsuz yönde insan müdahalesini en aza indirgeyen otomatik bir sistem sağlanmıştır.

Bu model tek bir mama kabı içerir ve prototip sistem olarak kediler üzerinde test edilmiştir. Prototip sistem, bilgisayarlı görme teknolojisi kullanılarak özel bir donanım ile tasarlanmıştır. Kedi, TensorFlow kütüphanesi ile algılandıktan sonra mama kabının doluluk oranının uygun seviyede olup olmadığı renk sensörü ile kontrol edilir. Mama kabı bitmek üzereyken, kullanıcıya Basit Posta Aktarım Protokolü (SMTP) kullanılarak e-posta ile bilgi verilir.

Geliştirilen modelin bir ihtiyaç olarak değerlendirilmesi, insan müdahalesinin en aza indirilerek olumsuz durumların önlenmesi veya mamanın çevre koşullarından etkilenmeden muhafaza edilebilmesi açısından büyük önem taşımaktadır. Aynı şekilde ev ve bahçe gibi ortamlarda insanın en büyük dostu olan ve doğal yollardan beslenme şansı olmayan hayvanların, insana ihtiyaç duymadan beslenebilmesini sağlayabilmek projenin ikinci büyük amacıdır. Ortaya çıkan ürün, açık ve kapalı ortamlarda bir veya birden fazla hayvan dostumuz için kullanılabilir.

Anahtar Kelimeler: Akıllı mama kabı, Arduino, bilgisayarla görme, TensorFlow.

ABSTRACT

M.Sc. Thesis

CAMERA CONTROLLED SMART FOOD CONTAINER DESIGN

Hatice Selenay ÇAKICI

**Istanbul Commerce University
Graduate School of Natural and Applied Sciences
Department of Computer Engineering**

**Supervisor: Assit. Prof. Dr. Metin TURAN
2022, 47 pages**

Today, stray animals can be exposed to poison through the sources they feed for various reasons or their food cannot be preserved due to unusual changes in seasonal weather conditions. In addition, when pet owners go on vacation/to work, cat food may become stale and runout. In both cases, pets which can be achieved with as much food as they need, thanks to a closed system that allows it to operate without human intervention or assistance.

In this thesis, it is aimed to create a closed system that detects the animal by providing the necessary equipment and automatically feeds in line with this detection. Thanks to a system that provides its own functioning through this learning, it is aimed to provide the animals with as much food as they need in an environment where they cannot be fed naturally. In this way, an automatic system that minimizes human intervention in a positive or negative way has been provided.

This model includes a single bowl and was tested on cats as a prototype system. The prototype system was designed with special hardware using computer vision technology. After the cat is detected with the TensorFlow library, it is checked with the color sensor whether the food container is full or not. When the lunch box is about to runout, the user is notified by e-mail using Simple Mail Transfer Protocol (SMTP).

Evaluating the developed model as a need is of great importance in terms of minimizing human intervention, preventing negative situations or preserving the food without being affected by environmental conditions. Likewise, the second goal of the project is to ensure that animals, which are man's best friend in environments such as home and garden, and who do not have the chance to feed naturally, can be fed without the need for humans. The resulting product can be used for one or more of our animal friends in indoor and outdoor environments.

Keywords: Arduino, computer vision, smart food container, TensorFlow.

TEŞEKKÜR

Bu araştırma için beni yönlendiren, karşılaştığım tüm zorlukları samimiyeti, bilgi ve tecrübesi ile aşmamda yardımcı olan, her desteğine ihtiyacım olduğunda kolaylıkla ulaşabildiğim, beni cesaretlendiren, kıymetli zamanını ayırıp tez sürecimde yaşadığım tüm eksikliklere çözüm bulan, desteğini hem akademik hem sosyal alanda benden esirgemeyen çok kıymetli Danışman Hocam Dr. Öğr. Üyesi Metin TURAN'a sonsuz teşekkür ederim.

Hayat boyu öğrenme ve geliştirmenin devam etmesi gerektiğini, insanı ve toplumu ileri götürecek şeylerin bunlar olduğunu bana öğreten canım aileme, çalışma arkadaşlarıma, desteğini esirgemeyen kıymetli dostum Şahincan KANBER'e sevgi ve saygılarımı sunarım.

Hatice Selenay ÇAKICI
İSTANBUL, 2022



ŞEKİLLER DİZİNİ

	Sayfa
Şekil 1.1 Akıllı uygulama avantajları.....	4
Şekil 2.1. Tipik bir Nesne Takip Sistemi süreci.....	8
Şekil 2.2. Yapay sinir ağının genel düzeni.....	9
Şekil 2.3. Çalışmada kullanılan Arduino yazılım IDE'sinden bir kesit.	12
Şekil 2.4. RFID-RC522 sensor	13
Şekil 2.5. Nem ve sıcaklık sensörü (DHT22)	14
Şekil 2.6. Hareket sönsörü (PIR HC-RS501).....	14
Şekil 2.7. Arduino için Reed Switch sensörü.....	16
Şekil 3.1. ATmega328 Mikro Denetleyici Görüntüsü	17
Şekil 3.2. MG996R Servo Motor	18
Şekil 3.3. Renk sensörü (TCS3200).....	19
Şekil 3.4. Sistemin genel akış şeması.....	21
Şekil 3.5. Mama kabının detaylı görünümü. (a) Mama kabının hazneleri ve mama indirmenin yapıldığı bölge (b) Mama kabı kapalıyken Servo Motor, Renk Sensörü ve çarkın konumu. (c) Mama kabının dış görünüşü.....	23
Şekil 3.6. Çarkın ve dış kapağın görüntüsü (a) 8 odacıklı çarkın görüntüsü (b) Dış kapağın alttan görüntüsü (c) Dış kapağın üstten görüntüsü.....	24
Şekil 3.7. Arduino UNO'nun MG 996R servo motor, bilgisayar ve TCS230/TCS3200 renk sensörü ile bağlantısı.....	25
Şekil 3.8. Algoritmanın gerçek hayattaki ekran çıktısı	29
Şekil 4.1. Boş ve dolu değerlerin grafiği	33
Şekil 5.1 Önerilen sistem modelinin bir prototipi.....	34
Şekil C.1. 45 derece bölmeli çark	45
Şekil C.2. Çark haznesi	45
Şekil C.3. Mama haznesi.....	46
Şekil C.4. Mama indirme haznesi	46

ÇİZELGELER DİZİNİ

	Sayfa
Çizelge 3.1. Arduino UNO ATmega 328 teknik özellikleri	17
Çizelge 3.2. TCS230 / TCS3200 teknik özellikleri	19
Çizelge 4.1. Mama kabı boş ve dolu olduğunda RGB Değerleri.....	32



1. GİRİŞ

Günümüz rekabet koşullarında hızlı, ucuz ve güvenilir çözümler üretmek mühendisler için bir hedeftir. Yapay zekanın gelişmesi ve bu teknolojinin hemen her alanda kullanılmaya başlanması, bilgisayarla görme alanında yapay zeka kullanılarak insan faktörünün en aza indirilmesinin yanı sıra zamandan ve emekten tasarruf edilmesi ihtiyacını doğurmuştur. Türkiye'nin tüm büyük illerinde hayvanlar yerel belediyelerden hizmet almaktadır. Barınma, beslenme gibi ihtiyaçları kısmen yerel çalışanlar tarafından karşılanmaktadır. Dr. Satioğlu'nun açıklamasına göre, Türkiye'nin her yerinde belediyeler 90'ların sonu ve 2000'lerin başında yüzlerce köpeği zehirledi (NYT, 2022). Yakınlardaki diğer vakalarda ise Denizli'de 28 sokak kedisinin zehirlenerek öldürüldüğü (D20HABER, 2022) ve Adana'nın Kozan ilçesinde bir sokakta 3 zehirli kedinin bulunduğu görülmüştür (HBR, 2022).

Bugüne kadar hayvanlar için pek çok mama kabı tasarımı yapılmıştır. Kevin Shibley, çalışmasında sensörler ve zamanlayıcılar kullanarak otomatik bir evcil hayvan maması dağıtıcısı tasarladı. Besleme kapısı, 180° dönebilen, 5V'ta çalışan ve sinyalini bir Atmega328P mikrodenetleyiciden (Shibley, 2022) alan bir servo motor tarafından tetiklenmektedir.

Diğer yandan İbrahim vd., (2019), evcil hayvan mamasını belirli bir zaman diliminde teslim edebilen otomatik bir evcil hayvan besleyici için bir mekanizma tasarladı. Arduino kullanarak otomatik evcil hayvan yemliği yaptılar. Besleme süresi ayarlanarak motor istenilen zamana göre döner ve mekanizma çalıştırılır (İbrahim, Zakaria ve Wei Xian, 2019).

Vineeth vd., (2020), Raspberry pi, DC motor, kamera ve ultrasonik sensör kullanarak bir tasarım yaptılar. Sistemde hoparlör tarafından kaydedilen bir sesli arama ile evcil hayvan besleme periyodu başlar. Daha sonra sistemin önüne yerleştirilen ultrasonik sensör ile hayvan tespiti yapılır. Evcil hayvan sistemin önüne geldiğinde sensör hayvanı algılar ve kamerayı çalıştırır. Hayvan tespiti daha sonra CNN'deki TensorFlow kütüphanesi kullanılarak yapılır. Evcil hayvanın doğru evcil hayvan

olduğu tespit edildiğinde, DC motor mamayı dağıtır. Daha sonra evcil hayvan besleme işlemi gerçekleştirilir (Vineeth vd, 2020).

Yapay zekâ kullanımının yanı sıra insan müdahalesi ile de çalışan mama kapları da tasarlanmıştır. Khatavkar vd., (2019), cep telefonu ile geliştirdikleri sisteme mesaj göndererek evcil hayvanlarını beslediler. Solenoid valf ve servo motor, GSM teknolojisi kullanılarak alınan bir mesajla etkinleştirilir. Ardından servo motor döner ve besleme işlemi başlar. Besleme işlemi tamamlandığında evcil hayvan sahibine bir mesaj gönderilir (Khatavkar vd., 2019).

Mahanty vd., görüntü tabanlı bir hayvan algılama sistemi önermek için tam yüz PCA algoritmasını kullandı. Görüntüyü yeniden boyutlandırdılar ve görüntünün öz yüzünü ve öznitelik değerlerini hesapladılar. Daha sonra ortadaki görüntünün yüz uzayına izdüşümü hesaplanmıştır. Son olarak test görüntüsü ve izdüşümü bir önceki görüntüye göre hesaplanmıştır. İki projeksiyonu karşılaştırmak için Öklid mesafesi kullanılmıştır. Veri seti görüntüsünün Öklid mesafesi en küçük ise, bu bizim tanıma resmimizdir (Mahanty vd., 2019).

Bu çalışmada hayvan hareketlerini algılayan ve otomatik olarak besleyen kapalı bir sistem oluşturulması amaçlanmıştır. Ana fikir, hayvanlara doğal olarak beslenemeyecekleri bir ortamda ihtiyaç duydukları kadar yiyecek imkanı sağlamaktır. Yürütülen çalışmada OpenCV kütüphanesi kullanılarak kameradan görüntüler alınır. Daha sonra alınan görüntüde TensorFlow kütüphanesi ve MobileNet eğitilmiş modeli kullanılarak kedi tespiti yapılmıştır. Kedi algılandığında, sensör mama haznesinin dolu olup olmadığını kontrol eder. Mama kabında belirtilen seviyenin üzerinde mama varsa Arduino mikro denetleyici servo motorun çalışmasını tetikler. Servo motor çalışır ve mama kabının besleme haznesi doldurulur. Sensör, kabin mama seviyesinin düşük olduğunu algıladığında, Basit Posta Aktarım Protokolü (SMTP) ile e-posta gönderilerek hayvan sahibine bilgi verilir.

Proje kedilerin kullanımı için hazırlanmıştır. Tasarlanan ürün, kedileri %90'ın üzerinde doğrulukla besleme yeteneğine sahiptir.

1.1. Akıllı Uygulama Nedir?

Ev Otomasyonu (Akıllı Ev) kavramı, evlerimizde kullanılan ev lambaları, nem ve sıcaklık, fanlar, kapı kilitleri vb. elektronik cihazların ve elektrikli eşyaların otomatikleştirilmesi ve kontrol edilmesi anlamına gelir. Ev Otomasyonu (Akıllı Ev) konusu aslında 1960'lardan beri Bilgisayar Tabanlı Sisteme dayalı olarak inşa edilmiş ve geliştirilmiştir.

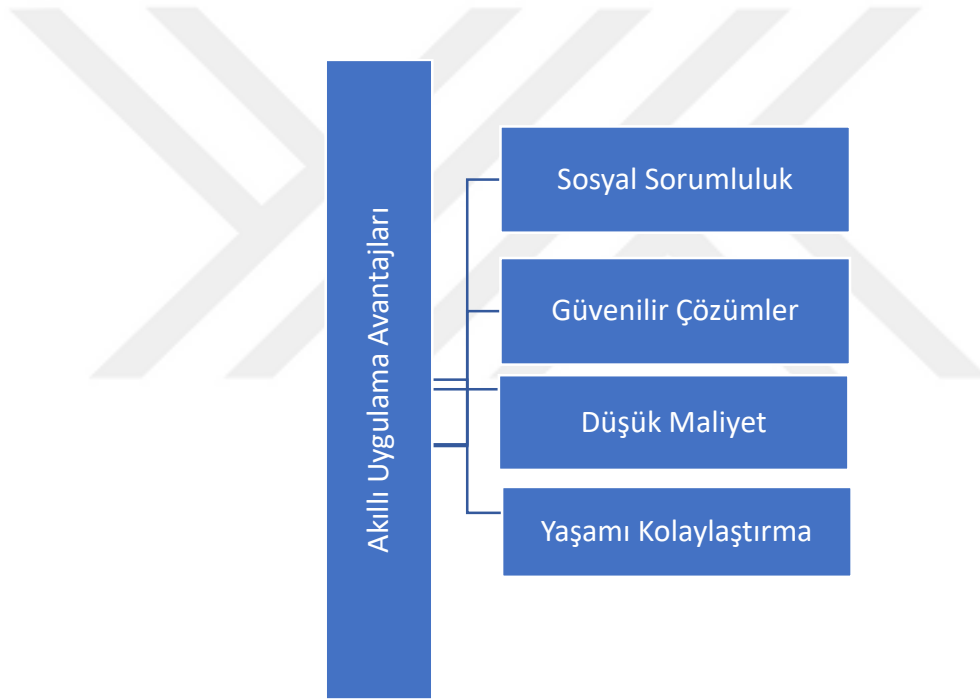
Ev Otomasyonu (Akıllı Ev) ve Nesnelerin İnterneti (IoT) konuları birbiriyle yakından ilişkili ve bağlantılıdır. IoT'nin ögeleri, örneğin; akıllı sıcaklık sensörü, akıllı cihazlar (akıllı telefon, tablet vb.) tarafından kontrol edilebilir ve izlenebilir. Sonuç olarak, internet üzerinden bu kolaylıkların sağlanabilmesi sayesinde, bunların kapsamı dünya çapında genişletilebilir. Akıllı telefonların popülaritesinin artmasıyla birlikte IoT araçları, ev otomasyonu kullanım alanları günümüzde hızla artmaktadır. Ev Otomasyonu cihazlarıyla birlikte IoT ögeleri sağlanır ve bir modu etkinleştirerek (Otomatik Enerji Tasarrufu) veya kullanıcılara gerçek zamanlı olarak enerji tüketimi hakkında kısa ayrıntılar vererek enerji tüketimi için optimizasyon sağlanır. Ayrıca, klasik elektronik cihazların bir IoT akıllı ürünle değiştirilmesi durumunda maliyet yüksek olabilir. Ancak akıllı eve geçişte en büyük sorun, IoT olmayan ögelerin akıllı ev sistemi ile uyumlu hale getirilmesi ve uygun şekilde çalışmasının sağlanmasıdır (Brush, vd., 2017).

