

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

**3D NOKTA BULUTU VERİLERİ KULLANILARAK OTONOM SÜRÜŞ İÇİN
NESNE ALGILAMA YÖNTEMİ İLE KARAYOLU ENVANTERLERİNİN
TESPİT EDİLMESİ**



DOKTORA TEZİ

Hilal GEZGİN

Bilişim Uygulamaları Anabilim Dalı

Coğrafi Bilgi Teknolojileri Programı

HAZİRAN 2025

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

**3D NOKTA BULUTU VERİLERİ KULLANILARAK OTONOM SÜRÜŞ İÇİN
NESNE ALGILAMA YÖNTEMİ İLE KARAYOLU ENVANTERLERİNİN
TESPİT EDİLMESİ**

DOKTORA TEZİ

**Hilal GEZGİN
706182006**

Bilişim Uygulamaları Anabilim Dalı

Coğrafi Bilgi Teknolojileri Programı

Tez Danışmanı: Prof. Dr. Reha Metin ALKAN

HAZİRAN 2025

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

**DETERMINATION OF HIGHWAY INVENTORIES WITH OBJECT
DETECTION METHOD FOR AUTONOMOUS DRIVING USING 3D POINT
CLOUD DATA**

Ph.D. THESIS

**Hilal GEZGİN
(706182006)**

Department of Information Technology Applications

Geographic Information Technologies Programme

Thesis Advisor: Prof. Dr. Reha Metin ALKAN

JUNE 2025

İTÜ, Lisansüstü Eğitim Enstitüsü'nün 706182006 numaralı Doktora Öğrencisi Hilal GEZGİN, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı "3D Nokta Bulutu Verileri Kullanılarak Otonom Sürüş İçin Nesne Algılama Yöntemi ile Karayolu Envanterlerinin Tespit Edilmesi" başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı : **Prof. Dr. Reha Metin ALKAN**
İstanbul Teknik Üniversitesi

Jüri Üyeleri : **Prof. Dr. Mustafa YANALAK**
İstanbul Teknik Üniversitesi

Prof. Dr. Engin GÜLAL
İstanbul Atlas Üniversitesi

Prof. Dr. Turan ERDEN
İstanbul Teknik Üniversitesi

Prof. Dr. Cemal Özer YİĞİT
Gebze Teknik Üniversitesi

Teslim Tarihi **: 25 Nisan 2025**
Savunma Tarihi **: 16 Haziran 2025**





Eşime ve oğluma,



ÖNSÖZ

Bu doktora çalışmam süresince engin tecrübesi, bilgisi ile bana yol gösteren, danışmanım Sayın Prof.Dr. Reha Metin ALKAN hocama, sabrı ve ilgisinden dolayı en içten teşekkürlerimi sunarım.

Değerli zamanlarını, bilgilerini ve tecrübelerini paylaşan Tez İzleme Jüri üyesi Prof.Dr. Mustafa YANALAK ve Prof.Dr. Engin GÜLAL hocalarıma ayrı ayrı bu süreçteki emekleri, yapıcı geri bildirimleri ve cesaretlendirici sözleri için çok teşekkür ederim.

Çalışmalarımı 43066 numaralı proje ile destekleyen İTÜ BAP Koordinasyon Birimi Proje Bölümü'ne teşekkür ederim.

Çalışmamızın uygulama kısmında kullandığımız Mobile Mapping System ile toplanan verilerin teminindeki katkılarından ve çalışmamız sürecinde desteklerinden dolayı Dr. Hüseyin KURŞUN'a ve nezdinde Anka Coğrafi Bilgi Teknolojileri firmasına teşekkür ederim.

Hayatım boyunca en büyük destekçim olan, maddi ve manevi destekleriyle her zaman yanımda olan babam Salih TÜRK'e, annem Emine TÜRK'e minnettarım. Çalışmam boyunca sürekli yanımda olan, her an yardımına koşan bilgisi ve tecrübesi ile beni destekleyen, bana inanan sevgili eşim Dr. Öğr. Üyesi Ahmet Talha GEZGİN'e teşekkür ederim. Onların yardımı olmadan bu tez çalışmasını tamamlamak imkânsız olurdu. En büyük motivasyon kaynağım, uğurum olan, bu süreçte doğan, büyüyen oğlum Atlas'a sevgilerimi sunarım.

Son olarak, doktora sürecim boyunca beni sabırla destekleyen ve her zaman yanımda olan herkese en içten teşekkürlerimi sunarım.

Haziran 2025

Hilal GEZGİN
(Öğretim Görevlisi)

İÇİNDEKİLER

	<u>Sayfa</u>
ÖNSÖZ	ix
İÇİNDEKİLER	xi
KISALTMALAR	xiii
SEMBOLLER	xv
ÇİZELGE LİSTESİ	xvii
ŞEKİL LİSTESİ	xix
ÖZET	xxi
SUMMARY	xxv
1. GİRİŞ	1
1.1 Tezin Amacı	4
1.2 Literatür Araştırması	6
2. METODOLOJİ	11
2.1 Yol Envanterleri Toplama Yöntemleri.....	11
2.2 Teknik Arka Plan	16
2.2.1 Derin öğrenme.....	16
2.2.2 Sinir ağı	17
2.2.3 Evrişimsel sinir ağı.....	18
2.2.3.1 Evrişimsel katman	19
2.2.3.2 Havuzlama.....	19
2.2.3.3 Tam bağlantılı katman.....	19
2.2.3.4 Bırakma katmanı	20
2.2.4 Aktivasyon fonksiyonları	20
2.2.4.1 ReLU	21
2.2.4.2 Softmax	21
2.2.5 Optimizasyon algoritması	22
2.2.5.1 Adam	22
2.2.6 Loss fonksiyonu	22
2.2.7 Bilgisayarla görü	22
2.2.8 Görüntü işleme	23
2.2.9 Deney ortamı.....	23
2.2.9.1 Python	23
3. YOL ENVANTERİ TESPİTİ İÇİN ÇALIŞMADA UYGULANMAK ÜZERE ÖNERİLEN YÖNTEM	27
3.1 Önerilen Algoritma YOLOv4-Tiny	27
3.2 Uygulama Verilerine Genel Bakış	32
3.3 Veri Seti Görüntülerinin İşlenmesi	36
3.4 CNN Modelinin Oluşturulması ve Eğitilmesi	38
3.5 Modelin Gerçek Zamanlı Olarak Test Edilmesi	41

3.5.1 Açık kaynak görüntüler üzerinde gerçekleştirilen deneyler.....	41
3.5.2 MHS sisteminden elde edilen gerçek ortam görüntüleri üzerinde gerçekleştirilen deneyler	43
4. TARTIŞMA	47
5. SONUÇ VE ÖNERİLER.....	55
ÖZGEÇMİŞ.....	65



KISALTMALAR

ADAM	: Adaptive Moment Estimation (Adaptif Moment Tahmini)
AUS	: Akıllı Ulaşım Sistemleri
CNN	: Convolutional Neural Network (Evrişimsel Sinir Ağı)
CBS	: Coğrafi Bilgi Sistemi
EEG	: Elektroensefalografi
GIS	: Geographic Information System
GNSS	: Global Navigation Satellite System (Küresel Navigasyon Uydu Sistemleri)
ITS	: Intelligent Transportation Systems
LIDAR	: Light Detection and Ranging (Lazer Görüntüleme Algılama ve Menzil Belirleme)
MMS	: Mobile Mapping System (Mobil Haritalama Sistemi)
MHS	: Mobil Haritalama Sistemi
mAP	: Ortalama Hassasiyet
ReLU	: Rectified Linear Unit (Doğrultulmuş Lineer Ünite)
SVM	: Support Vector Machine (Destek Vektör Makineleri)
TÜİK	: Türkiye İstatistik Kurumu
YOLO	: You Only Look Once (Sadece Bir Kez Bakarsın)
YSA	: Yapay Sinir Ağı



SEMBOLLER

C_x, C_y : Sınıfları ifade eden parametre

σ_{tx}, σ_{ty} : Sigmoid fonksiyonu

b_x, b_y : Nesnenin orta noktasının koordinatları

b_h, b_w : Sınırlayıcı kutunun yüksekliği ve genişliği

P_w : Sınıf olasılığını tanımlayan çapa kutusunun genişliği

P_h : Sınıf olasılığını tanımlayan çapa kutusunun yüksekliği

e^{th}, e^{tw} : Logaritmik ölçeklenen boyut tahmini yüksekliği ve genişliği

$s(t)$: Çıktı sinyali

$x(a)$: Giriş fonksiyonu

z^i : Giriş vektörü bileşeni

K : Giriş vektöründeki toplam bileşen sayısı



ÇİZELGE LİSTESİ

Sayfa

Çizelge 2.1 : Yol envanteri bilgisi toplama yöntemleri.....	15
Çizelge 3.1 : Test ölçümlerinde kullanılan MHS bileşenlerinin ana özellikleri.....	34
Çizelge 3.2 : Seçilen Trafik İşaretleri ve Sınıf IDleri.....	36
Çizelge 3.3 : TT100K veri kümesi kullanılarak çeşitli trafik işareti algılama modellerinin performanslarının değerlendirilmesi [65]......	40
Çizelge 3.4 : Trafik işaretlerinin sınıflarına göre <i>mAP</i> değerleri.....	41



ŞEKİL LİSTESİ

Sayfa

Şekil 3.1 : YOLO tespit sistemi.....	29
Şekil 3.2 : Yolov4-Tiny'nin ağ mimarisi.	30
Şekil 3.3 : MHS aracı görüntüleri.....	33
Şekil 3.4 : MHS Aracı.	35
Şekil 3.5 : Trafik işareti etiketleme örneği.	37
Şekil 3.6 : Etiketlenmiş trafik işareti.	38
Şekil 3.7 : YOLOv4-Tiny kullanılarak eğitim kaybı değeri, <i>mAP</i> ve AP sonucu. ...	39
Şekil 3.8 : Trafik işareti algılama sonucu örneği.....	42
Şekil 3.9 : Trafik işareti algılama sonucu örneği.....	43
Şekil 3.10 : Modelin gerçek MHS görüntüleri üzerinde test edilmesi.	44
Şekil 3.11 : Bir web kamerasından trafik işareti algılama örneği.....	45
Şekil 3.12 : Bir web kamerasından trafik işareti algılama örneği.....	46
Şekil 4.1 : Renk değişiminin trafik işareti algılama üzerindeki etkisi.	47
Şekil 4.2 : Renk değişiminin trafik işareti algılama üzerindeki etkisi.	48
Şekil 4.3 : Farklı açı ve mesafelerde artan <i>mAP</i> değeri ile elde edilen test sonuçları.	50
Şekil 4.4 : Farklı açı ve mesafelerde artan <i>mAP</i> değeri ile elde edilen test sonuçları.	51
Şekil 4.5 : Farklı açı ve mesafelerde artan <i>mAP</i> değeri ile elde edilen test sonuçları.	52
Şekil 4.6 : Farklı ışık ortamlarının ve hava koşullarının trafik işaretlerinin algılanması üzerindeki etkisi.	53



3D NOKTA BULUTU VERİLERİ KULLANILARAK OTONOM SÜRÜŞ İÇİN NESNE ALGILAMA YÖNTEMİ İLE KARAYOLU ENVANTERLERİNİN TESPİT EDİLMESİ

ÖZET

Trafik işaretleri, karayollarında düzeni sağlamak, sürücü ve yayalara rehberlik etmek, trafik güvenliğini artırmak ve trafik akışını düzenlemek amacıyla kullanılan semboller, levhalar ve işaretlemelerdir. Yapılan araştırmalar, trafik işaretlerinin eksikliği veya yetersizliği nedeniyle meydana gelen kazaların oranının önemli ölçüde arttığını göstermektedir. 2023 yılında yayımlanan raporda, her yıl yaklaşık 1.19 milyon insanın trafik kazalarında hayatını kaybettiği ve 17.8 milyon kişinin ciddi şekilde yaralandığı belirtilmiştir. Trafik işaretleri, olası tehlikelere karşı sürücü ve yayalara uyarıda bulunarak kazaların önlenmesine yardımcı olmakta ve trafik kurallarının anlaşılmasını kolaylaştırmaktadır. Yol envanteri ise, yolların fiziksel özelliklerini, trafik işaretlerini, sinyalizasyon sistemlerini ve diğer altyapı unsurlarını içeren kapsamlı bir veri bütünüdür. Yol envanterinin düzgün ve güncel olması, trafik işaretlerinin doğru konumda olması sürücülerin güvenli bir şekilde seyahat edebilmesi için hayati önem taşımaktadır. Ayrıca, trafik yönetimi ve şehir planlaması açısından da yolların bakım ve geliştirme süreçlerinin daha verimli hale getirilmesi için gereklidir.

Türkiye’de trafik işaretleri ve yol envanteri yönetimi, Karayolları Genel Müdürlüğü (KGM), Ulaştırma ve Altyapı Bakanlığı, Belediyeler ve İçişleri Bakanlığı’na bağlı Emniyet Genel Müdürlüğü ve Jandarma Genel Komutanlığı gibi kurumlar tarafından yönetilmektedir. Yol bilgileri, trafik yoğunluğu izleme, kavşak düzenlemeleri sistemleri ve bakım-onarım ihtiyaçları gibi işlemler için gerekli veriler çoğunlukla saha çalışmalarıyla toplanmaktadır. Karayolları Genel Müdürlüğü, "Yol Envanter Sistemi" aracılığıyla tüm verileri merkezi bir veritabanında saklamaktadır. Dünyada ise manuel çalışmalara ek olarak Geographic Information System (GIS) tabanlı veri sistemleri, uydu görüntüleri, otomatik veri toplama sistemleri, dronlar, LiDAR ölçmeleri ve yapay zeka destekli sistemler ile yol envanteri bilgi sistemleri güncel tutulmaya çalışılmaktadır. Trafik işareti ve yol envanteri bilgi sistemi oluşturmak için geleneksel yöntemlerle sahada ölçüm yapmak, bilgilerin dijital ortama aktarılması ve yönetilmesi oldukça zaman alıcı ve maliyeti yüksek olan bir süreçtir. Bunun yanı sıra, bu geleneksel yöntemler veri üretimini zorlaştırırken, iş güvenliği açısından da çeşitli riskler taşımaktadır. Bu risklerin de etkisiyle hızlı veri toplamaya olan ihtiyaç, yeni araştırma alanlarının ortaya çıkmasına neden olmuştur. Özellikle 3D yol envanterinin hazırlanması, oluşturulması ve güncellenmesi gibi süreçleri, sahaya gitmeye gerek kalmadan hızlı ve verimli bir şekilde gerçekleştirilebilmek mümkündür. Bu noktada Mobil Haritalama Sistemleri (MHS), High Definition (HD) harita üretimi, simülasyon tabanlı senaryo geliştirme ve otonom sürüş teknolojileri gibi çeşitli alanlarda kritik bir rol üstlenmektedir. Dünya genelinde yaygın olarak kullanılan bu sistemler, yol envanterinin oluşturulması ve yönetilmesinde giderek daha önemli hale gelmektedir.

Özellikle nüfus artışı, kentleşme ve ulaşım altyapısındaki değişimler, akıllı şehir teknolojilerinin daha da önem kazanmasına neden olmaktadır. Trafik yoğunluğunu azaltmak, ulaşım güvenliğini artırmak ve çevresel sürdürülebilirliği desteklemek amacıyla geliştirilen çeşitli akıllı ulaşım çözümleri, şehirlerin daha yaşanabilir hale gelmesini sağlamaktadır. Akıllı Şehir konseptinin en kritik bileşenlerinden biri olan Akıllı Ulaşım Sistemleri (AUS), trafik yönetimini optimize ederek ulaşımı daha güvenli, verimli ve sürdürülebilir hale getirmeyi amaçlamaktadır. AUS, yapay zeka, büyük veri analitiği ve nesnelerin interneti gibi yenilikçi teknolojileri kullanarak trafik akışını düzenlemekte, sürücülere anlık trafik bilgileri sunmakta ve trafik kazalarını önleyici önlemler almaktadır. Bu sistemler sayesinde trafik ışıkları ve kavşak yönetimi daha akıllı hale getirilerek şehir içi ulaşımın daha akıcı olması sağlanmakta, gereksiz trafik yoğunluğunu azaltılmaktadır. Otonom araçların başarılı bir şekilde çalışabilmesi için, yalnızca GPS/GNSS verileri yeterli olmamakta, yüksek hassasiyetli harita verilerine de ihtiyaç duyulmaktadır. HD haritalar, otonom araçlara yalnızca coğrafi konum bilgisi sağlamakla kalmayıp, yol eğimleri, kavşaklar, trafik tabelaları ve yaya geçitleri gibi detaylı yol bilgilerini de sunarak, araçların çevresini daha iyi algılamasına olanak tanımaktadır. Geleneksel navigasyon sistemlerine kıyasla santimetre hassasiyetine sahip olan bu haritalar, otonom sürüş güvenliğini artırmakla kalmayıp, otonom taksi ve toplu taşıma hizmetleri gibi yeni nesil ulaşım çözümlerinin de temelini de oluşturmaktadır. Ayrıca ileri seviye sürücü destek sistemleri için hassas yol bilgileri sunmayı amaçlamaktadır. Bu sistemler, şerit takip asistanları, acil fren destek sistemleri ve trafik işareti algılama teknolojileri gibi birçok gelişmiş özelliği bünyesinde barındırmaktadır. HD haritalar ve ADAS teknolojileri birlikte çalışarak, sürücülerin çevresel farkındalığını artırmakta ve kazaların önlenmesine önemli ölçüde katkı sağlamaktadır. HD haritalar, geleneksel haritalardan çok daha yüksek doğruluk sunarak yol yüzeyi detaylarını, şerit bilgilerini, trafik işaretlerini ve yol eğimlerini hassas bir şekilde tanımlayarak, akıllı trafik yönetiminin daha verimli çalışmasına yardımcı olmaktadır. Yol üzerindeki hız limitleri, kavşak yapıları ve yol çalışmaları gibi dinamik veriler de sürekli olarak güncellenerek sürücülerin en güncel bilgilere erişmesini sağlamaktadır. 3D yol envanterleri ise yol altyapısının dijital ikizini oluşturarak, trafik akışı analizleri, bakım-onarım planlamaları ve şehir içi ulaşım optimizasyonu gibi işlemler için kapsamlı bir veri tabanının oluşmasını sağlamaktadır. Bu teknolojiler, akıllı şehirlerde dinamik trafik yönlendirmeleri, güvenli sürüş asistanları ve otonom mobilite çözümlerinin temelini oluşturmaktadır. Akıllı ulaşım sistemlerine verilen önemin artmasıyla birlikte, trafik işaretlerinin otomatik tespit edilmesi üzerine yapılan çalışmalar da artmakta, birçok ülke kendine ait trafik işareti veri setleri oluşturarak bu sistemlerin doğruluğunu ve verimliliğini artırmayı hedeflemektedir.

