

**SÜRÜCÜ DESTEK SİSTEMLİ ARAÇLAR İÇİN DİJİTAL İKİZ
TEKNOLOJİSİ VE GERÇEK ZAMANLI VERİ ENTEGRASYONU**

Abdulmuttalip DURAN

YÜKSEK LİSANS TEZİ

SAVUNMA TEKNOLOJİLERİ (DİSİPLİNLERARASI) ANA BİLİM DALI

**SİVAS BİLİM VE TEKNOLOJİ ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**

HAZİRAN 2025

ETİK BEYAN

Sivas Bilim ve Teknoloji Üniversitesi Lisansüstü Eğitim Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
 - Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
 - Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
 - Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
 - Bu tezde sunduğum çalışmanın özgün olduğunu,
- bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

.....
Abdullah DURAN
12/06/2025

SÜRÜCÜ DESTEK SİSTEMLİ ARAÇLAR İÇİN DİJİTAL İKİZ TEKNOLOJİSİ VE GERÇEK ZAMANLI VERİ ENTEGRASYONU

(Yüksek Lisans Tezi)

Abdulmuttalip DURAN

SİVAS BİLİM ve TEKNOLOJİ ÜNİVERSİTESİ

LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

Haziran 2025

ÖZET

Yapılan çalışmada, normal yol koşullarında meydana gelen engellerin kamera ve derin öğrenme tabanlı bir sistem geliştirilerek yüksek doğrulukta tespit edilmesi ve bu verilerin 5G altyapısı kullanılarak dijital ikiz platformuna aktarılması amaçlanmıştır. Çalışmada öncelikle araç için elektronik devre, PCB tasarımları, mekanik tasarım ve iskelet yapısı sıfırdan geliştirilmiştir. Daha sonra STM32 tabanlı mikrodenetleyici ve mini bilgisayar temelli bir kontrol altyapısı oluşturulmuştur. Yola düşen engelleri tespit edebilmek için YOLOv10 modeli 2.000 adet görselden oluşan veri seti ile eğitilmiştir. Eğitim sonucunda elde edilen %98-%99 seviyesine ulaşan mAP değeri, modelin nesnelere oldukça yüksek performansla tespit ettiğini göstermektedir. Tespit edilen nesnelere ve araçtan alınan sensör verileri 4G altyapısı kullanılarak bulut sunucuya, 5G altyapısı kullanılarak da uç sunucuya gönderilmiştir. Veri aktarımı sırasında 4G ve bulut sunucusu kullanımında 106 ms seviyesinde bir gecikme meydana gelirken, 5G ve uç sunucu kullanıldığı durumda bu gecikme 23 ms seviyesine kadar düşürülerek neredeyse anlık olarak verilerin aktarılması ve dijital ikizde gösterilmesi sağlanmıştır. Dijital ikizin oluşturulması için Unity oyun motoru kullanılmıştır. Ayrıca aracın nesne tespiti yaparken belirlenen parkurda ilerleyebilmesi için şerit takibine ihtiyaç duyulmuştur. Bu aşamada antrenman konileri kullanılarak şerit oluşturulmuştur. Bu konilerin dizilişleri ve ekrandaki konumlarına göre regresyon tabanlı bir açı denklemi geliştirilmiştir. Yapılan testlerde araç başarılı bir şekilde şerit takibi yapmış ve engelleri doğru bir şekilde tespit edip dijital ikizde göstererek belirtilen güzergahta seyredebilmiştir.

Bilim Kodu : 93445

Anahtar Kelimeler : Dijital ikiz, yapay zekâ, YOLOv10, 5G, Unity

Sayfa Adedi : 59

Danışman : Doç. Dr. Hacı Mehmet GÜZEY

DİGİTAL TWİN TECHNOLOGY AND REAL-TİME DATA İNTEGRATİON FOR MİLİTARY AUTONOMOUS VEHİCLES

(M. Sc. Thesis)

Abdulmuttalip DURAN

SİVAS UNIVERSITY OF SCIENCE AND TECHNOLOGY
INSTITUTE OF GRADUATE STUDIES

June 2025

ABSTRACT

The aim of the study is to detect obstacles occurring in road conditions by developing a camera and deep learning-based system and transfer this data to the digital twin platform using a 5G infrastructure. First of all, electronic circuit, PCB designs, mechanical designs and skeletal structure were developed for the vehicle in the study. A control infrastructure based on STM32-based microcontroller and minicomputer was created. In order to detect obstacles on the road, the YOLOv10 model was trained with a data set consisting of 2.000 images. The mAP value, which reached %98-%99 as a result of the training, shows that the model detects objects with very high performance. The detected objects and sensor data received from the vehicle were sent to the cloud server using the 4G infrastructure and to the end server using the 5G infrastructure. While a delay of 106 ms occurred during data transfer when 4G and cloud server were used, the delay was reduced to 23 ms when 5G and end server were used, allowing the data to be transferred and displayed on the digital twin almost instantly. Unity was used to create the digital twin. In addition, lane tracking was needed for the vehicle to proceed on the specified track while detecting objects. At this stage, a lane was created using training cones. A regression-based equation was developed according to the arrangement of these cones and their positions on the screen.

Science Code : 93445

Key Words : Digital twin, artificial intelligence, YOLOv10, 5G, Unity

Page Number : 59

Supervisor : Assoc. Prof. Dr. Hacı Mehmet GUZEY

TEŐEKKÜR

Bu tez alıőmasının planlanması, gerekleőtirilmesi ve sonuçlarının yorumlanması aőamalarında akademik, teorik ve bilimsel bilgi birikimiyle desteęini esirgemeyen danıőman hocam Do. Dr. Hacı Mehmet Güzey'e teőekkür ederim.

Araőtırma sürecindeki yoğun alıőma dönemlerinde göstermiş olduęu sabır, anlayıő ve destekleriyle motivasyonumu yüksek tutmamı saęlayan niőanlım Kübra elik'e ok teőekkür ederim.

Bu tez alıőması, Avrupa Birlięi destekli bir araőtırma projesi kapsamında yürütölmüş olup, saęlanan finansman ve araőtırma altyapısı sayesinde başarıyla tamamlanmıştır. Destekleri için ilgili program yönetimine ve emeęi geen tüm paydaőlara içten teőekkür ederim.

İÇİNDEKİLER

ÖZET	iv
ABSTRACT.....	v
TEŞEKKÜR.....	vi
ÇİZELGELERİN LİSTESİ.....	ix
ŞEKİLLERİN LİSTESİ.....	x
SİMGELER VE KISALTMALAR.....	xii
1. GİRİŞ.....	1
2. KAYNAK ARAŞTIRMASI.....	4
3. MATERYAL ve YÖNTEM.....	7
3.1 Sistemin Planlanması	7
3.2 Motor, Sürücü ve Güç Sistemi Seçimi.....	7
3.3 Elektronik Kart (PCB) Tasarımı ve Elektronik Bileşenler.....	9
3.3.1 Ana kontrol kartı	13
3.3.2 Servo motor kontrol kartı.....	14
3.3.3 Kablosuz acil durum kumanda kartı	15
3.3.4 Kablosuz kumanda alıcı kartı.....	16
3.3.5 Güç dağıtım kartı.....	17
3.4 Mikrodenetleyici Yazılım Geliştirme Süreci	18
3.5 Mekanik Tasarım ve Montaj	19
3.6 Yapay Zekâ Tabanlı Nesne Tespiti	20
3.6.1 YOLOv10.....	20
3.6.2 Gerçek zamanlı nesne tespiti.....	21

3.6.3 Derinlik kamerası entegrasyonu.....	24
3.6.4 Şerit takip sistemi.....	25
3.7 Unity Arayüzü ve Dijital İkiz.....	26
3.8 Sunucu Haberleşme Katmanı.....	28
3.9 Yazılım ve Tasarım Geliştirme Araçları.....	29
4. BULGULAR ve TARTIŞMA.....	31
4.1 Yapay Zekâ (YOLO) Eğitimi ve Test Sonuçları.....	31
4.1.1 Karışıklık matrisi analizi.....	32
4.1.2 F1-Puanı eğrileri.....	33
4.1.3 Veri dağılımları ve sınırlayıcı kutu analizleri.....	34
4.2 Dijital İkiz Platformunda Gecikme Ölçümleri.....	36
4.2.1 4G bağlantısı ve merkezi sunucu.....	36
4.2.2 5G bağlantısı ve uç sunucu (edge server).....	37
4.3 Şerit Takip Sistemi.....	37
5. SONUÇ VE ÖNERİLER.....	39
KAYNAKLAR.....	41

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 3.1. YOLOv10 eğitim hiperparametreleri	21
Çizelge 3.2. Antrenman konilerinin konum ve açı değerleri	26
Çizelge 3.3. Kullanılan program ve yazılımlar	29
Çizelge 3.4. Önceki çalışmalar ile karşılaştırma.....	36



ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 3.1. Elektrikli araçlar için motor ve aks seti sistemi.....	7
Şekil 3.2. Scooter motoru.....	8
Şekil 3.3. Scooter motor sürücü kontrol kartı.....	8
Şekil 3.4. Motor sürücü ekran bağlantı kablosunun kısa devre edilmesi.....	9
Şekil 3.5. KiCad devre şeması çizimi.....	10
Şekil 3.6. KiCad kütüphane oluşturma sayfası.....	10
Şekil 3.7. KiCad PCB çizimi için kural belirleme sayfası.....	11
Şekil 3.8. Araç kontrol kartı PCB şeması.....	11
Şekil 3.9. Devrenin üç boyutlu görünümü.....	12
Şekil 3.10. FlatCAM ile delik açma sistemi.....	13
Şekil 3.11. Araç kontrol kartı.....	14
Şekil 3.12. Servo motor kontrol kartı.....	15
Şekil 3.13. Kumanda kontrol kartı.....	16
Şekil 3.14. Kumanda alıcı kart.....	17
Şekil 3.15. Güç dağıtım kartı.....	17
Şekil 3.16. DAC modülü adres seçimi.....	18
Şekil 3.17. Aracın şasesi.....	19
Şekil 3.18. Lineer aktüatör.....	19
Şekil 3.19. Servo motor.....	20
Şekil 3.20. Veri seti görselleri.....	22
Şekil 3.21. Veri seti etiketleme.....	22
Şekil 3.22. Veri seti dağılımı.....	23

Şekil	Sayfa
Şekil 3.23. RGB kamerada nesne tespiti.....	23
Şekil 3.24. Derinlik kamerası görüntüsü.....	24
Şekil 3.25. Araç içerisindeki bilgisayarın arayüzü	27
Şekil 3.26. RUTX50 modem	27
Şekil 4.1. Eğitimin sonuç grafikleri	31
Şekil 4.2. Karışıklık matrisi	33
Şekil 4.3. F1 puan grafiği.....	34
Şekil 4.4. Etiket dağılım grafikleri.....	35
Şekil 4.5. Dijital ikiz ekran fotoğrafı	36

SİMGELER VE KISALTMALAR

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

Simgeler

V	Voltaj (Volt)
A	Akım (Amper)
Ah	Amper-saat (Batarya Kapasitesi)
Ms	Milisanıye
Cm	Santimetre
%	Yüzde
°	Derece

Kısaltmalar

YOLO	Yapay Zekâ Algoritması (You Only Look Once)
PV-YOLO	Kısmi Görüş YOLO Modeli (Partial Vision YOLO)
mAP	Ortalama Doğruluk Oranı (Mean Average Precision)
F1	F1 Skoru – Harmonik Ortalama (F1 Score)
AI	Yapay Zekâ (Artificial Intelligence)
LİDAR	Mesafe Algılayıcı Sensör (Light Detect and Ranging)
RGB	Kırmızı-Yeşil-Mavi Renk Modeli (Red Green Blue)
3B	Üç Boyutlu (Three-Dimensional)
4G	4. Nesil Mobil Ağ (4th Generation Mobile Network)
5G	5. Nesil Mobil Ağ (5th Generation Mobile Network)
IoV	Araçların İnterneti (Internet of Vehicles)
HTTP	Köprü Metni Aktarım Protokolü
HTTPS	Güvenli Köprü Metni Aktarım Protokolü
Vb.	Ve benzeri
VPN	Sanal Özel Ağ (Virtual Private Network)
JSON	Veri Listeleme Formatı (JavaScript Object Notation)
AWS	Amazon Web Hizmetleri (Amazon Web Services)
PCB	Baskı Devre Kartı (Printed Circuit Board)
STM32	STMicroelectronics 32-bit Mikrodenetleyici Ailesi

UART	Evrensel Asenkron Alıcı-Verici (Universal Asynchronous Receiver Transmitter)
USB	Evrensel Seri Veriyolu (Universal Serial Bus)
TTL	Transistör-Transistör Mantığı (Transistor-Transistor Logic)
USB-TTL	USB-den TTL'-ye Dönüştürücü (USB-to-TTL Converter)
İC	Entegre Devreler Arası İletişim (Inter-Integrated Circuit)
EEPROM	Elektrikle Silinebilir Programlanabilir Salt Okunur Bellek (Electrically Erasable Programmable ROM)
DAC	Dijital'den Analog'a Çevirici (Digital-to-Analog Converter)
PWM	Darbe Genişlik Modülasyonu (Pulse Width Modulation)
RC (filtre)	Direnç-Kapasitör Filtresi (Resistor-Capacitor)
OLED	Doğal Işık Yayan Diyot (Organic Light Emitting Diode)
GPS	Küresel Konumlama Sistemi (Global Positioning System)
NMEA-TCP	NMEA tabanlı TCP Protokolü (National Marine Electronics Association – TCP)
IDE	Tümleşik Geliştirme Ortamı (Integrated Development Environment)
RAM	Rastgele Erişimli Bellek (Random Access Memory)
PC	Kişisel Bilgisayar (Personal Computer)

1. GİRİŞ

Günümüzde otonom araç teknolojileri hızla gelişmekte ve bu araçların güvenli bir şekilde seyretmesini sağlayacak güvenlik sistemlerine olan ihtiyaç artmaktadır [1]. Sürücü destek sistemli araçlar için yola düşen yükler, araç lastikleri, trafik ekipmanları, ağaç kütükleri vb. çeşitli engeller büyük tehlike oluşturmaktadır. Bu engellerin tespit edilmesi trafik güvenliği açısından kritik hale gelmiştir. Bu tür engeller trafik akışını bozup sürücülerin ve yolcuların olumsuz etkilenmesine sebep olabilmektedir.

Yola düşen engellerin tespiti üzerine yapılan çalışmalar daha çok radar, LİDAR veya kamera gibi sensör sistemlerinin karmaşık entegrasyonu üzerine dayanmaktadır [2]. Buna karşın maliyet ve kullanım kolaylığı açısından optik kamera ve derinlik kamerası ile çalışan, derin öğrenme tabanlı nesne tespit yöntemi birçok proje açısından daha ulaşılabilir bir seçenek sunmaktadır. Nesne tespiti konusunda YOLO (You Only Look Once) büyük ilgi görmüştür ve gerçek zamanlı veri işleme konusunda oldukça başarılı olmuştur [3]. Özellikle YOLOv10 sürümünde hız ve doğruluk alanında yapılan geliştirmeler daha kısa zamanda daha çok hedef yakalayabilme kabiliyetini artırmıştır. Bu sebeplerden ötürü yola düşen çeşitli engellerin tespit edilmesi konusunda YOLO tabanlı bir algoritma kullanmak, düşük seviye donanımlara sahip bilgisayarlarda dahi uygulanabilir bir seçenek olarak değerlendirilmektedir.