Ev otomasyonu (akıllı ev), IoT'nin önemli parçalarından biri olan akıllı ev sistemi, IoT'nin farklı elektronik ögeleriyle iletişim kurarak kullanıcılara hizmet vermiştir. Nesnelerin İnterneti'ne (IoT) dayalı akıllı ev teknolojisi ile yer ve zaman gözetmeksizin herkese bağlanabilirlik sağlanarak insanların hayatı değişmiştir (Samuel, 2016). Son zamanlarda, akıllı ev sistemleri giderek daha gelişmiş hale gelmiştir. Akıllı ev sistemleri, her türlü ögenin bilgi ve hizmetlerini değiştirmek için sağlanan yöntemlere ve protokollere sahiptir. Ev otomasyonu, Nesnelerin İnterneti'nin (IoT) bir alanıdır. Aynı zamanda ev içinde sensörler, elektronik nesneler, yazılımlar ve ağ bağlantısı sağlayan bir donanım ögeleri grubudur (Kim vd., 2015).

Modern akıllı ev sistemleri, genellikle ağ geçitleri olarak adlandırılan ve merkezi birime bağlanabilen akıllı sensörler ve anahtarlardan oluşur. Ağ geçitleri ile birlikte kullanıcı ara yüzü, akıllı ev sistemlerini kontrol etmek için akıllı telefonlar, tabletler veya bilgisayarlar ile etkileşime girer; bu sistemlerin bağlantıları, Nesnelerin İnterneti (IoT) tarafından kontrol edilir ve yönetilir (Galinina vd., 2015).

1.2. Akıllı Uygulama Avantajları

Bu bölüm, ev otomasyon sistemleri ile ilgili literatürde bildirilen ve benzer avantajlara göre gruplara ayrılmış, hayatımız için önceliğe dayalı birçok önemli kullanım alanını içermektedir (Şekil 1.1.).



Şekil 1.1. Akıllı uygulama avantajları (Stolojescu-Crisan vd., 2021)

1.2.1. Akıllı uygulamalar ile sosyal sorumluluk

Birçok özel kurum veya şirket öncülük edip evcil hayvanlara sağlıklı bir hayat sunabilmek için yenilikçi ürünler geliştirmektedir. Aynı zamanda sosyal medya üzerinden kampanyalar yürütülmektedir. Yine özel firmalar sokak hayvanları için farkındalık yaratacak programlar başlatmışlardır (Campaign, 2022 ; Haytap, 2022).

Diğer yandan hayvan sevgisinin çocukların gelişimi üzerinde olumlu etkileri olduğu ileri sürülmektedir (Arıkan vd., 2019). Bu yönüyle çocuk gelişiminde temel bir rol oynayan hayvan sevgisi, duygusal erişkinlikte olduğu kadar onların topluma uyum sağlayan sosyal bir varlık olarak yetişmesinde de kritik yer tutmaktadır. Hayvanı beslemek, onu sevmek, ona zarar vermemek, onun iyiliğini düşünmek, çocuğun sadece hayvanlara olan olumlu duygularını beslemekle kalmamakta, başkalarının varlığının da kendisinininki kadar değerli olduğunu hatırlatmaktadır.

Bu yönüyle tasarlanıp prototip olarak geliştirilen proje, kedilerin beslenmesinde akıllı uygulama örneği olarak yerini alacaktır. Bu proje ile sokak hayvanlarına dair en temel sorunlar olarak ortaya çıkan Beslenme – Barınma – Sahiplendirme konularından beslenme konusuna akıllı sistemle çözüm bulunması projenin güçlü yanlarındanır.

1.2.2. Akıllı uygulama ile güvenilir çözümler

Günümüzde birçok araştırma ve teknoloji şirketi akıllı evler üzerine çalışmaktadır. Akıllı ev kavramını uygulamak ve geliştirmek için; MIT, Siemens, Cisco, IBM, Xerox, Microsoft gibi araştırma grupları tarafından birçok örnek evler oluşturulmuştur (Gügöl ve Sarıtaş, 2008). Bunların yanı sıra akıllı sistemlerle hayvanların ihtiyaçları konusunda da çözümler üretilmektedir.

Gelişen teknoloji ile hayatımıza giren teknoloji harikaları hayatımızı kolaylaştırmaya hızla devam etmektedir. Akıllı teknolojiler, daha rahat ve konforlu bir yaşam sunarken aynı zamanda güvenilir olması onları daha cazip hale getirmektedir. Akıllı uygulama, dünya genelinde pek çok alanda kullanılan kontrol sistemlerini evlere, yaşam alanlarına getiren bir sistemdir (Örselli ve Akbay, 2019).

Güvenilir kontrol sistemi, kullanıcıların ihtiyaçları doğrultusu göz önüne alınarak ihtiyaç ve isteklere göre hazırlanmaktadır. Diğer bir deyişle akıllı otomasyon adını alan sistemler gerek insanların gerekse hayvanların yaşamlarına teknolojik katkılar sunarak hayatı kolaylaştırmaktadır (Aşar, 2018). Akıllı mama kabı projesi de bu yönüyle kediler için tasarlanmıştır. Kullanılan ekipman ve sistemin kedilere hiçbir

zararı olamayacak şekilde tasarlanmıştır. Bu yönüyle güvenlik sağlar ve konforlu bir beslenme sistemi sunmaktadır.

1.2.3. Akıllı uygulama ile düşük maliyet

Akıllı uygulamalar, kullanıcıların tasarruf ederek temel ihtiyaçlarını karşılamak amacıyla birçok farklı alanda kullanılmaktadır. Günümüzde uzaktan sağlık takibi yapılarak sağlık hizmeti sunumunun maliyeti azaltılmaktadır. Aynı zamanda yaşlıların ve engellilerin sağlık hizmetlerine erişimi kolaylaşmıştır. Akıllı evdeki IoT cihazlarının sağlık ağlarına bağlanmasıyla, yaşlı veya engellilerle doktora gitme sıklığı daha düşük olacak, ayrıca tıbbi maliyet azalacaktır (Shamszaman vd., 2014). Kliniklerde ve hastanelerde kullanılan akıllı sistem teknolojisi, maliyeti düşürerek, tıbbi hizmetlere erişimi arttırabilir ve kolaylaştırabilir.

Akıllı evde maliyeti düşürmek, akıcılığı ve esnekliği artırmak için kablolu ağ yerine kablosuz ağ teknolojisi kullanılmaktadır. Buna ek olarak, enerji tasarrufu maliyeti düşürmektedir. Ayrıca, güvenlik sistemi tarafından enerji tasarrufu sağlayan bir cihazın arızalanması önlenmiştir. Güvenlik sistemi ile kullanılan enerjide bir karışıklık ortaya çıkmadan önemli bir enerji tüketiminin önüne geçilir (Moser vd., 2014).

1.2.4. Akıllı uygulama ile yaşamı kolaylaştırma

Akıllı ev, kullanıcılara konforlu bir yaşam sunar ve kullanıcıların güvenli ve emniyetli olmasını sağlar. Akıllı ev sistemindeki tüm cihazlar her an çalıştırılabilir. Bu cihazların tümü sensörlerle sağlanır ve her sensörün IoT teknolojisine dayalı kablosuz bağlantıyla farklı işlevi vardır (Bhide ve Wagh, 2015). Örneğin akıllı ev içerisinde kullanılan cihazlar, kullanıcı evden çıktığında otomatik olarak kapanabilmektedir. Ayrıca kullanıcılar akıllı ev ile faturalarını rahatlıkla ödeyebilmektedir. Mobil cihazlar ile akıllı ev, kullanıcılar için uygundur ve bu mobil cihazlar, fiziksel anahtar yerine cihazları uzaktan açıp kapatmak için kullanılabilir (Pandey vd., 2015).

2. LİTERATÜR ÖZETİ

Bugüne kadar hayvanlar için birçok mama kabı tasarımı yapılmıştır. Kevin Shibley (2014) yaptığı çalışmada, sensörler ve zamanlayıcılar kullanarak otomatik bir evcil hayvan maması dağıtıcısı tasarlamıştır. Besleme kapısı, 180° dönebilen, 5V'da çalışan ve sinyali bir Atmega328P mikro denetleyiciden alan bir servo motor tarafından kontrol edilmektedir.

Görüntüler, görüntü işleme uygulamalarında önemli bir kaynaktır. Görüntü işleme, gelecekte insan-bilgisayar etkileşimini değiştirecektir. Birçok görüntü işleme uygulaması, görüntülerden karmaşık özelliklerin çıkarılmasına yardımcı olabilir. Görüntü işleme, tek boyutlu görüntüler veya çok boyutlu görüntüler için uygulanabilir ve birçok gerçek zamanlı iyileştirme tekniğinin temelidir (Howard vd., 2018).

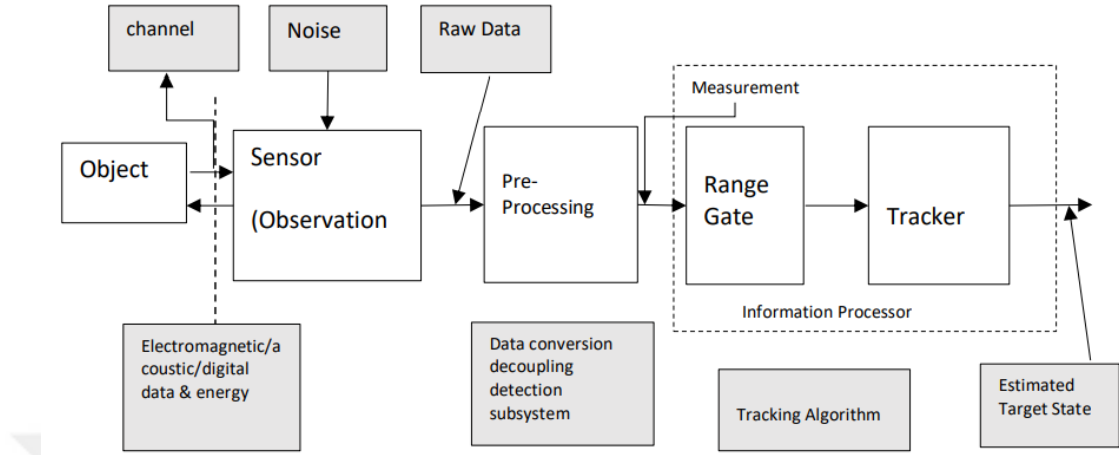
2.1 Nesne Algılama ve İzleme Sistemleri

Bu bölümde, nesne algılama ve izleme konusundaki kavramlardan bahsedilecektir.

Hedef veya nesne izleme, “Nesne İzlemenin Temelleri” kitabı tarafından basit terimlerle tanımlanmaktadır. Bu, modelin belirli bir süre için belirli bir ortamdaki her benzersiz hedef, konumlar ve hareket kalıpları hakkında öğrenme sağlayarak geliştirmesini sağlayan teknolojidir (Subhash vd., 2011). Aynı zamanda, bir çevredeki nesneler hakkında kimlikleriyle birlikte bilgi almaya, işlemeye dayanır. Bu kimlikler, nesne sayısı, konum, hız, koordinatlar vb. gibi metriklerden oluşur (Lo vd., 2010).

Bu metrikler, izleme olarak adlandırılan her belirli hedefin kimliğine göre hesaplanmaktadır (Subhash vd., 2011). İzleme, radarlar, kameralar, kızılötesi veya ultraviyole sensörler gibi çoklu sensörler aracılığıyla gerçekleştirilebilir (Richter vd., 2008). Ancak çeşitli aşamalara paralel olarak ortaya çıkan öngörülemez belirsizlikler ve gürültüler nedeniyle izleme işlemi kolay bir işlem değildir. Bu tür olaylar arasında yanlış okumalar (görünürde hiçbir nesne olmadığında ancak model yine de algılanan bir nesneyi gösterdiğinde), algılanabilir nesnelerin sayısının bir örnekten diğerine ani ve rastgele değişmesi, ölçüm yanlışlıkları (ölçülen mesafe veya iki nesne arasındaki hız, fiziksel olandan farklıdır), birden çok hedef arasındaki girdi

girişimi gibi sebeplerden nesneler algılanamamaktadır. Şekil 2.1.1., bir nesne takip sisteminin genel mimarisini göstermektedir.

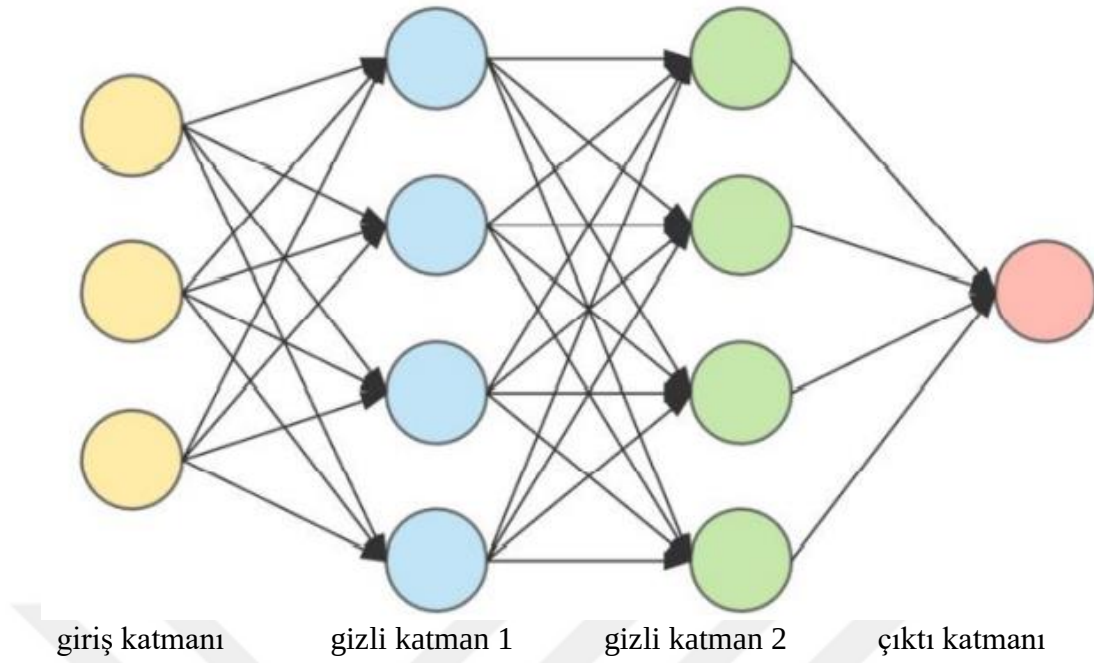


Şekil 2. 1. Tipik bir Nesne Takip Sistemi süreci (Subhash, vd., 2011)

Nesne algılama olmadan izlemenin gerçekleştirilemeyeceğine dikkat edilmelidir. Bu nedenle, sonraki bölümlerin daha alakalı olması için derinlik tabanlı nesne algılama ilkeleri ele alınacaktır.