Bu çalışmada, trafik işaretlerinin otomatik tespit edilebilmesi için MHS'ye dayalı bir yöntem önerilerek, daha önce mevcut olmayan ve ilk kez bu çalışmada oluşturulan Türkiye'ye ait trafik işaretleri veri kümesi hazırlanmıştır. Yapay zeka ile otomatik olarak tespit edilmesi ve tanınması istenilen 10 farklı trafik işaretinin seçimi KGM Trafik İşaretleri El Kitabından yapılmıştır. Çalışma için seçilen trafik işaretlerinin daire, üçgen, sekizgen gibi farklı şekillerde, farklı renklerde olmasına dikkat edilmiştir. Ayrıca trafik işaretlerinin içerisinde yazı ve küçük şekiller bulunması ve tüm Dünya'da kullanılan trafik işaretleri arasında yer alması dikkate alınmaya çalışılmıştır. 2021 yılı Temmuz ayında, yaklaşık 80 km/s hızla seyreden bir araca monte edilen MHS ile Karadeniz Bölgesi'nde bir otoyolun yaklaşık 6 km'lik bölümünde ölçmeler yapılarak veri toplanmıştır. MHS kullanılarak gerçek trafik sahnelerinden görüntüler elde edilmesi çalışmada ortaya konulan modelin

güvenilirliğini arttırmaktadır. Bu çerçevede, trafik işaretlerinin tespiti ve tanınması için mobil araçlar için daha uygun olduğu öngörülen, bu çalışma ile dünyada ilk defa MHS sistemi verileri üzerinde kullanılan nesne tespit algoritmalarından YOLOv4-Tiny kullanılmıştır. Bu algoritma diğer algoritmalara göre basit sinir ağı yapısından dolayı düşük işlem maliyeti sunmaktadır. Donanımsal yapısı açısından gömülü cihazlar için daha uygundur. Ayrıca gerçek zamanlı algılama için diğer yaklaşımlara göre daha iyi bir seçenektir. Geliştirilen yöntemde modelin eğitimi amacıyla, bir kısmı gerçek saha ölçümlerine dayanan MHS kaynaklı görüntülerden, bir kısmı ise açık kaynaklı veri setlerinden temin edilen görüntülerden oluşan bir veri kümesinden yararlanılmıştır. Bu geliştirilen modelin eğitimi sonucunda Ortalama Hassasiyet (*mAP*) değeri %98.18 olarak elde edilmiştir. Elde edilen *mAP* değeri, nesne tespiti modelleri üzerine yapılan diğer çalışmalara göre oldukça yüksektir. Buna ek olarak özellikle trafik işareti gibi karmaşık senaryoların içinde yer alan küçük nesnelerin tespitinde büyük bir başarı oranını göstermektedir. Eğitim süreci tamamlanan model, öncelikle açık kaynaklı görüntüler ile MHS verileri üzerinde test edilmiştir. Ardından, laboratuvar koşullarında sabit bir web kamerası aracılığıyla gerçek zamanlı testler gerçekleştirilmiştir. Elde edilen test bulguları, önerilen yöntemin trafik işaretlerini hızlı ve isabetli biçimde algılayarak sürüş güvenliğine katkı sağlayabileceğini ortaya koymuştur. Böylece trafik işaretlerinin de içinde bulunduğu yol envanterlerinin otomatik olarak tespit edilmesi ve tanınmasına katkı sağlayarak, süreçleri hızlı ve daha az riskli bir hale getirebileceği görülmüştür. Önerilen yöntemin mobil ve otonom sistemler için gerçek hayatta uygulanmasının kolay olması sebebiyle, otonom araçlarda kullanım için uygun olduğu ve akıllı şehir planlamalarında yer alan akıllı ulaşım sistemlerine entegre edilerek katkı sağlayacağı düşünülmektedir.

Çalışmanın giriş bölümünde konuyla ilgili detaylı bir literatür taraması ve çalışmanın amaç ve hedefleri ortaya konulmuştur. İkinci bölümde çalışmada kullanılan yapay zeka teknolojileri ile ilgili detaylı bilgiler paylaşılmıştır. Üçüncü bölümde yapay zeka nesne tespit algoritmalarından YOLOv4-Tiny'nin tarafımızca modellenmesi, eğitilmesi, MHS verileri kullanılarak trafik işareti veri seti üretimi ve model üzerinde yapılan testlerin uygulanması ile ilgili bilgiler verilmiştir. Dördüncü bölümde ise gerçek hayatta karşılaşılabilecek çeşitli senaryolara göre farklı testler yapılarak önerilen yöntemin performansına etkisi olabilecek durumlar tartışılarak değerlendirilmiştir. Sonuç kısmında ise çalışmanın sonuçları ile ilgili bilgiler verilmiş ve gelecekte konuyla ilgili yapılacak benzer çalışmalar için bazı önerilerde bulunulmuştur.



DETERMINATION OF HIGHWAY INVENTORIES WITH OBJECT DETECTION METHOD FOR AUTONOMOUS DRIVING USING 3D POINT CLOUD DATA

SUMMARY

Traffic signs are symbols, signs and markings used to maintain order on highways, guide drivers and pedestrians, increase traffic safety and regulate traffic flow. Studies show that the rate of accidents caused by the lack or inadequacy of traffic signs has increased significantly. In 2023, it was reported that approximately 1.19 million people lose their lives and 17.8 million people are seriously injured in traffic accidents every year. Traffic signs help prevent accidents by warning drivers and pedestrians of potential hazards and facilitate understanding of traffic rules. A road inventory is a comprehensive set of data that includes the physical characteristics of roads, traffic signs, signaling systems and other infrastructure elements. A proper and up-to-date road inventory and the correct location of traffic signs are vital for drivers to travel safely. It is also necessary for traffic management and urban planning to make road maintenance and development processes more efficient. In Turkey, traffic signs and road inventory management are managed by the General Directorate of Highways (KGM), the Ministry of Transport and Infrastructure, municipalities, and the General Directorate of Security and Gendarmerie General Command under the Ministry of Interior. Data for road information, traffic density monitoring, intersection arrangement systems and maintenance and repair needs are mostly collected through field studies. The General Directorate of Highways stores all data in a central database through the “Road Inventory System”. In the world, in addition to manual studies, Geographic Information System (GIS) based data systems, satellite images, automatic data collection systems, drones, LiDAR measurements and artificial intelligence supported systems are used to keep road inventory information systems up to date. Traditional methods for creating a traffic sign and road inventory information system are time-consuming and costly, including field measurements, digitizing and managing the information. In addition, these traditional methods make data generation difficult and carry various risks in terms of occupational safety. The need for rapid data collection due to these risks has led to the emergence of new research areas. In particular, processes such as the preparation, creation and updating of 3D road inventories can be carried out quickly and efficiently without the need to go to the field. At this point, Mobile Mapping Systems (MMS) play a critical role in various fields such as High Definition (HD) map production, simulation-based scenario development and autonomous driving technologies. These systems, which are widely used around the world, are becoming increasingly important in the creation and management of road inventory. Especially population growth, urbanization and changes in transportation infrastructure cause smart city technologies to gain more importance. Various smart transportation solutions developed to reduce traffic density, increase transportation safety and support environmental sustainability make cities more livable. Intelligent Transportation Systems (ITS), one of the most critical

components of the Smart City concept, aims to make transportation safer, more efficient and sustainable by optimizing traffic management. ITS use innovative technologies such as artificial intelligence, big data analytics and the Internet of Things to regulate traffic flow, provide instant traffic information to drivers and take measures to prevent traffic accidents. Thanks to these systems, traffic lights and intersection management are made smarter, making urban transportation smoother and reducing unnecessary traffic congestion. For autonomous vehicles to operate successfully, not only GPS/GNSS data is not enough, but also high-precision map data is needed. HD maps not only provide autonomous vehicles with geolocation information, but also detailed road information such as road gradients, intersections, traffic signs and pedestrian crossings, allowing vehicles to better perceive their surroundings. With centimeter accuracy compared to traditional navigation systems, these maps not only improve autonomous driving safety, but also form the basis of next-generation transportation solutions such as autonomous taxi and public transport services. It also aims to provide precise road information for advanced driver assistance systems. These systems incorporate many advanced features such as lane keeping assistants, emergency braking support systems and traffic sign detection technologies. HD maps and ADAS technologies work together to increase drivers' environmental awareness and contribute significantly to accident prevention. HD maps offer much higher accuracy than traditional maps, helping intelligent traffic management work more efficiently by precisely identifying road surface details, lane information, traffic signs and road gradients. Dynamic data such as speed limits, intersection structures and road works are also continuously updated to ensure that drivers have access to the most up-to-date information. 3D road inventories create a digital twin of the road infrastructure, providing a comprehensive database for traffic flow analysis, maintenance-repair planning and urban transportation optimization. These technologies form the basis of dynamic traffic routing, safe driving assistants and autonomous mobility solutions in smart cities. With the increasing importance given to intelligent transportation systems, studies on automatic detection of traffic signs are also increasing, and many countries aim to increase the accuracy and efficiency of these systems by creating their own traffic sign data sets.

In this study, a method based on MHS for automatic detection of traffic signs is proposed and a dataset of traffic signs of Turkey, which is not available before and created for the first time in this study, is prepared. The selection of 10 different traffic signs to be automatically detected and recognized by artificial intelligence was made from KGM Traffic Signs Manual. The traffic signs selected for the study should have different shapes such as circles, triangles, octagons and different colors. In addition, it was tried to take into account that the traffic signs contain text and small shapes and are among the traffic signs used all over the world. In July 2021, data was collected by measuring a 6 km section of a highway in the Black Sea Region with a MHS mounted on a vehicle traveling at a speed of approximately 80 km/h. Obtaining images from real traffic scenes using the MHS increases the reliability of the model presented in the study. In this framework, YOLOv4-Tiny, one of the object detection algorithms, which is predicted to be more suitable for mobile vehicles for the detection and recognition of traffic signs, was used for the first time in the world on MHS system data. This algorithm offers low computational cost due to its simple neural network structure compared to other algorithms. It is more suitable for embedded devices in terms of hardware structure. It is also a better option for real-time detection than other approaches. In the proposed method, a dataset consisting partly of images taken with MHS based on realistic field measurements and partly of images obtained from open

datasets was used to train the model. As a result of the training of this developed model, the Average Precision (mAP) value was obtained as 98.18%. The obtained mAP value is quite high compared to other studies on object detection models. In addition, it shows a great success rate especially in the detection of small objects in complex scenarios such as traffic signs. The trained model was first tested on existing open source images and MHS images. Then, real-time tests were conducted in a laboratory environment using a simple fixed webcam. The test results show that the proposed method can improve driving safety by detecting traffic signs quickly and accurately. Thus, it can contribute to the automatic detection and recognition of road inventories, including traffic signs, making the processes faster and less risky. Since the proposed method is easy to implement in real life for mobile and autonomous systems, it is thought that it is suitable for use in autonomous vehicles and will contribute to smart transportation systems in smart city planning.

In the introduction section of the study, a detailed literature review on the subject and the aims and objectives of the study are presented. In the second section, detailed information about the artificial intelligence technologies used in the study is shared. In the third section, the modeling and training of YOLOv4-Tiny, one of the artificial intelligence object detection algorithms, the generation of a traffic sign dataset using MHS data, and the implementation of tests on the model are presented. In the fourth section, the performance of the proposed method is discussed and evaluated by performing different tests according to various scenarios that can be encountered in real life. In the conclusion, the results of the study are presented and some suggestions are made for future similar studies on the subject.



1. GİRİŞ

Dünya nüfusu her geçen gün artmaktadır ve bu nüfusun yarısından fazlasının kentlerde yaşadığı bilinmektedir. Hızlı nüfus artışı ile enerji, barınma, sağlık, ulaşım gibi ihtiyaçların karşılanması giderek zorlaşmaktadır. Büyük kentlerde bu sorunların çözümü daha fazla baskı oluşturmaktadır ve kaynaklar gittikçe azalmaktadır. Yaşanan bu sorunların üstesinden gelmek için teknolojiye yararlanmak gerekmektedir. Sorunların çözülmesinde akıllı şehir planlamaları büyük önem teşkil etmektedir. Bu nedenle, özellikle büyük şehirlerde (metropollerde) güvenlik ve sağlık gibi faktörler göz önünde bulundurularak sürdürülebilir, etkili ve yaşanabilir sistemler inşa etmek gerekmektedir. Bu sorunlara çözüm bulmak ve vatandaşlara daha iyi hizmet verebilmek için kentin tüm kaynaklarının etkin ve sürdürülebilir kullanımı amacıyla “Akıllı Şehir” kavramı ortaya çıkmıştır. Dünyada büyük ve orta ölçekli pek çok şehirde, 10, 20, 30 yıllık akıllı şehir planları yapılmaktadır. Akıllı şehir planlarında, dünyanın birçok yerinde uygulama safhasına geçilmiştir. Avrupa Birliği'nin de tanımlamış olduğu gibi, akıllı şehrin bileşenlerini akıllı hareketlilik, akıllı yönetim, akıllı ekonomi, akıllı çevre, akıllı yaşam ve akıllı insan oluşturmaktadır. Akıllı şehirlerin temel amaçlarından biri, farklı sistemleri ve alt sistemleri kendi aralarında birbirine bağlayarak yaşam kalitesini artırmak, sürdürülebilirlik, enerji tasarrufu sağlamak ve karbon ayak izini azaltmaktır. Türkiye Cumhuriyeti Çevre ve Şehircilik Bakanlığı tarafından 2019 yılında yürürlüğe konulan '2020-2023 Ulusal Akıllı Şehirler Stratejisi ve Eylem Planı'nda, akıllı şehir kavramının; şehirde mevcut ve gelecekte ortaya çıkabilecek ihtiyaçları ve sorunları tüm kentsel alanlarda ve sistemlerde itici bir güce dönüştürmek, fiziksel, sosyal ve dijital planlamayı entegre şekilde değerlendirmek, karşılaşılan zorlukları öngörüp tanımlayarak sistematik, esnek ve sürdürülebilir yöntemlerle çözmek, şehirdeki kurumsal yapılar arasında iş birliğini güçlendirerek entegre hizmet sunumu ve yenilik üretme kapasitesini artırmak olduğu ifade edilmiştir. Plansız kentleşmenin neden olduğu en büyük sorunlardan birisi de ulaşımdır. Özellikle metropollerde günümüzde de yaşanmakta olan ulaşım sorunlarının çözümü için, akıllı kentin bir bileşeni olan Akıllı Ulaşım Sistemlerinden (AUS) yararlanıldığı görülmektedir. Akıllı Ulaşım sistemleri sayesinde, yol

planlamalarının yapılması, var olan yolların en verimli şekilde kullanılması, seyahat sürelerinin azaltılması, ulaşım güvenliğinin sağlanması, ulaşım araçları arasında geçişin kolaylaştırılması amaçlanmaktadır. Ayrıca akıllı ulaşım ile enerji verimliliği elde edilerek ülke ekonomisine katkı sağlanması, çevreye verilen zararın en aza indirilmesi mümkündür. Akıllı bir ulaşımın olması için, akıllı altyapının oluşturulması gereklidir. Akıllı Altyapı; Akıllı Çevre, Akıllı Ulaşım ve İletişim Teknolojileri bileşenleri kapsamında kullanılan sensörlerle toplanan veriyi ileten, analiz eden, ölçen, izleyen ve daha gelişmiş performans ve kullanıcı deneyimi için kullanıcı talepleri ve çevredeki değişikliklere akıllı şekilde yanıt verebilen ve kamusal değer oluşturan sistemlerdir [1]. Akıllı Şehirler Stratejisi Eylem Planı'nda 15.14 numaralı eylemde, ilgili eylem ve politikalar bölümünde, "2.2.1.2.5. Otomotiv" başlığı altında "385. Yeni nesil araçlar için uygun altyapı oluşturulacaktır.", "385.1. Alternatif güç sistemli araçlara yönelik etkin bir altyapı kurulmasına ilişkin araştırma yapılacaktır.", "385.2. Otonom ve AKILLI ŞEHİRLER STRATEJİSİ VE EYLEM PLANI 539 / 658 bağlantılı araçların geliştirilmesi ile kullanılmasına ilişkin teknik mevzuat ve altyapı ihtiyacı belirlenecektir." şeklinde akıllı ulaşım altyapısına verilen önem görülmektedir. T.C. Çevre, Şehircilik ve İklim Değişikliği Bakanlığı tarafından, Ulusal Akıllı Şehir Açık Veri Platformu (ULASAV) kurulmuştur. Ülkemizin başarılı üniversitelerinden biri olan, Gebze Teknik Üniversitesi'nin Akıllı Şehir ve Ulaşım Teknolojileri Yüksek Lisans Programı ile akıllı şehir ve ulaşım teknolojileri alanında, sektör uygulama ihtiyaçlarına yönelik ve yenilikçi araştırma alanlarında lisansüstü eğitim verilmesi planlanmaktadır. Yine T.C. Çevre, Şehircilik ve İklim Değişikliği Bakanlığı tarafından akıllı şehir faaliyetlerine yön vermek üzere 2024-2030 Ulusal Akıllı Şehirler Stratejisi ve Eylem Planı hazırlanmaktadır. Ülkemizdeki akıllı şehir faaliyetlerine yön vermek üzere "İleri toplum, ileri teknoloji, sürdürülebilir ve dirençli dünya" vizyonu ile hazırlanan 2024-2030 Ulusal Akıllı Şehirler Stratejisi ve Eylem Planına ilişkin çalışmalarda, 9 Ocak 2024 tarihinde merkezi kamu kurumları ve üniversitelerden 115 kişi, 10 Ocak 2024 tarihinde yerel yönetimler ve özel sektörlerden 106 kişi katılım sağlamıştır. Etkinlik kapsamında, çalışmada yer alan 30 eylem, 271 uygulama adımı ve yeni eylem önerilerine yer verilmiştir [2]. Bütün bu gelişmeler akıllı şehrin önemini vurgulamaktadır. Amerika Birleşik Devletleri Ulusal Karayolu Trafik Güvenliği İdaresi (NHTSA) yoldaki ciddi kazaların %94'ünün insan hatasından kaynaklandığını bildirmiştir [3]. Bu bağlamda, otonom araç konsepti, başta kazaları azaltmak ve çok daha güvenli bir ulaşım için büyük bir potansiyele sahiptir. Trafik

altyapısının durumu ile ilgili güncel bilgiler yerel yönetimler ve trafik büroları için büyük öneme sahiptir. Yol işaretleri, özellikle geçiş durumlarında trafik akışının düzenlenmesi için gerekmektedir. Bu nedenle, yol işaretleri hakkındaki bilgileri verimli bir şekilde elde etmeye ve ayrıntılı yol haritalarının otomatik olarak oluşturulmasına ihtiyaç duyulmaktadır. Otonom araçların beklenen şekilde çalışması günümüzde kullanılan haritalar ve trafik verileri ile yetersiz kalmaktadır. Geleneksel haritalar insanların okuması içindir. İnsanlar, çevrelerini sürekli olarak değerlendirir, olası riskli durumlarda anlık olarak tepki verebilirler. Otonom arabalar insanlarla aynı yeteneklere sahip olmadığından, sürüş anında riski en aza indirecek ve olası risk anında zararı en aza indirecek şekilde uyum sağlamalarını destekleyecek haritalara ihtiyaçları vardır. Otonom araçların hareket kabiliyetlerini geliştirmek için mevcut verilerden farklı olarak işlenmiş ve gelişmiş yol, trafik ve trafik işaret verileri gerekmektedir. Yol şeridinin ve trafik işaretlerinin tanınması, araç yönlendirmesi için çok önemlidir. Yol güvenliği denetimlerinin sağlıklı bir şekilde gerçekleştirilebilmesi için, yol üzerindeki ve çevresindeki tüm nesnelerin geometrik özelliklerinin elde edilmesi gereklidir. LiDAR olarak bilinen mobil lazer tarama teknolojisinin, yol koridorları boyunca yüksek yoğunlukta nokta bulutları üretme konusunda oldukça etkili olduğu ortaya konmuştur. Mobil Haritalama Sistemine entegre edilmiş lazer tarayıcılarla toplanan veriler, hem yol çevresinin detaylı geometrik yapısını elde etmek hem de farklı nesnelerin tespiti için güçlü bir temel sunmaktadır. Mobil haritalama taramaları, gerekli bilgiler için yerleşik bir veri kaynağıdır. LiDAR aracılığıyla ölçülen 3D nokta bulutları, yol işaretlerinin yerleştirilmesi için önemli yoğunluk verilerini ve yol görüntülerini içermektedir ve kentsel bölgelerde kolayca yakalanabilmektedir. Yoğunluk değeri, tarama yapan lazerin yansımalarının gücünü temsil ederek yüzeyin yapısı veya malzemesi hakkında sonuçlara varılmasını sağlamaktadır. Bu sebeple yapılan çalışmada kullanılan MHS aracı ile yol envanterlerinden trafik işaretlerinin tespitini sağlayarak, çalışma kapsamında ülkemize ait trafik veri seti oluşturulması hedeflenmektedir. Ayrıca bu çalışma HD haritalama sisteminin geliştirilmesinde de otonom araçların yaygınlaşmasında bir alt taban oluşturacak, akıllı şehir akıllı ulaşım planlamalarına katkı sağlayacaktır. Bu çalışmanın içeriğinde kullanılan veri seti genişletilerek, diğer yol envanterlerini kapsayan bir hale getirilmesi mümkündür.