Optik kamera ile yakalanan görüntülerin yapay zekâ tabanlı modellerde işlenerek nesne tespitinin yapılması tek başına güvenli bir yöntem değildir. Buna ek olarak engellerin tespit edildikten sonra anlık olarak araç ile engel arasındaki mesafenin hassas olarak ölçülmesi ve dünya üzerindeki konumunun hassas olarak tespit edilmesi gerekmektedir. Bu aşamada optik kameraya ek olarak optik kamera görüntüsü ile senkronize şekilde çalışan derinlik kamerası kullanmak iyi bir yaklaşım olmaktadır.

Tespit edilen engellerin ve çevre koşullarının uzaktan izlenmesi önem taşımaktadır. Bu noktada dijital ikiz teknolojisi devreye girmektedir. Dijital ikiz teknolojisi fiziksel bir sistemin gerçek zamanlı olarak bir kopyasını sanal ortamda oluşturmayı kapsayan bir yaklaşımdır. Dijital ikiz temel olarak sensör verisi, konum ve tespit edilen engellerin verisi gibi gerçek dünya verilerini toplayıp sanal modelde görselleştirmeyi amaçlar. Bu teknoloji

uzaktan gözlem ve müdahaleyi kolaylaştırmakla beraber otonom araçların birbiriyle iletişim kurmasını ve olası engellere önceden önlem almasına da imkân tanımaktadır.

Projenin çıkış noktası, yola düşen çeşitli engellerin güvenlik riskini en aza indirecek şekilde ve bu tespitleri yaparken karmaşık ve yüksek maliyetli donanım altyapılarına ihtiyaç duymayan bir yönteme olan ihtiyaçtır. Araçtaki optik kamera üzerinden alınan görüntüler üzerinde YOLO algoritması ile yapılan nesne tespiti, derinlik kamerasından alınan uzaklık verisi ve araçta bulunan diğer çevresel sensör (GPS, rüzgâr sensörü vb.) verileri, 4G ile sunucuya veya 5G ile uç sunucuya aktarılmaktadır. Sunuculardan alınan veriler ile görselleştirme yapılacak bilgisayar üzerinde dijital ikiz oluşturulmakta ve anlık olarak da araçtan gelen veriler ile senkronize şekilde sanal aracı ve sanal çevre koşullarını oluşturmaktadır. Bu sayede uzaktaki bir operatör aracın, yolun ve çevre koşullarının incelemesini yapabilmektedir; ayrıca tespitin yapıldığı araçtan sonra o yolda seyredecek diğer otonom araçlar engeller konusunda erken uyarı alarak yolun durumuna uygun yeni senaryolarda seyredebilmektedir.

Proje kapsamında nesne tespitine yönelik gerekli verileri tek bir kaynaktan elde etmek ve bunu güçlü derin öğrenme modeli ile yorumlamak hedeflenmektedir. Derin öğrenme modellerinin yola düşen bir nesneyi normal yoldan ayırt edebilmesi eğitimi için kullanılan veri setinin güçlü olmasına bağlıdır. Veri setinin farklı ortamlarda, farklı aydınlatma koşullarında, günün farklı dilimlerinde oluşturulması derin öğrenme modelinin nesneyi doğru tespit edebilme performansını artırmaktadır.

Önerilen yaklaşımda veri setinin oluşturulması için belirli sayıda nesne kullanılmıştır. Yolda sıkça rastlanabilecek araç lastikleri, trafik dubası, ağaç kütükleri vb. nesnelerin görselleri kullanılmıştır. Projede sadece normal asfalt yola odaklanılmamış, farklı yol ve zeminlerde de tespit yapabilecek bir derin öğrenme modeli eğitilmiştir. Ayrıca dijital ikiz teknolojisinde kullanılan 5G altyapısının sunduğu uç sunucu haberleşme teknolojisi çok yüksek hızlarda düşük gecikme ile veri aktarma imkânı sağlamaktadır.

Bu tez, yapay zekâ algoritmaları ile gerçek zamanlı nesne tespitini, 5G altyapısındaki uç bilişim teknolojisi ile veri aktarımını, dijital ikiz teknolojisi ile anlık verilerin sanal ortamda gösterilmesini amaçlamaktadır. Tez kapsamında yapılan çalışma iki ana ekseninde ilerleyecektir. İlk olarak yola düşen çeşitli engelleri yüksek doğrulukta tespit edecek yapay

zekâ modelinin eğitilmesi ve bunun için gerekli olan veri setinin hazırlanması, ön işleme, etiketleme ve eğitim süreçlerinin tamamlanması yapılacak. Daha sonra ise bu veriler ile araca ait diğer sensör verilerinin sunucu üzerinden dijital ikizin oluşturulacağı bilgisayara aktarılıp geliştirilecek program ile anlık olarak gerçek dünyanın dijital ikizinin oluşturulması gerçekleştirilecek.



2. KAYNAK ARAŞTIRMASI

Otonom araçlar ve bu araçların dijital ikizlerinin oluşturulmasına yönelik son yıllarda çeşitli alanlarda çalışmalar yapılmaktadır. Dijital ikiz kavramı, fiziksel bir sistemin anlık olarak sanal bir ortama yansıtılmasını ifade eder. Bu bölümde ulusal ve uluslararası düzeyde yayımlanmış makaleler doğrultusunda dijital ikiz ve nesne tespiti alanındaki çalışmalar incelenmiştir.

Dijital ikiz yaklaşımları, başlangıçta endüstriyel tesisler ya da üretim hatlarında yaygınlaşmış olsalar da zamanla otonom araçlar ve akıllı ulaşım sistemleri alanlarında da kullanılmaya başlanmıştır [4]. Dijital ikiz oluşturma konusunda Unity oyun motorunun sağladığı üç boyutlu (3B) görselleştirme avantajı, gerçek zamanlı veri akışını sanal ortamla birleştirme konusunda önemli kolaylıklar sunmaktadır. Wang ve arkadaşları 2021 yılında Unity oyun motorunu kullanarak bağlı ve otonom araçların (connected and automated vehicles) dijital ikiz simülasyonunu gerçekleştirmiştir. Bu sayede araçların sensör verileriyle sanal modeli eş zamanlı güncellenmesini sağlamıştır [5]. Benzer şekilde, eğitim odaklı bir uygulamada Balla ve arkadaşları 2023 yılında üretim hattının dijital ikizini oluşturmuştur [4]. Bu şekilde kullanıcıların sistem parametrelerini anlık olarak gözlemleyebildikleri bir dijital ikiz sistemi altyapısı kurmuşlardır.

Dijital ikiz teknolojisinin ulaşım ve 5G destekli araç ağlarına uyarlanması, gerçek zamanlı veri analizi ve trafik yönetimi gibi konularda önemli katkılar sağlar. Hu ve arkadaşları 2021 yılında 5G destekli araçlarda trafik verilerinin öngörülmesinde dijital ikiz teknolojisinden yararlanmış ve elde edilen bulgular, bu yaklaşımın düşük gecikme ve yüksek doğrulukta tahminler sunduğunu göstermiştir [6]. Benzer şekilde Schwarz ve Wang, 2022 yılında dijital ikiz teknolojisinin otonom araçlar üzerinde kullanımını incelemiş; sensör verilerinin anlık olarak işlenmesiyle sanal kopyanın operasyonel karar süreçlerine katkı sağladığı vurgulanmıştır [7]. Özellikle enerji tüketimi ve analizi konusunda Zhang ve arkadaşları 2021 yılında elektrikli araçlarda dijital ikiz teknolojisini kullanarak enerji tüketim modellerinin doğruluğunu artırma konusunda IoV ortamında oluşan veri akışının anlık simülasyonla beslendiğinde daha isabetli sonuçlar verdiğini göstermiştir [8]. Zhang ve arkadaşları 2019 yılında elektrikli araç şarj altyapılarının davranış modellemesine odaklanarak internet tabanlı araç haberleşme ağlarında dijital ikiz

teknolojisinin güvenli bir izleme sistemi sunabileceğini ileri sürmüştür [9].

Geliştirilen dijital ikiz uygulamaları araç hareketliliğinin simülasyonunun yanı sıra süreç güvenliğini doğrudan etkileyen nesne veya engel tespiti mekanizmalarını da kapsar. Bu kapsamda, YOLO (You Only Look Once) ailesi, gerçek zamanlı nesne tespiti konusunda öne çıkan algoritmalarından biridir. Son zamanlarda yapılan çalışmalar, YOLO tabanlı sistemlerin otonom araçlarda hem çevresel engellerin algılanmasında hem de trafik işaretleri gibi tabelaların tanınmasını mümkün kılarak sürücü destek sistemlerinde önemli görevler üstlendiğini göstermektedir. Zaghari ve arkadaşları 2021 yılında YOLO tabanlı bir “non-maximum suppression fuzzy” algoritması önererek engel tespitini geliştirmiştir. Bu yöntemin özellikle trafiğin yoğun olduğu ortamlar gibi karmaşık sahnelerde daha doğru sonuçlar ürettiği gözlemlenmiştir [10]. Huu ve arkadaşları 2022 yılında şerit algılama ile engel tespiti konularını birlikte değerlendirerek otonom sürüş uygulamalarında YOLO’yu gelişmiş ağ protokolleriyle bütünleşmiş hale getirmiştir [11].

Nesne tespiti alanında YOLO konusunda çok sayıda inceleme, iyileştirme ve geliştirme çalışması bulunur. Rekna ve arkadaşları 2020 yılında YOLO sürümlerinin gelişimini özetleyerek otonom araçlardaki kullanım alanlarını tartışmıştır [12]. Alahdal ve arkadaşları ise 2024 yılında gerçek zamanlı nesne tespitini optimize eden ek modüllerle YOLO performansını iyileştirmeye yönelik yöntemler aktarmıştır [13]. Aynı şekilde Sukumar ve arkadaşları 2024 yılında engel tespiti ve sınıflandırma konularını ele alarak otonom araçların sürüş güvenliğini yükseltmeyi hedeflemiştir [14]. Trafik levhası tanıma alanında Flores-Calero ve arkadaşları 2024 yılında YOLO tabanlı yaklaşımların derin öğrenme modelleri arasındaki başarımını vurgulamıştır [15]. Fanthony ve arkadaşları ise 2021 yılında elektrikli otonom araçlarda çevre algısı için benzer sonuçlar elde etmiştir [16]. Gheorghe ve arkadaşları 2024 yılında yaptıkları çalışmalarında, bilgisayarla görme yöntemlerinin gerçek zamanlı nesne tespiti ve hedef takibine uyarlanabilirliğini incelemiştir. Bu bağlamda YOLO algoritmasının işlem hızını ve doğruluk düzeyini öne çıkarmışlardır [17]. Liu ve arkadaşları 2025 yılında PV-YOLO olarak adlandırdıkları bir model ile yaya ve araç tespitinde performans açısından donanım dostu bir yöntem sunmuştur [18]. Martinelli ve arkadaşları ise gerçek zamanlı nesne tespiti sisteminde tespit edilen araçlarla mesafenin tespitine yönelik bir yöntem geliştirmiştir [19].

Bu çalışmalar incelendiğinde, YOLO algoritmasının gerçek zamanlı nesne tespiti alanında gösterdiği performansı sayesinde yalnızca cadde veya yollarda değil aynı zamanda endüstriyel tesislerde de kullanımının mümkün olduğu göze çarpmaktadır. İncelenen kaynaklardaki çoğu araştırmacı nesne tespiti konusunda oldukça başarılı sonuçlar elde etmesine karşın bu sonuçların harici bir dijital ikiz platformunda eş zamanlı olarak modelinin oluşturulması konusunu henüz detaylı bir şekilde ele almamıştır. Unity başta olmak üzere çeşitli oyun motorları aracılığıyla oluşturulan dijital ikiz sistemleri, gerçek sistemin 3B olarak görselleştirilmesini sağlarken, geliştirilen arka uç yazılımları da araç verilerinin sunucuya ve oradan da sanal ortama aktarılmasını sağlamaktadır. Bu sayede aracın sürüş dinamikleri ve çevre algısı tek bir platformda birleştirilerek kullanıcıya daha kapsamlı bir gözlem ve analiz imkânı sağlayabilmektedir [20].

Yapılan çalışmalara bakıldığında bir yandan dijital ikiz teknolojisinin akıllı ulaşım ve otonom araç teknolojileri ile birleşmesiyle ortaya yeni teknoloji ve buna bağlı çeşitli fırsatlar ve sorunlar çıkmaktadır. Aynı zamanda YOLO algoritması gibi derin öğrenme ve yapay zekâ algoritmalarının sahada uygulanması tartışılmaktadır. Mevcut çalışmalara bakıldığında çoğunlukla trafik veri tahmini, enerji yönetimi veya trafik levhası tanıma gibi konular ele alınmaktadır [8, 21]. Fakat yola düşen engellerin anlık olarak tespiti ve bu bilgilerin sanal bir ortamda gösterilmesine ilişkin çalışmalar detaylı yapılmamıştır. Bu tezde ise, güçlü bir nesne tespiti altyapısını YOLO dijital ikiz mekanizmasıyla bütünleştirerek aracın konum, hız ve nesne tespiti gibi bilgileri gerçek zamanlı olarak sunucuya iletmesi ve sunucudan alınan bu verilerin dijital ikizde sanal ortam oluşturularak görselleştirilmesi amaçlanmıştır.

3. MATERYAL ve YÖNTEM

Bu bölümde, tezde kullanılan malzeme, bileşenler, programlar, yazılım mimarileri ve uygulanan yöntemler ayrıntılı biçimde tanımlanmıştır. Başlangıç olarak projede kullanılan aracın mekanik ve elektronik altyapısının nasıl oluşturulduğu, kullanılan yazılım ve donanımların seçim süreci açıklanmıştır. Ardından ise yapay zekâ tabanlı nesne tespiti, şerit takibi ve dijital ikiz sisteminin geliştirilme süreçleri açıklanmıştır.

3.1 Sistemin Planlanması

Tez çalışmasının başlangıcında ilk aşama olarak, yapılacak prototip otonom elektrikli aracın gereksinimleri belirlenmiştir. Kullanılacak motor türünün belirlenmesi, motora uygun motor sürücüsünün belirlenmesi, şasi malzemesinin belirlenmesi, dış gövde malzemesinin belirlenmesi, kontrol mekanizmasının belirlenmesi, kullanılacak sensörlerin seçilmesi, kullanılacak bilgisayar ve işletim sisteminin belirlenmesi şeklinde bir yol izlenmiştir.

3.2 Motor, Sürücü ve Güç Sistemi Seçimi

Bu aşamada ilk olarak, araç hareket sistemi için Şekil 3.1’de gösterilen 3 veya 4 tekerlekli küçük elektrikli araçlarda kullanılan elektrikli motor ve aks seti düşünülmüştür; fakat boyut, tork ve tedarik sorunları sebebiyle vazgeçilmiştir. Sonrasında Şekil 3.2’de gösterilen üç fazlı scooter motorları kullanım açısından daha uygun bulunmuş, sürücüsü olarak ise Şekil 3.3’te gösterilen scooter da kullanılan motor kontrol ünitesi (kontrol beyni) tercih edilmiştir.



Şekil 3.1. Elektrikli araçlar için motor ve aks seti sistemi.