Radar tabanlı (Richter vd., 2008) ve birbirine bağlı araç teknolojileri (Ren, vd., 2012) çok nesneli izleme gerçekleştirmek için kullanılan çeşitli teknolojiler arasında bulunmaktadır. Nesne izleme gerçekleştirebilmek için en verimli ve uygulanabilir yöntem, derin öğrenmeye dayalı nesne algılama modelleri kullanmaktır. Bu nedenle, bu bölümde, bilgisayarla görme ve nesne algılama için derin öğrenme teknikleri orta düzeyde incelenecektir (Hu, vd., 2018).

Yapay Sinir Ağları teknolojisi nesne algılama tekniği için kullanılabilir. Yapay Sinir Ağları (YSA) temelli derin öğrenmenin yapıtaşı büyük ölçüde sinirbilim disiplini kuran biyolojik muadilinden türetilmiştir (Krohn, vd., 2019). Teknik bir bakış açısıyla, bir nöronun eşdeğeri bir algılayıcı olarak adlandırılır (Widrow ve Lehr, 1990). Toplu olarak, standart bir YSA üç tür katman halinde incelenir (Şekil 2.2.) (Torres, 2018).



Şekil 2. 2.Yapay sinir ağının genel düzeni (Torres, 2018)

Derinliğe dayalı nesne algılama işleminden sonra nesne takibini gerçekleştirmek ikinci adım olacaktır. Görsel derinlik tabanlı çoklu nesne izleme mimarisi üzerine, Toronto Üniversitesi tarafından Honda Araştırma Enstitüsü ve Motorola Laboratuvarları ile birlikte bir Artımlı Görsel İzleme (IVT) algoritması geliştirilmiştir (Ross vd., 2008).

Artımlı Görsel İzleme (IVT), modelin performansını ve verimliliğini artırmak için sürekli olarak kullanılan girdi değeri ile beraber, bu girdiler üzerine gelişen ve bu yönetime önemli bir avantaj sağlayan artımlı öğrenme teorisine dayanır (Giraud-Carrier, 2019). Diğer bir avantajı, diğer algoritmaların aksine oldukça yüksek kare hızlarında (yaklaşık 33 fps) çalışabilmesidir (Wu vd., 2013). Algoritmada ilk olarak otomatik veya manuel olarak hedef görüntüden konum belirlemek amacıyla tek bir nokta kullanılır. İlk çerçevedeki hedef nesnenin ortalamasının yanı sıra öz bazı (Math, 2022) sıfırlanır. Filtre kullanılarak hareketli modele göre parçacıkları çizerken, aynı zamanda bir sonraki çerçeveye geçilir (Okuma vd., 2004). Pencerenin her parçacık için geçerli çerçeveden çıkarılarak ağırlıkları hesaplanır. Olasılığı yüksek ağırlıktaki pencere kaydedilir ve yeni görüntülerin sayısı istenen bir değere ulaştığında artımlı güncelleme gerçekleştirilir. Hareketli modelden görüntü olarak süreç tekrarlanır.

Bir diğerk çalışma olan; California Üniversitesi tarafından tanıtılan Çoklu Örnekli Öğrenme ile Görsel İzleme (Multiple Instance Learning), kısaca MIL olarak anılacaktır (Babenko vd., 2009), bir denetimli öğrenme kategorisi olan çok örnekli öğrenmeye dayanır (Maron ve Lozano-Perez, 1998). Bu yöntem, ayrı ayrı etiketlenmiş örneklerin aksine, her biri çeşitli örnekler içeren bir dizi çerçeve veya parçacık sağlayarak gelişir. Algoritma yapısı olarak ilk önce görüntü elde edilir ve bu görüntülerin özellik vektörlerini hesaplar. Argmax işlevini kullanarak izleyici konumunu günceller (Argmax, 2022). İlk adımı iki kez tekrarlar. Son olarak MIL modeli, her biri setten bir görüntü içeren bir pozitif ve negatif kütüphane ile güncellenir (Babenko vd., 2009).

Yapılan bir başka çalışma ise, Online Acceleration (Çevrimiçi Hızlandırma) aracılığıyla gerçek zamanlı izleme, adından da anlaşılacağı gibi, gerçek zamanlı görsel izleme elde etmek için bir algoritmadır (Grabner vd., 2006). MIL'den farklı olarak, bu yaklaşım yalnızca çevrimiçi artırmaya dayanır. Bu yöntem üç ana kısımdan oluşur. $t+1$ çerçevesindeki söz konusu nesne bölgesinin civarındaki t çerçevesindeki sınıflar değerlendirilir. Daha sonra çıktılarından bir güven haritası oluşturulur. Güven haritası işlenerek en olası pozisyon üretilir ve böylece görüntü güncellenir.

2.2. Nesne Takibi İçin Kullanılabilecek Araçlar

2.2.1. Arduino kart

Arduino, bir mikro denetleyici kartında kolayca programlanabilen, silinebilen ve yeniden program kayıt edilebilen bir donanım aracıdır. Arduino'nun birkaç girişi vardır (örneğin; anahtarlar ve/veya sensörler) ve birkaç çoklu çıkışı kontrol eder (örneğin; ışıklar, motor ve diğerleri). Arduino programı; Windows, Macintosh ve Linux işletim sistemlerinde (OS) çalışabilmektedir.

Arduino, programlanması, yeni başlayanlar dahil herkes için öğrenmesi ve uygulaması kolay bir mikro denetleyicidir. Arduino, farklı bir anahtar ve/veya sensör koleksiyonundan girdi alarak, ışık, motor ve diğer fiziksel çıktıları kontrol etmek için ve birbirleriyle etkileşimli cihazlar yapmak için de kullanılabilmektedir. Arduino aktiviteleri tek tek de kalabilir veya programlanabilir kameralar gibi belirli bir cihaz üzerinde çalışan programlara da bağlanabilir. Devre kartı elle parça parça toplanabilir veya önceden monte edilmiş olarak satın alınabilir. Ayrıca, açık kaynaklı IDE ücretsiz olarak indirilebilir (Arduino, 2022).

Arduino kurulu, 2005 yılında programlama ve elektronik kavramı hakkında bilgisi olmayan öğrenciler için önerilen Ivrea Etkileşim Tasarım Enstitüsü'nü kurmuştur. Atmel tarafından üretilen, 8-bit Atmel AVR mikro denetleyiciler veya 32-bit Atmel ARM ile oluşturulan bu mikro denetleyiciler, IoT (Nesnelerin İnterneti) uygulamalarında da kullanılabilmektedir.

Arduino, diğer devre kartlarının çoğunun aksine programlanabilirdir. Arduino'nun yeni bir kod programlamak için bir donanım parçasının bağlantısını kesmesine gerek yoktur. Doğrudan USB kablosu ile bağlantı kurulabilmektedir. (Arduino, 2022). Her Arduino kartı tamamen açık kaynaklıdır. Açık kaynaklı olması nedeniyle, deneyime sahip programcılar tarafından geliştirilebilmektedir. Yazılımın dili, C++ kütüphaneleri tarafından genişletilebilir. Aduino, kullanıcıların Arduino kartlarını ayrı olarak tasarlamalarına ve ortak bir ortamda geliştirebilmelerine, buna bağlı olarak da Arduino'nun tam ihtiyaçlarına göre uyarlanmasına imkan vermektedir (Arduino, 2022).

Arduino'nun sağladığı yukarıda sıralanan özelliklerinden ötürü bu çalışmada Arduino kartı ve platformu kullanılmıştır.

2.2.1.1. Arduino programlama ortamı

Arduino Yazılımı veya Arduino Entegre Geliştirme Ortamı (IDE), kod yazmak için bir metin düzenleyiciye, bir mesaj alanına, bir metin konsoluna, ortak işlevler için düğmeler içeren bir araç çubuğuna ve bir dizi menüye sahiptir. Arduino Yazılımının kodları (programları) yüklemek ve birbirleri arasında bağlantı kurmak için Arduino donanımına bağlı olması gerekmektedir (Arduino, 2022). Şekilde 2.3'de örnek olarak çalışmada yazılan Arduino yazılımının bir bölümü IDE görüntüsüne örnek olarak gösterilmektedir.



Şekil 2.3. Çalışmada kullanılan Arduino yazılım IDE'sinden bir kesit.

2.2.2. Sensörler

Sensörler, hayatımızdaki fiziksel durumlardan gelen çeşitli girdi türlerini algılayan ve bunlara yanıt veren araçlardır. Sensörlerin girdi (input) değerleri ışık, ısı, hareket, nem, kuvvet, uzaklık, PH, ses, renk, basınç vb. olabilir ve sensörler bu değişimleri

agılayabilirler. Sensörlerin çıkış (output) değerleri girdilere göre belirlenir ve daha sonra insanlar tarafından okunabilir. Sensörler, miktar değerindeki değişiklikleri algılayan ve çıkış değerini elektrik sinyali gibi yapan cihazlardır (Jacob, 2010, s. 19).

2.2.2.1. RFID sensörü

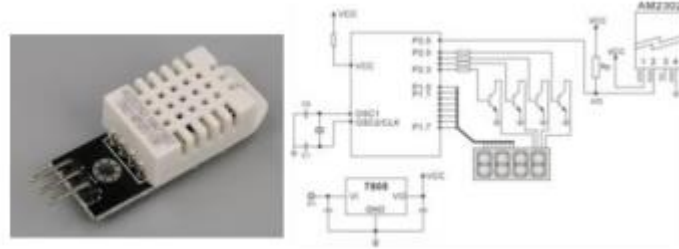
RFID (Radyo Frekansı Tanımlama), basit, düşük güç ve birçok kimlik ve erişim noktası seçeneği ile destekler. RFID teknolojisi, nesnelerle etkileşime giren işaretleri belirlemek ve izlemek için radyo TX (verici) ve RX (alıcı) olmak üzere iki yöntemden yararlanır (Wu, vd., 2009). Şekil 2.4’de RFID-RC522 sensörünün örnek fotoğrafı gösterilmektedir.



Şekil 2.4. RFID-RC522 sensör

2.2.2.2. Nem ve sıcaklık sensörü

DHT22 sensörü, düşük maliyetli ve enerjili bir dijital nem ve sensörüdür. Bu sensör, ortamı ölçmek ve veri pini üzerinde dijital bir sinyal iletmek için kapasitif bir nem sensörü ve bir termistör kullanır (dijital veya analog pinlere bağlanabilir). Kullanımı oldukça basittir, ancak veri toplaması gereken zamanla dikkatli olunmasını gerektirir. Şekil 2.5’de Nem ve Sıcaklık Sensörü tipi DHT22’nin resmi gösterilmektedir.



Şekil 2.5. Nem ve sıcaklık sensörü (DHT22)

2.2.2.3. Pasif Kızıl Ötesi (PIR) sensörü

Pasif Kızıl Ötesi (bundan sonra PIR olarak anılacaktır) sensörü, herhangi bir ortamdaki kızılötesi ışıktaki değişikliklere dayalı olarak bir hareket algılar. Hareket algılandığında otomatik alarm sağlamak için birkaç farklı sistemle birleştirilmiştir. Amacı, görüş alanındaki cisimlerden gelen kızılötesi ışığı ölçmektir. Bu sensör otomasyon, güvenlik, aydınlatma sisteminin kontrolü, otomatik kapılar vb. diğer sistemler için önemlidir (Budijono vd., 2014). Şekil 2.6’da Hareket sensörü PIR tipi HC-RS501’in resmi gösterilmektedir.



Şekil 2.6. Hareket sönsörü (PIR HC-RS501)

2.2.2.4. Manyetik (Reed) Anahtar sensörü

Reed Switch sensörü, basit bir yapı ile iletişim kuran, kontrolü kolay ve küçük boyutlu bir pasif elektronik anahtar türüdür. Ayrıca, bu sensör, bir manyetik alan sinyali tarafından kontrol edilen bir devre anahtarlama cihazıdır. Ayrıca, Reed anahtarı sensörü belirli bir alanı vb. sınırlamak ve saymak için kullanılabilir (özellikle kapıları ve pencereleri korumak için kullanılan güvenlik sistemlerinde de kullanılabilir). Şekil 2.7’de Reed Switch sensörünün örnek bir fotoğrafı gösterilmiştir.



Şekil 2.7. Arduino için Reed Switch sensörü

3. UYGULAMA

Bu tez çalışmasında hayvan hareketlerini algılayan ve otomatik olarak besleyen kapalı bir sistem oluşturulmuştur. Bu bölümde, tercih edilen donanım parçaları incelenerek devamında sistemin çalışma prensibi ve tasarımı, daha sonra yazılım bilgisi hakkında detaylı bilgi verilecektir.

3.1. Çalışmada Tercih Edilen Donanım Parçaları

Bu bölümde, çalışmada kullanılan donanım parçalarına detaylı olarak yer verilecektir.