1.1 Tezin Amacı

Kısa mesafeli ulaşımda en yaygın kullanılan altyapılar otoyollar ve yollardır. Bu altyapılarda güvenli ulaşımın sağlanması kritik bir öneme sahiptir. Türkiye karayolu ağı ise toplamda 68.689 kilometre uzunluğa sahiptir. [4]. Türkiye İstatistik Kurumu'nun (TÜİK) verilerine göre Türkiye'de 2022 yılında ölümlü yaralanmalı trafik kazasına neden olan toplam 234.688 kusuruna bakıldığında, kusurların %69.4'ünün sürücü kaynaklı olduğu görülmüştür. Kaza istatistiklerine göre, sürücü kaynaklı kazaların %29'u trafik işaretleri etkisi ile gerçekleşmiştir [5]. Söz konusu güvenliğin akıllı ulaşım sistemleri ile sağlanması için, otoyolların dijital olarak izlenebilmesi gerekmektedir. Otoyolların dijital olarak izlenebilmesi için farklı teknikler ve teknolojiler bulunmaktadır. Bu teknik ve teknolojik gelişmeler etkililik ve uygulanabilirliğe bağlı olarak, genellikle uydu görüntüleri, hava görüntüleri ve MHS çözümlerine dayanmaktadır.

Otonom araçların güvenli seyahat gerçekleştirebilmesi için trafik işaretlerini hızlı ve doğru bir biçimde algılaması gereklidir. Bu işaretler, araçların trafik kurallarına uygun hareket etmesini, rotalarını etkili şekilde planlamasını ve trafik akışını düzenli biçimde yönetmesini sağlar. Özellikle hız limitleri, yol koşulları ve riskli bölgeler gibi kritik bilgileri içeren trafik işaretleri hem araç güvenliği hem de çevresel güvenlik açısından büyük önem taşır. Dolayısıyla, otonom sistemlerin trafik işaretlerini doğru şekilde tanıyabilme kabiliyeti, trafik düzeninin sürdürülebilirliği açısından temel bir unsurdur. Mobil Haritalama Sistemleri sayesinde, özel donanım ve yazılımlar kullanılarak yolların ve çevresinin gerçek zamanlı, ayrıntılı üç boyutlu haritaları üretilebilmektedir. Bu haritalar; trafik yönetimi, otonom araçlar, acil durum hizmetleri ve yol bakımı gibi bir dizi uygulama için önemli veriler sağlamaktadır. MHS aracılığıyla, yol durumu, trafik yoğunluğu, altyapı özellikleri ve çevresel etkenler gibi veriler sürekli güncellenmektedir. Akıllı ulaşım sistemleri ise bu dinamik verileri kullanarak daha güvenli ve verimli bir seyahat deneyimi sunmaktadır. Bu teknolojik gelişmeler öncesinde, trafik işaretlerinin konumu, türü ve durumu hakkındaki bilgiler trafik denetçileri tarafından yerinde gözlem yapılarak manuel olarak belirlenmekteydi [6], [7]. Bu yöntemle yapılan çalışmalar, süreci zaman alıcı ve insan hatalarına duyarlı kılmaktaydı. Ayrıca, trafik işaretlerinin sayısı arttıkça verilerin güncel ve doğru olması, bakım sorunları ve erişim zorlukları ortaya çıkmıştır. Trafik işaretlerinin tanımlanması ve tespitini otomatikleştirmek için yapılan ilk çalışmalar, işaretlerin

renkleri, şekilleri ve bazı temel kurallar kullanılarak belirli kriterlere göre tanımlanmasını içermektedir. Ayrıca, şablon eşleştirme ve radyal simetri gibi basit yöntemler trafik işaretlerini algılamak için kullanılmıştır. Fakat bu yöntemler, işaretlerin artan karmaşıklığı ve çeşitliliği karşısında bazı kısıtlamalarla karşılaşmıştır. Ayrıca, trafik işaretlerinin algılanması ve tanımlanmasını zorlaştırabilecek birtakım etkenler mevcuttur; bunlar arasında işaretlerdeki renk solmaları, araçların hareket hâlinde olması ve olumsuz hava şartları sayılabilir. Özellikle kırmızı renge sahip trafik işaretleri, uzun süre güneş ışığına maruz kaldığında zamanla solabilmektedir. Hava kirliliği ve hava koşulları (yağmur, kar, sis, gölge ve bulut) trafik işaretlerinin görünürlüğünü azaltabilmekte, gündüzden geceye değişen dış aydınlatma koşulları trafik işaretlerinin renklerini etkileyebilmektedir. Araçlar, yayalar ve diğer trafik işaretleri gibi engeller, trafik işaretlerini kısmen kapatabilmektedir [8]. Bu durumlar, araştırmacıları trafik işaretlerinin algılanması ve veri toplama süreçlerinin daha hızlı, hassas ve etkili bir şekilde yapılması konusunda farklı teknolojilere yapay zeka ve otomasyon teknolojilerine yöneltmiştir.

Trafik işaretlerinin sınıflandırılması için geleneksel bilgisayarla görme ve makine öğrenimi tabanlı yöntemler yaygın olarak kullanılmıştır [8], [9]. Bu tip araştırmaların yapılabilmesinde veri setleri büyük rol oynamaktadır. GTSRB [10] trafik işareti veri seti, Çin trafik işareti veri seti ve Tsinghua-Tencent 100 K(TT100K) veri seti dışında diğer bir çok ülkenin de kendine ait trafik işaretleri veri setleri mevcuttur. Fakat Türkiye'ye ait kapsamlı bir veri seti bulunmamaktadır. Bu nedenle ilk olarak Türkiye'ye ait örnek trafik işaretleri veri seti oluşturulmuş ve çalışma kapsamında kullanılan veriler bu veri seti üzerinden temin edilmiştir.

Bu çalışmada, otonom bir aracın çevresindeki trafik işaretlerini daha hassas ve hızlı bir şekilde algılayarak gerçek bir yolda ilerlemesini sağlamak için, mobil araçlarda kullanımı daha uygun olan YOLOv4-Tiny algoritması kullanılarak trafik işaretlerinin tespit edilmesi amaçlanmıştır. Literatürde MHS ile toplanan gerçek verilerin kullanılması ile YOLOv4-Tiny algoritması kullanılarak yapılan öncü bir çalışmadır. Trafik işaretleri tespit edilmekte ve aynı zamanda tespit edilen trafik işareti hakkında kullanıcıya bilgi verilmektedir. YOLOv4-Tiny tabanlı yöntem, görüntüleri ön işleme tabi tutmadan özel veri kümesi üzerinde tespit ve tanımlama gerçekleştirmektedir. Ayrıca, bu öneri, hareketli bir platform (genellikle bir araç) üzerine entegre edilmiş sensörler, kameralar, LiDAR tarayıcılar, GNSS (Küresel Navigasyon Uydu Sistemi) alıcıları ve atalet ölçüm birimleri (IMU) gibi çeşitli veri toplama bileşenlerinden

oluşan, MHS ile toplanan verilerin de kullanılmasıyla daha da güçlendirilmiştir. MHS ile toplanan verilerden, bu çalışma için oluşturulan veri setinin benzeri bulunmadığı için, bu çalışma ile Türkiye trafik işaretleri veri seti de oluşturulmuştur. Bu entegre yaklaşım, trafik işaretlerinin tespitini geliştirerek otonom araçların hem güvenliğini hem de verimliliğini artırmayı amaçlamaktadır. Bununla birlikte, önerilen yöntemin kullanım alanı yalnızca trafik işaretleriyle sınırlı kalmamaktadır. Sunulan yöntem, akıllı şehirlerin planlanması ve otonom ulaşım çözümleri açısından da yüksek uyumluluk sergilemektedir. Sonuç olarak, bu çalışma hem çokgen biçimli hem de çokgen dışı trafik işaretlerinin sınırlarını etkili bir şekilde öngörmeyi hedeflemektedir.

1.2 Literatür Araştırması

Trafik işaretlerini belirlemek ve tanımak amacıyla geliştirilen yaklaşımlar; renge dayalı, şekil odaklı, makine öğrenmesi temelli ve derin öğrenme destekli olmak üzere dört ana kategoriye ayrılabilir. Bazı araştırmacılar [11], [12], trafik işaretini tespit etme ve renk eşikleme yöntemi ile sınıflandırma yapmak için, renk uzayını RGB'den HSV'ye dönüştürerek, destek vektör makinası üzerinde çalışmışlardır. Trafik işaretlerinin tespiti ve sınıflandırılması için Bahlmann ve diğ. [13] renk, şekil, hareket bilgisi ve haar dalgacık tabanlı özellikler kullanmışlardır. Redmon ve Farhadi'nin [14] ve Yuan ve diğ.'nin [15] şekil tabanlı yöntemi kullandıkları çalışmalarında, görüntüdeki çizgileri ve daireleri bulmak için Hough dönüşümü kullanılmıştır. Görüntüdeki çizgilerin oluşturduğu kapalı alanlara dayanarak trafik işaretlerinin konumları belirlenmiştir. Ancak bu süreçte kullanılan Hough dönüşüm tekniği, yüksek hesaplama maliyeti ve uzun işlem süresi nedeniyle karmaşıktır. Bu durum, yöntemin gerçek zamanlı trafik işareti tespiti için uygun olmamasına neden olmaktadır. Bu nedenle, renk ve şekil temelli trafik işareti algılama yaklaşımlarının bazı kısıtları söz konusudur. Öte yandan, makine öğrenmesi temelli tekniklerde ise sınıflandırıcılar genellikle görüntüden elde edilen özniteliklere dayanmaktadır. Saadna ve diğ. [16] hız sınırı işaretlerinin gerçek zamanlı tespiti ve doğru bir şekilde tanınması için bilgisayarla görme sistemi önermişlerdir. Bu sistemde sınıflandırıcı olarak iki doğrusal destek vektör makinesinden oluşan basamaklı bir mimari kullanılmıştır. Chen ve diğ. [17] AdaBoost ve destek vektör regresyonunu (SVR) ayırt edici dedektör öğrenimi için keşfederek doğru ve verimli bir trafik işareti algılama tekniği önermişlerdir. Makine öğrenmesine dayalı yaklaşımlar genellikle dönme, kayma,

ölçek deęiřimi ve biçim bozulması gibi etkenlerden etkilenmeyen bazı öznitelikleri (örneğin Haar, HOG gibi) kullanmaktadır. Bu öznitelikler, daha sonra önceden eğitilmiş sınıflandırıcılarla bir araya getirilerek trafik işaretlerinin sınıflandırılması sağlanmaktadır. Yaygın olarak kullanılan sınıflandırıcı destek vektör makinesidir (SVM), çünkü sınıflandırma görevlerini yerine getirirken hızlı ve yüksek sınıflandırma doğruluęu gibi avantajları bulunmaktadır. Zeng ve dię. [18] derin algısal özelliklere sahip çekirdek tabanlı bir aşırı öğrenen makine sınıflandırıcısı kullanarak bir trafik işareti tanıma yöntemi önermişlerdir. Makine öğrenimine dayalı geleneksel trafik işareti algılama yöntemleri renk ve şekillere dayalı yöntemlerden daha iyi sonuçlar elde etmiş olsa da bu makine öğrenimi yöntemlerinin HOG özelliklerini veya Harr özelliklerini çıkarması gerekmektedir. Bu durum, hesaplama işlemlerini daha da karmaşıklştırmaktadır. Bununla birlikte, kullanılan özniteliklerin ifade gücünün sınırlı olması, bu yöntemlerin karmaşık durumlar karşısında yeterli başarıyı göstermesini engellemektedir. Bu sebeple, bazı arařtırmacılar trafik işaretlerinin algılanması ve tanınması konusundaki çalışmalarda derin öğrenme tekniklerine yönelmişlerdir. Song ve dię. [15] derin öğrenme yöntemlerinin hesaplama maliyeti üzerine bir çalışma yapmak için model boyutu %10 azaltılmış bir CNN aęını, Tsinghua-Tencent 100K trafik işareti veri seti üzerinde denemişlerdir. Büyük trafik sahnelerinde küçük trafik işaretlerinin tespit edilebilmesi, akıllı cihazların güvenliğini artırmaktadır. Fakat küçük nesne tespiti bilgisayarla görmede zorlu bir problemdir. Doğru trafik işareti tespitindeki ana sorun işaretlerin küçük boyutlu olmasıdır. Liu ve dię. [16] yapmış olduęu çalışmada, bu sorunla başa çıkmak için bir dekonvolüsyon bölgesi tabanlı konvolüsyonel sinir aęı (DR-CNN) önerilmiştir. Önerilen yöntem MS COCO ve Tsinghua-Tencent 100K veriseti kümeleri üzerinde deęerlendirilmiş, Faster R-CNN yöntemine göre daha iyi sonuç verdięi görülmüřtür. Bayouhd ve dię. [17] önermiş oldukları derin öğrenme tabanlı sürüş destek sisteminde, transfer öğrenme paradigmasına dayalı hibrit 2D-3D CNN modelleri sunulmuřtur. Önerilen yöntemin performansının açık veri seti olan GTSRB ve KITTI üzerinde umut verici olduęu görülmüřtür. Lopez-Montiel ve dię. [18] otonom araçlarda trafik işareti algılama sorununda uygun algoritma ve donanım seçimi konusunda bir çalışma yapmışlardır. Standartlaştırılmış bir veri kümesinde (LISA) trafik işareti tespiti için Tek Atış Çoklu Kutu Dedektörü (SSD) algoritması ve Özellik Piramit Aęı (FPN) bileşeninin bir kombinasyonunda bazı derin öğrenme modellerinin kombinasyonlarını (MobileNet v1 ve ResNet50 v1) uygulamışlardır. Bu uygulamalarını farklı donanım mimarileri (CPU,

GPU, TPU ve Gömülü Sistem) üzerinde test etmişlerdir. Redmon ve diğ.[19] tarafından 2016 yılında literatüre, nesne tespiti alanına “You Only Look Once (YOLO)” adlı yeni bir yöntem kazandırılmıştır. YOLO, nesne konumlarını belirlemek için karmaşık olmayan bir işlem hattı kullanan ve bu süreci regresyon problemi olarak ele alan tutarlı bir mimari yapıya sahiptir. Bu sayede oldukça hızlı bir şekilde işlemin gerçekleşmesini sağlamaktadır. Wang ve Cheng-Yue [20] tarafından yapılan çalışmada YOLO’nun, Faster R-CNN’den yaklaşık altı kat daha hızlı olduğu ortaya konulmuştur. Hasegawa ve diğ. [21], Japonya'daki karmaşık trafik sahneleri için daha iyi bir trafik işareti tanıma yöntemi önermişlerdir. Deney sürecinde giriş modeli olarak YOLOv2 tercih edilmiş ve bu model, beş farklı görüntü boyutunda denenmiştir. Redman ve Farhadi [22]’nin gerçekleştirdiği YOLOv3 çalışması, gerçek zamanlı nesne tespiti alanında özellikle küçük boyuttaki nesnelerin tanınmasında kayda değer bir gelişme sağlamıştır. Liu ve diğ. [23] Tam Evrişimli Ağlar (FCN) yaklaşımından ilham alınarak optimize edilmiş yeni bir YOLO modeli geliştirilmiş ve bu süreçte çeşitli YOLO kombinasyonları kullanılmıştır. Bu birleşik modeller, özellikle trafik yoğunluğu yüksek sahnelerde daha hassas nesne tespiti yapılmasını mümkün kılmaktadır. VOC07 ve VOC12 veri setlerinde ortalama %69.3 doğruluk oranına ulaşılırken, klasik YOLO modeliyle bu oran yalnızca %64 civarında kalmıştır. Bochkovskiy ve diğ. [24] tarafından YOLOv4, 2020'de önerilmiştir. Günümüzdeki birçok bilimsel modelin eğitimi, genellikle büyük mini batch boyutları gerektirdiği için birden fazla GPU kullanımını zorunlu kılmaktadır. Ancak, yalnızca bir GPU ile gerçekleştirilen eğitim süreçleri çoğunlukla yavaşlamakta, verimsiz ve ağır bir hâl almaktadır. YOLOv4 ise bu soruna, daha küçük mini batch boyutlarıyla tek bir GPU üzerinde eğitilebilen bir nesne algılama modeli geliştirerek çözüm sunmaktadır.

Literatürde yapılan çalışmalar, trafik işaretlerinin tespiti ve tanınması konusunda farklı yöntemlerin geliştirilmesine odaklanmıştır. Yapılan çalışmaların, genellikle gerçek trafik sahnelerinden uzak, küçük boyutlu ve karmaşık olmayan trafik işareti görüntüleri üzerinde olduğu gözlemlenmiştir. Renk tabanlı ve şekil tabanlı yöntemlerin yaygın kullanıldığı bu çalışmalarda, özellikle Hough dönüşümü gibi ön işleme dayalı teknikler, hesaplama süreçlerini oldukça karmaşık ve zaman alıcı hale getirmiştir. Bu yöntemlerin gerçek zamanlı uygulamalarda sınırlı performans sergilediği görülmüştür. Makine öğrenimi tabanlı yöntemler, trafik işaretlerinin tespiti ve tanınmasında önemli bir ilerleme sağlamıştır. Bu yöntemler, genellikle görüntü

özelliklerinin çıkarılması ve sınıflandırılması sürecine dayanmaktadır. Ancak bu özelliklerin sınırlı ifade gücü ve karmaşık senaryolara karşı düşük performansı, derin öğrenme tabanlı yaklaşımların geliştirilmesine zemin hazırlamıştır. Derin öğrenme tabanlı yöntemler arasında CNN ve türevleri, karmaşık trafik sahnelerinde yüksek doğruluk sağlamış; küçük trafik işaretlerinin tespiti gibi zorlayıcı problemleri çözmeye önemli bir rol oynamıştır. YOLO serisi, Redmon ve Farhadi [22] tarafından başlatılan yenilikçi bir nesne tespit yaklaşımıyla literatürde dikkat çekmektedir. YOLO, hızlı ve etkili bir şekilde çalışarak diğer yöntemlere kıyasla daha düşük işlem karmaşıklığı ve daha yüksek performans sunmuştur. Örneğin, YOLOv3, küçük nesnelerin algılanmasında önemli bir ilerleme sağlamıştır. YOLOv4 ise daha küçük mini parti boyutlarıyla tek GPU üzerinde eğitilebilen bir sistem sunarak, modern derin öğrenme modellerinin eğitimine yönelik önemli bir çözüm geliştirmiştir. Buna rağmen, bu yöntemlerin bazı çalışmalarda sadece %64 doğruluk seviyesine ulaşabildiği görülmüştür. Gezgin ve Alkan [25] tarafından yapılan çalışmada mobil ve otonom araçlar için YOLOv4-Tiny algoritması ile nesne tespiti ve tanımlanmasının yüksek doğrulukla yapılabildiği görülmüştür. Bu bağlamda, önerilen çalışmada, YOLOv4-Tiny ve MHS aracı verileri birleştirilerek karmaşık olmayan, ön işlem gerektirmeyen ve kısa hesaplama süreleri sunan bir sistem tasarlanmıştır. Bu sistem, her boyuttaki görüntüyü işleyebilecek şekilde eğitilmiş, oluşturulan veri seti üzerinde gerçek ve karmaşık trafik sahnelerinde %98 doğrulukla trafik işaretlerini tespit etmeyi başarmıştır. Bu sonuçlar, YOLO'nun farklı versiyonlarındaki gelişmelerin, gerçek zamanlı nesne algılama ve trafik işareti tespitinde önemli bir potansiyel taşıdığını göstermektedir.