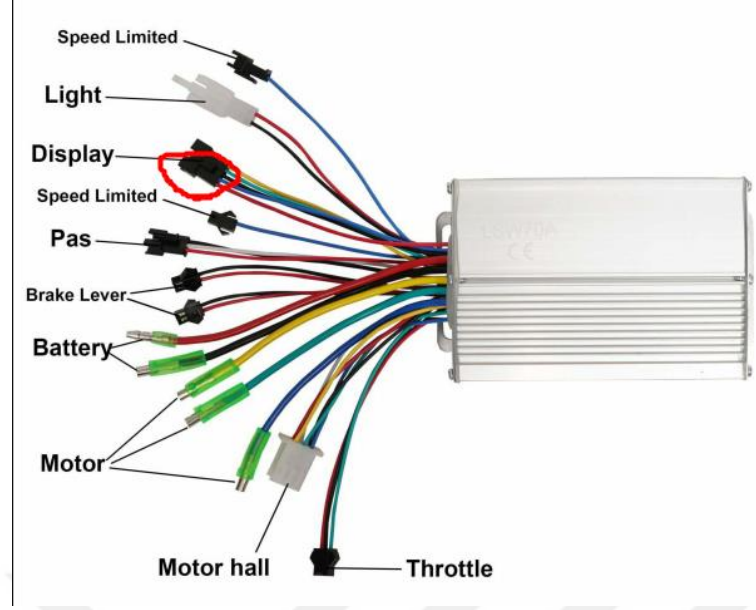


Şekil 3.2. Scooter motoru



Şekil 3.3. Scooter motor sürücü kontrol kartı

Denemeler sırasında, motor kontrol sürücüsünün kontak kilidi benzeri bir koruma devresi olduğu görülmüş, çözüm olarak ise sürücünün ekran bağlantısındaki ilk iki pin kısa devre edilerek güvenlik kilidi devre dışı bırakılmıştır (Şekil 3.4). Bu sayede gaz sinyal hattına manuel olarak potansiyometre bağlanarak motor sürücünün kontrollü bir şekilde motoru farklı hızlarda hareket ettirmesi test edilmiştir. Daha sonra bu kontrol sistemi potansiyometre yerine STM32'den sinyal alan bir DAC sinyal üreticisi ile sağlanmıştır.

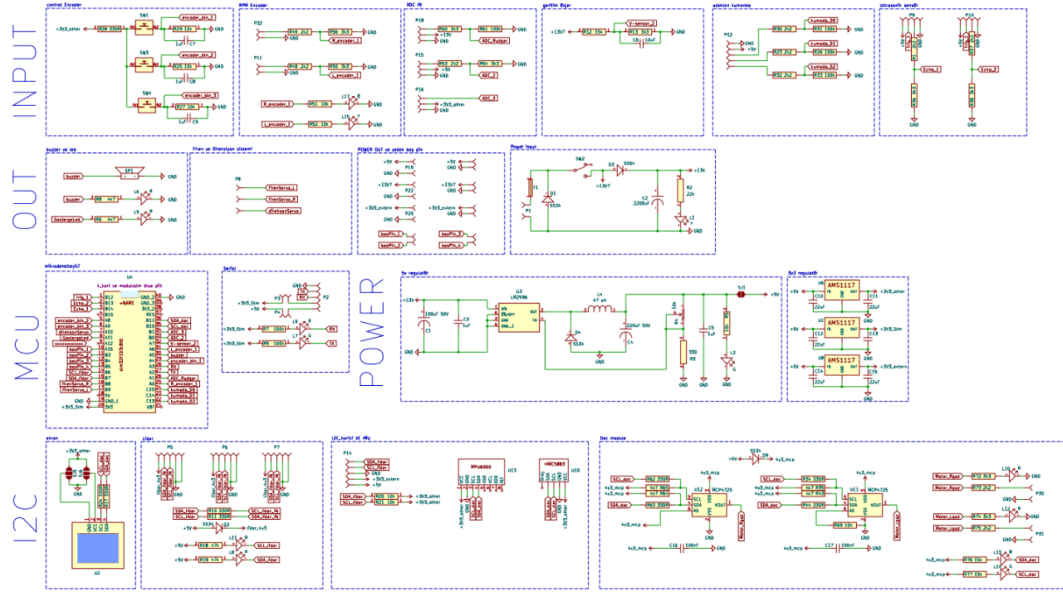


Şekil 3.4. Motor sürücü ekran bağlantı kablosunun kısa devre edilmesi.

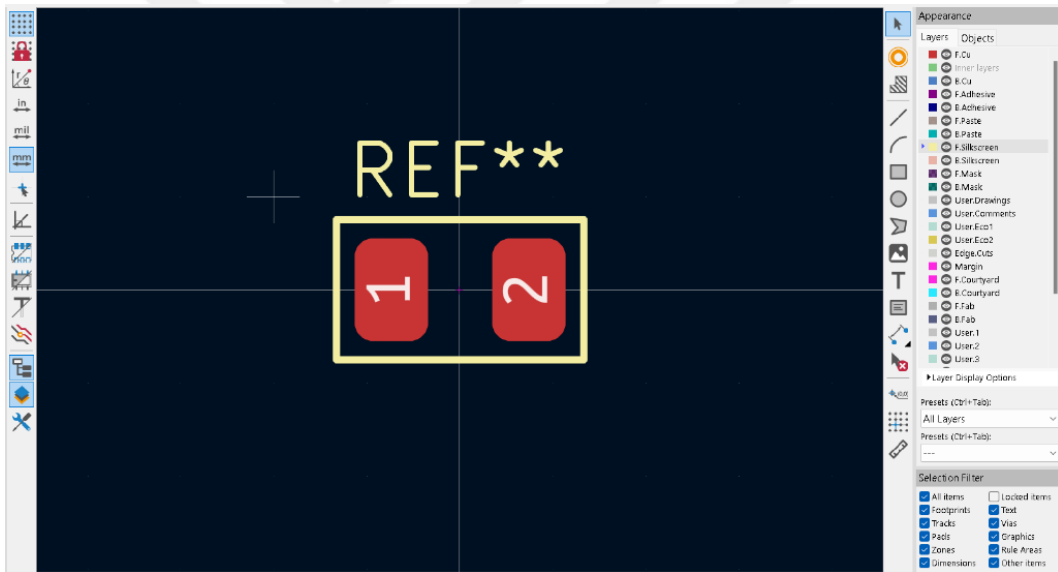
Güç kaynağı olarak motorlara ve diğer elektronik bileşenlere enerji sağlamak için 12 V 14 Ah jel aküler kullanılmıştır. Motorlara enerji sağlamak için 3 adet jel akü seri bağlanarak 36 V 14 Ah akü seti oluşturulmuştur. Diğer elektronik bileşenlere güç sağlamak için ise 2 adet jel akü paralel bağlanarak 12 V 28 Ah akü seti oluşturulmuştur. İki enerji sisteminin eksi uçları birleştirilerek haberleşme sistemlerinin motor sürücüleri ile kontrol sistemi arasında veri alışverişi yapabilmesi sağlanmıştır.

3.3 Elektronik Kart (PCB) Tasarımı ve Elektronik Bileşenler

Aracın kontrolünü sağlayacak tüm elektronik kartlar, KiCad programında tasarlanmıştır. Başlangıç olarak tasarlanacak devrelerin kâğıt üzerinde taslağı oluşturulmuştur. Sonraki aşamada tasarlanacak devrelerin şemaları KiCad programında çizilmiştir (Şekil 3.5). Bu aşamada devre tasarımı yapılırken gereken elektronik bileşenlerin kütüphaneleri kullanılmıştır. Kullanılacak kütüphaneler program içerisindeki araçlar ile oluşturulabilmektedir (Şekil 3.6). Ayrıca SnapEDA web sitesinden de temin edilebilmektedir. Devre şeması oluşturulduktan sonra ilgili devreler uzay montaj olarak breadboard üzerinde kurulmuştur ve testleri yapılmıştır. Testler sonrasında olası hata çıkması durumunda devre şeması düzeltilmiştir.

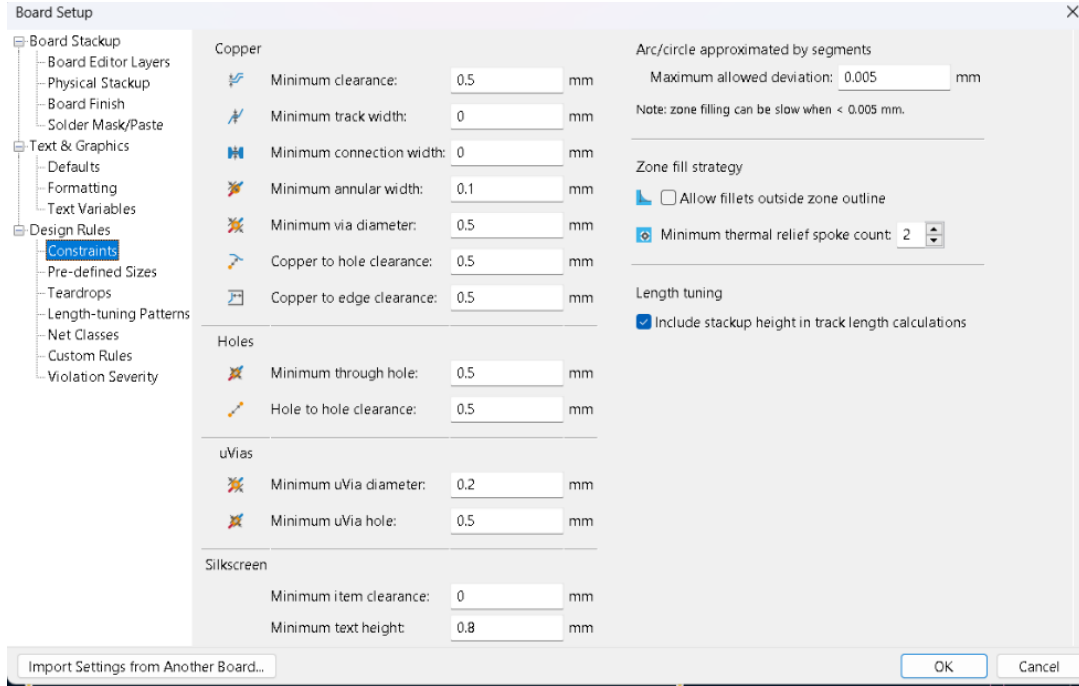


Şekil 3.5. KiCad devre şeması çizimi

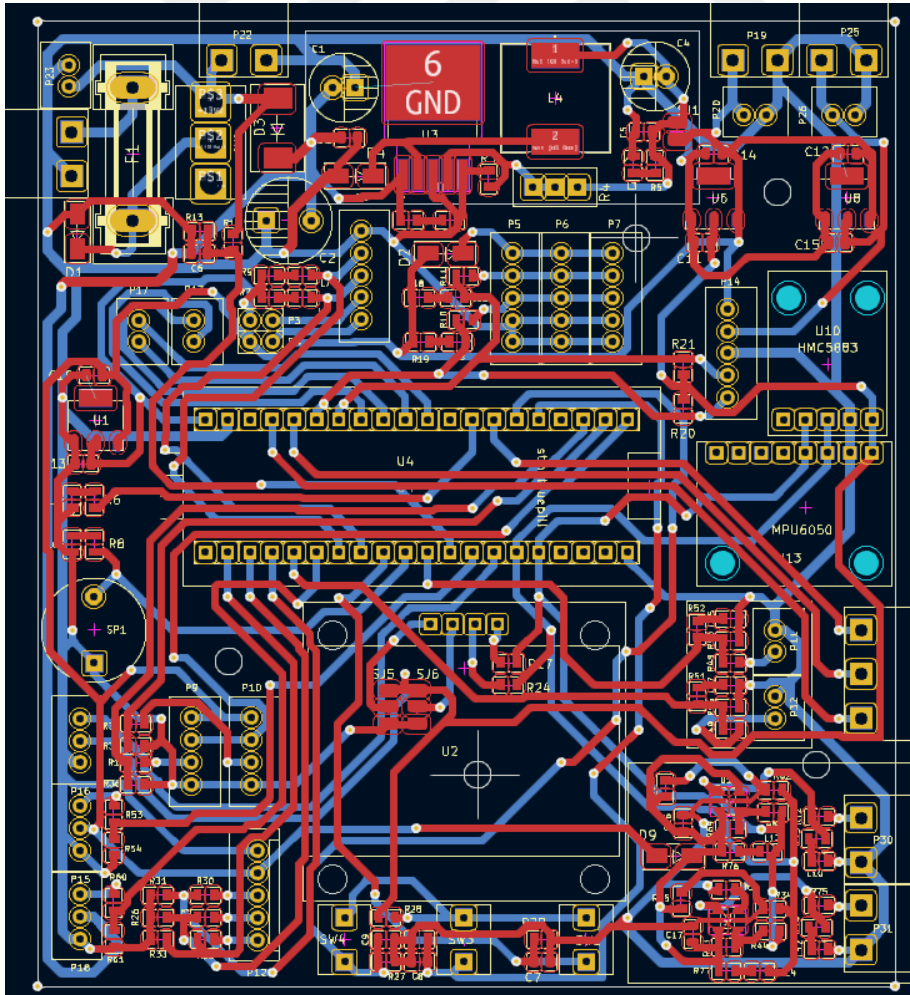


Şekil 3.6. KiCad kütüphane oluşturma sayfası

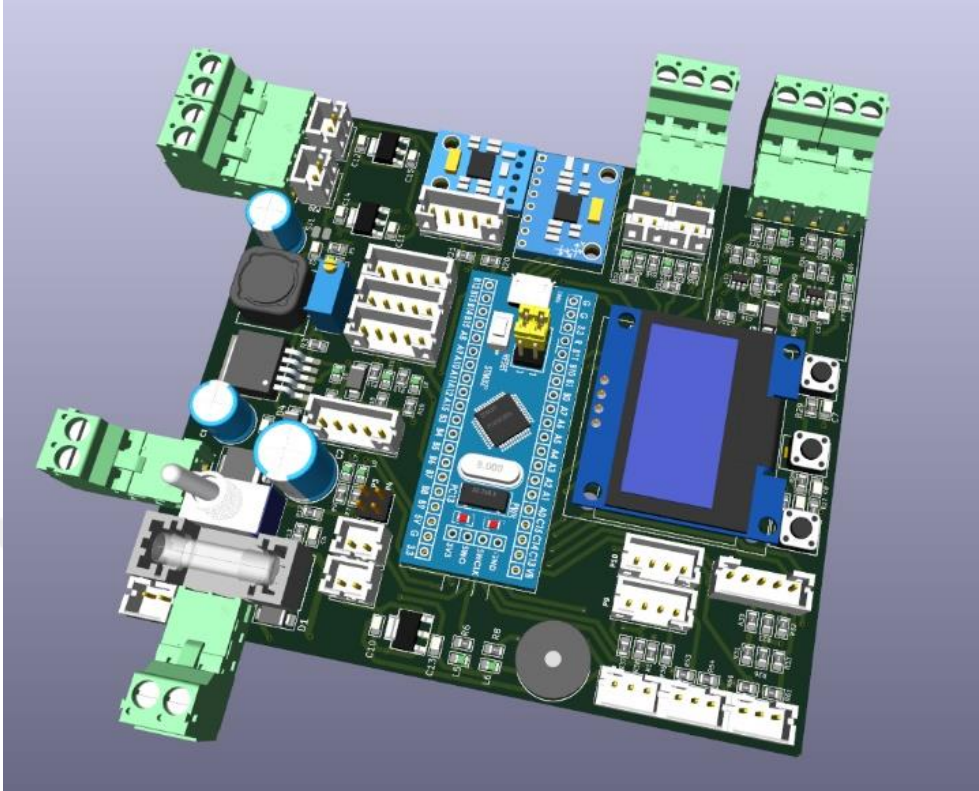
Sonraki aşamada, üretilecek devre kartlarının tasarımları yapılmıştır. Bu aşamada ilk olarak Şekil 3.7’de gösterildiği gibi tasarlanacak devre için minimum yol kalınlığı, yollar arası minimum mesafe, minimum delik çapı vb. gerekli PCB kuralları tanımlanmıştır. Daha sonra kullanılacak devre elemanları uygun bölgelere yerleştirilmiştir. Devre elemanları konumlandırıldıktan sonra elektrik ve sinyal taşıyacak olan akım yolları çizilmiştir (Şekil 3.8). Tasarlanan devre kartının üç boyutlu görünümü Şekil 3.9’da gösterilmiştir.