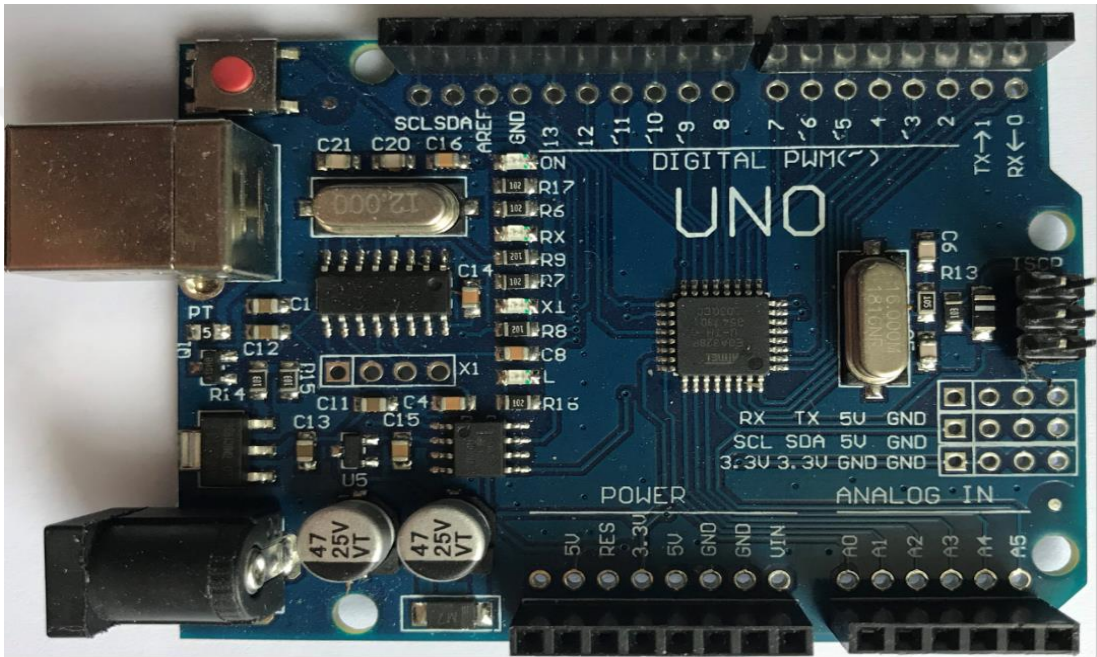
3.1.1. Arduino UNO R3 USB Chip CH340 (ATmega328 mikro denetleyici)

Bu çalışmada Arduino UNO, ATmega328 tercih edilmiştir. Arduino UNO ATmega328'e dayanan bir karttır. 14 dijital giriş/çıkış pimi (6'sı PWM çıkışları olarak kullanılabilir), 6 analog giriş, 16 MHz seramik rezonatör, bir USB bağlantısı, bir güç jakı, bir ICSP başlığı ve bir sıfırlama düğmesi vardır.

UNO versiyonu bir USB seri dönüştürücü olarak programlanmış Atmega16U2 (Atmega8U2up sürüm R2) içerir ve detaylı teknik özellikleri Çizelge 3.1.'de belirtilmiştir. Kullanılan Arduino UNO modeli Şekil 3.1.'de gösterilmektedir.

Çizelge 3.1. Arduino UNO ATmega 328 teknik özellikleri

Mikroçip	ATmega328
Çalışma Voltajı	5V
Giriş Voltajı (Limitler)	7-12V
Digital I/O Pin Sayısı	14
Analog I/O Pin Sayısı	6
I/O Pin DC Akım	40 Ma
Flash Memory	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
İşlemci Hızı	16 MHz



Şekil 3.1. ATmega328 Mikro Denetleyici Görüntüsü

3.1.2. MG996R servo motor

Servo motor, yapılması planlanan veya tanımlanması gereken belirli bir görev ve konum sırasında, parçaları iterek ve döndürerek çalışan bir elektrik motorudur. Bir servo motor tetiklendiğinde, motorun belirli bir parçası belirli bir konuma hareket ettirilir ve yeni bir görev verilene kadar orada kalır. Servo motorun input ve output değerleri, normalde 4,5V–6V arasında güce, topraklamaya ve kontrol için kablolarla ihtiyaç duyar. Şekil 3.2.’de Arduino için kullanılan, MG996R Servo Motor modeli gösterilmiştir.



Şekil 3.2. MG996R Servo Motor

MG996R Dijital Servo Motor, 10 kilografa kadar duraksama torkuna sahiptir. Mama haznesinin ağırlığı göz önüne alınarak tork değeri yüksek olan bir servo motor tercih edilmiştir. Yapılan çalışmada akım 500 mA ve çalışma hızı 4.8 V olarak tercih edilmiş olup, aynı zamanda 6V'ta 10Kg'ın üzerinde tork elde edebilir ve 120 ° tarama açısına sahiptir. Bu çalışmada 45 °'lik dönme açısı tercih edilmiş olup, Arduino UNO mikro denetleyici ile entegrasyonu sağlanarak programlanmıştır. 45 °'lik dönme açısı tercih edilmesinin sebebi, gerçek boyutlu tasarlanan mama haznesinin, mama indireceği miktarın ölçüsüne uygun olmasıdır. Taneli mama kullanıldığında 45 °'lik bölme haznesi, 60 gram mama indirme kapasitesine tekabül etmektedir.

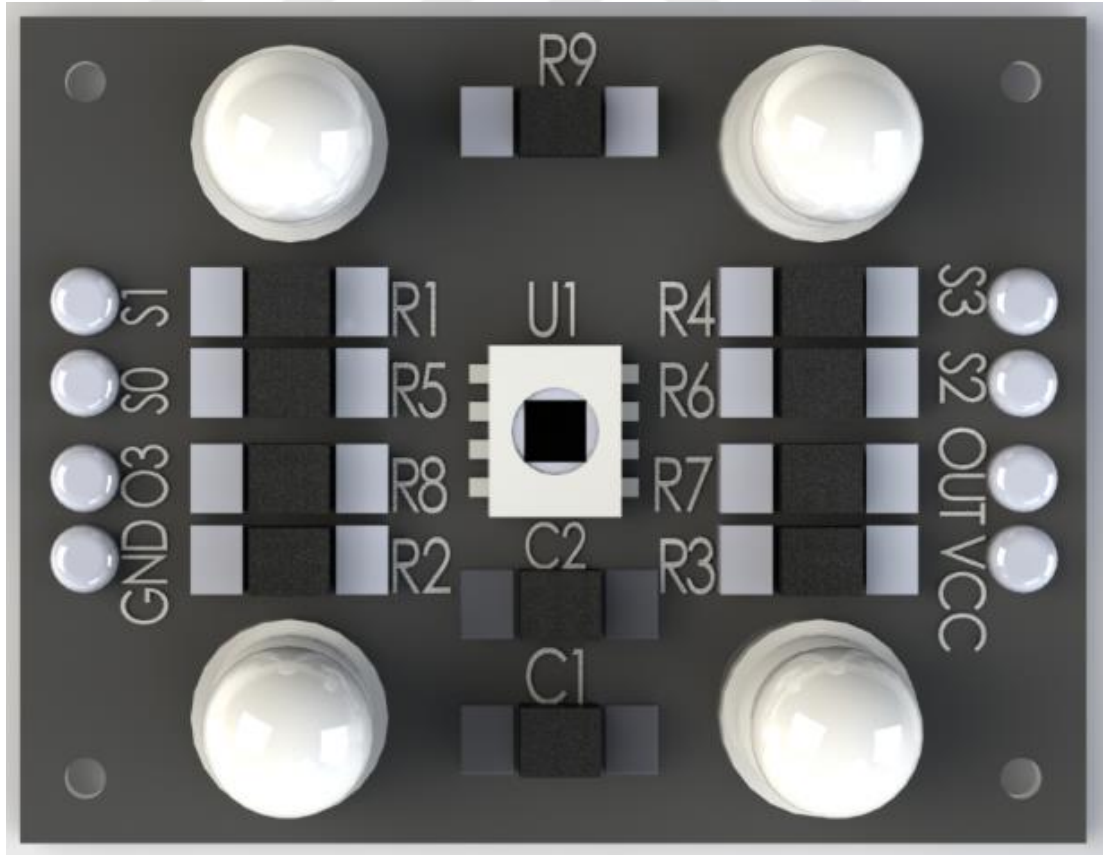
3.1.3. TCS230 / TCS3200 renk sensorü

TCS3200, programlanabilir renkli ışık-frekans dönüştürücüsüdür; tek bir CMOS (Bütünleyici Metal Oksit Yarı İletken) devrede, konfigüre edilebilir silikon foto diyotların birleştirilmesinden oluşmaktadır. Çıkış (output) değeri, ışık yoğunluğu (ışınlama) ile doğru orantılı frekansa sahip bir kare dalgadır. Tam ölçekli çıkış frekansı, iki kontrol giriş pimi aracılığıyla üç önceden ayarlanmış değerden herhangi biri tarafından ölçeklendirilebilir. Çalışmanın ilerleyen kısımlarında tercih edilen değer ve ölçekler detaylı açıklanacaktır. TCS3200'de, ışık-frekans dönüştürücü bir 8 x 8 dizi fotodiyot okur. On altı foto diyot mavi filtreler, 16 fotodiyot yeşil filtreler, 16 fotodiyot kırmızı filtreler ve 16 fotodiyot da berrak yani filtresizdir.

Fotodiyotların dört tipi (renkleri), insidentirradiansın tekdüzeliğinin etkisini en aza indirmek için interdijitalleştirilmiştir. Aynı renkteki tüm fotodiyotlar paralel olarak bağlanır. Pinlerden S2 ve S3, hangi fotodiyot grubunun (kırmızı, yeşil, mavi, berrak) aktif olduğunu seçmek için kullanılır. Diğer teknik özellikler Çizelge 3.2’de detaylı olarak gösterilmiştir. Şekil 3.3.’de ise, 3 Boyutlu (3D) ortamda çizilmiş gerçek boyutlu TCS3200 renk sensörü örneği bulunmaktadır.

Çizelge 3.2. TCS230 / TCS3200 teknik özellikleri

Terminal Name	Terminal No	I/O	Açıklama
GND	4		Güç kaynağı toprak
QE	3	I	f0 için etkinleştir (aktif düşük)
OUT	6	O	Çıkış frekansı (f0)
S0, S1	1,2	I	Çıkış frekansı ölçekleme seçimi girişleri
S2, S3	7,8	I	Fotodiyot tipi seçim girişleri
V _{DO}	5		Besleme gerilimi

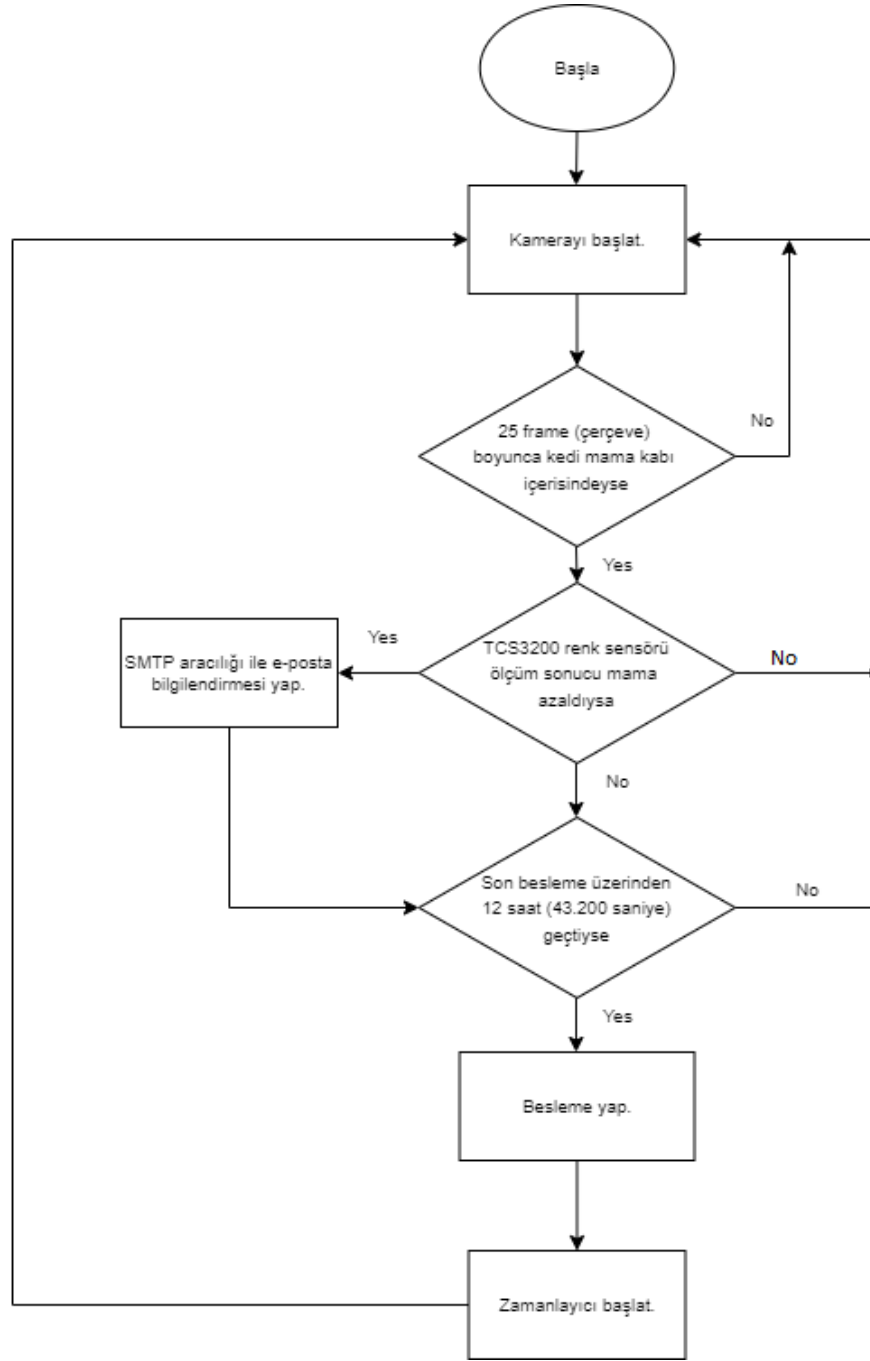


Şekil 3.3. Renk sensörü (TCS3200)

TCS230 / TCS3200 renk sensörü, rengi algılamak için TAOS TCS3200 RGB sensör çipini kullanır. Sensörün önünde nesneyi aydınlatan dört beyaz LED vardır. Çıkış frekansı 2Hz ile 500kHz arasındadır. Çalışma mantığı, HCSR04 ultrasonik mesafe algılama sensörüne benzerdir.

3.2. Kamera Kontrollü Akıllı Mama Kabı Çalışma Prensibi ve Tasarımı

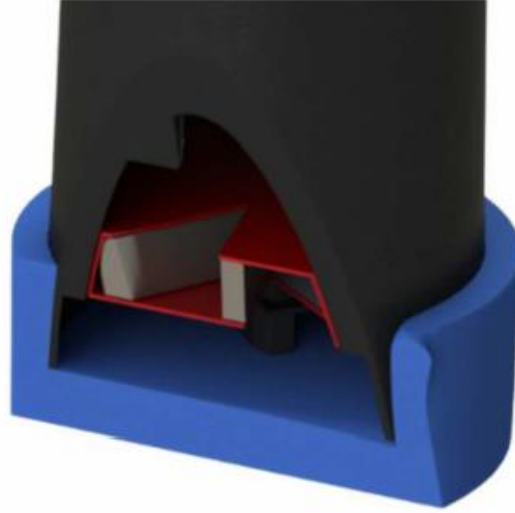
Prototip geliştirme için bilgisayar kamerası ve özel olarak tasarlanmış mama kabı tasarımı kullanılmıştır. Sistemin uygulama süreci bilgisayar ve mama kabının açısının ayarlanması ile başlamaktadır. Kamera, mama kabının konumuna odaklanmaktadır. Kamera başlatılır ve kedinin mama kabının bulunduğu alana girdiğini algılar. 25 çerçeve (frame) boyunca mama kabının çerçevesi içinde bir kedi tespit ederse, mama kabının toplam mama limiti TCS3200 renk sensörü tarafından kontrol edilmektedir. Mama kabı yeterince dolu değilse mama kabı sahibi SMTP (Basit Mail Aktarım Protokolü) kullanılarak e-posta aracılığıyla bilgilendirilir. Renk sensöründen alınan veriye göre mama kabı doluyorsa, Arduino UNO, seri portu üzerinden uyarılır. Son mama verilme (besleme) süresinin üzerinden 43.200 saniye (12 saat) geçtiyse Arduino UNO'ya bağlı MG996R servo motor tetiklenir. Servo Motor tetiklenme eylemini, belirtilen miktarda yiyeceği 45 derecelik bir git gel hareketi yaparak besleme haznesine dökmek için mama kabının içindeki özel açma kapama çarkını döndürerek sonlandırır. Böylece besleme işlemi gerçekleştirilir. Besleme işleminin gerçekleştirmesi ile tekrar yaklaşık 12 saatlik zamanlayıcı başlatılır. Tercih edilen bilgisayarın teknik özelliklerine göre saniye/hız durumu değişiklik gösterebilir. Sistemin genel akış şeması Şekil 3.4.'de gösterilmiştir.