2. METODOLOJİ

Yol geometrisi, trafik işaret ve levhaları, yol yüzey durumu, köprüler, tüneller, kavşaklar, aydınlatma sistemleri ve yol kenarı güvenlik ekipmanları gibi birçok unsur yol envanteri kapsamına girmektedir. Yolun uzunluğu, genişliği, eğimi, şerit sayısı, yol kaplama türü (asfalt, beton v.b.), yol aydınlatmaları, trafik işaretleri gibi bilgileri içeren yol envanter bilgisi, yol ağının etkin bir şekilde planlanması, bakımı ve güvenliğinin sağlanması için kritik veri kaynaklarıdır. Yol envanteri bilgileri, sadece mevcut yolların yönetimi için değil, aynı zamanda yeni ulaşım projelerinin planlanması, otonom araç teknolojileri, HD harita üretimi ve akıllı şehir uygulamaları için de büyük önem taşımaktadır. Yol envanter bilgilerinin doğru bir şekilde toplanması ve veri tabanında depolanması gerekmektedir. Bu bilgilerin toplanması için farklı yöntemler kullanılmaktadır: Teknolojinin gelişmesi ile yol envanteri bilgilerinin daha hızlı, güvenli, maliyeti az bir şekilde toplanabilmesi için de yeni yöntemler ortaya çıkmıştır.

2.1 Yol Envanterleri Toplama Yöntemleri

Yol envanteri verilerinin toplanması, yol altyapısının yönetimi ve bakım süreçlerinin optimize edilmesi açısından önemli bir adımdır. Bu doğrultuda başvurulan yöntemler arasında; görsel kontroller, taşınabilir ve mobil cihazlar, daha gelişmiş teknolojik sistemler ile görüntü işleme tabanlı algoritmalar yer almaktadır. Günümüz teknolojisi ile, bu yöntemlerin daha etkin ve maliyet etkin bir şekilde kullanılması mümkün hale gelmiştir. Genel olarak yol envanteri bilgisi toplama yöntemleri 3 ana gruba ayrılabilir. Bunlar:

- i-) Yol envanterinin manuel olarak toplanması,
- ii-) Mobil Haritalama Sistemi (MHS) ile yol envanterinin toplanması,
- iii-) Yapay zeka yöntemleri ile yol envanterinin toplanması.

Bu yöntemler kısaca aşağıda açıklanmıştır.

i-) Yol envanterinin manuel olarak toplanması: Yol envanteri bilgileri, ulaşım altyapısının etkin yönetimi ve güvenliğin artırılması için kritik bir rol oynamaktadır. Bu envanter bilgilerini toplarken genellikle GNSS entegreli saha veri kaydedicileri ve geleneksel tarama yöntemleri kullanılarak manuel veri toplama yöntemlerine başvurulmaktadır. Bu yöntemin oldukça doğru sonuçlar verdiği, ancak zaman alıcı olması ve yoğun insan kaynağı gerektirmesi nedeniyle operasyonel maliyetlerin arttığı belirtilmiştir. Bu durum, özellikle büyük ölçekli yol ağlarının yönetiminde ciddi bir zaman ve iş gücü ihtiyacını beraberinde getirmektedir [26]. Yapılan araştırmalarda, envanter veri toplama sürecinin bir envanter yönetim sistemi oluşturulmasında en çok zaman alan adımlardan biri olduğu vurgulanmıştır. Yol ağının büyüklüğüne bağlı olarak kaynakların ve zamanın büyük bir kısmının saha çalışmalarına harcanması gerekliliği, manuel yöntemlerin sürdürülebilirliğini azaltmaktadır. Ayrıca, veri toplama işlemi sırasında oluşabilecek insan hataları, envanter verilerinin doğruluğunu ve güncelliğini etkileyebilir [27]. Teknolojinin gelişmesinden önce, trafik işaretlerinin yeri, türü ve durumu gibi bilgiler, trafik görevlileri tarafından sahada yapılan doğrudan gözlemlerle manuel şekilde tespit edilmekteydi [6], [7]. Bu tür yöntemler, düşük yoğunluklu yol ağları için yeterli olsa da modern şehirlerde trafik işaretlerinin sayısının artmasıyla bakım zorlukları, güncel ve doğru veri ihtiyacı gibi sorunları beraberinde getirmiştir.

ii-) Mobil Haritalama Sistemi (MHS) ile yol envanterinin toplanması: Mobil Haritalama Sistemleri (MHS), saha veri toplama süreçlerini hızlandırmak ve doğruluğu artırmak için kullanılan yenilikçi bir teknolojidir. MHS, bir mobil platforma entegre edilmiş haritalama sensörlerinden oluşan bir sistemi tanımlamakta ve jeo-uzamsal verilerin toplanması sırasında platformun konumunun belirlenmesine olanak tanımaktadır [28]. Bu sistemler, yol altyapısı ve çevresi hakkında detaylı veri toplamak için özellikle uygundur ve günümüzde yaygın olarak kullanılmaktadır. MHS aracı bir yol üzerinde sürülür ve veriler sürekli olarak toplanır. Bu ham veriler daha sonra uygun işleme yazılım paketleri kullanılarak bir bilgisayarda işlenir ve bir envanter veri tabanına girilir [29]. MHS veri toplama sürecini büyük ölçüde hızlandırır da elde edilen verilerden gerekli bilgileri çıkarmak için çok fazla zaman ve kaynak gerekmektedir. Bu kaynakların çoğu özelliklerin sayısallaştırılması ve öznel detaylarının girilmesi için harcanmaktadır. Veri sayısallaştırma sırasında ofis işlerinin yoğunluğu, mobil haritalama verilerinden otomatik ve yarı otomatik özellik çıkarma

yöntemlerine yönelik arařtırmaların önemini artırmıřtır. Mobil haritalama verileri kullanarak, yol ve çevresindeki öğelerin (örneğin yol merkez çizgileri, sınırlar, řeritler, iřaretlemeler vb.) otomatik ya da yarı otomatik olarak tanımlanabilmesine yönelik çeřitli yöntemler geliřtirilmiřtir [30]. David ve dię. [31] tarafından yapılan çalışmada düzenli yol bakımına yönelik ihtiyaç duyulan yol envanterinin tespiti ve yeniden oluřturulması üzerine yoğunlařılmıřtır. Bu arařtırmada yayılan aęaç yapısı temel alınmıř olup, bu yaklařımın zorlu çevre kořullarında bile yol iřaretlerinin güvenilir şekilde belirlenmesini ve sınırlarının doęru biçimde yeniden modellenmesini saęladığı belirtilmiřtir. Makalede önerilen yöntem, yol řeritlerini tanımlamak için ortak bir vektör katmanı oluřturmak üzere LiDAR ölçümlerinin çıktıısı olan nokta bulutu verilerini kullanmaktadır. Vektör katmanı, çoęu coęrafi bilgi sistemi tarafından desteklenen ortak bir dosya formatında saklanmakta ve yol envanteri süreçlerine dayalı olarak karar vermede kullanılabilir bir çıktı üretmektedir. Solian ve dię. [32] LYNX MHS aracı ile elde edilen nokta bulutu ve görüntü verilerinden trafik iřaretlerini otomatik olarak tespit etmek ve sınıflandırmak için bir yöntem geliřtirmiřtir. Düşük çözünürlüklü nokta bulutu verilerinin doęruluęunu artırmak için, ilgili RGB görüntüleri analiz etmiř ve bu verilerle trafik iřaretlerinin tanımlanmasını saęlamıřtır. Yansıtılan noktalarla belirlenen sınırlayıcı kutu içerisindeki RGB verileri üzerinde analiz gerçekleştirilmiřtir. Bu analiz sonucunda %95'in üzerinde bir geri çağırma oranı ve %90'ın üzerinde bir F-skoru elde edilmiřtir. MHS, yalnızca yol envanteri varlıklarının tespitinde deęil, aynı zamanda yol kusurlarının belirlenmesinde ve CBS tabanlı veri tabanlarının oluřturulmasında da kritik bir role sahiptir. Özel yazılım ve donanımlar aracılıęıyla, MHS sayesinde yollar ile çevrelerinin detaylı üç boyutlu haritaları anlık olarak üretilebilmektedir. Bu haritalar, yalnızca yol bakım iřlemleri için deęil, aynı zamanda otonom araç navigasyonu, trafik yönetimi ve acil durum hizmetleri gibi birçok uygulama için kullanılabilir. Mobil Haritalama Sistemleri, yol altyapısının detaylı bir şekilde haritalanması ve envanter bilgilerinin güncellenmesi için güçlü bir araçtır. Yolların ve çevrelerinin 3D haritalarının oluřturulması, yalnızca trafik güvenlięi ve yol bakımı için deęil, aynı zamanda otonom araç teknolojilerinin geliřtirilmesi için de kritik bir veri saęlar. Teknolojinin sürekli ilerlemesiyle birlikte MHS'nin daha etkin ve otomatik bir yapıya kavuřması, onu akıllı ulařım sistemlerinin kilit unsurlarından biri haline getirmektedir.

iii-) Yapay zeka yöntemleri ile yol envanterinin çıkarılması: Yol envanterlerinin otomatik olarak tanımlanması ve tespit edilmesine yönelik ilk girişimler, trafik işaretlerini renk, şekil ve temel kurallara dayalı belirli özelliklerle tanımaya odaklanmıştır. Ancak işaretlerin artan çeşitliliği ve karmaşık yapıları, bu yaklaşımların etkinliğini sınırlamıştır. Özellikle, trafik levhalarının renklerinin zamanla solması, araçların hareket hâlinde olması ve olumsuz hava koşulları gibi etkenler, işaretlerin algılanmasını zorlaştırabilmektedir. Kırmızı renkli levhalar güneş ışığına uzun süre maruz kaldığında solarken; yağmur, sis, kar, hava kirliliği ve gölgeler de görünürlüğü önemli ölçüde azaltabilir. Ayrıca, gece aydınlatması da trafik işaretlerinin rengini etkileyebilir. Park edilmiş araçlar, yayalar ve diğer olası engeller trafik işaretlerini kısmen gizleyebilir [8]. Bu durum, araştırmacıları trafik işaretlerinin tespitini ve veri toplama süreçlerini daha hızlı, daha hassas ve daha etkili hale getirmek için farklı teknolojilere, yapay zeka ve otomasyon teknolojilerine yönlendirmiştir. Güncel çalışmalar, özellikle Evrişimsel Sinir Ağları (CNN) tabanlı derin öğrenme yaklaşımlarının etkili biçimde kullanılmasıyla trafik işaretlerinin algılanmasında kayda değer gelişmeler sağlandığını ortaya koymaktadır. Çeşitli iyileştirici stratejiler ve mimarilerle desteklenen bu yaklaşım, çeşitli kıyaslama veri kümeleri üzerinde doğruluk ve verimlilikte önemli gelişmeler sağlamıştır. Özellikle, trafik işareti sınıflandırmasında geleneksel makine öğrenmesi teknikleri ve bilgisayarla görme tabanlı yöntemler yaygın şekilde tercih edilmektedir. [8], [9]. Böyle araştırmaların yürütülmesinde veri kümeleri çok büyük bir rol oynamaktadır. GTSRB [10], Çin ve Tsinghua-Tencent 100 K (TT100K) trafik işareti veri setinin yanı sıra, diğer birçok ülkenin kendilerine ait trafik işareti veri kümeleri bulunmaktadır. Hienonen ve arkadaşları [33] otomatik trafik işareti tespiti ve sınıflandırmasını, trafik işareti envanteri oluşturma ve durum değerlendirmesiyle bütünleştiren yapay görme temelli bir yöntem önermiştir. Araştırmada, verilerin toplanması aşamasında yüksek çözünürlüklü bir kamera, iyi aydınlatma koşullarında kullanılmıştır. Araştırmacılar, yöntemin başarımının artırılabilmesi için düşük kaliteli işaretlere ait daha fazla görüntüye ihtiyaç duyulduğunu vurgulamışlardır. Wang ve arkadaşları [34] bilgi kaybını azaltmak ve temsil kabiliyetini artırmak amacıyla gerçek zamanlı trafik sinyali tespiti için AF-FPN adı verilen bir özellik piramidi modeli önermiştir. Bu makalede ayrıca otomatik bir veri artırma yöntemi de önerilmiştir. Bu tez çalışmasında ayrıca otomatik bir veri artırma yöntemi de önerilmiştir. Sirigineedi ve diğ. [35] tarafından yapılan çalışmada Hindistan'da trafik işareti tanıma için derin öğrenme kullanan

otonom bir sistem tanıtılmaktadır. Alman Trafik İşareti veri seti kullanılarak yapılan çalışma YOLOv5 algoritmasına dayanmaktadır. Daha Hızlı Bölge Tabanlı Evrişimsel Sinir Ağı (Faster R-CNN), Maske R-CNN ve YOLOv5 algoritmaları birlikte kullanılarak yaklaşık %97.80 doğruluk elde edilmiştir. Cui ve arkadaşları [36] YOLO ağ yapısından Conv3 modülünü çıkararak, özellik çıkarım katmanına 152×152 boyutlarında bir özellik füzyon modülü entegre etmiştir. TT100K veri kümesi üzerinde yapılan çalışmada %93.88 *mAP* değeri elde edilmiştir. Yang ve diğ. [37] çalışmalarında Faster R-CNN, R-FCN, SSD ve YOLO gibi farklı algoritmaların performansını karşılaştırmalı olarak analiz etmişlerdir. Microsoft COCO ve German veri kümeleri üzerinde yapılan karşılaştırmalarda Faster R-CNN %95.77 *mAP* değeri ile en yüksek değere ulaşmıştır.

Yukarıda ana hatlarıyla verilen 3 bu yöntemin genel bir değerlendirmesi ve her birinin güçlü ve zayıf yönleri Çizelge 2.1’de verilmiştir.

Çizelge 2.1 : Yol envanteri bilgisi toplama yöntemleri.

Yöntemler	Avantajlar	Dezavantajlar
1 Yol envanterinin manuel olarak toplanması	Konum ve envanter türü bilgilerine ek olarak, veri toplama yöntemleri görünür kusurların belirlenmesine olanak tanmaktadır. Özellikle yol işaretlerinin gece koşullarında algılanması zor olduğundan, elde taşınabilir mobil retro-reflektometre cihazları kullanılmaktadır.	Yerel olarak sınırlı bilgi kapsamı. Tüm yol ağı verilerini toplamak için çok uzun süreler gereklidir. Ölçümün gerçekleştirilmesi için şerit veya yolun kapatılması gereklidir. Bu yöntem önemli ölçüde insan kaynağı gerektirir; görevlinin her bir trafik işaretinin ayrı ayrı analiz edilmesi zorunludur.

Çizelge 2.1 (devamı) : Yol envanteri bilgisi toplama yöntemleri.

Yöntemler	Avantajlar	Dezavantajlar
2 Mobil haritalama sistemi ile yol envanterinin toplanması	Kamera ve GNSS gibi teknolojiler kullanılarak otomatik olarak toplanabilmekte, çeşitli algoritmalar ve yazılım paketleri ile manuel yöntemlere kıyasla daha hızlı bir değerlendirme yapılabilmektedir.	Başlangıçta yatırım maliyeti yüksek olsa da, uzun vadeli toplam maliyet manuel yöntemlere kıyasla daha düşüktür. Elde edilen veri miktarının fazlalığı, özellikle nokta bulutu gibi veri türlerinin işlenmesinde özel yazılım çözümleri ve yetkin personel gereksinimini beraberinde getirmektedir.
3 Yapay zeka yöntemleri ile yol envanterinin toplanması	Mobil haritalama sistemleri veya daha önce toplanmış veri setleri yapay zeka algoritmaları ve görüntü işleme teknikleri ile otomatik olarak işlenebilir.	Veri kümelerinde bulunan görüntülerin çözünürlüğü iyi, algoritmanın hızı ve doğruluğu yüksek olmalıdır.

2.2 Teknik Arka Plan

2.2.1 Derin öğrenme

Derin Öğrenme, yapay zekanın bir alt dalı olarak, insan müdahalesine gerek kalmadan veriye dayalı kararlar verebilen ve öğrenme süreçlerini gerçekleştirebilen sinir ağı modellerinin geliştirilmesiyle ilgilenmektedir [38]. Bu teknoloji, büyük veri setlerinden otomatik olarak anlamlı bilgiyi çıkarma yeteneğine sahiptir ve karmaşık yapıları kavramlar hiyerarşisi içinde temsil ederek dünyayı anlamlandırabilir. Bilgisayar bilgisi toplama konusunda kendi kendine yeterli olduğundan, bilgisayar için gerekli bilgiyi belirlemek için insan müdahalesine gerek olmamaktadır. Derin öğrenmeyi özel kılan en önemli unsurlardan biri, özellik çıkarma işleminin doğrudan öğrenme sürecine entegre edilmesidir. Özellik çıkarma, modele sağlanan giriş verilerinden anlamlı bilgilerin otomatik olarak elde edilmesini ifade eden bir kavramdır. Geleneksel makine öğrenimi yöntemlerinde, uzmanların problemi anlaması ve özellikleri manuel olarak belirlemesi gerekmektedir. Ancak derin öğrenme, bu süreci otomatikleştirerek çok daha karmaşık veri yapılarında bile etkin

sonuçlar elde etmektedir [39]. Bir görüntü tanıma problemi ele alındığında, derin öğrenme algoritmaları, görüntünün piksel tabanlı yapısından başlayarak, daha soyut ve üst düzey özellikler öğrenebilmektedir. Bu, sinir ağlarının katmanlı yapısı sayesinde mümkündür; her katman daha yüksek soyutlama seviyelerinde özellikler çıkarmaktadır. Her görüntü piksellerden oluşmaktadır ve bu pikseller komşu piksellerle ilişkilidir. Derin öğrenme yöntemleri, geniş bir görüntünün küçük parçalarından anlamlı öznitelikler elde ederek bu avantajı kullanmaktadır. Bu yaklaşımlar, karmaşık ve soyut özellikleri öğrenebilme kapasitesine sahiptir. Böylece, her yeni problem için ayrı bir özellik çıkarım mekanizması tasarlama gereksinimi ortadan kalkmaktadır. Bu katmanlı öğrenme süreci, her yeni problem için bir özellik çıkarıcı geliştirme gerekliliğini ortadan kaldırarak, modelin genelleştirme kabiliyetini arttırmaktadır. Derin öğrenme modelleri, çok sayıda parametre içermesi ve genellikle büyük veri setleri ile eğitilmesi nedeniyle yüksek işlem gücü gerektirmektedir. Bu nedenle, derin öğrenme modellerinin eğitimi, geleneksel makine öğrenimi algoritmalarına kıyasla daha fazla zaman alabilir. Bununla birlikte, güçlü donanımların (örneğin, GPU'lar ve TPU'lar) ve paralel işlem tekniklerinin kullanımı ile eğitim süresini önemli ölçüde azaltabilir [40]. Modelin karmaşıklığı, özellikle hiperparametre ayarlama süreçlerini ve model optimizasyonunu zorlu hale getirmektedir. Bununla birlikte, modern derin öğrenme çerçeveleri (örneğin, TensorFlow ve PyTorch) bu zorlukları yönetmek için geniş bir araç seti sunarak ve araştırmacılara daha verimli bir geliştirme ortamı sağlamaktadır [41].