Şekil 3.7. KiCad PCB çizimi için kural belirleme sayfası

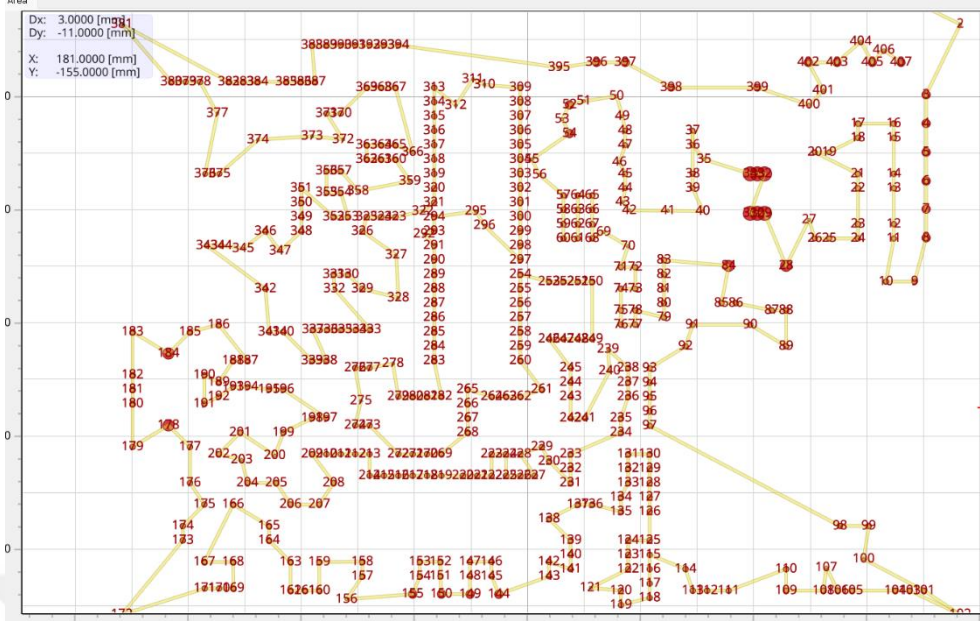


Şekil 3.8. Araç kontrol kartı PCB şeması



Şekil 3.9. Devrenin üç boyutlu görünümü

Devre kartlarının şematik ve PCB tasarımları tamamlandıktan sonraki aşamada kartların üretimi yapılmıştır. Bu aşamada ilk olarak KiCad programında devre kartlarının Gerber dosyaları oluşturulmuştur. Bu dosyalar FlatCAM programına aktarılmıştır. FlatCAM programı devre kartını oluşturacak bakır plaketlerin delinmesi için gerekli olan G-code'un oluşturulması için kullanılmıştır (Şekil 3.10). Devre kartlarının via ve bileşenlerin takılacağı pinler FlatCAM'de oluşturulan G-code'u kullanılarak CNC makinesinde delinmiştir.



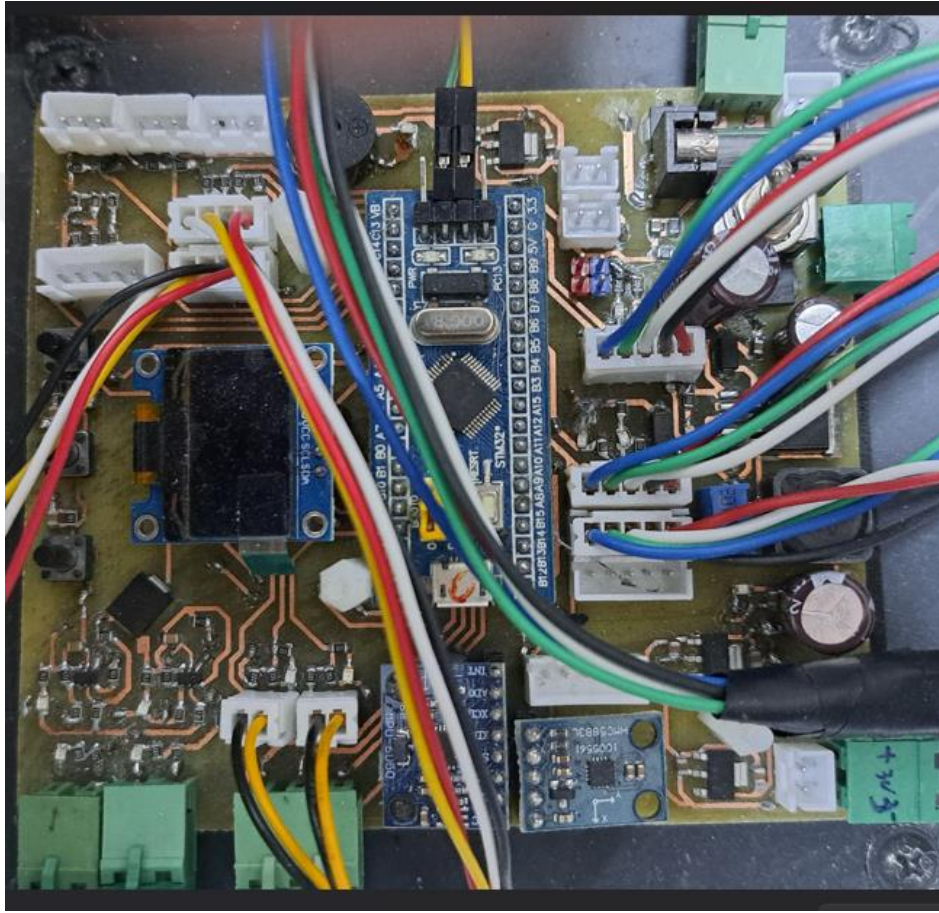
Şekil 3.10. FlatCAM ile delik açma sistemi

Sonraki aşamada iz yollarının oluşturulması için bakır plakette maskleme işlemi ve asitleme işlemi yapılmıştır. Bakır plaketten PCB levha üretildikten sonra gerekli devre elemanları PCB üzerine montajları yapıp lehimlenerek devre kartı tamamlanmıştır. Toplamda 5 farklı devre kartı oluşturulmuştur.

3.3.1 Ana kontrol kartı

Sistemde bilgisayardan gelen sinyalleri işleyip motor sürücülere hareket sinyali veren, çeşitli çevre birimlerindeki sensörlerden gelen verileri işleyip bilgisayara gönderen, aracın hareketi, veri işleme vb. faaliyetleri gerçekleştirebilmesi için STM32 ailesinden STM32F103C8T6 mikrodenetleyicisi kullanılarak ana kontrol kartı geliştirilmiştir (Şekil 3.11). Bu kart CH340 entegrasyonu içeren USB-TTL dönüştürücüsü aracılığıyla, UART haberleşme protokolünü kullanarak mini bilgisayar ile çift taraflı veri aktarımı yapabilmektedir. Scooter motorlarını sürmek için kullanılan motor sürücüsü 2-4 V arasında uygulanan analog sinyale göre hareket sağlamaktadır. Fakat STM32F103C8T6 kartının analog sinyal üretmek için kullanabileceği dahili bir DAC birimi bulunmamaktadır. Bundan ötürü harici bir DAC modülüne ihtiyaç doğmaktadır. Bu aşamada başlangıçta STM32'den alınan PWM sinyalleri belirli RC filtrelerden geçirerek stabil bir analog sinyal elde edilmesi denenmiştir. Bu sistem temel düzeyde çalışmasına karşın her zaman istenilen kararlılıkta çalışması başarısız olmuştur. Bunun üzerine STM32 ile kolay bağlantı

kurabilmesi, güç tüketiminin düşük olması, giriş ve çıkış gerilimlerinin istenilen aralığa uygun olması sebebiyle, I²C haberleşme protokolü ile çalışan MCP4725 DAC modülü tercih edilmiştir. Anakart üzerine MPU6050 eğim ve ivme ölçer, QMC5883 pusula gibi sensörler entegre edilmiştir; temel düzey log çıktıları alınabilmesi için OLED ekran eklenmiştir. Karta gelen elektrik enerjisini uygun seviyelere dönüştürebilmek için ise LM2596 ve AMS1117 gerilim regülatörleri eklenmiştir. Bu sayede kart üzerinde 5 V ve 3,3 V gerilim elde edilmiştir.

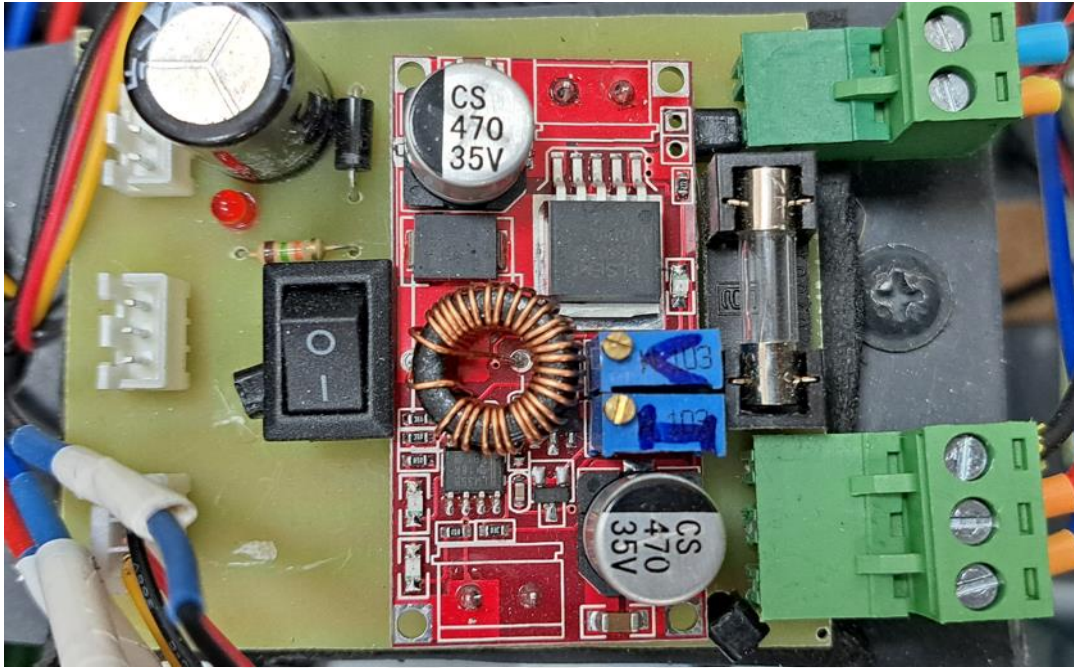


Şekil 3.11. Araç kontrol kartı

3.3.2 Servo motor kontrol kartı

Geliştirilen araçta direksiyon ve frenlerin kontrolü için 25 kg/cm gücüne sahip servo motor kullanılmıştır. Bu servo motorlar çalışırken yüksek güce ihtiyaç duymaktadır. Bu gücü ana kontrol kartından almak hem ana kontrol kartının ısınmasına ve elektrik dalgalanmalarından kaynaklanan gürültülere sebep olabilmektedir. Bu sebepten ötürü servo motorlara güç vermek ve ana kontrol kartından gelen PWM sinyallerini servo

motorlara iletmek için servo kontrol kartı geliştirilmiştir (Şekil 3.12). Karta güç vermek için 5 A çıkış akımına sahip XL4015 gerilim regülatörü kullanılmıştır.



Şekil 3.12. Servo motor kontrol kartı

3.3.3 Kablosuz acil durum kumanda kartı

Aracın kontrolden çıkması durumunda uzaktan müdahale edebilmesi için Şekil 3.13'te gösterilen uzaktan kumanda sistemine ihtiyaç duyulmuştur. Bu aşamada kablosuz haberleşme modülü olarak kullanım kolaylığı ve yüksek menzili sebebiyle nRF24L01 kablosuz haberleşme modülü tercih edilmiştir. Yapılan testlerde 500 metrenin üzerinde veri aktarımı başarıyla gerçekleşmiştir. Mikrodenetleyici olarak ise ileride internete bağlanabilme ihtiyacı oluşabilmesine karşın internete bağlanabilme özelliği olan ESP8266 NodeMCU LoLin mikrodenetleyicisi kullanılmıştır. Bu sistem yüksek menzilde, düşük maliyetli kablosuz kontrol imkânı sağlamaktadır.



Şekil 3.13. Kumanda kontrol kartı

3.3.4 Kablosuz kumanda alıcı kartı

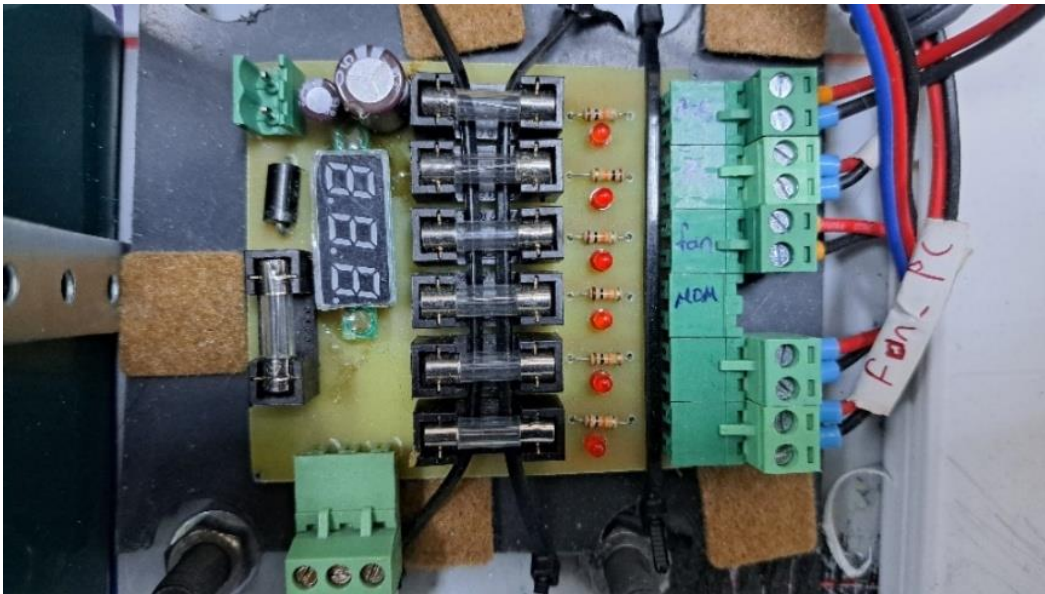
Kablosuz acil durum kumandasından gönderilen sinyallerin yakalanması noktasında bir adet de sinyal alıcı sisteme ihtiyaç duyulmaktadır. Bu aşamada gürültü oluşumu ve sinyal kaybını en aza indirmek için, ana kontrol kartından bağımsız bir kontrol ünitesine sahip alıcı kart tasarlanmıştır (Şekil 3.14). Bu kart içerisinde de ileride internete ihtiyaç olabileceğinden ötürü ESP8266 mikrodenetleyicisi kullanılmıştır.