Şekil 3.4. Sistemin genel akış şeması

Mama kabı için özel bir tasarım geliştirilmiştir. Mama kabının tasarımında 8 odacıklı ve 45 derece ile ayrılmış özel hazneli çark tasarlanmıştır. Hazneye ağırlık merkezinden bağlanan MG996R servo motor, çarkı merkezden döndürme yetkisine sahiptir. 8 odacığın mama haznesinin ve indirme işleminin yapıldığı bölmesinden başka, kalan 6 haznesinin kullanılmasının nedeni, servo motorun ağırlık merkezinden bağlı olması ve dengenin sağlanabilmesidir. Servo motor 45 derecelik bir açıyla açılır ve üstte

istiflenen mamanın besleme haznesine dolan kısmını boşaltır. 45 derecelik bir açıyla geriye doğru hareket ettirilerek açılan kısım tekrar kapanır ve mama dökme işlemi sona erer. Mama kabının hazneleri ve mama indirmenin yapıldığı bölge (a), mama kabı kapalıyken Servo Motor, Renk Sensörü ve çarkın konumu (b), mama kabının dış görünüşü (c) 3 boyutlu ortamda gerçek boyutlarıyla tasarlanmıştır ve Şekil 3.5.'de detaylı olarak gösterilmektedir.



(a)



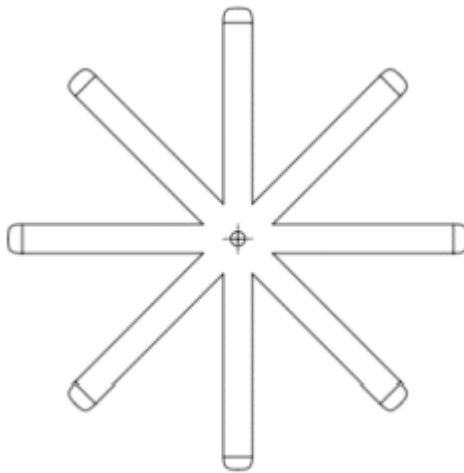
(b)



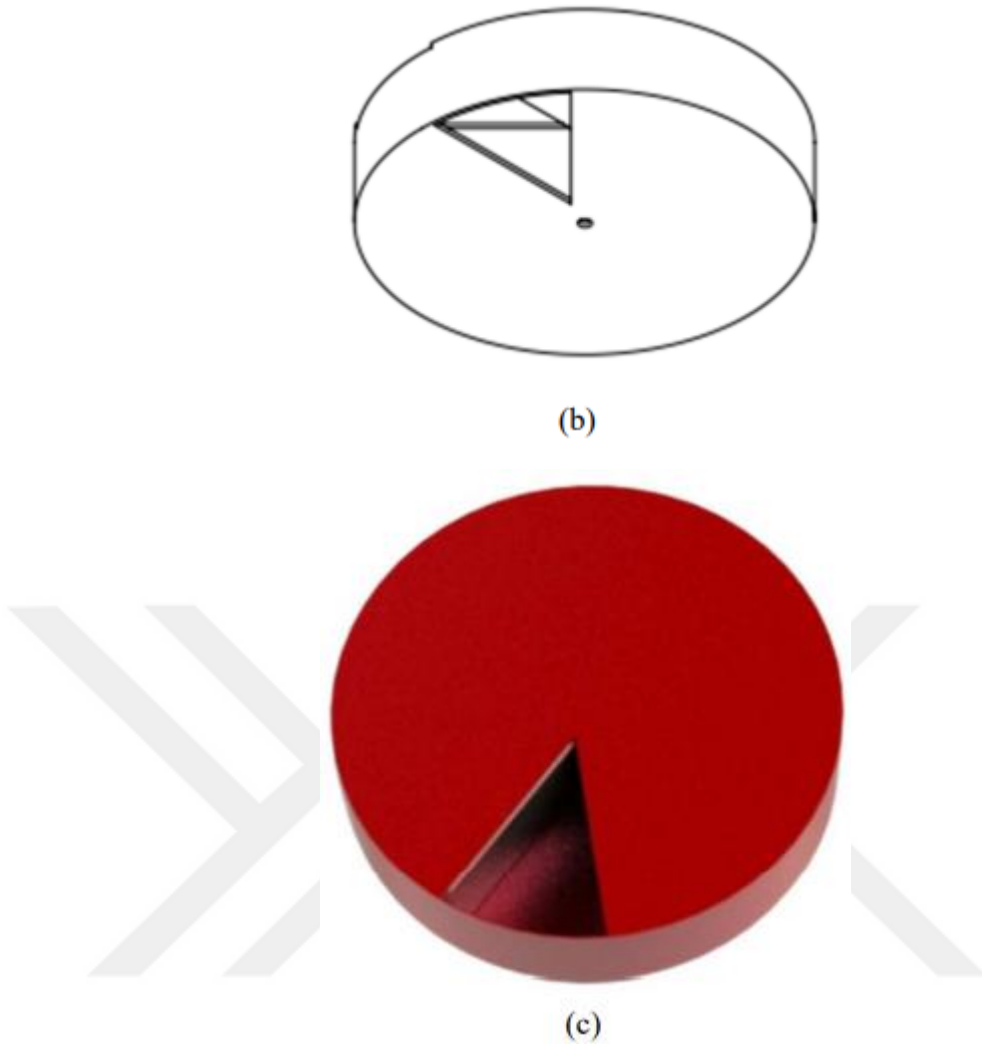
(c)

Şekil 3.5. Mama kabının detaylı görünümü. (a) Mama kabının hazneleri ve mama indirmenin yapıldığı bölge (b) Mama kabı kapalıyken Servo Motor, Renk Sensörü ve çarkın konumu. (c) Mama kabının dış görünüşü

Mama kabında oluşacak ağırlığı önlemek için çarkın üzerine destekleyici bir kapak yerleştirilmiştir. 3 boyutlu ortamda gerçek boyutlarıyla çizilen, 8 odacıklı çarkın görüntüsü (a), dış kapağın alttan görüntüsü (b) ve dış kapağın üstten görüntüsü (c) Şekil 3.6'da gösterilmiştir.



(a)



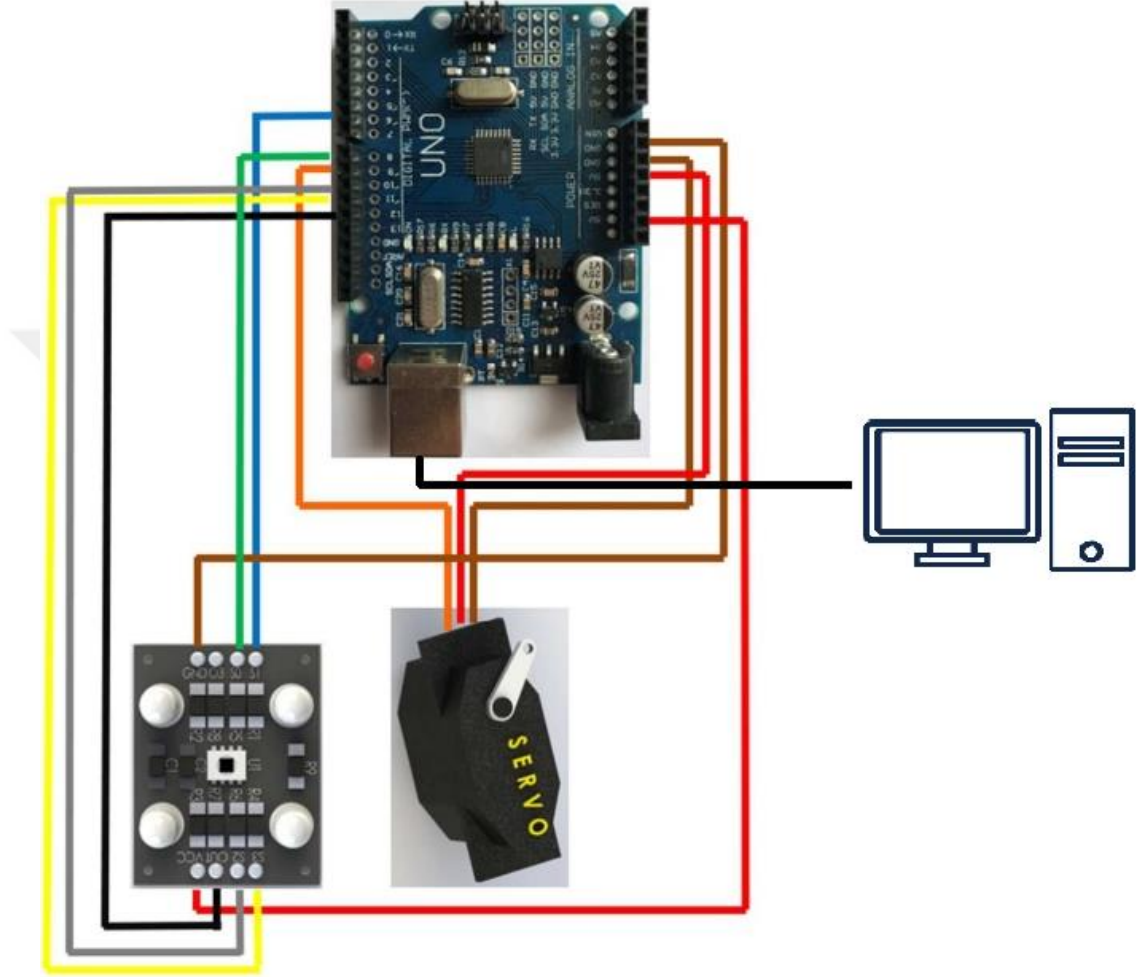
Şekil 3.6. Çarkın ve dış kapağın görüntüsü (a) 8 odacıklı çarkın görüntüsü (b) Dış kapağın alttan görüntüsü (c) Dış kapağın üstten görüntüsü

TCS230 / TCS3200 renk sensörü kullanılarak, mama kaptaki belirtilen seviyenin altına düştüğünde kullanıcı e-posta ile bilgilendirilir.

Renk sensörünün çıktılarında, mamaları mama kabına yerleştirerek RGB renk skalasında en fazla farkı yaratan ayırt edici renklerin yeşil olduğu görülmüş, böylece performans verimliliği için yalnızca sensör çıkışı için belirleyici olan yeşil çıkışlar seçilmiştir.

Arduino UNO, ATmega328 mikro denetleyici kullanır. Servo motoru tetiklemek ve renk sensörünü etkinleştirmek için Arduino UNO mikro denetleyici tercih edilmiştir. Arduino Uno'nun 9 numaralı PWM pini servo motorun kontrolü için kullanılmaktadır.

Topraklama için GND pini ve elektrik alması için 5V pini kullanılmıştır. Sistem, Arduino'nun seri portu üzerinden bilgisayar kamerasına bağlıdır ve bilgisayar kamerası ile sürekli olarak kontrol edilir. Projenin donanım parçalarının genel bağlantı şekli Şekil 3.7.'deki gibidir.



Şekil 3.7. Arduino UNO'nun MG 996R servo motor, bilgisayar ve TCS230/TCS3200 renk sensörü ile bağlantısı

3.3.Programlar ve Kütüphaneler

Bu bölümde, çalışmada kullanılan Python sistem yazılımı ve Arduino sistem yazılımı hakkında detaylı bilgi verilecektir.

3.3.1. Python sistem yazılımı

Python, 1990 yılında Hollandalı bir programcı olan Guido Van Rossum tarafından yazılmaya başlanmış bir programlama dilidir. Açık kaynak kodlu ve özgür yazılımı destekleyen ücretsiz bir yazılım olması geliştiriciler için tercih sebebidir. Python makine kodlarına derlenen bir dil değil, yorumlanan bir dildir. Derleme adımı olmadığı için, düzenleme-test-hata ayıklama süreci hızlıdır. Unix, Linux, Mac, Windows, Amiga, Symbian gibi farklı platformlarda çalıştırılabilir. Makine öğrenmesi ve robotik uygulamaları, oyun geliştirme, sistem ve ağ programlama, web programlama, veri analizi ve veri tabanı yazılımı programlama gibi birçok alanda kullanılır. Hızlı prototipleme imkânı vermesi C ya da C++ gibi dillere tercih edilmesine sebep olmuştur. Özellikle yapay zeka alanında açık kaynaklı ve deneyimli geliştiriciler tarafından paylaşılan python dilinde yazılmış birçok proje bulunmaktadır. Yapay zeka alanında geliştirilen çeşitli kütüphaneleri dolayısıyla bu alanda çoğunlukla tercih edilen yazılım dilidir.

Bu çalışmada, Python programlama dili ve TensorFlow, OpenCV kütüphanelerinden faydalanılmıştır.

3.3.2. TensorFlow

TensorFlow, yüksek performanslı sayısal hesaplama için kullanılan açık kaynaklı bir yazılım kütüphanesidir. Aslen Google'daki araştırmacılar ve mühendisler tarafından geliştirilen bu kütüphane, makine öğrenimi ve derin öğrenme için güçlü bir kütüphanedir (TF, 2022).

Bir kediye ayırt edebilmek için nesnelerdeki farklılıkları öğrenmek ve özellikle kedi olduğu açık olan kusursuz görüntülerle ilgili unsurları beslemek gerekir. Evcil hayvan dedektörü, bilgisayar aracılığı ile kamerada nesne algılama gerçekleştirmek için eğitilmiş olan SSD MobileNet modeli kullanılmıştır. Yapay sinir ağı, her karedeki nesneleri algılar ve görüntüdeki konumlarını elde eder. Algılanan her nesnenin üç özelliği vardır, algılama yüzdesi, konum ve sınıflandırma (Lin, 2014). Çalışmanın

yazılım kısmında, TensorFlow kütüphanesinin nesne tespit listesinde 17 numarada kedi sınıfı seçilerek daha sonra doğruluk değeri %90'ın üzerine çıkarılmıştır.