2.2.2 Sinir ağı

Sinir ağları, biyolojik sinir sistemlerinden esinlenerek tasarlanmış, bilgi işleme ve öğrenme kapasitesine sahip sistemlerdir. Biyolojik nöronlardan ilham alan bu yapılar, öğrenmeyi uygulamak ve bilgiyi hafızaya almak için çok katmanlı bir mimariyi kullanmaktadır. Yapay zeka (YZ) ile ilgili sorunları çözmek için tasarlandıklarında, bu ağlara Yapay Sinir Ağları (YSA) adı verilmektedir [42]. YSA, yapay nöronlar olarak adlandırılan bilgi işleme birimlerinden oluşmakta ve öğrenme sürecini optimize etmek için katmanlı bir yapı ile çalışmaktadır. Bir YSA, genellikle üç temel bileşenden oluşmaktadır. Bunlar her girişin, bir nöronun giriş değerini temsil ettiği giriş katmanı, girdiyi işlemek için çeşitli algoritmaları ve aktivasyon fonksiyonlarını kullanan, ağın karmaşık özellikler öğrenebilmesini sağlayan gizli katmanlar ve işlenen bilginin nihai bir çıktıya dönüştüğü çıkış katmanı olarak özetlenebilir.

2.2.3 Evrişimsel sinir ağı

Konvolüsyon ağları (CNN veya ConvNet), bilinen ızgara benzeri bir topolojiye sahip verileri işleyebilen sinir ağları ailesine aittir [43]. Bu tezde kullanılan Evrişimsel Sinir Ağları (CNN'ler), özellikle görüntü işleme ve bilgisayarla görme alanlarında büyük başarılar elde etmiş bir YSA türüdür. CNN'ler, görüntülerdeki mekansal ilişkileri ve düzenleri öğrenmek için tasarlanmışlardır [44]. Görüntüler söz konusu olduğunda, 2 boyutlu bir piksel ızgarası bulunmaktadır. CNN bir girdi görüntüsünü alarak, görüntüdeki çeşitli yönlere/nesnelere ağırlıklar ve önyargılar ekleyerek ve bir görüntüyü diğerinden ayırt ederek özellik katmaktadır [45]. Geleneksel sinir ağlarından farklı olarak CNN'ler, verilerdeki yerel bağıntıları anlamak için özel katmanlar içermektedirler. CNN bir girdi katmanı ve bir çıktı katmanı ile bunların arasında yer alan çoklu evrişimsel, havuzlama veya tam bağlantılı katmanlardan oluşmaktadır. CNN'ler, görüntülerden düşük seviyeli özelliklerden (kenarlar vb.) başlayarak yüksek seviyeli özelliklere (nesne sınıfları vb.) kadar anlamlı bilgi çıkarabilmektedir [46]. Bu katmanlı yapı, CNN'lerin hem doğruluk oranını arttırmakta hem de genelleştirme kabiliyetini güçlendirmektedir. CNN, katmanlarında matris çarpımı yerine basitçe konvolüsyon adı verilen matematiksel bir işlem kullanılmaktadır. Konvolüsyon işlemi matematiksel olarak Eşitlik 2.1'de gösterildiği şekilde ifade edilebilir:

$$s(t) = (x * w)(t) \quad (2.1)$$

Burada x giriş değişkeni, yani görüntü pikselleri, w çekirdek veya piksellere uygulanan filtre, t ise uzamsal konum değişkenini temsil etmektedir. Çekirdek, görüntüden daha küçük boyutlu bir matristir. Bu işlem, özellik haritası adı verilen bir çıktı elde etmek için çekirdeği görüntü piksellerinin ilgili komşuluğuna uygulamaktadır. Bir katmanın çıktısı diğer katmana girdi olarak gönderilir. Bu şekilde, görüntü pikselleri için konvolüsyonlar gerçekleşmekte ve modelin özellikleri çıkardığı karşılık gelen özellik haritaları oluşturulmaktadır.

2.2.3.1 Evrişimsel katman

Bir CNN, giriş görüntülerinden özellikler çıkarmak için kullanılan birkaç evrişimsel katmandan oluşmaktadır. CNN'ler, girdilerdeki mekansal ilişkileri anlamak ve görsel desenleri öğrenmek için bu katmanları temel almaktadır. Evrişimsel katman, çekirdek adı verilen küçük öğrenilebilir filtrelerin görüntü piksellerine uygulanmasıyla çalışır. Çekirdek matrisi üzerinde çapraz korelasyon gerçekleştirildikten sonra görüntü piksellerine uygulanmaktadır. Sonuç olarak bir özellik haritası üretilmektedir. CNN, giriş görüntüsünden özellikler çıkarmak için birkaç evrişimsel katman kullanılmaktadır. Evrişimsel işlemin ilk adımı olarak, giriş görüntüsü, her biri farklı boyutlara ve ağırlıklara sahip çekirdekler ile işlenmektedir. Çapraz korelasyon gerçekleştirilerek, çekirdek görüntünün komşu piksellerine kaydırmalı bir şekilde uygulanması ile belirli bir bölgedeki özelliklerin yoğunluğunu temsil eden bir sonuç elde edilmektedir. Bir sonraki adımda ağırlıkların öğrenilmesi gerçekleştirilmektedir. Çekirdekteki ağırlıklar eğitim sırasında öğrenilmekte ve eğitim tamamlandıktan sonra ağırlıklar sabitlenerek ve test sırasında kullanılmaktadır [46]. Konvolüsyon eşlemesinden sonra çıktı, gradyan problemini ele almak için doğrusal olmayan aktivasyon fonksiyonlarına gönderilmektedir. Bu, modelin karmaşık ve doğrusal olmayan ilişkileri öğrenmesine olanak tanımaktadır. Bu süreçte elde edilen çıktıya özellik haritası (feature map) denilmektedir. Özellik haritası, giriş görüntüsündeki desenleri ve ilişkileri temsil ederek, sonraki katmanlarda daha yüksek seviyeli özelliklerin öğrenilmesi için kullanılmaktadır.

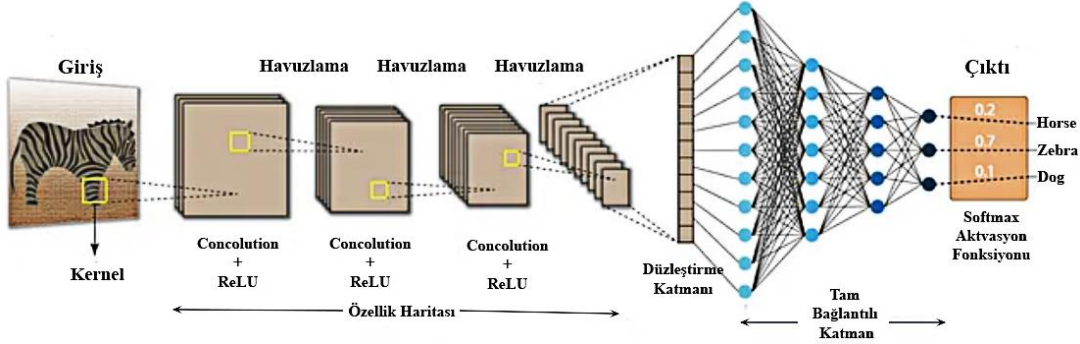
2.2.3.2 Havuzlama

Havuzlama, bir pencere içindeki değerlere istatistiksel fonksiyonların uygulandığı bir kayan pencere tekniğidir. Maksimum havuzlama, pencere içindeki maksimum değer alınacağı yaygın olarak kullanılan işlev olarak tanımlanabilir. Havuzlama, alt örnekleme işlemi için, temel olarak parametre sayısını azaltmak ve özellik tespitini daha sağlam hale getirmek için kullanılmaktadır.

2.2.3.3 Tam bağlantılı katman

Tam bağlı veya yoğun katman, havuzlama katmanından elde edilen çıktıdan nesnelere sınıflandırılmasını yapmaktadır. CNN mimarisinin son birkaç katmanını oluşturmaktadır. Önceki katmandan gelen giriş verisi düzleştirilmektedir. Düzleştirilmiş vektör daha sonra genellikle matematiksel işlemlerin gerçekleştirildiği

birkaç tam bağlantılı katmanından daha geçmektedir böylece sınıflandırma süreci gerçekleşmeye başlamaktadır. Şekil 2.1'de bu katmanın sinir ağı üzerindeki görevi görülmektedir.



Şekil 2.1 : Evrişimsel sinir ağı ve katmanlar.

2.2.3.4 Bırakma katmanı

Tüm özellikler tam bağlı katmana bağlandığında, eğitim veri kümesinde aşırı uyum sağma sorunu yaşanabilir. Aşırı uyum, belirli bir modelin eğitim verilerinde o kadar iyi çalışması ve yeni verilerde kullanıldığında modelin performansında düşüş yaşanarak olumsuz bir etkiye sebep olması durumudur. Bu sorunu aşmak için, eğitim süreci sırasında sinir ağından birkaç nöronun düşürülmesiyle sonuçlanan bir düşürme katmanı kullanılmaktadır ve bu da modelin boyutunun küçülmesine neden olmaktadır. 0.3'lük bir bırakma geçildiğinde, düğümlerin %30'u sinir ağından rastgele düşürülür. Bırakma katmanı, böylece aşırı uyum sorununu azaltmaktadır ve sinir ağlarının performansını arttırmaktadır.

2.2.4 Aktivasyon fonksiyonları

Aktivasyon fonksiyonları YSA'da giriş verilerini çıkış verilerine dönüştürmek için kullanılmaktadır ve bu veriler bir sonraki katmana giriş olarak gönderilmektedir. Bir sinir ağı modelinin tahmin doğruluğu, kullanılan aktivasyon fonksiyonunun türüne bağlıdır [2]. Aktivasyon fonksiyonları kullanılmazsa, çıktı verileri basit bir doğrusal polinom fonksiyonu olacaktır. Model, verilerden karmaşık eşlemeleri öğrenemeyecek ve tanıyamayacaktır. Bu nedenle sinir ağı görüntü, konuşma, metin, video, ses vb. gibi karmaşık veri türlerinin modellenmesinde etkili olmayacaktır. ReLU ve Softmax bu tezde kullandığımız iki aktivasyon fonksiyonudur.

2.2.4.1 ReLU

ReLU (Rectified Linear Unit) aktivasyon fonksiyonu, derin öğrenme modellerinde yaygın olarak kullanılan bir aktivasyon fonksiyonudur. ReLU fonksiyonu, negatif girdileri sıfır olarak döndürürken, pozitif girdileri doğrudan aktive etmektedir. Bu özelliği özellikle CNN'lerde ve diğer derin sinir ağlarında tercih edilmesini sağlamaktadır.

ReLU aktivasyon fonksiyonu matematiksel olarak Eşitlik 2.2'de gösterildiği şekildedir:

$$f(x) = \max(0, x) \quad (2.2)$$

Burada x fonksiyon girdisi olarak tanımlanmaktadır. Bu fonksiyonun türevi girdi pozitif olduğunda 1, negatif olduğunda ise 0'dır. Bu basit formülasyonu ve hesaplama kolaylığı, ReLU'nun tercih edilme sebeplerinden biridir. Ayrıca, ReLU'nun düzgün olmayan bazı aktivasyon fonksiyonlarına (sigmoid veya tanh) göre daha hızlı ve etkin bir şekilde öğrenme sağladığı da gözlemlenmiştir [47].

2.2.4.2 Softmax

Softmax aktivasyon fonksiyonu, genellikle çoklu sınıflı sınıflandırma problemlerinde kullanılan bir aktivasyon fonksiyonudur. Softmax fonksiyonu, bir sinir ağının çıktılarını olasılık dağılımı olarak yorumlamamızı olanak vermektedir, yani her sınıf için ayrı bir olasılık değeri üretmektedir.

Softmax aktivasyon fonksiyonu matematiksel olarak Eşitlik 2.3'te gösterildiği şekildedir:

$$\text{softmax}(z)_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}} \quad (2.3)$$

Burada, z vektörü sinir ağının çıktılarını, K sınıf sayısını, i indeksi çıktı vektöründeki belirli bir elemanı, e üssel fonksiyonu temsil etmektedir. Softmax fonksiyonunun temel özelliği, çıktı vektöründeki her elemanı 0 ile 1 arasında bir değere dönüştürmesi ve tüm elemanların toplamının 1 olmasını sağlamasıdır. Bu sayede çıktılar olasılık dağılımını temsil etmektedir ve en yüksek olasılığa sahip sınıf tahmin edilmektedir [38], [48], [49].

2.2.5 Optimizasyon algoritması

Optimizasyon algoritmaları sinir ağlarındaki modelin kaybını (loss) minimize etmek için model parametrelerini güncellemektedir. Kayıpların azalması ile sonucun doğruluğu artmaktadır.

2.2.5.1 Adam

Adaptif Moment Tahmini, Adam optimizasyon algoritması, derin öğrenme modellerinde sıkça kullanılan ve gradient inişin geliştirilmiş bir versiyonu olan bir optimizasyon algoritmasıdır. Adam, gradient inişin momentum ve adaptif öğrenme oranı gibi parametrelerini otomatik olarak ayarlayarak eğitim sürecini hızlandırmayı ve stabiliteyi artırmayı amaçlamaktadır. Adam algoritması, her bir model parametresi için ayrı ayrı bir adaptif öğrenme oranı tutmaktadır. Bu, her parametrenin aynı hızda güncellenmesine gerek olmadığı durumlarda faydalı olabilmektedir. Ayrıca, Adam momentumu da hesaba katarak gradyanların geçmişini takip etmektedir ve böylece eğitim sürecini daha hızlı yapmaktadır. Bu optimizasyon yönteminin uygulanması kolaydır ve daha az bellek gerektirmektedir. Bu algoritma, stokastik amaç fonksiyonlarının gradyan tabanlı optimizasyonu için verimli bir algoritmadır [43], [50].

2.2.6 Loss fonksiyonu

Loss fonksiyonu, bir makine öğrenimi modelinin tahminlerinin gerçek değerlerden ne kadar uzak olduğunu ölçen bir fonksiyondur. Bu fonksiyonun değeri, modelin ne kadar iyi veya kötü performans gösterdiğini belirlemektedir. Genellikle, modelin öğrenmesini yönlendirmek ve modelin tahminlerini gerçek değerlere yaklaştırmak için loss fonksiyonu kullanılmaktadır. Bir modelin başarısını ölçmek için kullanılan loss fonksiyonları, problem türüne ve modelin çıktısına bağlı olarak değişiklik gösterebilmektedir [43], [51].

2.2.7 Bilgisayarla görü

Bilgisayarla görü (computer vision) ile, bilgisayarların dijital görüntüler üzerinde işlem yapabilmesi ve bu işlem sonucunda görüntülerden anlamlı bilgiler çıkartabilmesi amaçlanmaktadır. Bu alandaki temel amaç, bilgisayarların insan gözü gibi görüntüleri algılayabilmesini ve yorumlayabilmesini sağlamaktır. Bu amaç genellikle dijital görüntü işleme tekniklerini kullanarak gerçekleştirilmektedir. Bu teknikler,

görüntüleri işleyerek kenarlar, şekiller, renkler ve desenler gibi özellikleri tanımlamayı ve anlamayı amaçlamaktadır. Bu sayede, nesne tanıma, yüz tanıma, hareket algılama, görüntü sınıflandırma ve benzeri görevler gerçekleştirilebilmektedir.

Bilgisayarla görü alanındaki çalışmalar, derin öğrenme yöntemlerinin gelişmesiyle büyük bir ivme kazanmıştır. Özellikle derin evrişimli sinir ağları (CNN'ler), bilgisayarla görü alanında çığır açan başarılar elde edilmesini sağlamıştır. Bu ağlar, görüntü işleme problemlerinde insanların performansını hatta bazen geçebilecek düzeyde sonuçlar elde etmeye imkân tanımıştır [47], [52].

2.2.8 Görüntü işleme

Görüntü işleme, dijital görüntüler üzerinde bilgisayar algoritmaları kullanarak bilgi çıkarma veya işlem yapma sürecidir. Bu algoritma ve teknikler, dijital görüntülerden anlamlı bilgiler çıkarmak, görüntüleri iyileştirmek, analiz etmek, tanımak veya anlamak gibi çeşitli amaçlar için kullanılabilir. Örneğin, tıbbi görüntüleme, otomotiv endüstrisi, güvenlik ve gözetim sistemleri, robotik, tarım, eğlence ve daha birçok alanda görüntü işleme teknikleri kullanılmaktadır. Görüntü işleme genellikle şu ana başlıklar altında incelenmektedir: Düşük seviye görüntü işleme: Bu, görüntüleri daha temel işlemlerle işleme sürecidir. Örneğin, görüntü boyutunu değiştirme, parlaklık ve kontrast ayarlama, kenar tespiti gibi işlemler bu kategoriye girmektedir. Orta seviye görüntü işleme: Bu, görüntülerdeki nesnelere ve özelliklere tanıma yönelik işlemleri içermektedir. Örneğin, nesne tanıma, yüz tanıma, hareket algılama gibi işlemler bu kategoriye dahil edilebilir. Yüksek seviye görüntü işleme: Bu, görüntülerdeki bilgiyi analiz ederek anlamaya çalışmaktadır. Örneğin, görüntü içeriğinin anlaşılması, anlamsal bölütleme gibi işlemler bu kategoriye girmektedir [49], [52].

2.2.9 Deney ortamı

2.2.9.1 Python

Python, 1990'ların sonlarında geliştirilmeye başlanmış, yüksek seviyeli, genel amaçlı bir programlama dilidir. Genellikle okunması kolay ve anlaşılır sözdizimi ile öne çıkmaktadır. Python, çok çeşitli platformlarda çalışabilmekte ve nesne yönelimli, işlevsel ve prosedürel programlama gibi birçok farklı programlama modelini desteklemektedir. Python'un temel avantajlarından biri de zengin ve yerleşik

kütüphanelere sahip olmasıdır. Bu kütüphaneler, dosya işleme, ağ programlama, veri işleme, veritabanı yönetimi, GUI programlama ve daha pek çok alanda kullanılabilir işlevler içermektedir. Python, Veri Bilimi, Makine Öğrenimi ve Yapay Zeka çalışmalarında yaygın kullanılan bir dildir. Python'un geniş üçüncü taraf kütüphane ve araç ekosistemine örnek olarak NumPy, pandas, TensorFlow, Django, Flask gibi kütüphaneler verilebilir. Bu tezde kullanılan python kütüphaneleri aşağıdaki gibi açıklanmıştır [53]–[55].

Numpy

NumPy, Python programlama dili ile sayısal hesaplamayı mümkün kılan açık kaynaklı bir projedir. 2005 yılında Numeric ve Numarray kütüphanelerinin ilk çalışmaları temel alınarak oluşturulmuştur [56].

Keras

Keras'ın amacı, Makine Öğrenimi destekli uygulamalar göndermek isteyen tüm geliştiricilere hata ayıklama hızı, kod zarafeti ve kısalığı, sürdürülebilirlik ve dağıtılabilirliğe odaklanarak avantaj sağlamaktır. Keras daha küçük, daha okunabilir ve üzerinde yineleme yapma kolaylığı olan kod tabanı sağlamaktadır. Keras, sinir ağları oluşturmak için Google tarafından geliştirilmiş yüksek seviyeli bir derin öğrenme API'sidir. Keras, tutarlı ve basit API'ler sunarak, yaygın kullanım durumları için gereken kullanıcı eylemlerinin sayısını en aza indirmektedir. Keras, JAX, TensorFlow ve PyTorch ile çalışmaktadır. Çerçeve sınırlarını aşabilen ve bu çerçevelerin üçünün de ekosisteminden yararlanabilen modeller oluşturmaya olanak tanımaktadır [57].

Matplotlib

Matplotlib, nörobiyolog John Hunter tarafından EEG verileriyle çalışmak üzere oluşturulmuştur. Daha sonra birçok farklı alanda birçok kişi tarafından kullanılmaya ve geliştirilmeye başlanmıştır. Matplotlib Python'da statik, animasyonlu ve etkileşimli görselleştirmeler oluşturmak için kapsamlı bir kütüphanedir. Matplotlib ile kaliteli grafikler oluşturulabilir ve yakınlaştırma, kaydırma ve güncelleme yapabilen etkileşimli şekiller oluşturulabilmektedir. Oluşturulan grafik ve şekiller birçok dosya formatına aktarılabilir [58].