Şekil 3.14. Kumanda alıcı kart

3.3.5 Güç dağıtım kartı

Geliştirilen araçta çevre birimleri farklı akımlarda elektrik enerjisine ihtiyaç duyduğu için her bir çevre biriminin güvenliği için kullanılacak sigorta değerleri de farklı olmaktadır. Bu aşamada hem güvenliği sağlamak hem de kablo karışıklığının önüne geçmek için aküden alınan elektrik enerjisini mini bilgisayar, fanlar, modem vb. çevre birimlerine dağıtacak bir karta ihtiyaç duyulmuştur (Şekil 3.15). Kart üzerine koruma sigortaları, güç çıkışını gösteren LED'ler ve anlık olarak akünün şarj seviyesini izleyebilmek için mini voltmetre eklenmiştir.

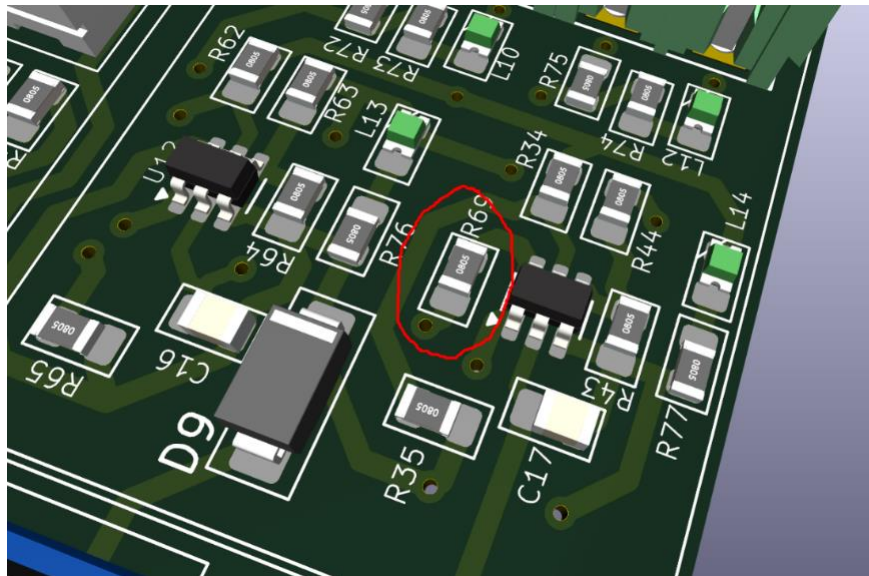


Şekil 3.15. Güç dağıtım kartı

3.4 Mikrodenetleyici Yazılım Geliştirme Süreci

Kontrol sistemleri için seçilen STM32F103C8T6 mikrodenetleyicisi, STM32CubeIDE programı kullanılarak programlanmıştır. Tasarlanan elektronik devre ve geliştirilen mikrodenetleyici yazılımı, oluşabilecek olası sorunları gidermek amacıyla öncelikle breadboard üzerinde uzay montaj testleri yapılmıştır. DAC sinyalinin stabil şekilde üretilebilmesi, motor sürücülerin motorları istenilen hızlarda döndürebilmesi, servo motorlara PWM gönderilmesi, bu PWM sinyalinin gürültülerden etkilenmeden motorlara ulaşabilmesi ve I²C tabanlı sensörlerin haberleşmesi uzay montaj olarak test edilmiştir.

Yapılan testlerde MCP4725 DAC modülünün fabrika çıkışındaki EEPROM ayarı sebebiyle kontrol kartı henüz çalışmaya başlamadan motor sürücülerin ani olarak motorları kısa bir süreliğine yüksek hızda hareket ettirdiği gözlemlenmiştir. Bu sorunun çözülmesi için EEPROM sıfırlaması yapan ek bir devre kartı ve EEPROM sıfırlama yazılımı geliştirilmiştir. Yeni MCP4725 DAC modülleri sisteme dahil edilmeden önce bu karta takılarak EEPROM sıfırlama işlemi yapılmıştır. Ayrıca aracın sol ve sağ motorunu ayrı ayrı kontrol edilebilmesi için Şekil 3.16'da gösterildiği gibi aynı I²C hattına bağlı MCP4725 modüllerinin adresleri üzerinde bulunan adres pinlerinin direnç kullanılarak lojik 0 veya lojik 1 yapılmasıyla değiştirilmiştir.



Şekil 3.16. DAC modülü adres seçimi

3.5 Mekanik Tasarım ve Montaj

Üretilen prototip aracın modeli FreeCAD programı kullanılarak üç boyutlu olarak tasarlanmıştır. Aracın gövde ve iskeletinin, şekli ve boyutları belirlendikten sonra üretim aşamasına geçilmiştir. Araç iskeletinde 5×20 mm lama profiller kullanılmıştır. Dış gövdede ise kompozit malzeme kullanılmıştır. Ön teker olarak iki adet scooter tekeri kullanılmış, arka teker olarak ise serbest döner teker kullanılarak aracın dış gövde ve şasesi yapılmıştır (Şekil 3.17). Başlangıçta direksiyon sistemi olarak scooter tekerlerini farklı hızlarda döndürerek yönlendirme yapılması denenmiştir. Fakat aracın boyutu, ağırlığı ve gövde yapısından ötürü bu denemede başarısız olunmuştur. Daha sonra arka tekerlere Şekil 3.18’de gösterilen lineer aktüatör kullanılarak bir hareket mekanizması oluşturulmuştur. Bu aşamada da lineer aktüatörün hızı yetersiz kalmış ve araca istenilen şekilde yönlendirme yapılamamıştır. Daha sonra lineer aktüatör yerine Şekil 3.19’da gösterilen 25 kg/cm gücünde servo motor takılmasına karar verilmiştir. Bu aşamadan sonra aracın kontrolü başarılı bir şekilde sağlanmıştır.



Şekil 3.17. Aracın şasesi



Şekil 3.18. Lineer aktüatör



Şekil 3.19. Servo motor

3.6 Yapay Zekâ Tabanlı Nesne Tespiti

Yola düşen yabancı cisimleri tespit etmek için YOLOv10 derin öğrenme algoritması kullanılmıştır.

3.6.1 YOLOv10

YOLO (You Only Look Once), nesne tespiti problemini tek bir ileri beslemeli sinir ağı olarak ele alan devrimci bir mimardır. Görüntüyü sabit boyutlu ağ girişine yeniden boyutlandırıp tek geçişte hem sınıf hem de konum bilgilerini çıkararak gerçek zamanlı çalışmaya imkân verir. YOLOv10, gerçek zamanlı uçtan uca nesne tespiti alanında hem performans hem de verimlilik sınırlarını zorlayan yeni bir nesil YOLO modelidir. Geleneksel YOLO sürümleri, tespit doğruluğunu artırmak için sıklıkla Yüksek Sahne Baskınlığı (Non-Maximum Suppression- NMS) işlemi kullanırken, YOLOv10 bu bağımlılığı ortadan kaldıran “uyumlu ikili atama” mekanizmasını benimser; böylece hem eğitimi hem de çıkarımı adım adım indirger ve tepki gecikmesini düşürür. Ayrıca, modelin bileşenleri verimlilik-doğruluk dengesine odaklanan bütünsel bir tasarım stratejisiyle yeniden düzenlenmiş; büyük çekirdekli evrişimler, kanalsal-uzaysal ayrışık aşağı örnekleme ve diğer mimari optimizasyonlarla hesaplama yükü önemli ölçüde azaltılmıştır. Sonuç olarak, YOLOv10, COCO veri setinde benzer AP değerlerinde önceki yaklaşımlara kıyasla yüzde 80’e kadar daha hızlı çıkarım yaparken, parametre ve FLOP sayısını üç kat azaltabilmektedir [22].

Bu çalışmada Çizelge 3.1’de listelenen hiperparametrelerle YOLO-v10 Nano modeli eğitilmiştir. Eğitim 100 epoch sürmüş, batch boyutu ise bellek taşmasına (Out-of-Memory) yol açmadan mümkün olan en büyük mini-paket değeri olacak şekilde 35 olarak belirlenmiştir. 0.01 başlangıç öğrenme oranı (lr0) eğitim ilerledikçe lineer olarak 0.0001’e düşürülmüş; böylece ilk safhada hızlı öğrenme sağlanırken son safhada ağırlık güncellemeleri küçültülerek kararlı yakınsama elde edilmiştir. SGD optimizasyonu ve 0.937 momentum değeri, YOLO’nun varsayılan ayarlarıyla tutarlıdır ve eğitim kaybındaki salınımları sönümlemeye yardımcı olmuştur. Veri artırmada tam mozaik (mosaic = 1.0) kullanılarak her adımda dört görüntü birleştirilmiş, böylece sınırlı veri kümesinde örnek çeşitliliği artırılmıştır; karışık örnekleme (mixup = 0.0) ise doğrulukta düşüşe neden olduğu için devre dışı bırakılmıştır. 640×640 piksel giriş boyutu, nesne ayrıntısını korurken gerçek-zamanlı çıkarım hızını da desteklemiştir. Tespit-gerçek kutu çakışması için %70 IoU eşiği kullanılmış, bu sayede yalancı pozitifler azaltılmıştır.

Çizelge 3.1. YOLOv10 eğitim hiperparametreleri

Parametre	Değer
epochs	100
batch	35
imgsz	640×640 px
lr0	0.01
lr_final	0.0001
lrf	0.01
optimizer	SGD
momentum	0.937
mosaic	1.0
mixup	0.0
iou_threshold	0.7
model	yolov10n.pt (Nano)

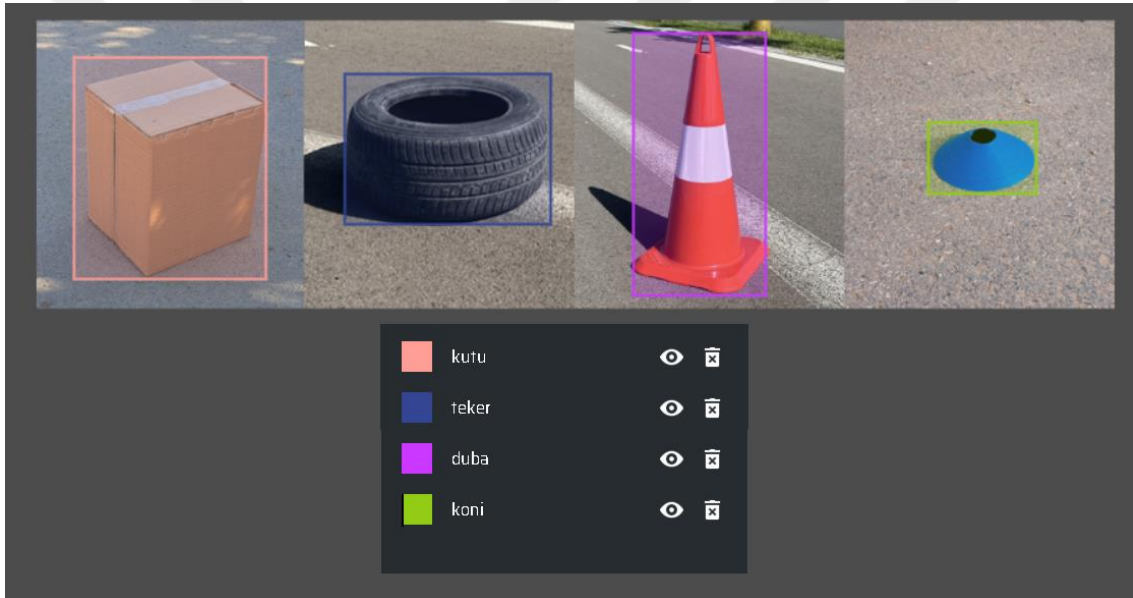
3.6.2 Gerçek zamanlı nesne tespiti

Derin öğrenme modelini eğitmek için gerekli olan veri seti için kutu, teker, antrenman konisi, trafik dubası ve odun olmak üzere 5 farklı sınıf belirlenmiştir. Bu nesnelerin fotoğrafları farklı zeminlerde, farklı ışık şiddetlerinde, günün farklı saatlerinde, farklı mesafelerde ve farklı çekim açılarında çekilmiştir (Şekil 3.20). Her bir sınıf için 400 adet olmak üzere toplamda 2.000 adet görsel çekilmiştir. Fotoğraf çekimi için iPhone 13 ve Samsung A53 telefonları kullanılmıştır.



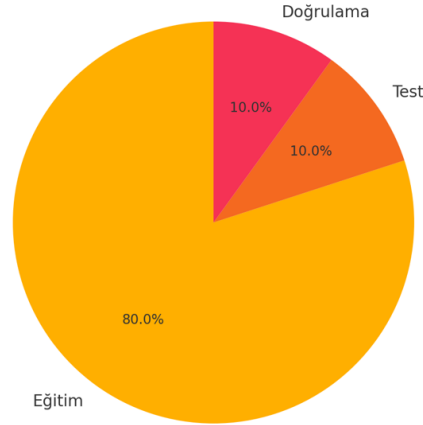
Şekil 3.20. Veri seti görselleri

Çekilen fotoğraflar www.maksense.ai platformu kullanılarak Şekil 3.21’de gösterildiği gibi nesne tespitine uygun olarak etiketlenmiştir.



Şekil 3.21. Veri seti etiketleme

Etiketlenen görseller %80 eğitim, %10 test ve %10 doğrulama olacak şekilde ayrılmıştır (Şekil 3.22). Veri seti ayırma işleminde eğitim, test ve doğrulama işlemlerinden her biri için her sınıftan eşit sayıda görsel alınması için özel bir Python yazılımı geliştirilmiştir. Bu sayede homojen bir dağılım sağlanarak sonuçların iyileştirilmesi hedeflenmiştir.