3.3.3. OpenCV (Açık kaynak Bilgisayarla Görme)

OpenCV, (Açık Kaynak Bilgisayarla Görme) temel olarak gerçek zamanlı bilgisayarlı görüş sistemlerine odaklanan açık kaynak kodlu bir kütüphanedir. OpenCV projesi ilk olarak 1999'da Intel tarafından başlatıldı ve projenin ilk versiyonu 2000'de çıktı. Proje daha sonra Willow Garage tarafından desteklendi ve şu anda Itseez tarafından sürdürülmektedir (OpenCV, 2022).

OpenCV'nin ana uygulama alanları görüntü işleme, video analizi, yüz tanıma ve nesne algılamadır. Bu özelliklerini açık kaynaklı ücretsiz bilgisayar görsel kitaplığı olmasından alır.

Platform orijinal olarak C/C++ kullanılarak geliştirilmiştir. Birincil arayüz hala C++'dır, ancak aynı zamanda daha eski, daha az kapsamlı bir C arayüzünü de korur. OpenCV, MATLAB, Java ve Python gibi birçok programlama dilinde kullanılabilen API (Uygulama programlama arayüzü) sunmuştur (Python, 2022).

Kameradan alınan görüntünün sürekli olarak algılanmasını sağlamak amacıyla OpenCV kütüphanesi tercih edilmiştir. İnsan beyni ve gözü gibi çalışan bu kütüphane ve içeriğindeki makine öğrenim algoritmaları sayesinde kameradan alınan görüntü bilgisayar gücü devam ettiği sürece devamlı kamera kontrolü yapabilmektedir. Model bir çerçevede kedi algıladığında, kedinin en az 25 çerçeve (frame olarak anılır) için önceden tanımlanmış kutu içinde algılanıp algılanmadığını kontrol eder.

3.3.4. Görüntü işleme algoritması

Görüntü işleme algoritması, yazılımın en önemli parçalarından birisidir. Programlama kodları, kamera ile iletişime geçmekten ve kameradan gelen verileri modeller yardımı ile işleyerek evcil hayvan tespiti için kullanılmaktadır. Algoritma, kamera kontrolünün sağlandığı bilgisayarda çalıştırılır.

Kodun ilk kısmında TensorFlow başlatılır ve görüntü tespiti için kullanılacak olan görüntü modelini sistem belleğine yükler.

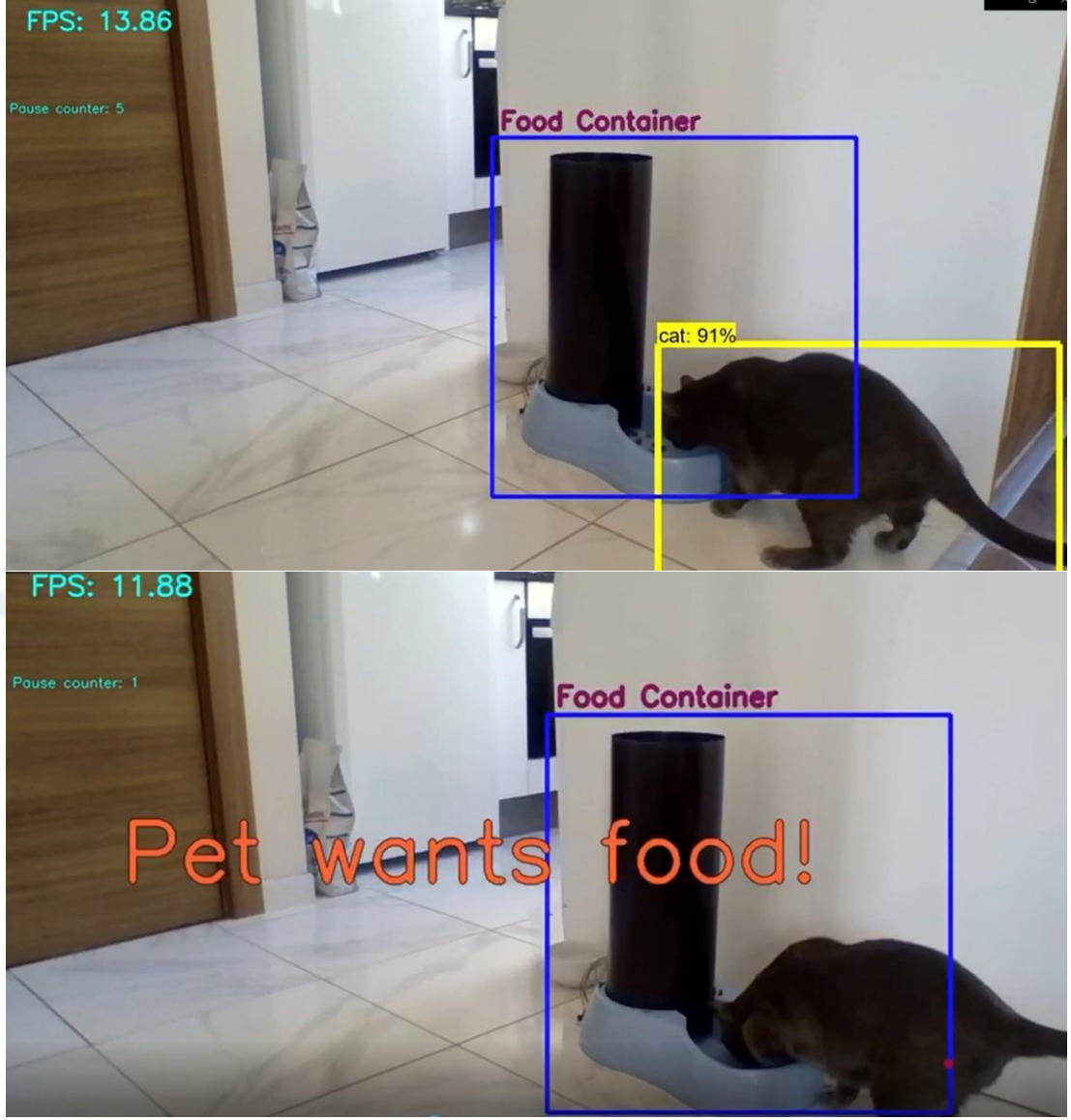
Ardından kamera ile iletişime geçer, kamerayı başlatır ve belirlenmiş FPS (Saniyedeki kare sayısı) hızında kameradan görüntü yakalamaya başlar. Program, kameradan alınan her kareyi, TensorFlow kütüphanesi kullanarak eğitilmiş görüntü işleme algoritmasında işler ve görüntü içerisinde evcil bir hayvan olup olmadığını kontrol eder.

Program, işlenen görüntü içerisinde bir evcil hayvan tespit ettiğinde, tespit edilen modeli programın belleğinde yer alan “classes[0][0]” değişken matrisine kayıt eder.

Ardından program, işlenen görüntülerin kayıt edildiği classes[0][0] değişkeninde depolanan bir değer olup olmadığını kontrol eder, eğer değişken matriste algılanan bir model varsa, algılanan nesnenin sınıfının bir kedi olup olmadığını sorgular (classes[0][0] matrisinde yalnızca algılanan nesnenin kimlik numarası kayıt edilir. Eğitilmiş MobileNet modeli, okunan değer, cat (kedi) sınıfına karşılık gelen 17 kimlik değeri (ID) olup olmadığını kontrol ederek hangi nesnenin algılandığını tespit eder.) .

Evcil hayvanın akıllı mama kabının önünde beklediğini doğrulamak için program evcil hayvanın 25 frame boyunca belirlenmiş alanda olup olmadığını kontrol eder (classes[0][0] değişkeninin 25 frame boyunca 17 numaralı sınıfa eşit olup olmadığını takip eder.).

Algoritmanın gerçek denemelerde kedi tespit edildiğindeki ekran çıktısı Şekil 3.8.’de gösterilmiştir.



Şekil 3.8. Algoritmanın gerçek hayattaki ekran çıktısı

3.3.5. Aksiyonların e-posta ile raporlanması

Programın bu bölümü akıllı mama kabındaki mama miktarı belirlenmiş seviyenin altına düşerse kullanıcıyı e-posta aracılığıyla bilgilendirilmesini sağlamaktadır.

Mama kabının seviyesini tespiti donanım tarafında gerçekleşmektedir. Mama miktarını tespit etmek için, donanım içerisindeki renk sensörü RGB (Kırmızı Yeşil Mavi) dedektörlerinden yeşil rengin output(çıktı) değerleri kontrol edilir ve veriler Arduino UNO kartına aktarılır. Arduino UNO, sensörden iletilen veriyi okur ve eğer

hedef deęerin altında bir renk deęeri tespit ederse, mama kabının belirlenmiř seviyenin altında dolu olduęunu algılar.

Mama miktarının belirli seviyenin altına düřtüęünü tespit ettięi durumda, Arduino UNO, seri baęlantı üzerinden bilgisayara bu tespiti iletir.

Arduino'dan veri okuyan sunucuya mama miktarı uyarısı iletildięi durumda, sunucudaki Python kodu, akıllı mama kabı kullanıcısına Basit Posta Aktarım Protokolü (SMTP) kullanılarak daha önce kaydedilmiř e-posta adresine bilgilendirme yapar.

3.3.6. Arduino Yazılımı

Donanım tarafında bulunan Arduino yazılımı, sunucudan aldıęı istek neticesinde mama verme iřlemini gerekleřtirmekten ve kendisine baęlı renk sensöründen okuduęu veriler neticesinde sunucuyu mamanın belirlenmiř seviyenin altına düřtüęünü bildirmekten sorumludur. Arduino yazılımı bilgisayardan aldıęı 5V ile sürekli olarak aktiftir.

Sunucu tarafı ile veri iletiřimine gemek için, “Serial.begin(9600)” komutu ile bilgisayar ile seri iletiřim bařlatılır. Arduino sürekli bir döngü halinde sunucudan ve renk sensöründen veri okur.

Eęer sunucudan iletilen SerialData deęeri “1” olursa, bu kedinin mama kabının önünde 25 frame boyunca bekledięi anlamına gelir ve besleme iřlemi gerekleřtirir.

Mama seviyesinin tespiti için, Arduino UNO kartında S0, S1, S2, S3 pinleri renk sersörü olarak tanımlanmıřtır. Daha sonra output deęeri olaran 12. pin seilmiřtir. S2 ve S3 deęerlerinin “HIGH” konumundayken renk sensöründen yeřil renk okuması gerekleřtirir. Renk sensöründen okunan deęer “greenValue” deęiřkenine kayıt edilir.

Eğer sensörden okunan değer 75'ten küçük olursa mama sınırın altında kalmıştır ve haznenin doldurulması gerekmektedir. Bu durumda Arduino Serial haberleşme ile sunucu tarafına veri iletimi sağlar ve sunucuya bilgilendirme gönderir.



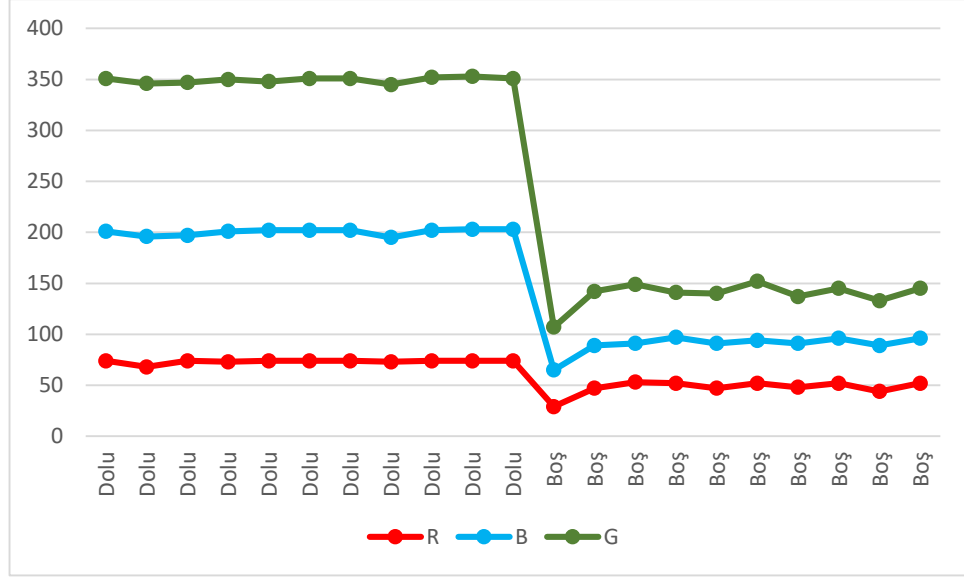
4. ARAŞTIRMA BULGULARI VE TARTIŞMA

Yapılan çalışmada sensörden algılanan renklerde en değer farkı yaratan renk yeşil olduğu tespit edilmiştir. Bu sebeple yeşil renk değeri referans alınmıştır. TCS230 / TCS3200 sensör kontrolü altında mama kabının boş ve dolu olduğundaki RGB değerleri Çizelge 4.1. 'de verilmiştir. RGB rengindeki boş ve dolu değerlerin grafiksel gösterimi Şekil 4.1.'de verilmiştir.

Mama kabının içerisinde tamamen karanlık bir ortam yaratabilmek için siyah yüzeyli mama haznesi tercih edilmiştir. Şeffaf bir mama kabı üretilmesi durumunda, renk sensörü ortam aydınlatması ve renk tonundan etkileneceği için tercih sebebi olmayacaktır.

Çizelge 4.1. : Mama kabı boş ve dolu olduğunda RGB değerleri

Doluluk	R	B	G
Dolu	74	127	150
Dolu	68	128	150
Dolu	74	123	150
Dolu	73	128	149
Dolu	74	128	146
Dolu	74	128	149
Dolu	74	128	149
Dolu	73	122	150
Dolu	74	128	150
Dolu	74	129	150
Dolu	74	129	148
Boş	29	36	42
Boş	47	42	53
Boş	53	38	58
Boş	52	45	44
Boş	47	44	49
Boş	52	42	58
Boş	48	43	46
Boş	52	44	49
Boş	44	45	44
Boş	52	44	49



Şekil 4.1. Boş ve dolu değerlerin grafiği

Modelde karşılaşılan olası problemlerden ilki, 0,39 inçten (1 cm) daha küçük mamaların mama kabının çarkla teması sonucunda çarka takılabilmesi durumudur. Genel amaçlı kullanım için malzeme cinsini değiştirmek ve daha güçlü ve pürüzsüz bir malzeme ile çalışmak gerekli görünmektedir. Ayrıca algılanan kare sayısı FPS'ye göre değişir. Bu nedenle gerçekte 12 saate yakın olan mama indirme için gerekli bekleme süresi, kullanılan cihazın donanım özelliklerine göre değişiklik gösterebilir. Dolayısıyla net bir süre belirlenemez. Bu durum, model tasarımının geliştirilmesi gereken özelliklerinden biridir.

5. SONUÇ VE ÖNERİLER

Yapılan çalışmada tamamen kapalı bir sistem içerisinde hayvan mamasını muhafaza eden, akıllı bir mama kabı geliştirilmiştir. Önerilen sistem modelinin prototipi Şekil 5.1.'de görülmektedir.