Open CV/CV2

OpenCV (Open Source Computer Vision Library), açık kaynak kodlu bir bilgisayarla görme ve makine öğrenimi kütüphanesidir. Bu kütüphane, bilgisayarla görme uygulamaları için ortak bir altyapı sunmayı ve makine algısının ticari ürünlerde kullanımını hızlandırmayı amaçlamaktadır. OpenCV, hem geleneksel hem de güncel bilgisayarla görme ve makine öğrenimi algoritmalarını içeren 2500'den fazla optimize edilmiş algoritma sunmaktadır. Bu algoritmalar insan yüzleri tespit etmek ve tanımak, nesnelere tanımlamak, videolardaki insan eylemlerini kategorilendirmek, kamera hareketlerini izlemek, hareketli nesnelere izlemek, nesnelere 3D modellerini çıkarmak, stereo kameralardan 3D nokta bulutları üretmek, tüm bir sahnenin yüksek çözünürlüklü bir görüntüsünü üretmek için görüntüleri birleştirmek, bir görüntü veritabanından benzer görüntüleri bulmak, fotoğraflardan kırmızı gözleri çıkarmak, göz hareketlerini takip etmek, manzarayı tanımak ve artırılmış gerçeklikle kaplamak için işaretçiler oluşturmak vb. için kullanılabilir. Google, Yahoo, Microsoft, Intel, IBM, Sony, Honda, Toyota gibi köklü şirketler OpenCV'yi kapsamlı bir şekilde kullanmaktadır. OpenCV'nin yaygın kullanım alanları, sokak görüntülerini bir araya getirmek, Türkiye'de pistlerde enkaz olup olmadığını kontrol etmek, İsrail'de güvenlik videolarındaki izinsiz girişleri tespit etmek, Willow Garage'da robotların gezinmesine ve nesnelere toplamasına yardımcı olmak, Çin'de maden ekipmanlarını izlemek, Avrupa'da yüzme havuzunda boğulma kazalarını tespit etmek, İspanya ve New York'ta interaktif sanatı yürütmek, Japonya'da hızlı yüz algılamak ve dünyanın dört bir yanındaki fabrikalarda ürünlerin üzerindeki etiketleri incelemek olarak örneklendirilebilir. C++, Python, Java ve MATLAB arayüzlerine sahiptir ve Windows, Linux, Android ve Mac OS işletim sistemlerini desteklemektedir. OpenCV ile çoğunlukla gerçek zamanlı görüntü uygulamalarına yönelik çalışmalar yapılmaktadır. OpenCV yerel olarak C++ dilinde yazılmıştır ve STL konteynerleri ile sorunsuz çalışan şablonlu bir arayüze sahiptir [59].

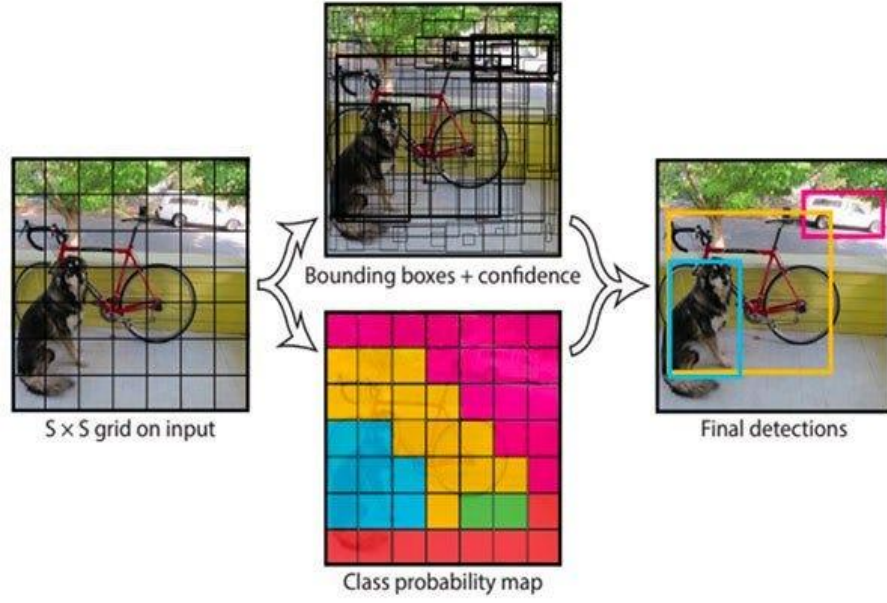


3. YOL ENVANTERİ TESPİTİ İÇİN ÇALIŞMADA UYGULANMAK ÜZERE ÖNERİLEN YÖNTEM

3.1 Önerilen Algoritma YOLOv4-Tiny

YOLO nesne algılama yöntemi ve versiyonları, kamu güvenliği için yüz algılama [60], uzay uygulamaları [61], sağlık uygulamaları [62] ve akıllı şehir uygulamaları [63] gibi birçok farklı alanda yaygın olarak kullanılmaktadır. 2016 yılında Redmon ve diğ. [19] You Only Look Once (YOLO) adlı yeni nesne tanıma yöntemi tanıtılmıştır. YOLO, nesne tespitini bir regresyon problemi olarak ele alan ve kutuları tespit etmek için oldukça sade bir işlem hattı kullanan tutarlı bir mimariye sahiptir. Bu yapı, işlemi son derece hızlı bir hale getirmektedir. Hasegawa ve diğ. [21], Japonya'nın karmaşık trafik koşullarına uygun, gelişmiş bir trafik işareti tanıma sistemi önerilmiştir. Çalışmada giriş verisi olarak YOLOv2 modeli kullanılmış ve bu model beş farklı görüntü boyutunda test edilmiştir. Redmon ve Farhadi [22] tarafından yapılan YOLOv3 çalışması, gerçek zamanlı nesne tespitinde, özellikle de çok küçük nesnelerin tespitinde önemli ilerlemeler göstermiştir. Liu ve diğ. [23] “Tam Evrişim Ağından (FCN)” esinlenmiş ve yeni bir optimize YOLO türü oluşturmak için YOLO kombinasyon modellerini kullanmıştır. Önerilen birleşik modeller, trafik yoğunluğunun yüksek olduğu senaryolarda nesne tespitinde daha iyi bir doğruluk oranı sunmaktadır. VOC07 ve VOC12 veri setlerinde ortalama doğruluk oranı %69.3 olarak raporlanırken, klasik YOLO mimarisi bu oranın %64'ünde kalmıştır. YOLOv4 algoritması, 2020 yılında Bochkovskiy ve diğ. [24] tarafından önerilmiştir. Günümüzde yaygın olarak kullanılan birçok bilimsel yöntem, büyük mini batch boyutlarıyla eğitim yapabilmek için birden fazla GPU'ya ihtiyaç duyar. Eğitim yalnızca tek bir GPU ile gerçekleştirildiğinde süreç yavaşlamakta, performans düşmekte ve verimlilik azalmaktadır. YOLOv4 ise bu duruma, daha küçük bir mini batch boyutuyla tek bir GPU üzerinde eğitilebilen bir nesne algılama modeli sunarak çözüm getirmektedir.

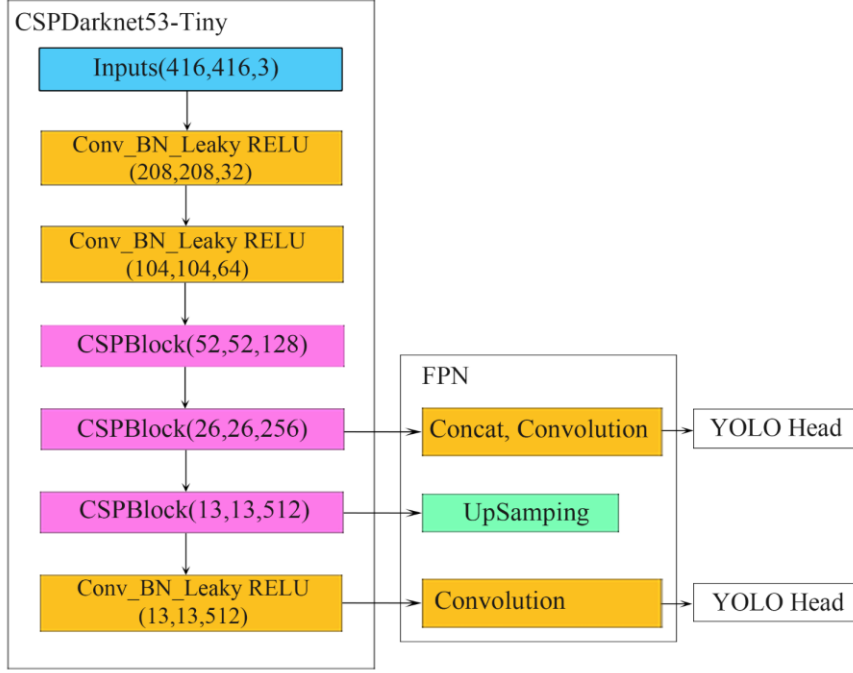
YOLO gibi derin öğrenme modelleri, nesne algılama uygulamalarında önerilen en yeni yöntemler arasındadır. YOLO, gerçek zamanlı uygulamalarda kullanılan ve arzu edilen performansa sahip güncel bir nesne algılayıcıdır. Gerçek zamanlı uygulamalarda sunduğu yüksek hız ve doğruluk oranı ile öne çıkmaktadır. YOLO, nesne algılama görevini bir regresyon problemine dönüştürerek, geleneksel yöntemlere kıyasla önemli performans avantajları sunmaktadır. YOLO algoritması, görüntüyü küçük parçalara bölerek her bir bölge için sınırlayıcı kutular ile ilgili sınıf olasılıklarını üretir. Elde edilen bu kutular, tahmin edilen olasılık değerlerine göre ağırlıklandırılarak değerlendirilir. Geleneksel nesne algılama modelleri, her bir olası nesne bölgesi için ayrı ayrı tahminler yaparken, YOLO tüm görüntüyü tek seferde işleyerek tahminlerini gerçekleştirmektedir. Bu yaklaşım sayesinde, model hem uzamsal bağlamı daha iyi kavrayabilir hem de gereksiz tekrarları önleyerek işlem süresini önemli ölçüde azaltabilir. Bu nedenle, görüntüdeki tüm bilgiler kullanılarak nesne tespiti mümkündür. YOLO nesne dedektörü, bir görüntü için binlerce bölge gerektiren, hesaplama maliyetleri yüksek olan R-CNN ve Fast-RCNN'den farklıdır. Bunun nedeni tek bir ağ değerlendirmesi kullanmasıdır. Model, görüntü giriş özelliklerini Şekil 3.1'de gösterildiği gibi elde eder. YOLO mimarisinde, giriş görüntüsü Şekil 3.1'de gösterildiği üzere $S \times S$ boyutlarında bir ızgara yapısına bölünmektedir. Her bir ızgara hücresi, ilgili bölgedeki nesneye ait verileri analiz ederek sınırlayıcı kutular üretmektedir ve bu kutular, tespit edilen nesneye ilişkin güven skorlarıyla desteklenmektedir. Her hücre, bir sınırlayıcı kutu ve buna ait bir güven skoru tahmininde bulunmaktadır. Bu kutular; (x, y, w, h) koordinatları ile güven değeri olmak üzere toplam beş parametre içermektedir. Burada (x, y) , kutunun merkezini tanımlarken, (w, h) görüntü boyutlarına göre kutunun genişliğini ve yüksekliğini temsil etmektedir. Güven skoru ise, mimarinin ilgili kutuda gerçekten bir nesne bulunduğuna dair duyduğu güvenin derecesini ifade etmektedir.



Şekil 3.1 : YOLO tespit sistemi.

YOLO, Darknet'e dayanmaktadır ve bir dizi derin öğrenme mimarisi üreten çeşitli geliştirmelere sahiptir. İlk üç YOLO sürümü (YOLO, YOLOv2 ve YOLOv3) Joseph Redmon tarafından sırasıyla 2016 [19], 2017 [14] ve 2018 [22] yıllarında yayınlanmıştır. Bochkovskiy ve diğerleri [24] YOLOv4 modelini yüksek tespit doğruluğu ve etkileyici işlem hızıyla başarılı bir nesne algılama yöntemi olarak değerlendirmiştir. YOLOv4, önceki sürümlerine kıyasla kayda değer performans artışları sunmaktadır. COCO veri seti üzerinde Tesla V100 GPU ile gerçekleştirilen deneylerde, model saniyede 65 kare işleme (FPS) kapasitesine ulaşmış ve %43.5 ortalama doğruluk (AP) elde etmiştir. Hız ve hassasiyet açısından değerlendirildiğinde, YOLOv4 en güçlü modellerden biri olarak öne çıkmaktadır. Lopez-Montiel ve arkadaşları [64] tarafından geliştirilen YOLOv4-Tiny, YOLOv4 mimarisinin daha sade ve hafifletilmiş bir versiyonudur. Bu yapının kurulumu, diğer modellere kıyasla daha az karmaşıktır ve nesne tespiti görevlerinde tatmin edici bir başarı sunmaktadır. Bu sürüm, sinir ağı modelinin doğruluğunu artırırken hesaplama karmaşıklığını en aza indirebilir. YOLOv4-Tiny, daha hızlı bir model olduğu için gömülü sistemler veya sınırlı bilgi işlem gücüne sahip cihazlar için daha uygundur. YOLOv4-Tiny sürümü, omurga ağı olarak CSPDarknet53-tiny ağını kullanır. Bu ağ artık ağdaki ResBlock modülü yerine aşamalar arası segmental ağda CSPBlock modülünü kullanmaktadır. YOLOv4-Tiny, düşük maliyetli gömülü sistemlerde yüksek hızda çalışabilmesi amacıyla optimize edilmiştir; bu doğrultuda, CSP omurgasında yer alan konvolüsyon katmanları sadeleştirilmiştir. Ayrıca, orijinal

modelde bulunan üç YOLO katmanı ikiye indirilmiş ve tahmin süreci birkaç bağlantı kutusu aracılığıyla gerçekleştirilmiştir. Şekil 3.2, bu hafifletilmiş modelin genel ağ mimarisini görselleştirmektedir. YOLOv4-Tiny yapısında birden fazla ölçeklenmiş YOLO katmanı yer almakta olup, bu katmanlar farklı boyutlardaki nesnelere algılayabilmek için çeşitli çözünürlüklerdeki özellik haritalarını kullanmaktadır.



Şekil 3.2 : Yolov4-Tiny'nin ağ mimarisi.

YOLOv4-Tiny algoritması başlıca 3 modüle ayrılabilir: Bu modüller Şekil 3.2'de gösterildiği gibi CSPDarknet53-Tiny, Özellik Piramidi Ağı (FPN) ve YOLO Head modülleridir. Giriş verileri, genellikle 416x416 piksel gibi ağa uygun boyutlarda tensör biçiminde modele sunulmaktadır. Cross Stage Partial Network (CSPNet) mimarisine göre, CSPBlock adlı temel yapı katmanı iki ayrı kola ayrılmaktadır. Bu kollardan ilki, doğrudan artık bağlantıyı (residual edge) oluşturmaktan sorumludur. İkinci kol ise, konvolüsyon işlemleriyle işlenerek nihai çıktının elde edilmesine katkı sağlamaktadır.

FPN yapısı, derin ağın anlamsal verilerini ve düşük seviyeli ağın geometrik bilgilerini ayırabilen çeşitli ağ katmanlarının özelliklerini birleştirebilmektedir. Böylece, bu özellik çıkarma yeteneğini geliştirmektedir. YOLO Head, özellik çıktı sonuçları için mimarinin son modülüdür. Tek aşamalı bir dedektör durumunda YOLO Head'in rolü yoğun tahmini gerçekleştirmektir. Yoğun tahmin, tahmin edilen sınırlayıcı kutuların koordinatlarını (yükseklik, genişlik, merkez), etiketini ve tahminin güven puanını

içeren bir vektörden oluşan nihai sonuçtur ve Eşitlik 3.1’de gösterildiği gibi hesaplanmaktadır:

$$\begin{aligned}
 b_x &= \sigma_{tx} + C_x, \\
 b_y &= \sigma_{ty} + C_y, \\
 b_w &= P_w * e^{tw} \\
 b_h &= P_h * e^{th}
 \end{aligned}
 \tag{3.1}$$

Burada P_w ve P_h sınırlayıcı kutuların genişliğini ve yüksekliğini, C_x ve C_y görüntünün sol üst köşesinin koordinatlarını, b_x ve b_y nesnenin orta noktasının koordinatlarını, b_w ve b_h sınırlayıcı kutunun genişliği ve yüksekliğini, σ_{tx} ve σ_{ty} sigmoid fonksiyonunu temsil etmektedir.

YOLOv4-Tiny’de aktivasyon fonksiyonu olarak "Leaky ReLU" kullanılmaktadır. Leaky ReLU (Leaky Rectified Linear Unit), klasik ReLU fonksiyonunun geliştirilmiş bir türevidir. Bu aktivasyon fonksiyonu, negatif giriş değerlerine karşı sıfır yerine küçük bir eğimle yanıt vererek bilgi akışının devamını sağlamaktadır. Matematiksel olarak, Leaky ReLU fonksiyonu Eşitlik 3.2’de gösterildiği şekilde ifade edilmektedir.

$$f(x) = \max(0.1x, x) \tag{3.2}$$

Burada x , fonksiyon girdisini temsil etmektedir. Leaky ReLU aktivasyon fonksiyonu, bazı durumlarda diğer fonksiyonlara kıyasla daha etkili sonuçlar verebilir ve aşırı öğrenme (overfitting) problemini hafifletmeye yardımcı olabilir. Bu avantajları nedeniyle, YOLOv4-Tiny mimarisinde Leaky ReLU hem Darknet-53 altyapısında hem de diğer katmanlarda tercih edilen aktivasyon fonksiyonu olarak kullanılmaktadır.

Şekil 3.2’de de gösterildiği üzere, bu modelde “CSPBlock” (Cross Stage Partial Network) adı verilen özel bir yapı yer almaktadır. CSPBlock, ağın belirli katmanlarında kullanılan bir modüldür ve temel olarak modelin hızını ve genel performansını artırmaya yöneliktir. Bu yapı aynı zamanda modelin boyutunu düşürerek daha kompakt ve verimli bir mimari oluşturmayı hedeflemektedir. CSPBlock, Cross-Stage (Çapraz Aşama) bağlantılarını kullanarak ağın verimliliğini artırmaktadır. Ağın bir önceki aşamasının (backbone) çıktısını kullanarak, ağın farklı noktalarına geri besleme yapmaktadır. Böylece öğrenme sürecini optimize etmektedir. Bu geri besleme ve çapraz aşama bağlantıları, ağı daha güçlü ve daha etkili hale

getirmektedir. CSPBlock yapısı, birçok derin öğrenme modelinde kullanılan Residual Blokların geliştirilmiş bir versiyonu olarak değerlendirilebilir. Residual bloklar, ağırlık katmanları arasında kurduğu çapraz bağlantılar sayesinde öğrenme sürecini kolaylaştırmakta ve eğitim performansını artırmaktadır. CSPBlock ise bu yapıyı daha verimli hale getirerek, performans kayıplarını en aza indirmektedir. Bu özellikleriyle, özellikle sınırlı donanım kaynaklarına sahip gömülü sistemler için ideal bir mimari çözüm sunmaktadır. YOLO, düşük gecikmeli uygulamalar için uygundur ve gömülü sistemler gibi kısıtlı kaynaklara sahip platformlarda verimli çalışabilmektedir. Tüm görüntüyü tek bir işlemde analiz ettiği için, nesnelere arasındaki ilişkileri daha iyi anlamaktadır.