Şekil 3.22. Veri seti dağılımı

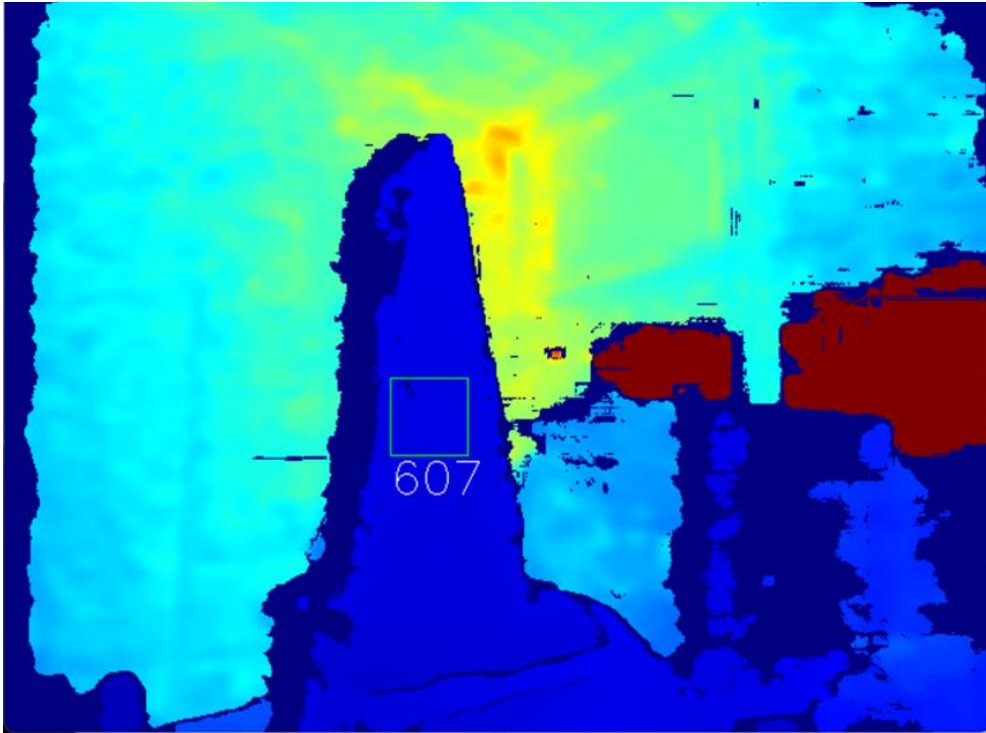
Oluşturulan veri seti Google Colab Pro ortamına yüklenerek YOLOv10 algoritması eğitilmiştir. Eğitilen model yerel bilgisayara indirilerek, gerçek zamanlı olarak nesne tespiti yapan yazılım geliştirilmiştir. Testler sonucunda kamera önüne getirilen nesnelere dikdörtgen çerçeve içerisinde sınıflandıran bir gerçek zamanlı tespit yazılımı geliştirilmiştir (Şekil 3.23). Daha sonra, derinlik kamerası entegre edilerek RGB ve derinlik eksenleri birleştirilmiş nesnenin konum uzaklık, boyut vb. özellikleri analiz edilmiştir



Şekil 3.23. RGB kamerada nesne tespiti

3.6.3 Derinlik kamerası entegrasyonu

Yapay zekâ ile tespit edilen nesnelerin mesafesini ölçmek için RealSense derinlik kamerası kullanılmıştır. Bu kamera mesafeyi siyah tonlar halinde göstermektedir. Geliştirilen yazılım ile bu siyah tonlar kırmızıdan maviye doğru bir renk skalasına dönüştürülmüştür. RGB kamerada nesnelere tespit edildiği anda tespit edilen nesnenin olduğu piksellere karşılık gelen bölgedeki derinlik bilgisinden cismin araca olan mesafesi hesaplanmıştır (Şekil 3.24). Başlangıçta iki görüntü arasında kamera çözünürlük farkından kaynaklı piksel uyumsuzluğu olmasına karşın daha sonra iki görüntü senkronize edilerek piksel kayması sorunu giderilmiştir. Derinlik kamerasına ek olarak RGB kameradan da eğitim konilerini referans alarak mesafe ölçümü yapan bir yazılım geliştirilmiştir. Bu aşamada eğitim konileri yol şeridi oluşturmak için birer metre aralıklarla aracın sağ ve sol tarafına dizilmiştir. RGB kamera bu konilerin piksel değerlerini analiz ederek her bir koninin araca ne kadar uzaklıkta olduğunu tespit etmiştir. Daha sonrasında tespit edilen nesnelerin konilere olan yakınlık durumuna göre ortalama bir mesafe tespiti gerçekleştirmiştir. Bu sistemde, cisimler araca yaklaşıncaya derinlik kamerası ile mesafe ölçülmesi; cisimler derinlik kamerası menzili dışındayken de RGB kamera ile mesafe ölçülmesi hedeflenmiştir.



Şekil 3.24. Derinlik kamerası görüntüsü

3.6.4 Şerit takip sistemi

Geliştirilen aracın nesnelere tespit ederken belirlenen yolda otonom ilerleyebilmesi için şerit takip yazılımı geliştirilmiştir. Bu aşamada şerit olarak eğitim konileri kullanılmıştır. Derin öğrenme algoritması ile tespit edilen eğitim konilerinin diziliş açıları ve aracın anlık yönelimi tespit edilerek aracı şeridin ortasına hizalayacak bir yazılım geliştirilmiştir. RGB kameraların perspektif açıda görüntü yakaladıkları için, eğitim konilerinin anlık konumlarında açı hesabı yapılabilmesi için Eş. 3.1 ve Eş. 3.2’de verilen denklem kullanılmıştır. Bu denklem sabitleri ise regresyon analizi kullanılarak tespit edilmiştir. Bu aşamada, antrenman konilerinden oluşturulan şeritte araç farklı konumlara yerleştirilerek çeşitli ölçümler yapılmış; bu ölçümler Çizelge 3.1’de gösterilmiştir. Bu çizelgede belirtilen yerleşim değeri aracın şeridin ortasındayken sağ veya solda iken alınan verileri gösteriyor. Açı değeri ise kamera kadrajında belirli bölgeye giren konilerden en baştaki koni ile en sondaki koni arasına hayali bir çizgi çekiliyor ve bu çizginin normal ile yaptığı açığı göstermektedir. X fark ve Y fark değerleri ise bu çizginin x eksenini ve y eksenindeki konumunu belirtiyor. Bu değer sağ taraftaki şerit için görüntünün sağ kenarına olan mesafe iken sol taraftaki şerit için ekranın sol tarafına olan mesafeyi piksel cinsinden belirtmektedir.

$$sol_{tahminiaçı} = k1 + (k2 * sol_{xfark}) + (k3 * sol_{yfark}) + (k4 * sol_{xfark}^2) + (k5 * sol_{xfark} * sol_{yfark}) + (k6 * sol_{yfark}^2) \quad (3.1)$$

$$sag_{tahminiaçı} = k1 + (k2 * sag_{xfark}) + (k3 * sag_{yfark}) + (k4 * sag_{xfark}^2) + (k5 * sag_{xfark} * sag_{yfark}) + (k6 * sag_{yfark}^2) \quad (3.2)$$

Çizelge 3.2. Antrenman konilerinin konum ve açı değerleri

Yerleşim	Sol			Sağ		
	Açı	X fark	Y fark	Açı	X fark	Y fark
Orta	44	31	299	136	24	334
	44	62	257	138	44	363
	47	13	334	135	76	390
	48	25	323	133	10	312
	42	74	242	139	56	385
	43	59	260	137	62	372
	49	21	322	139	81	402
	44	38	281	137	32	343
	42	86	239	138	67	384
	43	36	272	138	43	351
	47	35	294	135	27	328
	41	49	263	133	70	368
41	57	247	137	68	379	
Sağ	30	26	184	118	203	383
	32	8	200	120	206	356
	28	62	186	110	181	315
	33	41	170	119	176	317
	29	31	175	120	204	395
	29	24	181	115	221	392
	29	22	183	115	217	382
Sol	82	275	295	152	75	482
	82	268	300	152	21	451
	77	262	257	147	10	436
	80	252	310	150	70	479
	78	273	291	152	67	481
	82	276	284	153	21	457
	86	276	253	148	65	473
	80	257	291	151	33	460

3.7 Unity Arayüzü ve Dijital İkiz

Aracın içerisinde 16 GB RAM'e sahip Acemagic S1 mini PC kullanılmıştır. STM32 mikrodenetleyicisi ile USB-TTL üzerinden UART protokolü kullanılarak haberleşmektedir. Araç bilgisayarı tarafında verilerin izlenmesi ve kontrolü için Unity tabanlı bir arayüz tasarlanmıştır (Şekil 3.25). Bu arayüz hız, direksiyon, fren vb. fonksiyonların izlenmesi ve kontrolünü sağlamaktadır. Mikrodenetleyiciden gelen sensör verilerinin anlık olarak izlenmesi, bu arayüz içerisinde sağlanmaktadır. Bu arayüze uzaktan erişmek için AnyDesk uzak masaüstü uygulaması kullanılmıştır.



Şekil 3.25. Araç içerisindeki bilgisayarın arayüzü

Dijital ikiz oluşturma aşamasında, Unity’de ikinci bir arayüz daha tasarlanmıştır. Bu arayüz sayesinde, harici bir bilgisayarda araç verilerinin sunucudan çekilmesi ve dijital ikizde bulunan sanal model üzerinde gösterilmesi sağlanmıştır. Aracın sunucu ile haberleşmesi Python ile yazılan bir arka uç yazılımı ile sağlanmıştır. Araçta bulunan bilgisayar anlık GPS verisini Şekil 3.26’da gösterilen, internet erişimi için kullanılan RUTX50 modemden almaktadır. GPS verileri Python ile geliştirilen arka uç yazılımı ile NMEA-TCP protokolü kullanılarak alınmakta ve diğer veriler ile sunucuya iletilmektedir.



Şekil 3.26. RUTX50 modem

3.8 Sunucu Haberleşme Katmanı

Başlangıçta sunucu haberleşmesi için Amazon AWS üzerinden bir sunucu kurulumu yapılmıştır. Python ile yazılan bir HTTP/HTTPS tabanlı arka uç kodu PuTTY programı kullanılarak sunucuya aktarılmıştır. Postman programı kullanılarak sunucu iletişimi test edilmiş, başarılı bir şekilde veri alışverişi yapıldıktan sonra Unity ile sunucuyu haberleştirecek Python arka uç yazılımı geliştirilmiştir. Proje kapsamında dijital ikiz teknolojisi 5G tabanlı uç sunucu teknolojisi ile de test edileceği için bir sonraki aşamada Barselona'da bulunan uç sunucu için sunucu yazılımı geliştirilmiştir. Türkiye'de bu uç sunucuya bağlanmak için GlobalProtect VPN uygulaması kullanılmıştır. Tez kapsamında geliştirilen sistem Barselona'ya da götürülerek 5G ve uç sunucu üzerinden de test edilmiştir. Test aşamaları oldukça uzun bir süreç olduğu için Amazon AWS kullanmak maliyetli bulunmuştur. Test yaparken Barselona'daki uç sunucuyu kullanmak ise sunucuya bağlanmak için VPN vb. ek doğrulamalar gerektiği ve 4G ile 5G altyapılarının hız karşılaştırması için çok kullanılabilir bir yöntem olmadığı için sistemde kullanılanlarda üçüncü Bir yerel bilgisayarda sunucu yazılımı geliştirilmiştir. Bu yerel sunucu Cloudflare platformu kullanılarak global sunucuya dönüştürülmüştür. Araç için yapılan testlerde Türkiye'deyken bu sunucu kullanılmıştır.

Geliştirilen tüm sistemde veri akışı şu şekilde işlemektedir: Araç içerisindeki STM32 sensör verileri UART protokolü ile mini bilgisayarda bulunan Unity arayüzüne gönderir, mini Unity bu verileri işler ve araca gaz, direksiyon vb. hareket komutlarını gönderir, araçtan aldığı sensör verilerini kendi içerisinde işlediği diğer veriler ile birleştirerek Python'da çalışan arka uç yazılımına gönderir, arka uç yazılımı modemden aldığı GPS verisi ile birlikte JSON formatında sunucuya gönderir, sunucuda çalışan Python yazılımı bu verileri JSON formatında depolar, dijital ikiz bilgisayarında çalışan Python arka uç yazılımı sunucudan bu verileri çekip Unity'de geliştirilen arayüze gönderir. Unity arayüzü bu verileri kullanarak dijital ikizi yani aracın ve yolun sanal modellerini oluşturur.

3.9 Yazılım ve Tasarım Geliştirme Araçları

Sistemin geliştirilmesinde çeşitli program ve yazılımlar kullanılmıştır; bunlar Çizelge 3.2’de gösterilmiştir.

Çizelge 3.3. Kullanılan program ve yazılımlar

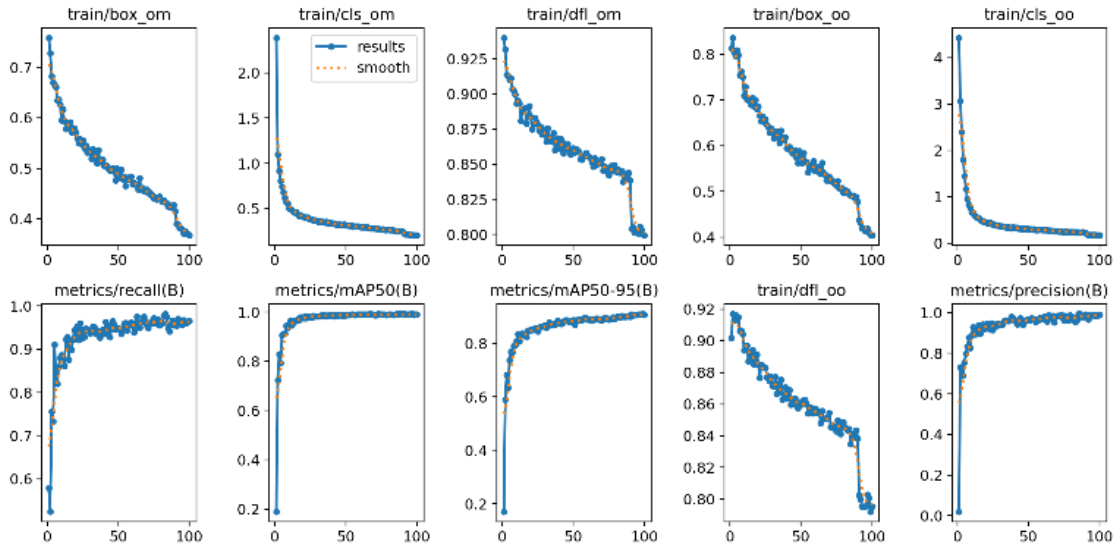
Kullanılan program/yazılım	Kullanım amacı
KiCad	Devre şeması ve PCB tasarımı
FlatCAM	PCB üretim dosyalarının oluşturulması
FreeCAD	Aracın 3B mekanik tasarımının yapılması
Blender	Dijital arayüzdeki 3B nesnelerin tasarımı
Unity	Dijital ikiz ve araç kontrol arayüzünün geliştirilmesi
Visual Studio	Dijital ikiz ve araç kontrol arayüzünün yazılımının geliştirilmesi
STM32CubeIDE	STM32F103C8T6 mikrodenetleyicisinin yazılımının geliştirilmesi
Python	Derin öğrenme, sunucu haberleşmesi, görüntü işleme ve veri işlemede kullanılan yazılım
Anaconda / Spyder	Python yazılımının geliştirildiği tümleşik geliştirme ortamı
Krita	Dijital ikiz arayüzündeki kullanıcı arayüz görsellerinin tasarlanması
Makesense.ai	Veri seti görsellerinin etiketlenmesi

4. BULGULAR ve TARTIŞMA

Bu bölümde, geliştirilen sistemin eğitim ve test aşamalarında elde edilen yapay zekâ ile nesne tespiti sonuçları, dijital ikiz platformundaki gecikme süreleri ve şerit takibi için oluşturulan algoritmanın performansı detaylı bir şekilde ele alınmaktadır.

4.1 Yapay Zekâ (YOLO) Eğitimi ve Test Sonuçları

Yapay zekâ modeli; eğitim konisi, trafik dubası, teker, kutu ve ağaç kütüğünden oluşan 5 farklı nesne sınıfını tespit etmek üzere YOLOv10 tabanlı model ile eğitilmiştir ve eğitim sonrası oluşan sonuç (Result) grafikleri Şekil 4.1’de gösterilmiştir. Eğitim süreci boyunca “box” ve “cls” (sınıflandırma) kayıpları (train/box_om,train/cls_om vs.) düzenli olarak azalma eğilimi gösterirken, “dfl” olarak adlandırılan dağılım kaybı (distribution focal loss) da giderek düşmüştür. 100 epoch’tan oluşan eğitim sonunda sınıflandırma hatalarının büyük oranda minimize edildiği gözlenmiştir.