Şekil 5.1 : Önerilen sistem modelinin bir prototipi

Önerilen sistem modelinde kedi dışında farklı nesneler algılanırsa besleme işlemi gerçekleşmemektedir. Bunu sağlamak için eşik değeri artırılmış ve eğitilmiş modelden sadece kedi (cat) sınıfı kullanılarak %90'ın üzerinde bir doğrulukla çalışma sağlanmıştır. Bu şekilde, farklı nesnelerin veya hayvanların sistem tarafından algılanması önlenir. Kullanılan model ve geliştirilen yazılım sayesinde sadece kedi tespit edilir ve kedi maması dökülür.

Projenin sokak hayvanları için kedi dışında birden fazla hayvan için çoklu besleyici şeklinde tasarlanabilmesi ileride yapılacak bir çalışma konusu olup, farklı disiplinlerin katılımını gerektirmektedir. Bu bağlamda ürün sosyal sorumluluk ürünü olarak da bağış niteliğinde mahallerde devreye alınabilir. Bu ürün hem evdeki arkadaşlarımız için hem de sokaktaki kedi ve köpek dostlarımız için sosyal sorumluluk projesi olarak

kullanılabilir. Önerilen modelin fabrikasyon modelleri daha uygun maliyetle üretilebilir.

Günümüzde pek çok kişinin ev arkadaşı olan kediler düşünüldüğünde düşük maliyet ön planda tutulmuştur. Önerilen model ayrıca kolayca üretilebilir ve güvenilir bir tasarıma sahiptir.

Bilgisayar kontrolü yerine geliştirme kartı kullanılarak ürün dış ortamda yeterli muhafaza yöntemleri sağlanarak da kullanılabilir.

Model, sokak hayvanlarının kullanımında gelecekteki çalışmalar için kedi ve köpek sınıfı seçilerek, çift hazneye çıkarılabilir. Geniş açıdan bakıldığında model aynı zamanda bir sosyal sorumluluk ürünüdür ancak ürün bilgisayar kontrolü yerine geliştirme kartı kullanılarak dış ortamlarda da kullanılabilir.

KAYNAKLAR

- Arduino, 2018. What is Arduino? Erişim Tarihi: 01.03.2022
<https://www.arduino.cc/en/Guide/Introduction>
- Argmax., 2022, argmax() function - IBM. Erişim Tarihi: 14.03.2022
<https://www.ibm.com/docs/en>
- Arıkan, A., Bakır , Y., Özden, M., 2019. Çocuklarda Sosyal Duyarlılığın Geliştirilmesi, Sokak Hayvanlarının Yardımcıları Projesi. Mehmet Akif Ersoy Üniversitesi Eğitim Fakültesi Dergisi, 463-490.
- Aşar, H. 2018. Hayvan Haklarına Yönelik Temel Görüşler ve Yanılgıları. Kaygı 3, 239-251.
- Babenko, B., Belongie, S., Yang, M. H., 2009. Visual tracking with online multiple instance learning 2009 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work. CVPR Work. vol. IEEE Computer Society Conference on Computer Vision and Pattern, 983–990.
- Bhide, V. H., Wagh, S., 2015. I-learning IoT: An Intelligent Self Learning System for Home Automation Using IoT. International Conference on Communications and Signal Processing (ICCSP), 15-20.
- Brush, A. B., Lee, B., Mahajan, R., Agarwal, S., Saroiu, S., Dixon, C., 2017. Home Automation in the Wild. Challenges and Opportunities.
- Budijono, S., Andrianto, J., & Noor, M. A. 2014. Design and implementation of modular home security system with short messaging system. 68. EPJ Web of Conferences, EDP Sciences, 1-5.
- Campaign. 2022, Ocak 22. “Canımız Sokakta” projesi sokak hayvanları için ortak değer yaratıyor.
Erişim Tarihi: 13.04.2022. <https://www.campaigntr.com/canimiz-sokakta-projesi-sokak-hayvanlari-icin-ortak-deger-yaratiyor/>
- D20HABER. 2022, Şubat 5. 28 Cats Killed by Poisoning in Denizli.
Erişim Tarihi: 13.03.2022. <https://d20haber.com/denizli/ilceler/cal/denizlide-28-kedi-zehirlenerek-olduruldu/>
- Galinina, O., Mikhaylov, K., Andreev, S., Turlikov, A., & Koucheryavy, Y. 2015. Smart Home Gateway System Over Bluetooth Low Energy with Wireless Energy Transfer Capability. EURASIP Journal on Wireless Communications and Networking 2015, 1–18.
- Giraud-Carrier, C., 2019. A note on the utility of incremental learning.
Grabner, H., Grabner, M., & Bischof, H., 2006. Real-time tracking via on-line boosting. Proc. Br. Mach. Vis. Conf, 47–56.

- Güçül, G. N., Sarıtaş, M., 2008. Akıllı Ev Sistemleri Ve Uygulaması. Journal of Science and Technology of Dumlupınar University, 49-60.
- Hammarstrand, L., Svensson, L., Sandblom, F., & Sörstedt, J., 2012. Extended object tracking using a radar resolution model. IEEE Trans. Aerosp. Electron. Syst., 2371–2386.
- Haytap., 2008, Ocak 2. Erişim Tarihi: 13.03.2022. <https://www.haytap.org/tr/bagislarinizi-nerede-kullaniyoruz/far-kindalik-calismalarimiz>
- Haberler.com, 2022, Şubat 10. 3 Stray Cats Found Poisoned. Erişim Tarihi:22.04.2021. <https://www.haberler.com/guncel/adana-da-3-sokak-kedisi-zehirlenmis-halde-bulundu-14077118-haberi/>
- Howard, A., Zhmoginov, A., Chen, L.-C., Sandler, M., & Zhu, M., 2018. Inverted Residuals and Linear Bottlenecks. Mobile Networks for Classification, Detection and Segmentation. CoRR, 43-81.
- Howard, A., Zhmoginov, A., Chen, L.-C., Sandler, M., & Zhu, M., 2018. Inverted Residuals and Linear Bottlenecks. Mobile Networks for Classification, Detection and Segmentation. CoRR , 18-20.
- Hu, R., Dollar, P., He, K., Darrell, T., Girshick, R., 2018. Learning to Segment Every Thing. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit, 4233–4241.
- Ibrahim, M., Zakaria, H., Wei Xian, E., 2019. Pet Food Autofeeder by Using Arduino. Int. Conf. on Applications and Design in Mechanical Engineering Penang Island: Malaysia., 1-5.
- Jacob, F. 2010 Handbook of Modern Sensors. 4th Edition, New York, JSTF. TensorFlow.js. Erişim Tarihi: 12.03.2022. <https://www.tensorflow.org/js/>
- Khatavkar, H., Kini, R., Pandey, S., Gijare, V., 2019. Intelligent Food Dispenser (IFD). Journal of Engineering (IOSRJEN), 65-70.
- Kim, J. Y., Lee, H. J., Son, J. Y., 2015. Smart Home Web of Objects Based IoT Management Model and Methods for Home Data Mining. Asia Pacific Network Operations and Management Symposium (APNOMS), 15-20.
- Krohn, J., Beyleveld, G., Bassens, A., 2019. Deep Learning Illustrated, Addison-Wesley Professional, 416, Boston, Massachusetts, USA.
- Lin, T. Y., 2014. Microsoft COCO, Common Objects in Context. CoRR , 14-20.
- Lo, P., Masters, M. B., Studies, E., & Science, C., 2010. Contextual Bayesian Inference for Visual Object Tracking and Abnormal Behavior Detection.

- Mahanty, A., Engavle, A., Bootwala, T., Jaiswal, I., 2019. Proposed System for Animal Recognition using Image Processing. *International Research Journal of Engineering and Technology (IRJET)*, 2955-2957.
- Maron, O., Lozano-Perez, T., 1998. A framework for multiple-instance learning. *Adv. Neural Inf. Process. Syst.*, 570–576.
- Math., 2022, Ocak 30. What is an Eigenbasis and how do I calculate it with the information below. Erişim Tarihi: 01.03.2022. <https://canvas.harvard.edu>
- Moser, K., Harder, J., Koo, S. G., 2014. Internet of Things in Home Automation and Energy Efficient Smart Home Technologies. *International Conference on Systems, Man, and Cybernetics (SMC)*, 14-20.
- NYT., 2022, Şubat 10. A New Deal for Turkey’s Homeless Dogs. *nytimes.com*: Erişim Tarihi: 12.04.2022. <https://www.nytimes.com>
- Ogawa, T., Sakai, H., Suzuki, Y., Takagi, K., & Morikawa, K., 2011. Pedestrian detection and tracking using in-vehicle lidar for automotive application. *IEEE Intell. Veh. Symp. Proc.*, 734–739.
- Okuma, K., Taleghani, A., De Freitas, N., Little, J. J., & Lowe, D. G., 2004. A boosted particle filter: Multitarget detection and tracking. *Lect. Notes Comput. Sci.*, 28–39.
- OpenCV, Erişim Tarihi: 01.05.2022. <http://www.opencv.org/about.html>
- Örselli, E., Akbay, C., 2019. Teknoloji ve Kent Yaşamında Dönüşüm, Akıllı Kentler. *Uluslararası Yönetim Akademisi Dergisi*, 228-241.
- Pandey, S., Paul, A., Chanu, L. J., 2015. Life-Cycle Tracking System of Home Automation Devices (LED Bulbs), *IEEE. International Conference on Green Computing and Internet of Things (ICGCIoT)*, 15-20.
- Python., OpenCV.org. How OpenCV-Python Bindings Works, OpenCV community. Erişim Tarihi: 01.02.2022. http://docs.opencv.org/3.2.0/da/d49/tutorial_py_bindings_basics.html
- Ren, X., Jiang, H., Wu, Y., Yang, X., Liu, K., 2012. The internet of things in the license plate recognition technology application and design. *Int. Conf. Bus. Comput. Glob. Informatiz*, 969–972.
- Richter, E., Schubert, R., Wanielik, G., 2008. Radar and vision based data fusion . Advanced filtering techniques for a multi object vehicle tracking system. *IEEE Intell. Veh. Symp. Proc.*, 120–125.
- Ross, D. A., Lim, J., Lin, R. S., Yang, M. H., 2008. Incremental learning for robust visual tracking. *Int. J. Comput. Vis.*, 125–141.

- Samuel, S. S., 2016. A Review of Connectivity Challenges in IoT-Smart Home, International Conference on Big Data and Smart City (ICBDSC), IEEE. International Conference on Big Data and Smart City , 16-20.
- Sathya, R., Abraham, A., 2013. Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification. Int. J. Adv. Res. Artif. Intell., 13-20.
- Shamszaman, Z. U., Lee, S., Chong, I., 2014. WoO Based User Centric Energy Management System in The Internet of Things. International Conference on Information Networking 2014 (ICOIN2014), 14-20.
- Shibley, K., 2022, Şubat 5. Automated Pet Food Dispenser. digitalcommons.calpoly.edu. Erişim Tarihi: 12.03.2022. <https://digitalcommons.calpoly.edu/cgi/viewcontent.cgi?article=>
- Stoloiu-Crisan, C., Crisan, C., Butunoi, B.-P., 2021. An IoT-Based Smart Home Automation System. Sensors , 1-23.
- Subhash, S., Morelande, M. R., Musicki, D., Evans, R. J., 2011. Fundamentals of Object Tracking. Cambridge : Cambridge University Press, 103-132, Melbourne.
- Torres, J., 2018. Learning process of a neural network. Erişim Tarihi: 12.04.2022. <https://towardsdatascience.com/learning-process-of-a-deep-neural-network-5a9768d7a651>
- Vineeth, S., Renukumar, B. R., Sneha, V. C., Prashant, G., Rani, B., 2020. Automatic Pet Food Dispenser using Digital Image Processing. Journal of Engineering Research & Technology (IJERT), 588-593.
- Widrow, B., Lehr, M. A., 1990. 30 Years of Adaptive Neural Networks: Perceptron, Madaline, and Backpropagation. Proc. IEEE, 1415–1442.
- Wu, D. L., Wing, W. Y., Yeung, D. S., Ding, H. L., 2009. A Brief Survey on Current RFID Applications. International Conference on Machine Learning and Cybernetics, 2330-2335.
- Wu, Y., Lim, J., & Yang, M. H., 2013. Online object tracking: A benchmark. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., 2411–2418.