3.2 Uygulama Verilerine Genel Bakış

Trafik işareti algılama ve tanımlama çalışmaları, bilgisayarla görme ve derin öğrenme tekniklerinin gelişmesiyle birlikte önemli ilerlemeler kaydetmiştir. Bu alanda yaygın olarak kullanılan Belgian Traffic Signs (BelgianTS) ve German Traffic Sign Recognition Benchmark (GTSRB) açık veri kümeleri, birçok makine öğrenimi ve derin öğrenme modelinin eğitilmesi ve test edilmesi için temel oluşturmuştur. Gerçekte, bu iki veri kümesi karmaşık trafik işareti algılama senaryolarını yeterince temsil edememektedir. Bunun nedenlerinden biri veri kümelerindeki görüntülerin çözünürlüğü ve boyutları, özellikle küçük trafik işaretlerini algılamada yetersiz kalabilmektedir. Buna ek olarak mevcut veri setleri, farklı hava koşulları, değişen ışık seviyeleri ve trafik yoğunluğu gibi gerçek yol ortamlarında karşılaşılan değişkenlikleri yeterince yansıtmamaktadır. Ayrıca trafik işaretleri, ülkeden ülkeye farklılık gösterebildiğinden, belirli bir bölge için optimize edilmiş veri setleri daha doğru sonuçlar sağlayabilir. Bu çalışmada veriler 2 farklı şekilde kullanılmıştır. Yukarıda belirtilen zorlukları aşmak amacıyla veri seti olarak Şekil 3.3'te gösterildiği gibi mobil haritalama sisteminden elde edilen görüntüler kullanılmıştır. Günümüzde, yüksek doğruluklu konumlandırma ve görüntüleme teknolojileri ile donatılmış MHS'leri, coğrafi bilgi sistemleri, akıllı ulaşım sistemleri ve otonom araçlar gibi birçok alanda yaygın olarak kullanılmaktadır.



Şekil 3.3 : MHS aracı görüntüleri.

Temmuz 2021 tarihinde Karadeniz Bölgesi'nde yer alan 6 kilometrelik bir otoyolda, ortalama 80 km/s hızla hareket eden bir araca entegre edilmiş Mobil Haritalama Sistemi (MHS) aracılığıyla veriler toplanmıştır. Tipik bir MHS; Küresel Navigasyon Uydu Sistemi (GNSS), Atalet Ölçüm Ünitesi (IMU) ve mesafe ölçüm cihazı gibi sensörleri bünyesinde barındırmaktadır. Bu çalışmada kullanılan MHS donanımının şematik gösterimi Şekil 3.4'te sunulmaktadır. Veri toplama amacıyla kullanılan araçta GNSS, IMU, LiDAR (Lazer Tarayıcı), yol yüzeyi tarayıcısı, kilometre sayacı, 360° panoramik kamera ve tünel içi görüntüleme kamerası gibi bileşenler yer almaktadır.

Kullanılan MHS sistemine ait temel teknik özellikler Çizelge 3.1'de sunulmuştur. Gerçek ortandan toplanan bu verilere ek olarak, açık kaynaklı uygulamalardan temin edilen görüntülerle desteklenen bir veri seti oluşturulmuştur. Veri seti hazırlanırken, güneşli, yağışlı, karlı ve sisli gibi farklı hava koşullarında kaydedilmiş görüntülerin dâhil edilmesine özen gösterilmiştir. Ayrıca, değişken aydınlatma durumlarının (gündüz, gece, alacakaranlık) trafik işareti algılaması üzerindeki etkisini değerlendirebilmek amacıyla bu tür senaryolara ait görüntüler de veri setine katılmıştır. Ayrıca, aynı trafik işaretinin farklı varyasyonları (farklı tasarımlar, renkler, boyutlar) ve kamera açılarına göre farklı konumları da veri kümesi görüntülerine dahil edilmiştir. Veri kümesinde ayrıca trafik yoğunluğu veya hava koşulları nedeniyle işaretin kısmen engellendiği görüntüler ve trafik işaretinin ağaç gibi nesnelere veya insanlar tarafından kısmen engellendiği veya engellendiği görüntüler de yer almaktadır. Veri kümesini genişletmek için, elde edilen görüntülere görüntü döndürme ve yansıtma (rotation & flipping), yeniden ölçeklendirme (rescaling), çevirme ve parlaklık/kontrast ayarlamaları gibi teknikler uygulanmıştır.

Çizelge 3.1 : Test ölçümlerinde kullanılan MHS bileşenlerinin ana özellikleri.

Bileşenler	Kullanılan Cihaz	Özellikler
Panoramik Kamera	Ladybug5+	2464 x 2048 Çözünürlük 30 FPS Resim Karesi Hızı 30 MP (5 MP x 6 sensör) 3.45 µm Piksel Boyutu 4.4 mm Odak Uzaklığında Lensler
Tünel Kamera	HIKVISION DS-2CD2T45FWD-I5	2688 × 1520 Çözünürlük 2.8; 4; 6; 8; 12 mm Odak Uzaklığı 1/2.5" Aşamalı Tarama CMOS Görüntü Sensörü 1/3 s to 1/100000 s Deklanşör Hızı 850nm Dalgaboyu IR range: 50 m ve 80 m Seçenekleri
LİDAR	Velodyne VLP-32C	Sensör 32 Kanal 200 m ± 3 cm Ölçüm Aralığı ve Doğruluğu Görüş Alanı: 360° Yatay; 40° Düşey. 5 Hz to 20 Hz Dönme Hızı Lazer ~903 nm Dalgaboyu 3D Lidar Veri Noktaları Çıktısı ~600000 Nokta/s (Tekli Dönüş Modu) ~1200000 Nokta/s (Çiftli Dönüş Modu)
IMU ve GNSS	Applanix POS LV V5	GNSS'li Performans Özellikleri X, Y, Z Konum (m) : 0.020, 0.020, 0.030 R ve P açıları (derece): 0.008 Başlık (derece): 0.020 (gerçek için) GNSS Özellikleri GPS, GLONASS, Galileo, BDS, QZSS, MSS L-band, SBAS TRM AT1675-540TS Trimble Anteni Veri Hızı: 1-200 Hz

Çizelge 3.1 (devamı): Test ölçümlerinde kullanılan MHS bileşenlerinin ana özellikleri.







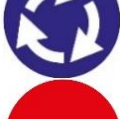



Bileşenler	Kullanılan Cihaz	Özellikler
Odometre	DYNAPAR DMI	1 to 20,000 PPR (darbe/devir) Çözünürlük Kademeli, Optik Kod 400G Şok ve 20G Titreşim 90° ± 30° Elektriksel Dörtlü Fazlama

Mevcut açık kaynaklı trafik işareti veri setleri Avrupa ülkelerine özgü işaretleri içermektedir. Ancak, Türkiye'ye özel bir veri setinin eksikliği, bu çalışmanın önemli hedeflerinden biri olarak belirlenmiştir. Bu kapsamda, elde edilen veriler etiketlenerek Türkiye'de kullanılan trafik işaretlerine odaklanan yeni bir veri kümesi oluşturulmuştur. Seçilen trafik işaretleri, Avrupa genelinde ortak kullanılan işaretlerle uyumlu olacak şekilde belirlenmiştir. Farklı renklere ve şekillere sahip on trafik işareti sınıfı oluşturulmuştur. Bu çalışma için belirlenen trafik işaretlerinin anlamları, sayıları ve sınıf kimlikleri Çizelge 3.2'de sunulmuştur.



Şekil 3.4 : MHS Aracı.

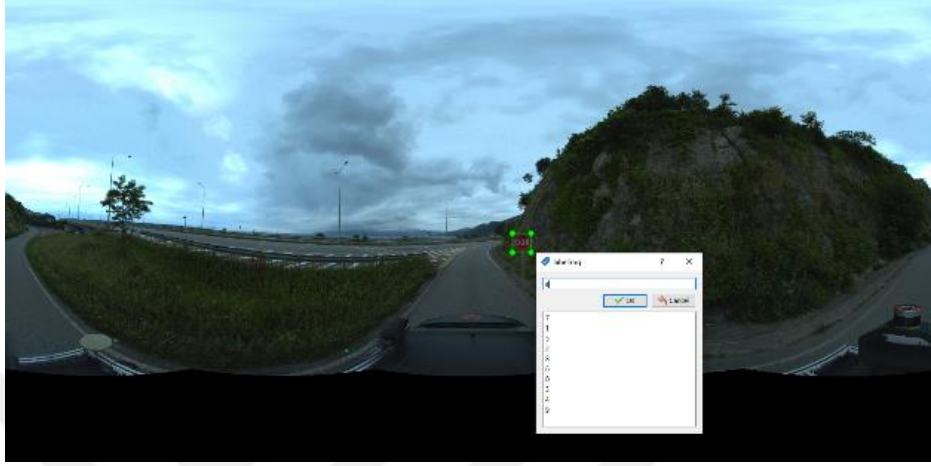
Çizelge 3.2 : Seçilen Trafik İşaretleri ve Sınıf IDleri.

Sınıf ID	Trafik İşareti	İşaretin Adı	Toplam Görüntü
0		Pedestrian Crossing	166
1		50 km	103
2		100 km	105
3		No overtaking	80
4		Stop	89
5		Cycle Crossing	139
6		Roundabout	86
7		No entry	88
8		120 km	58
9		90 km	57

3.3 Veri Seti Görüntülerinin İşlenmesi

YOLO gibi derin öğrenme tabanlı nesne algılama algoritmalarında görüntü etiketleme, modelin doğruluğunu artırmada kritik bir rol oynar. Etiketleme sürecinde, her görüntüdeki trafik işaretleri gibi nesnelere için sınırlayıcı kutular (bounding boxes) belirlenir ve bu kutular ilgili sınıf etiketleri ile eşleştirilir. YOLO, nesnelere belirli grid hücrelerine ayırarak tahmin yaptığından, etiketleme sırasında kutuların koordinatları (x, y, genişlik, yükseklik) doğru şekilde belirtilmelidir. Özenli bir etiketleme işlemi,

YOLO'nun daha yüksek doğrulukla nesne tespiti yapmasını ve karmaşık sahnelerde daha başarılı sonuçlar elde etmesini sağlamaktadır. Belirlenen trafik işareti sınıflarının tespiti amacıyla, 971 görüntüden oluşan bir veri kümesi kullanılmıştır. Bu görüntülerin her biri farklı çözünürlüklere sahiptir. Tüm görüntüler, Şekil 3.5'te gösterildiği üzere etiketleme sürecinden geçirilmiştir.



Şekil 3.5 : Trafik işareti etiketleme örneği.

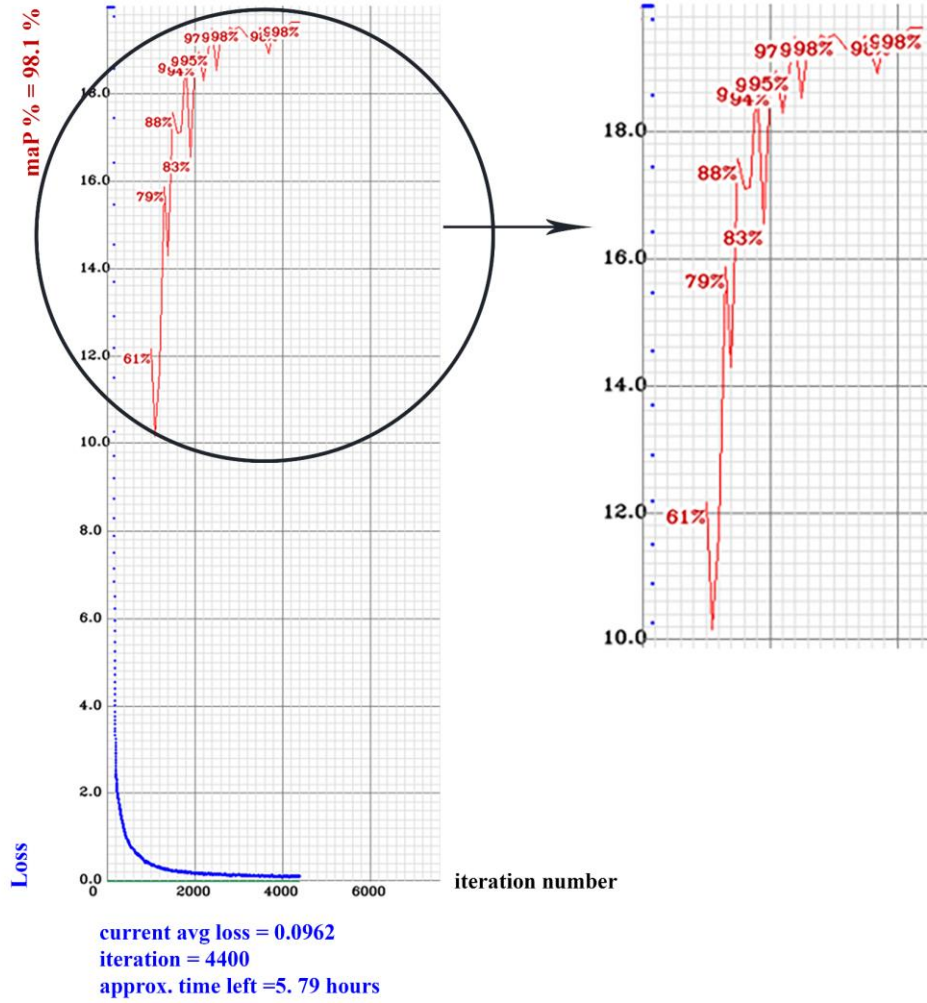
YOLO algoritması ile etiketleme işlemi yapılabilmesini sağlayan çeşitli yazılımlar mevcuttur. Örnek olarak Supervise.ly, Hasty.ai, CVAT, Darwin, Heartex, Scalable, Segments.ai, Make-Sense, LinkedAI, LabelImg, RectLabel, LabelBox, LabelMe, DataTurks ve makesense.ai gibi araçlar gösterilebilir. Bu çalışma kapsamında, tüm görsellere yönelik etiketleme işlemleri LabelImg programı kullanılarak gerçekleştirilmiştir. Şekil 3.6'da yer alan örnek .txt dosyasında, yaya geçidi trafik işareti için oluşturulan etiket bilgileri yer almaktadır. Dosyada ilk değer, işaretin ait olduğu sınıfı temsil ederken; koordinat bilgileri ve sınıf ID'si, görüntüyle aynı isme sahip bir metin dosyasına kaydedilmiştir.



Şekil 3.6 : Etiketlenmiş trafik işareti.

3.4 CNN Modelinin Oluşturulması ve Eğitilmesi

Veri setinin eğitim ve test olarak ayrılması, modelin genelleştirme yeteneğini değerlendirmek için kritik bir adımdır. Eğitim sürecinde kullanılan donanım özellikleri, modelin işlem hızını ve performansını doğrudan etkilemektedir. YOLOv4-Tiny modeli için belirlenen hiperparametreler, modelin dengeli ve verimli bir şekilde öğrenmesini sağlamayı amaçlamaktadır. Veri seti görüntüleri %90'ı eğitim, %10'u test olacak şekilde rastgele ikiye ayrılmıştır. Bu çalışmanın eğitim işlemleri CUDA desteğine sahip NVIDIA GeForce GTX 1660 GPU ekran kartı, 2.90 GHz hızında i7-10700 işlemci ve 32 GB DDR2 bellek içeren bir sistem üzerinde gerçekleştirilmiştir. Eğitim 20000 iterasyon (`max_batches`), `policy = steps` ve `steps = 16000, 18000` olarak işlenmiştir. Ağın ağırlıklarının güncellenme hızını belirleyen hiperparametre olan öğrenme oranı, bu çalışmada 0.00261 olarak seçilmiştir. Şekil 3.7'de, YOLOv4-Tiny modeli ile yürütülen eğitim süreci görselleştirilmiştir. YOLOv4-Tiny ile geliştirilen modele ait *mAP*, kayıp (*loss*) ve iterasyon değerlerine ait grafik görülmektedir. Eğitim sürecinin ilerleyişi, belirlenen iterasyonlar boyunca modelin hata oranının düşmesi ve doğruluk oranının artması ile takip edilmiştir. Şekil 3.7'de görüldüğü üzere, 4400 iterasyon sonunda mavi renk ile gösterilen kayıp (*loss*) değerleri azalmış, kırmızı renk ile gösterilen *mAP* değerleri ise artış göstermiştir.



Şekil 3.7 : YOLOv4-Tiny kullanılarak eğitim kaybı değeri, mAP ve AP sonucu.

Doğruluk değerlendirme metriği olan mAP , Eşitlik 3.3'te gösterildiği gibi, her bir sınıf için hesaplanan AP değerlerinin ortalamasını ifade etmektedir.

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (3.3)$$

Burada, AP_i i. sınıfın ortalama hassasiyetini, N ise sınıf sayısını temsil etmektedir.

Sınıflandırma modelleri mAP , Kesinlik, Geri Çağırma veya F1-Skoru kullanılarak değerlendirilir. Geri Çağırma, modelin tüm pozitif örnekleri bulma düzeyini hesaplamak için kullanılır. Kesinlik, modelin tahmin sonuçlarının doğruluğunu hesaplamak için kullanılır. F1 puanı, doğruluk ve geri çağırmanın kapsamlı bir oranıdır. Bu ölçütlerin hesaplanmasında Eşitlik 3.4 ve Eşitlik 3.5'te gösterildiği gibi karışıklık matrisi kriterleri kullanılmıştır.

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$
(3.4)

$$F1 - score = 2 * \frac{precision * recall}{precision + recall}$$
(3.5)











Burada TP doğru pozitif, TN doğru negatif, FP yanlış pozitif, FN ise yanlış negatif sayısını temsil etmektedir. Çizelge 3.3’de TT100K veri seti üzerinde test edilen çeşitli yöntemlerin *mAP* sonuçları karşılaştırılmaktadır. Bu çalışmada elde edilen performans ölçütleri sırasıyla: precision %86, recall %92, F1-skoru %89 ve *mAP* %98.18 olarak hesaplanmıştır. Model eğitimleri sonucu elde edilen *mAP* değeri, Çizelge 3.3’de *mAP* değerleri verilen çalışmalar ile karşılaştırıldığında tüm değerlerden daha yüksek olduğu gözlemlenmektedir.

Çizelge 3.3 : TT100K veri kümesi kullanılarak çeşitli trafik işareti algılama modellerinin performanslarının değerlendirilmesi [65].

Yöntemler	<i>mAP</i> (%)
SSD	59.8
YOLO v3	74.3
YOLO v4-Tiny	76.8
YOLOX	84.9
Önerilen Yöntem	98.1

Eğitim süreci sonunda trafik işaretleri için sınıf bazında elde edilen *mAP* değerleri Çizelge 3.4’te sunulmuştur. Çizelgedeki veriler incelendiğinde yöntemin ele aldığı trafik işareti sınıflarında *mAP* değerinin % 90.0’ın üzerinde olduğu görülmektedir.

Çizelge 3.4 : Trafik işaretlerinin sınıflarına göre *mAP* değerleri.

Sınıf ID	Trafik İşareti	İşaretin Adı	<i>mAP</i>	Sınıf ID	Trafik İşareti	İşaretin Adı	<i>mAP</i>
0		Pedestrian Crossing	%90.56	5		Cycle Crossing	%100.00
1		50 km	%100.00	6		Roundabout	%100.00
2		100 km	%100.00	7		No Entry	%91.28
3		No Overtaking	%100.00	8		120 km	%100.00
4		Stop	%100.00	9		90 km	%100.00

Model mimarisi ve hiperparametre ayarları, YOLOv4-Tiny algoritmasının optimizasyonunu ayarlamak için ayarlanabilir. YOLOv4-Tiny modelinin kapasitesi, daha derin bir mimari ya da filtre sayısının artırılmasıyla geliştirilebilir; ancak bu durum, eğitim süresinin uzamasına neden olabilir [24]. Ayrıca, yığın boyutu ve öğrenme oranı gibi hiperparametrelerin farklı kombinasyonlarıyla yapılacak denemeler, modelin eğitim performansını olumlu yönde etkileyebilir. Öğrenme hızının kademeli olarak azaltılması (öğrenme hızı düşüşü) genellikle avantajlıdır.