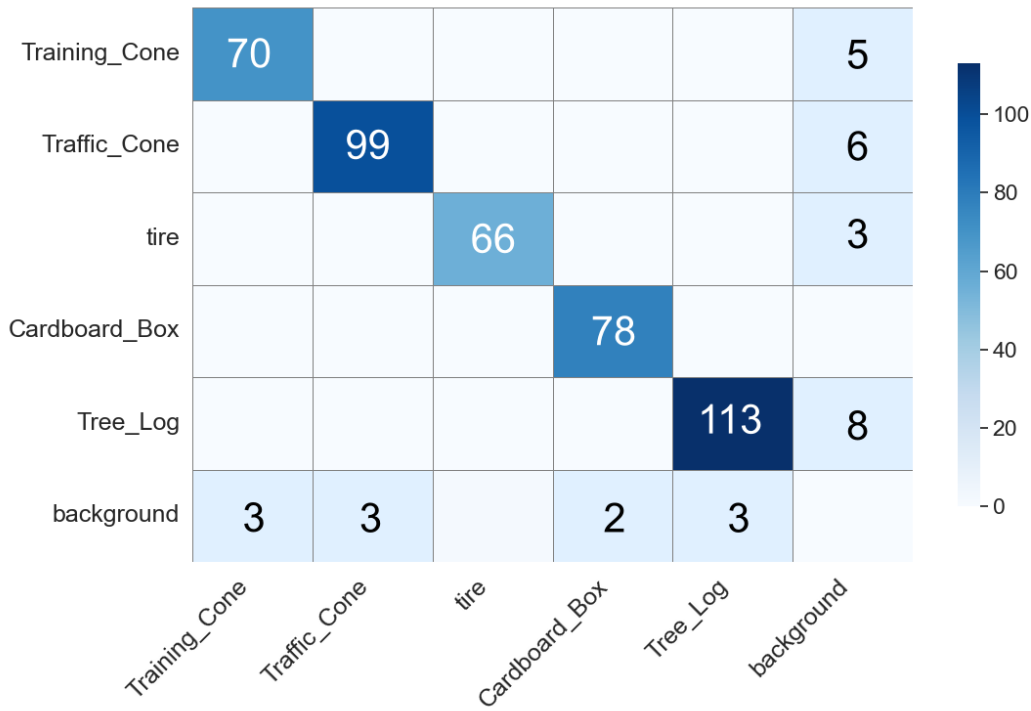
Şekil 4.1. Eğitimin sonuç grafikleri

Tüm loss değerlerinin epoch sayısı ilerledikçe azaldığı gözlenmiştir. Bu durum modelin öğrenme sürecince aşırı veya yetersiz öğrenme yaşamadan stabil bir biçimde öğrenme gerçekleştirdiğini göstermektedir. Elde edilen mAP (Mean Average Precision) değerleri mAP@0.50 için 0,99 ve mAP@0.50–0.95 için yaklaşık 0,91 düzeyine ulaşarak oldukça başarılı bir tespit performansı yakaladığını göstermektedir. Seçilen hiperparametreler,

doğruluk-hız dengesini gözeterek modelin küçük veri kümesi üzerinde yüksek performansla ($mAP@0.50 \approx 0.99$) kararlı biçimde yakınsamasını sağlamıştır; son epoch'larda ölçülen mAP değişimi %0,1'in altında kalarak modelin stabil hale geldiğini göstermiştir. Recall değerlerinde ise 0,60 seviyelerinden hızlıca 0,95 üzeri değerlere yükselen bir eğri gözlemlenmiş ve son aşamada 0,96 civarında bir değer elde edilmiştir. Bu, modelin yanlış-negatif tahminlerini büyük oranda azalttığı anlamına gelmektedir. Eğitimin erken safhalarında yaklaşık 0,50 seviyesinde başlayan precision, giderek artmış ve nihayetinde 0,97–0,98 aralığında seyretmiştir. Bu modelin yanlış-pozitif tahminlerinde önemli bir azalma olduğunu ortaya koymaktadır.

4.1.1 Karışıklık matrisi analizi

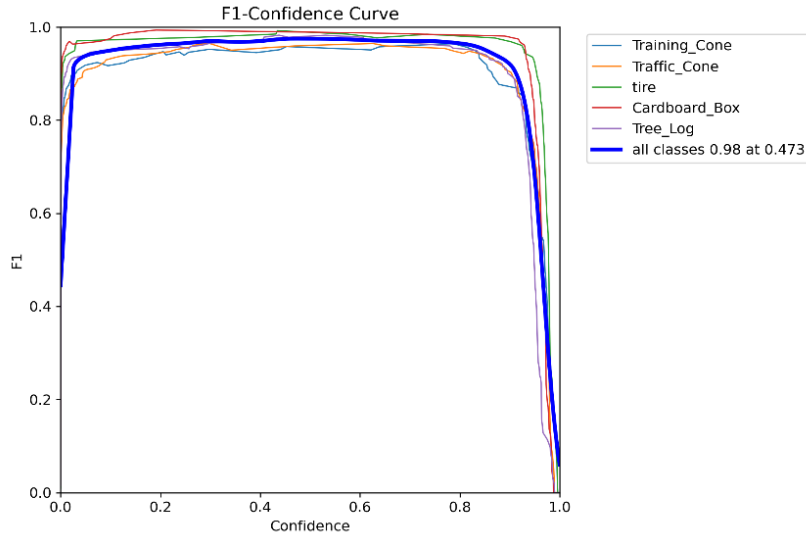
Yapay zekâ modelinin eğitimi sırasında yapılan testler sonucunda oluşturulan normalleştirilmiş karışıklık matrisi Şekil 4.2'de gösterilmiştir. Bu değerlendirme parametresi her bir nesne sınıfı için modelin ne kadar doğrulukta tahminleme yaptığını göstermektedir. Karışıklık matrisi incelendiğinde eğitim konisi, trafik dubası gibi sınıflarda 0,96-0,97 değerinde doğruluğa ulaşıldığı gözlemlenmiştir. Buna karşın bu sınıflarda 0,23-0,27 oranında karışıklık görülebilse de bu karışıklığın sebebi çoğunlukla arka plandan kaynaklanmaktadır.



Şekil 4.2. Karışıklık matrisi

4.1.2 F1-Puanı eğrileri

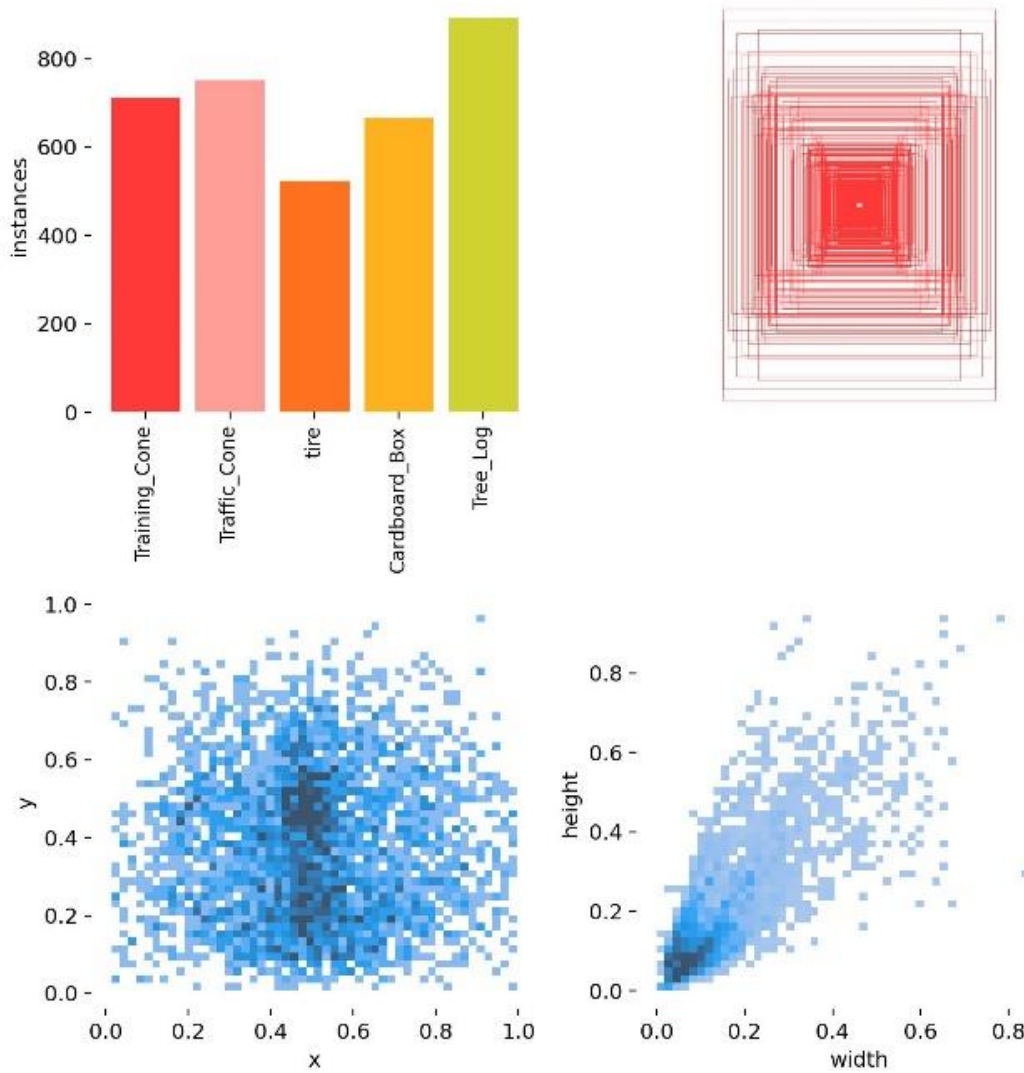
Modelin güven değerleri ile F1 puanları Şekil 4.3'te gösterilmiştir. Bu değerler incelendiğinde yaklaşık 0,4-0,5 aralığında neredeyse tüm sınıflar için F1 değerinin 0,95 seviyesinin üzerine çıktığı gözlemlenmiştir. Optimum eşik noktası civarında ise F1 değerinin yaklaşık 0,98 değerine ulaştığı gözlemlenmiştir. Bu veriler, anlık tespit sisteminde nesne tespiti için güven bakımından eşik değeri belirlemede yol gösterici olmaktadır. Sistemlerde çok fazla yanlış tespit istenmeyen senaryolarda bu değer 0,6 -0,7 civarında tercih edilirken engellerin kesinlikle kaçırılmaması gereken senaryolarda eşik değeri 0,4-0,5 seviyesinde tercih edilebilmektedir. Tez kapsamında yapılan çalışmada bu değer 0,5 olarak tercih edilmiştir.



Şekil 4.3. F1 puan grafiği

4.1.3 Veri dağılımları ve sınırlayıcı kutu analizleri

Şekil 4.4'te gösterilen veri seti içerisindeki etiketlerin dağılımı incelendiğinde ağaç kütüğü sınıfının yaklaşık 900 adet etiket ile en çok örneğe sahip sınıf olduğu görülmüştür. Bununla beraber araç tekeri sınıfının ise yaklaşık 550 adet etiket ile en az örneğe sahip olan sınıf olduğu görülmüştür. Çekilen görüntülerdeki sınırlayıcı kutuların (Bounding box) x-y konum ve genişlik-yükseklik dağılımları, etiketlerin oldukça geniş bir aralığa yayıldığını göstermiştir. Bu sayede modelin farklı konum ve farklı boyut varyasyonlarında yüksek başarı göstermesi sağlanmıştır. Dağılım ve histogram analizleri de verilerin genellikle merkez bölgelere yoğunlaşmakla beraber, kenar alanlarda da yeterince örnek bulunduğunu göstermiştir.



Şekil 4.4. Etiket dağılım grafikleri

Literatürde yer alan bu çalışmaya benzer diğer çalışmaların sonuçları Çizelge 3.4'te gösterilmiştir. Çizelge incelendiğinde, çalışmaların doğrudan karşılaştırılmayacak ölçüde farklı deneysel koşullarla raporlandığı; dolayısıyla metriklerin model mimarisinden çok kullanılan veri kümesinin kapsamı, görüntü çözünürlüğü, IoU eşiği, öğrenme-hızı politikası ve epoch sayısı gibi etkenlere duyarlı olduğu görülmektedir. COCO gibi çok sınıflı, çeşitliliği yüksek veri kümelerinde $mAP@0.50-0.95$ değerleri görece sınırlı kalırken; belirli engel tiplerine odaklanan, çözünürlüğü yükseltilmiş veya sınıf sayısı daraltılmış özel veri kümeleri tek IoU eşiğinde oldukça yüksek doğruluklar sunabilmektedir. Precision ve recall değerleri de benzer biçimde, veri kümesinin dağılımı ve yanlış-pozitif/negatif toleransını dengelemek üzere seçilen eşiklere bağlı geniş bir aralıkta raporlanmıştır.

Çizelge 3.4. Önceki çalışmalar ile karşılaştırma

Makale	mAP@0,50-0,95	mAP@0,50	Precision	Recall	Çözünürlük	ne tespit edildi	Model	Öğrenme oranı
[3]	%56,8	%74,4	0,667	0,839	1280x1280	COCO-80	YOLOv7-E6E	lr0=0,01 lrf=0,2
[10]	%89	%93	—	—	320x65	Araç + yaya + engel (özel set)	YOLOv3 + Fuzzy NMS	—
[13]	—	%92,7	0,918	0,903	800x800	Yolda engel tespiti (özel set)	Yolo v8	0,00125
[18]	%34,2	%57,4	0,659	0,525	640x640	Araç ve yaya tespiti	PV-YOLO	lr0=0,01 lrf=0,01
[22]	%54,4	%71,3	—	—	640x640	COCO-80	YOLOv10-X	lr0=0,01 lrf=0,0001
Bu çalışma	%90	%99	0,98	0,96	640x640	Yolda engel tespiti (özel set)	Yolo v10-nano	lr0=0,01 lrf=0,0001

4.2 Dijital İkiz Platformunda Gecikme Ölçümleri

Geliştirilen dijital ikiz sisteminde, araçtan alınan sensör ve nesne tespiti verileri sunucuya gönderilmekte ve dijital ikizin oluşturulacağı bilgisayar da bu verileri sunucudan çekerek oluşturulan sanal model üzerinde göstermektedir (Şekil 4.5). Bu veri akışı sırasında verinin gönderici birimden çıktıktan sonra alıcı birime ulaşma süresi kritik önem taşımaktadır. Yapılan tez çalışması kapsamında oluşturulan sistemde bu gecikmenin 5G teknolojisi ve uç sunucu teknolojileri kullanılması halinde 4G haberleşmesine göre ciddi bir düşüş yaşandığı gözlemlenmiştir.



Şekil 4.5. Dijital ikiz ekran fotoğrafı

4.2.1 4G bağlantısı ve merkezi sunucu

4G haberleşme teknolojisi kullanılarak yapılan testlerde gecikme yaklaşık 106 ms olarak ölçülmüştür. Ağ trafiğinin yoğun olduğu saatlerde ise bu değerin daha yüksek miktarlara ulaştığı gözlemlenmiştir.

4.2.2 5G bağlantısı ve uç sunucu (edge server)

4G ile yapılan testler sonucunda 5G ile kıyaslama yapılabilmesi açısından araç sistemi İspanya'nın Barcelona şehrinde bulunan IDIADA otomobil test merkezine götürülmüştür. Burada yapılan testlerde gecikme değeri yaklaşık 23 ms ölçülmüştür. Yüksek bant genişliği ve düşük gecikmeli bu haberleşme altyapısı sayesinde araç hareketleri neredeyse anında dijital ikize yansıtılmıştır.

Bu bulgu sistemin kontrol ve takibi açısından 5G altyapısının büyük avantaj sağladığını göstermektedir. Yüksek hızlı veri iletişimi tespit edilen engellerin ve araç verilerinin dijital ikize aktarılması ve gerekli durumlarda araca uzaktan hızlı müdahale imkânı sağlamıştır.