EKLER

EK A. Smart_food_container.py

EK B. Sensor_with_arduino.ino

EK C. Ürünün Gerçek Boyutlu Teknik Çizimleri



EK A. Smart_food_container.py

```
1. import os
2. import cv2
3. import numpy as np
4. import sys
5. import utils
6. import tensorflow as tf
7. import serial
8. import time
9. import threading
10.
11. from email.mime.multipart import MIMEMultipart
12. from email.mime.text import MIMEText
13. import smtplib
14.
15. arduinoData = serial.Serial(port='COM3', baudrate=9600)
16.
17.
18. IM_WIDTH = 1280
19. IM_HEIGHT = 720
20. camera_type = 'usb'
21. sys.path.append('.')
22.
23. isReadyToFeed = True;
24.
25. # Araçlar içe aktarılır.
26. from object_detection.protos import string_int_label_map_pb2
27. from object_detection.utils import label_map_util
28. from object_detection.utils import visualization_utils as vis_util
29.
30.
31. MODEL_NAME = 'ssdlite_mobilenet_v2_coco_2018_05_09'
32.
33. CWD_PATH = os.getcwd()
34.
35. PATH_TO_CKPT = os.path.join(CWD_PATH,MODEL_NAME,'frozen_inference_graph.pb')
36.
37. PATH_TO_LABELS = os.path.join(CWD_PATH,'data','muscoco_label_map.pbtxt')
38.
39. NUM_CLASSES = 20
40.
41. label_map = label_map_util.load_labelmap(PATH_TO_LABELS)
42. categories = label_map_util.convert_label_map_to_categories(label_map,
43. max_num_classes=NUM_CLASSES, use_display_name=True)
44. category_index = label_map_util.create_category_index(categories)
45.
46. detection_graph = tf.Graph()
47. with detection_graph.as_default():
48.     od_graph_def = tf.compat.v1.GraphDef()
49.     with tf.io.gfile.GFile(PATH_TO_CKPT, 'rb') as fid:
50.         serialized_graph = fid.read()
51.         od_graph_def.ParseFromString(serialized_graph)
52.         tf.import_graph_def(od_graph_def, name='')
53.     sess = tf.compat.v1.Session(graph=detection_graph)
54.
55.
56. image_tensor = detection_graph.get_tensor_by_name('image_tensor:0')
57.
58. detection_boxes = detection_graph.get_tensor_by_name('detection_boxes:0')
59.
60. detection_scores = detection_graph.get_tensor_by_name('detection_scores:0')
61. detection_classes = detection_graph.get_tensor_by_name('detection_classes:0')
62.
63. num_detections = detection_graph.get_tensor_by_name('num_detections:0')
64.
65. frame_rate_calc = 1
```

```

66. freq = cv2.getTickFrequency()
67. font = cv2.FONT_HERSHEY_SIMPLEX
68.
69.
70. TL_container = (int(IM_WIDTH*0.46),int(IM_HEIGHT*0.25))
71. BR_container = (int(IM_WIDTH*0.8),int(IM_HEIGHT*0.85))
72.
73. detected_container = False
74. container_counter = 0
75. pause = 0
76.
77. def cat_in_action():
78.     arduinoData.write(b'1')
79.     cv2.putText(frame,'Food
was
given!',(int(IM_WIDTH*.1),int(IM_HEIGHT*.5)),font,3,(45,101,255),5,cv2.LINE_AA)
80.
81. def pet_detector(frame):
82.
83.     global detected_container
84.     global container_counter
85.     global pause
86.
87.     frame_expanded = np.expand_dims(frame, axis=0)
88.
89.     (boxes, scores, classes, num) = sess.run(
90.         [detection_boxes, detection_scores, detection_classes, num_detections],
91.         feed_dict={image_tensor: frame_expanded})
92.
93.     vis_util.visualize_boxes_and_labels_on_image_array(
94.         frame,
95.         np.squeeze(boxes),
96.         np.squeeze(classes).astype(np.int32),
97.         np.squeeze(scores),
98.         category_index,
99.         use_normalized_coordinates=True,
100.        line_thickness=8,
101.        min_score_thresh=0.90)
102.
103.     cv2.rectangle(frame,TL_container,BR_container,(255,20,20),3)
104.     cv2.putText(frame,"Food
Container",(TL_container[0]+10,TL_container[1]-
10),font,1,(90,20,125),3,cv2.LINE_AA)
105.
106.
107.
108.
109.     if (((int(classes[0][0]) == 17)) and (pause == 0)):
110.         x = int(((boxes[0][0][1]+boxes[0][0][3])/2)*IM_WIDTH)
111.         y = int(((boxes[0][0][0]+boxes[0][0][2])/2)*IM_HEIGHT)
112.
113.         cv2.circle(frame,(x,y), 5, (75,13,180), -1)
114.
115.         if ((x > TL_container[0]) and (x < BR_container[0]) and (y > TL_container[1]) and (y <
BR_container[1])):
116.             container_counter = container_counter + 1
117.             if container_counter > 25 and pause == 0:
118.                 detected_container = True
119.                 container_counter = 0
120.                 pause = 1
121.
122.             if pause == 1:
123.                 if detected_container == True:
124.                     cv2.putText(frame,'Pet
wants
food!',(int(IM_WIDTH*.1),int(IM_HEIGHT*.5)),font,3,(0,0,0),7,cv2.LINE_AA)
125.                     cv2.putText(frame,'Pet
wants
food!',(int(IM_WIDTH*.1),int(IM_HEIGHT*.5)),font,3,(45,101,255),5,cv2.LINE_AA)
126.                     detected_container = False
127.                     global isReadyToFeed
128.                     if isReadyToFeed:
129.                         isReadyToFeed = False
130.                     cat_in_action()

```

```

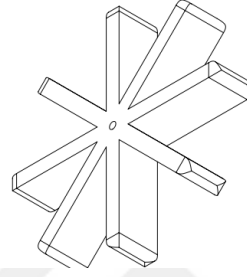
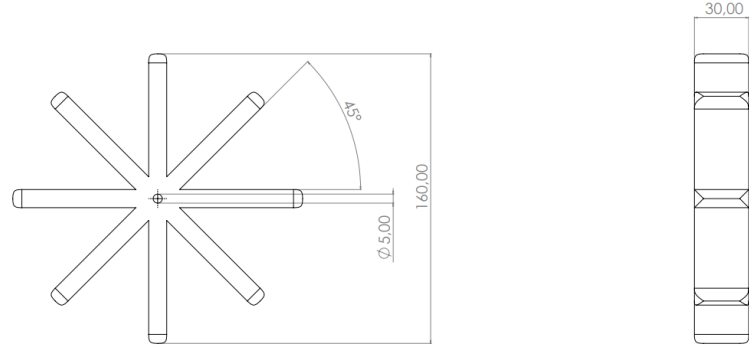
131.         x = threading.Thread(target=checkReadyToFeed, args=())
132.         x.start()
133.
134.         arduinoReadData = arduinoData.readline().decode('ascii')
135.         val = arduinoReadData.split("_")
136.
137.         if(val[0]=="empty"):
138.             print(arduinoReadData + "BOTTOM")
139.             mailGonder()
140.         return frame
141.
142. def checkReadyToFeed():
143.     time.sleep(43200)
144.     global isReadyToFeed
145.     global detected_container
146.     global container_counter
147.     global pause, pause_counter
148.     isReadyToFeed = True
149.     pause = 0
150.     detected_container = False
151.
152. def mailGonder():
153.     msg = MIMEMultipart()
154.     message = "Akıllı mama kabınızdaki mama miktarı azalmıştır."
155.     password = "*****"
156.     msg['From'] = "x*****@gmail.com"
157.     msg['To'] = "y*****@gmail.com"
158.     msg['Subject'] = "Akıllı Mama Kabı Bilgilendirme"
159.
160.     msg.attach(MIMEText(message, 'plain'))
161.
162.     server = smtplib.SMTP('smtp.gmail.com: 587')
163.     server.starttls()
164.
165.     server.login(msg['From'], password)
166.     server.sendmail(msg['From'], msg['To'], msg.as_string())
167.     server.quit()
168.
169.     print("successfully sent email to %s:" % (msg['To']))
170.
171. if __name__ == '__main__':
172.     camera = cv2.VideoCapture(0)
173.     ret = camera.set(3,IM_WIDTH)
174.     ret = camera.set(4,IM_HEIGHT)
175.
176.     while(True):
177.
178.         t1 = cv2.getTickCount()
179.
180.         ret, frame = camera.read()
181.
182.         frame = pet_detector(frame)
183.
184.         cv2.putText(frame,"FPS: {0:.2f}".format(frame_rate_calc),(30,50),font,1,(255,255,0),2,cv2.LINE_AA)
185.
186.         cv2.imshow('Object detector', frame)
187.
188.         t2 = cv2.getTickCount()
189.         time1 = (t2-t1)/freq
190.         frame_rate_calc = 1/time1
191.
192.         if cv2.waitKey(1) == ord('q'):
193.             break
194.         camera.release()
195.     cv2.destroyAllWindows()
196.

```

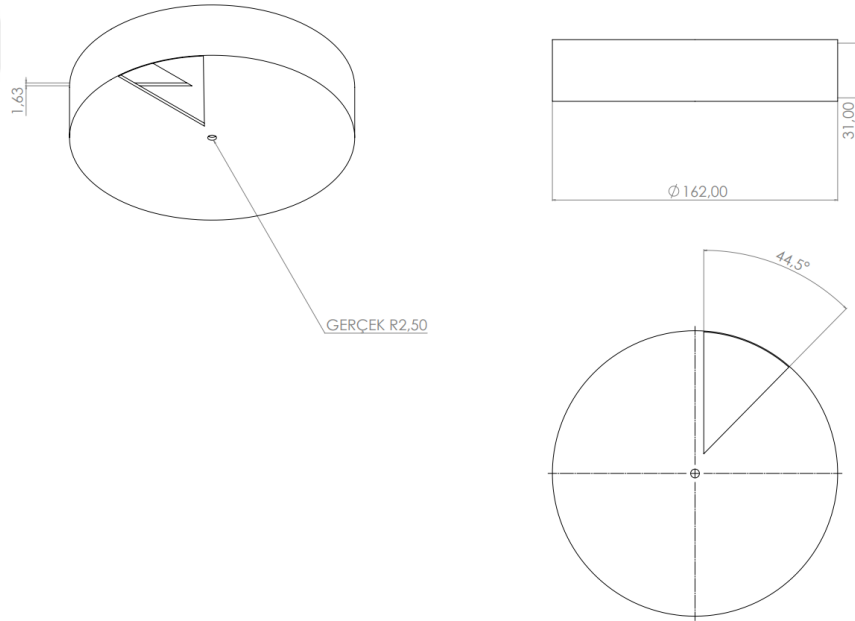
EK B. Sensor_with_arduino.ino

```
1.  #include <Servo.h>
2.
3.  #define s0 8
4.  #define s1 6
5.  #define s2 10
6.  #define s3 11
7.
8.  #define out 12
9.
10. int data = 0;
11.
12. Servo servoMain;
13. char serialData;
14.
15. void setup() {
16.   pinMode(s0, OUTPUT);
17.   pinMode(s1, OUTPUT);
18.   pinMode(s2, OUTPUT);
19.   pinMode(s3, OUTPUT);
20.
21.   pinMode(out, INPUT);
22.   digitalWrite(s0, HIGH);
23.   digitalWrite(s1, HIGH);
24.
25.   servoMain.attach(9);
26.   servoMain.write(0);
27.   Serial.begin(9600);
28. }
29.
30. void loop() {
31.
32.   while (Serial.available() > 0) {
33.     serialData = Serial.read();
34.
35.     if (serialData == '1') {
36.       servoMain.write(45);
37.       delay(1000);
38.       servoMain.writeMicroseconds(500);
39.       delay(4000);
40.       exit(0);
41.     }
42.   }
43.   digitalWrite(s2, HIGH);
44.   digitalWrite(s3, HIGH);
45.
46.   int greenValue = GetData();
47.   if(greenValue < 75){
48.     Serial.println("empty_");
49.   }
50.
51.   delay(2000);
52. }
53.
54. int GetData() {
55.   data = pulseIn(out, LOW);
56.   delay(10);
57.   return data;
58. }
59.
60.
```

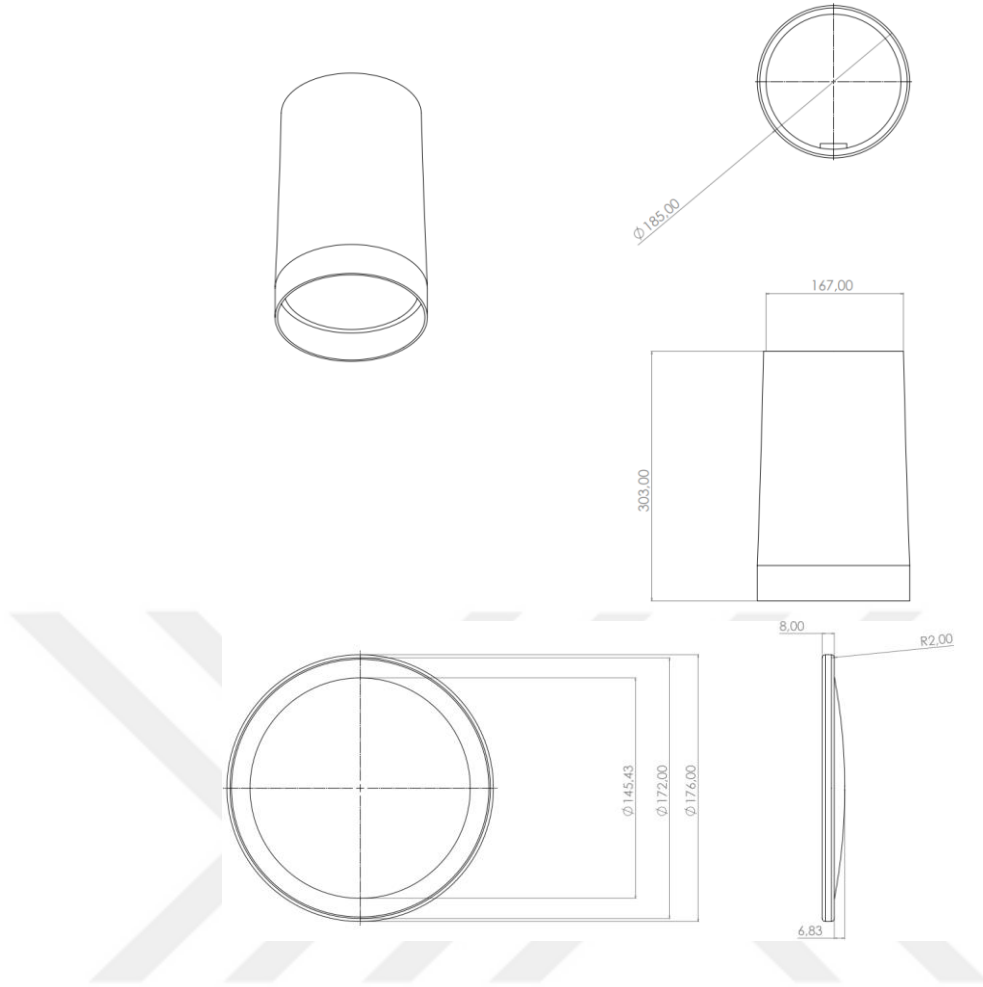
EK C. Ürünün Gerçek Boyutlu Teknik Çizimleri



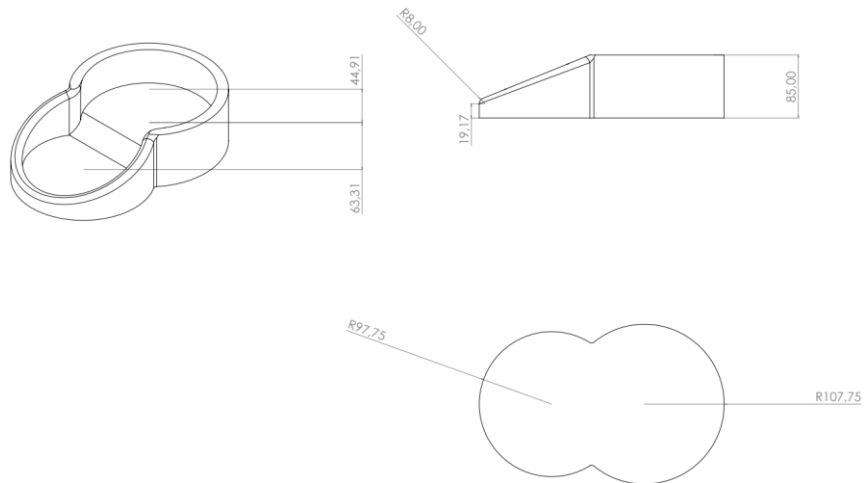
Şekil C.1. 45 derece bölmeli çark



Şekil C.2. Çark haznesi



Şekil C.3. Mama haznesi



Şekil C.4. Mama indirme haznesi

ÖZGEÇMİŞ

Adı Soyadı : Hatice Selenay ÇAKICI

Eğitim Durumu

Lise : Halit Armay Anadolu Lisesi, 2012

Lisans : İstanbul Ticaret Üniversitesi, Mühendislik Fakültesi,
Mücevherat Mühendisliği

Yüksek Lisans : İstanbul Ticaret Üniversitesi,
Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı

Yayınları

Turan M., Çakıcı S.H., 2021. Camera Controlled Smart Food Container Design. 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), Ankara, 11-13 Haziran.