4.3 Şerit Takip Sistemi

Aracın engel tespiti esnasında belirli bir rota izleyebilmesi için antrenman konileri kullanılarak şerit oluşturulmuştur. Aracın bu şeritleri ortalayarak seyretmesi hedeflenmiştir. Araç üzerine yerleştirilen kameranın perspektif açısından ötürü antrenman konilerinin şerit takibi açısından farklı konumlarda farklı değerler verdiği gözlemlenmiştir. Bu duruma çözüm olarak antrenman konilerinin konumları analiz edilerek hedef konum verisini tespit etmek için Eş. 4.1 ve Eş. 4.2'de belirtilen denklem değişkenleri için regresyon analizi kullanılarak katsayılar bulunmuştur.

$$sol_{tahminiaçl} = 90.71 + (-0.2644 * sol_{xfark}) + (0.0751 * sol_{yfark}) + (0.0005 * sol_{xfark}^2) + (-0.0003 * sol_{xfark} * sol_{yfark}) + (-0.0002 * sol_{yfark}^2) \quad (4.1)$$

$$sag_{tahminiaçl} = 77.57 + (0.2271 * sag_{xfark}) + (0.0402 * sag_{yfark}) + (-0.0003 * sag_{xfark}^2) + (0.0001 * sag_{xfark} * sag_{yfark}) + (0.0002 * sag_{yfark}^2) \quad (4.2)$$

Denklemlerde yer alan x_fark ve y_fark değişkenleri, kamerada tespit edilen konilerin referans noktasına göre yatay ve dikey eksenlerde olan konum farkını temsil etmektedir. Denklemlerde yer alan ikinci derece ve diğer çarpan değerleri ise konilerin kameradaki farklı açılardan çekilmiş varyasyonlarının regresyon analizi ile bulunmuştur. Bu fonksiyonlar

aracılıđıyla konilerden oluřturulan řerit bařarılı bir řekilde analiz edilmiř ve ara řeridin ierisinde kalarak hareket edebilmiřtir.

Geliřtirilen sistem genel olarak deęerlendirildięinde YOLOv10 ile oluřturulan nesne tespiti modeli, doęruluk ve geri aęırma parametreleri aısından yksek performansa eriřmiřtir. mAP deęeri yaklařık 0,99'a kadar ıkmıřtır. Bu bulgu, farklı zeminlerde ve aılarda ekilen veri setinin yeterince zengin olduęunu, modelin ařırı ęrenme (overfitting) veya yetersiz ęrenme (underfitting) meydana gelmeden bařarılı bir řekilde ęrenebildięini gstermektedir.

5G altyapısının 23 ms gecikme ile veri haberleřmesi saęlaması dijital ikiz sisteminin gerek zamana ok yakın bir deęerde alıřmasına imkân saęlamıřtır. 4G haberleřme altyapısında meydana gelen 106 ms olarak tespit edilen gecikme ise ara hareketlerinin sanal kopyaya daha ge yansımaya ve olası ani mdahale durumlarında olası sorunlar yařanabileceęi saptanmıřtır.

Regresyon tabanlı řerit takip sisteminde antrenman konilerinden elde edilen konum farklarından hedef aı bulunması doęrusal bir modelden ziyade regresyon analiz sonularına dayanan bir denkleme dayanması daha saęlıklı bir řerit takibi imkânı saęlamıřtır.

Devre řeması tasarımından PCB retimine, yazılım entegrasyonundan test ařamasına kadarki tm bu srete elde edilen bulgular sensr, kamera, mikrodenetleyici, bilgisayar, modem vb. eřitli evre birimlerinin uyum ierisinde alıřtıęını gstermektedir.

Tm bu sonular dikkate alındıęında yola dřen nesnelerin yksek doęrulukta tespit edilmesi ve 5G altyapısı kullanılarak dijital ikize neredeyse anlık olarak yansıtılması mmkndr. Ayrıca koni tabanlı bir řerit takip sistemi aracı basit dzeyde ynlendirmek iin yeterli olsa da farklı iklim kořulları, farklı yollarda ve karmařık durumlarda verimli sonular elde edebilmesi iin ek alıřmalar yapılmalıdır.

5. SONUÇ VE ÖNERİLER

Bu çalışmada, yola düşen engelleri gerçek zamanlı olarak tespit eden ve araçtan alınan sensör verileriyle birlikte dijital ikiz platformuna aktararak aracın uzaktan izlenmesi ve kontrolüne imkân sağlayan bir sistem geliştirilmiştir. Çalışma kapsamında araç donanımı, elektronik altyapı ve kontrol sistemi sıfırdan tasarlanmış, geliştirilen yazılım ile YOLO tabanlı yapay zekâ modeliyle nesne tespiti sağlanmış ve elde edilen verilerin dijital ikize aktarılması 4G ve 5G haberleşme altyapılarıyla test edilmiştir. Bu veriler Unity oyun geliştirme programı kullanılarak geliştirilen dijital ikiz modelinde oluşturulan sanal araçta görselleştirilmiştir.

Başlıca araştırma problemi; normal yol koşullarında yola düşen engellerin sürüş güvenliği açısından risk oluşturduğu bir durumda, düşük maliyetli kamera ve derin öğrenme yöntemleriyle söz konusu engellerin yüksek doğrulukta saptanabilmesi ve bu bilgilerin uzaktan değerlendirilebilmesidir. Bir diğer araştırma problemi, bu senaryoda 4G ve 5G haberleşme altyapılarının kullanılmasının veri aktarım hızı bakımından kıyaslanmasıdır. Bunların yanında üçüncü problem, aracın belirlenen istikamette konilerden oluşan bir şeridi takip edebilmesi problemidir. Bu amaçlar doğrultusunda geliştirilen prototip araç ve yazılım sistemlerinin sonucunda aşağıdaki çıkarımlar yapılmıştır.

YOLOv10 algoritmasıyla eğitilen model her bir nesne sınıfında ortalama %99 seviyesinde mAP@0.50 değerine ulaşmıştır. Bu sonuç, yola düşen nesnelerin önemli ölçüde doğru tespit edilebildiğini göstermektedir. Özellikle düşük ışık ve karmaşık ortamlarda nesneler hem hızlı hem de doğru bir şekilde tespit edilebilmiştir.

Çalışmanın bir diğer katkısı olan, tespit edilen nesnelerin ve sensör bilgilerinin bulut veya uç sunucuya iletilip Unity tabanlı dijital ikizde gösterilmesidir. 4G ve bulut sunucu kullanılarak yapılan testlerde 106ms, 5G ve uç sunucu kullanılarak yapılan testlerde ise 23 ms gecikme ölçülmesi gerçek zamanlı veri izleme ve kontrol için önemli bir göstergedir. Elde edilen bu sonuç, 5G altyapısının düşük gecikmeli veri aktarımıyla dijital ikiz uygulamalarında 4G'ye göre veri aktarım hızı bakımından çok daha kullanışlı olduğunu göstermektedir. Literatürde 5G'nin akıllı ulaşım senaryolarında gecikmeyi en aza getirdiği vurgulanmaktadır [6, 7]. Bu tezde ise söz konusu etkinin, prototip düzeyindeki sisteme

uygulandığındaki sonuçlar görülmektedir. Şerit olarak kullanılan konilerin kamera koordinatlarından faydalanılarak oluşturulan sistemdeki denklemlerin saha testlerinde aracın başarılı bir şekilde şeridin dışına çıkmadan hareket ettiği gözlemlenmiştir.

Geliştirilen sistem genel olarak değerlendirildiğinde devre şeması çiziminden, PCB üretimine, mikrodenetleyici yazılımından bulut ve dijital ikiz teknolojilerinin entegrasyonuna kadar bütünüyle ele alınmıştır. Elde edilen bulgular sistemin sorunsuz şekilde çalıştığını ve geliştirme amacına uygun bir prototip oluşturulduğunu ortaya koymaktadır.

Sistem genel olarak verimli çalışmakla beraber ileride sistemin performansını iyileştirecek öneriler bulunmaktadır. Farklı iklim şartlarında daha geniş bir veri seti oluşturularak modelin nesne tespiti doğruluğu artırılabilir. Kamera tabanlı tespitin LİDAR veya radar gibi ek sensörler ile birleştirilerek çoklu tespit sensörlerinden oluşan bir tespit sistemiyle daha güvenli bir sistem oluşturulabilir ve dijital ikiz açısından daha zengin görselleştirmenin yapılması sağlanabilir. Sistemde kamera titreşimlerinden kaynaklanan görüntü bulanıklığı giderilmesi açısından donanımsal ve yazılımsal geliştirmeler yapılarak yapay zekâ modelinin nesne tespiti performansı artırılabilir. Son olarak, dijital ikiz sisteminde yalnızca nesne tespiti ve çevresel sensörler yerine aracın motoru, bataryası ve diğer donanımlarının anlık durumu da dijital ikize yansıtılabilir. Bu sayede araçta meydana gelebilecek arıza durumları da tespit edilebilir.

KAYNAKLAR

1. Fagnant, D. J., Kockelman, K. (2015). Preparing a nation for autonomous vehicles: opportunities, barriers and policy recommendations. *Transportation Research Part A: Policy and Practice*, 77, 167-181.
2. Nabati, M. R. (2021). Sensor Fusion for Object Detection and Tracking in Autonomous Vehicles.
3. Wang, C.-Y., Bochkovskiy, A., Liao, H.-Y. M. (2023). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7464-7475.
4. Balla, M., Haffner, O., Kučera, E., Cigánek, J. (2023). Educational case studies: Creating a digital twin of the production line in TIA portal, unity, and game4automation framework. *Sensors*, 23(10), 4977.
5. Wang, Z., Han, K., Tiwari, P. (2021). Digital twin simulation of connected and automated vehicles with the unity game engine. *IEEE 1st International Conference on Digital Twins and Parallel Intelligence (DTPI)*, IEEE, 1-4.
6. Hu, C., Fan, W., Zeng, E., Hang, Z., Wang, F., Qi, L., Bhuiyan, M. Z. A. (2021). Digital twin-assisted real-time traffic data prediction method for 5G-enabled internet of vehicles. *IEEE Transactions on Industrial Informatics*, 18(4), 2811-2819.
7. Schwarz, C., Wang, Z. (2022). The role of digital twins in connected and automated vehicles. *IEEE Intelligent Transportation Systems Magazine*, 14(6), 41-51.
8. Zhang, Z., Zou, Y., Zhou, T., Zhang, X., Xu, Z. (2021). Energy consumption prediction of electric vehicles based on digital twin technology. *World Electric Vehicle Journal*, 12(4), 160.
9. Zhang, T., Liu, X., Luo, Z., Dong, F., Jiang, Y. (2019). Time series behavior modeling with digital twin for Internet of Vehicles. *EURASIP Journal on Wireless Communications and Networking*, 2019, 1-11.
10. Zaghari, N., Fathy, M., Jameii, S. M., Shahverdy, M. (2021). The improvement in obstacle detection in autonomous vehicles using YOLO non-maximum suppression fuzzy algorithm. *The Journal of Supercomputing*, 77(11), 13421-13446.
11. Huu, P. N., Pham Thi, Q., & Tong Thi Quynh, P. (2022). Proposing Lane and Obstacle Detection Algorithm Using YOLO to Control Self-Driving Cars on Advanced Networks. *Advances in Multimedia*, 2022(1), 3425295.
12. Rekha, B. S., Marium, A., Srinivasan, G. N., Shetty, S. A. (2020). Literature survey on object detection using YOLO. *International Research Journal of Engineering and Technology (IRJET)*, 7(06).
13. Alahdal, N. M., Abukhodair, F., Meftah, L. H., Cherif, A. (2024). Real-time Object Detection in Autonomous Vehicles with YOLO. *Procedia Computer Science*, 246, 2792-2801.

14. Sukumar, B. S., Khan, M. A., Swamy, T. J., Karthik, K., Reddy, R. S. (2024). Autonomous Vehicle: Obstacle Avoidance and Classification with YOLO. *2024 International Conference on Science Technology Engineering and Management (ICSTEM)*, IEEE, 1-5.
15. Flores-Calero, M., Astudillo, C. A., Guevara, D., Maza, J., Lita, B. S., Defaz, B., Ante, J. S., Zabala-Blanco, D., Armingol Moreno, J. M. (2024). Traffic sign detection and recognition using YOLO object detection algorithm: A systematic review. *Mathematics*, 12(2), 297.
16. Fanthony, I. V., Husin, Z., Hikmarika, H., Dwijayanti, S., Suprpto, B. Y. (2021). Yolo algorithm-based surrounding object identification on autonomous electric vehicle. *2021 8th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, IEEE, 151-156.
17. Gheorghe, C., Duguleana, M., Boboc, R. G., Postelnicu, C. C. (2024). Analyzing Real-Time Object Detection with YOLO Algorithm in Automotive Applications: A Review. *CMES-Computer Modeling in Engineering & Sciences*, 141(3).
18. Liu, Y., Huang, Z., Song, Q., Bai, K. (2025). PV-YOLO: A lightweight pedestrian and vehicle detection model based on improved YOLOv8. *Digital Signal Processing*, 156, 104857.
19. Martinelli, F., Mercaldo, F., Santone, A. (2024). A Method for Finding Distance in Real-Time Car Detection through Object Detection. *Procedia Computer Science*, 246, 5349-5358.
20. Nämerforslund, T. (2022). Digital twin performance: unity as a platform for visualizing interactive digital twins.
21. Ali, W. A., Fanti, M. P., Roccotelli, M., Ranieri, L. (2023). A review of digital twin technology for electric and autonomous vehicles. *Applied Sciences*, 13(10), 5871.
22. Wang, A., Chen, H., Liu, L., Chen, K., Lin, Z., & Han, J. (2024). Yolov10: Real-time end-to-end object detection. *Advances in Neural Information Processing Systems*, 37, 107984-108011.

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı

Uyruğu

Doğum tarihi ve yeri

Medeni hali

Telefon

Faks

e-mail

Eğitim

Derece	Eğitim Birimi	Mezuniyet Tarihi
Doktora		
Yüksek lisans	SBTÜ/ Savunma Teknolojileri	Devam ediyor
Lisans	KKÜ / Elektrik Elektronik Mühendisliği	2022
Lise	Halis Gülle Anadolu Lisesi	2018

İş Deneyimi

Yıl	Yer	Görev
2023-Halen	Ada Güzey Mühendislik LTD	AR-GE Mühendisi

Yabancı Dil

İngilizce

Yayımlar

Duran, A., ve Bakır, H. (2023). Hiper parametreleri Ayarlanmış Makine Öğrenimi Algoritmalarını Kullanarak Android Sistemlerde Kötü Amaçlı Yazılım Tespiti. *Uluslararası Sivas Bilim ve Teknoloji Üniversitesi Dergisi*, 2(1), 1-19.

Duran, A., Bakır, H., ve Guzey, H. M. (2023). Bilinen CNN mimarilerinin görsel Captcha sınıflandırması açısından değerlendirilmesi. In *International Conference on Pioneer and*

Innovative Studies (s. 379-385).

Duran, A., Çelik, K., Adem, K., ve Dümen, S. (2023). Yapay Zekâ Algoritmaları ile Dünya Mutluluğunu Tahminleme. In *International Conference on Pioneer and Innovative Studies* (s. 386-394).

Çelik, K., ve **Duran, A.** (2022). Akıllı Tarımda Nem ve Sıcaklık Sensörleri Kullanılarak Tarımsal Verilerin Algılanması İşlenmesi ve Transferi. *ISPEC JOURNAL OF SCIENCE INSTITUTE*, 1(1), 17-26.

Duran, A. ve Guzey, H. M. (2025). Real-Time Object Detection and Digital Twin Simulation: A YOLOv10 Approach for Autonomous Vehicles. *Modern Research In Computer Sciences And Engineering* (s. 5-15).

Hobiler

Film izlemek, Kitap okumak



**SIVAS
BİLİM VE TEKNOLOJİ
ÜNİVERSİTESİ**

KÖKLERDEN GÖKLERE...