3.5 Modelin Gerçek Zamanlı Olarak Test Edilmesi

3.5.1 Açık kaynak görüntüler üzerinde gerçekleştirilen deneyler

Model eğitimi sonucunda elde edilen ağırlıklar ile test edilen trafik işareti içeren görüntüler (açık veri) görsel ortama aktarıldığında Şekil 3.8'deki sonuçlar elde edilmiştir. Elde edilen sonuçlar, modelin tüm trafik işaretlerini başarılı bir şekilde algılayıp doğru sınıflandırdığını göstermektedir. Modelin performansı, nesne boyutu, aydınlatma koşulları, trafik işaretinin açısı ve çevresel faktörlerden bağımsız olarak büyük ölçüde tutarlı kalmıştır. Tüm trafik işaretleri sınıflarını temsil eden görüntülerde, “Pedestrian Crossing” 0.97, “Cycle Crossing” 0.99, “50 km” 0.97, “100

km” 0.64, “No Overtaking” 0.74, “Stop” 0.99, “RoundAbout” 0.71, “No Entry” 0.97, “120 km” 0.99 ve “90 km” 1.00 olarak yüksek seviye doğruluklar tespit edilmiştir.



Şekil 3.8 : Trafik işareti algılama sonucu örneği.



Şekil 3.9 : Trafik işareti algılama sonucu örneği.

Bazı trafik işaretlerinde göreceli olarak daha düşük doğruluk oranları gözlemlenmiştir. Örneğin, “100 km” işareti için 0.64 doğruluk oranı kaydedilmiş olup, bu düşüşün nedenleri arasında işaretin kısmen engellenmesi, düşük kontrastlı ortamlarda görüntülenmesi veya diğer trafik işaretlerine benzerlik göstermesi olabileceği düşünülmektedir. “100 km” trafik işaretini içeren başka bir görüntüde 1.00 doğruluk görülmesi de bu sonucu desteklemektedir. Genel olarak, YOLOv4-Tiny tabanlı modelin açık veri üzerinde gerçekleştirdiği testlerde yüksek performans gösterdiği ve trafik işaretlerini etkili bir şekilde sınıflandırdığı gözlemlenmiştir.

3.5.2 MHS sisteminden elde edilen gerçek ortam görüntüleri üzerinde gerçekleştirilen deneyler

Eğitilen modelin gerçek dünya koşullarında performansını değerlendirmek amacıyla, model verilerin toplandığı Mobil Haritalama Sistemi aracından elde edilen görüntüler üzerinde test edilmiştir. Gerçek ortamda çekilen bu görüntüler, modelin teorik eğitim aşamasındaki başarısının pratik uygulamalarda ne ölçüde geçerli olduğunu görmek için kritik bir değerlendirme sağlamaktadır. Eğitilen modelin gerçek ortamda, verilerin toplandığı MHS aracından alınan görüntüler üzerinden test edilmesi sonucu ulaşılan değerler Şekil 3.9’da görülmektedir. MHS aracının kullandığı 360 derece panoramik

ve yüksek çözünürlüklü görüntüler, geniş bir alanı kapsadığı için trafik işaretlerinin tespit edilmesini daha karmaşık hale getirmektedir. Büyük boyutlu görüntülerin işlenmesi yüksek hesaplama maliyeti gerektirmekte ve tespit süresini önemli ölçüde artırmaktadır. Ayrıca, bu tür görüntülerde trafik işaretlerinin göreceli boyutları oldukça küçük kalmakta, bu da modelin doğru tahmin yapmasını zorlaştırmaktadır. Küçük nesnelerin tespiti, özellikle uzak mesafedeki trafik işaretlerinin netliğinin azalması ve piksel yoğunluğunun düşmesi nedeniyle, nesne sınırlarının belirlenmesini güçleştirmektedir. Buna rağmen bu çalışmadan önerilen model, Şekil 3.9 a’da “50 km” trafik işaretini 0.97 doğruluk oranıyla başarılı bir biçimde algılayıp sınıflandırmıştır. Şekil 3.9 b’de ise “Pedestrian Crossing” trafik işaretini yine 0.97 oranı ile doğru bir şekilde tespit etmiş ve sınıflandırmıştır.



(a)

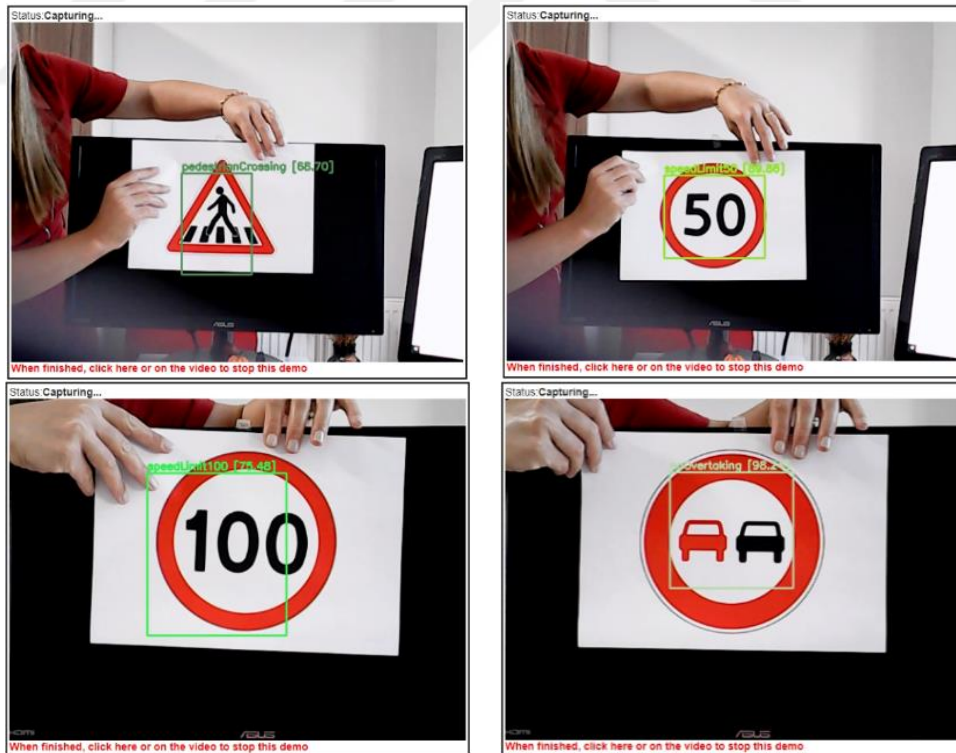


(b)

Şekil 3.10 : Modelin gerçek MHS görüntüleri üzerinde test edilmesi.

3.5.3 Laboratuvar ortamında gerçekleştirilen deneyler

Model eğitimi sonucunda performansını değerlendirmek için elde edilen ağırlıklar ile basit web kamerası kullanılarak gerçek zamanlı testler yapılmıştır. Testler sonucunda, trafik işareti içeren görüntünün YOLOv4-Tiny tarafından doğru tespit ettiği Şekil 3.10'da görülmektedir. Testler sırasında kullanılan kamera çözünürlüğünün düşük olması ve testlerin gerçek trafik işaretleri üzerinde yapılmaması nedeniyle gerçek zamanlı testlerinde doğruluk oranında düşme gözlemlenmiştir. Elde edilen sonuçlara göre; "Pedestrian Crossing" 68.70, "50 km" 89.88, "100 km" 75.48, "No Overtaking" 98.1, "Stop" 80.04, "Cycle Crossing" 90.04, "RoundAbout" 0, "No Entry" 62.43, "120 km" 77.37 ve "90 km" 91.66 olarak oranlarında başarılı tespitler gerçekleştirilmiştir. Bu doğruluk oranlarındaki farklılıklar, kamera çözünürlüğünün düşük olması, trafik işaretlerinin laboratuvar ortamında basılı kâğıt veya dijital ekran üzerinden test edilmesi ve test edilen görüntülerin gerçek sahne koşullarını tam olarak yansıtmaması gibi faktörlerle açıklanabilir. Gerçek zamanlı testlerin daha güvenilir sonuçlar vermesi için daha yüksek çözünürlüklü kameralar kullanılarak, farklı açılardan ve farklı ışık koşullarında çekilen gerçek trafik işaretleri ile ek testler yapılması önerilmektedir.



Şekil 3.11 : Bir web kamerasından trafik işareti algılama örneği.



Şekil 3.12 : Bir web kamerasından trafik işareti algılama örneği.

4. TARTIŞMA

Trafik işaretinin geliştirilen algoritmalar ile doğru tespit edilmesinde, aracın işareti gördüğü andaki ışık, çözünürlük ve trafik işaretinin rengi etkili olmaktadır. Şekil 4.1’de yer alan örnekte, a görüntüsündeki trafik işareti geliştirilen algoritma tarafından algılanamamıştır. Photoshop ile a görüntüsündeki sarı kısımlar çıkartılarak b görüntüsü elde edildiğinde ise b görüntüsündeki trafik işaretinin algoritma tarafından yüksek doğrulukla 0.97 doğru tespit edildiği görülmektedir.



(a)



(b)

Şekil 4.1 : Renk değişiminin trafik işareti algılama üzerindeki etkisi.

Benzer biçimde, Şekil 4.2’de 'RoundAbout' trafik işareti sınıfına ait örnek görüntüler yer almaktadır. Bu sınıfa ait işaretler orijinal olarak mavi renkte olup, eğitim veri setinde kullanılan örneklerde de aynı renk özelliği korunmuştur. Mavi renkte olan bu işaret algoritma tarafından 0.99 doğruluk oranı ile doğru tespit edilip sınıflandırılmıştır. Bu trafik işaretinin rengi photoshop ile değiştirildiğinde ise yine 0.36 oranı ile doğru tespit edilmiş fakat doğruluk oranı düşmüştür.



(a)



(b)

Şekil 4.2 : Renk değişiminin trafik işareti algılama üzerindeki etkisi.

Trafik işaretlerinin başarılı bir şekilde tanımlanmasında renk kadar, çözünürlük de belirleyici bir unsurdur. İşaretin çözünürlüğü, genellikle araç ile işaret arasındaki uzaklığa bağlı olarak farklılık göstermektedir. Şekil 4.3'te, MHS aracının trafik işaretine olan uzaklığına göre sırasıyla 'Uzak', 'Orta' ve 'Yakın' şeklinde sınıflandırılan, farklı çözünürlüklere sahip üç ardışık görüntüye yer verilmiştir. Bu görüntüler, farklı günlerde yapılan eğitimler sonucunda elde edilen *mAP* değerlerine göre test edilerek değerlendirilmiştir. Eğitim sürecinde elde edilen *mAP* sonuçları, eğitilen modelin parametrelerine göre farklılık gösterebilmektedir. Bu bağlamda, çalışmada modelin doğruluk seviyesini artırmak amacıyla çeşitli eğitim senaryoları uygulanmıştır. Testler, her bir eğitim sonunda elde edilen *mAP* değerlerine göre gerçekleştirilmiştir. Şekil 4.3'te yapılan 3 farklı eğitimin sonucunda elde edilen *mAP* değerleri referans alınarak gerçekleştirilen testler gösterilmektedir. Burada elde edilen *mAP* değerlerinin sırasıyla %77, %83 ve %98 olduğu görülmektedir. En yüksek başarıya ulaşan %98 *mAP* değeri olan sonuç, eğitim süresi, veri kümesinin zenginleştirilmesi ve epoch değerinin etkisi ile elde edilmiştir. Şekil 4.3'te bulunan a, b ve c görüntüleri MHS aracıyla kaydedilen aynı trafik işareti görüntülerine ait olup, "Uzak" kategorisinde gösterilmektedir. Bu görüntüler üzerinde farklı günlerde yapılan eğitim sonuçları test edilmiştir. Yapılan testler sonucunda model "Pedestrian Crossing" trafik işaretini tespit edip doğru sınıflandırırken, "RoundAbout" trafik işaretini tespit edememiş ve sınıflandıramamıştır. Ayrıca doğru sınıflandırmalarda, artan *mAP* değerine paralel olarak trafik işaretinin doğru sınıflandırma oranının da

arttığı görülmektedir. Şekil 4.3'te a görüntüsünde " Pedestrian Crossing" trafik işareti %77 *mAP* değerinde 0.82, %83 *mAP* değerinde 0.85 ve %98 *mAP* değerinde 0.99 oranında doğru tespit edilmiştir. Aynı trafik işaretine ait, MHS aracı tarafından daha kısa bir mesafeden alınmış görüntü ise 'Orta' mesafe kategorisinde yer almaktadır. Buradaki d, e ve f görüntüleri incelendiğinde %77 *mAP* değeri ile yapılan eğitim sonucunda gerçekleştirilen testte 0.90 oranı ile " Pedestrian Crossing" doğru tespit edilmiş ve sınıflandırılmıştır.

Ancak d görüntüsünde hatalı bir sınıflandırma yapıldığı da görülmektedir. Aynı görüntüyü içeren e görüntüsünde ise *mAP* değerinin %83 olduğu ve "Pedestrian Crossing" ve "RoundAbout" trafik işaretlerinin sırasıyla 0.73 ve 0.96 oranlarında doğru tespit edildiği ve sınıflandırıldığı görülmektedir. *mAP* değerinin %98 olduğu f görüntüsünde ise artan *mAP* değeri ile birlikte "Pedestrian Crossing" ve "RoundAbout" trafik işaretlerinin sırasıyla 0.94 ve 0.99 oranlarında doğru tespit edildiği ve sınıflandırıldığı görülmektedir. MHS aracının daha yakın mesafeden elde ettiği d, e ve f görüntülerine ait test sonuçları, trafik işaretlerinin çözünürlüğünün modelin doğru tespit ve sınıflandırma başarısı üzerindeki etkisini ortaya koymaktadır.

Aynı trafik işaretlerini içeren g, h ve i görüntüleri, MHS aracının trafik işaretine en yakın olduğu görüntüyü içermektedir. Bu görüntülerde de *mAP* değerinin etkisi diğer görüntülerde olduğu gibi trafik işaretinin doğru tespit edilme ve sınıflandırılma olasılığını artırmıştır. h ve i görüntülerindeki tüm trafik işaretleri doğru tespit edilmiş ve sınıflandırılmıştır. Bununla birlikte, bu yakın mesafeden elde edilen görüntülerde doğru tespit ve sınıflandırma performansında azalma olduğu fark edilmiştir. Şekil 4.3'te g görüntüsünde *mAP* değeri %77 olan "Pedestrian Crossing" trafik işareti 0.76 oranında doğru tespit edilirken "RoundAbout" trafik işareti tespit edilememiştir. h görüntüsünde, *mAP* değeri %83 olan "Pedestrian Crossing" trafik işareti 0.66 oranında doğru tespit edilirken, "RoundAbout" trafik işaretleri için bu oran 0.44'tür. Görüntü i'de ise %98 *mAP* değerine sahip "Pedestrian Crossing" trafik işareti için 0.60 ve " Pedestrian Crossing" trafik işareti için 0.99 oranlarına ulaşılmıştır. Bu sonuçlar MHS aracının trafik işaretini gördüğü görüntünün açısının etkisinin olduğunu göstermektedir.



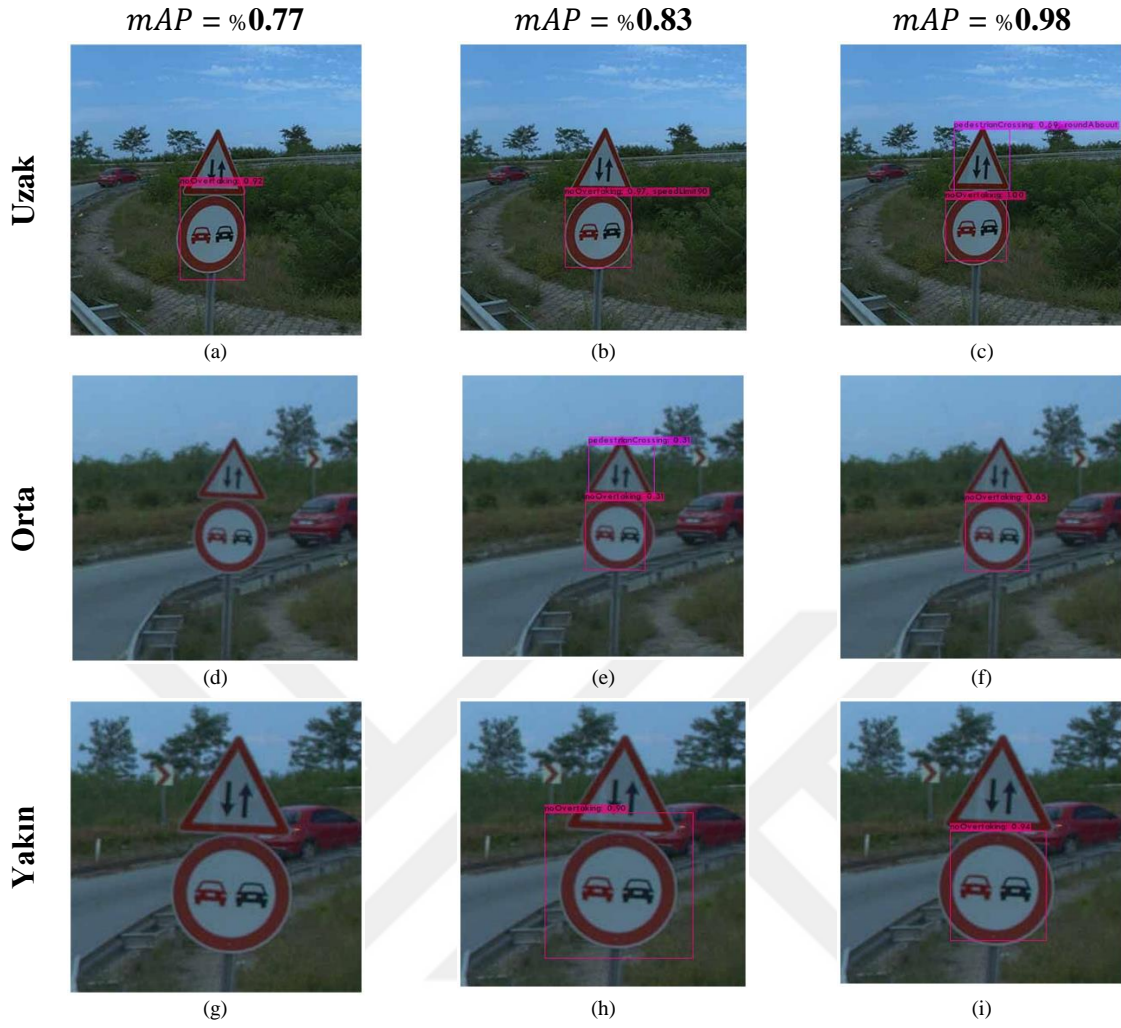
Şekil 4.3 : Farklı açı ve mesafelerde artan mAP değeri ile elde edilen test sonuçları.

Şekil 4.4'teki görseller incelendiğinde, trafik işaretinin araçla arasındaki mesafe, görüntü çözünürlüğü ve görüş açısının; modelin trafik işaretini doğru algılayıp sınıflandırmasında önemli bir rol oynadığı anlaşılmaktadır. Ayrıca mAP değerlerinin artışının da trafik işaretlerinin doğru tespit edilip sınıflandırılmasında doğruluk oranını arttırdığı görülmektedir. a, b, c, d, g görüntülerinde “Pedestrian Crossing” trafik işareti, g ve h görüntülerinde ise “50 km” trafik işaretleri model tarafında tespit edilememiştir. e, f, i görüntülerinde ise her iki trafik işareti de doğru tespit edilip sınıflandırılmıştır.



Şekil 4.4 : Farklı açı ve mesafelerde artan mAP değeri ile elde edilen test sonuçları.

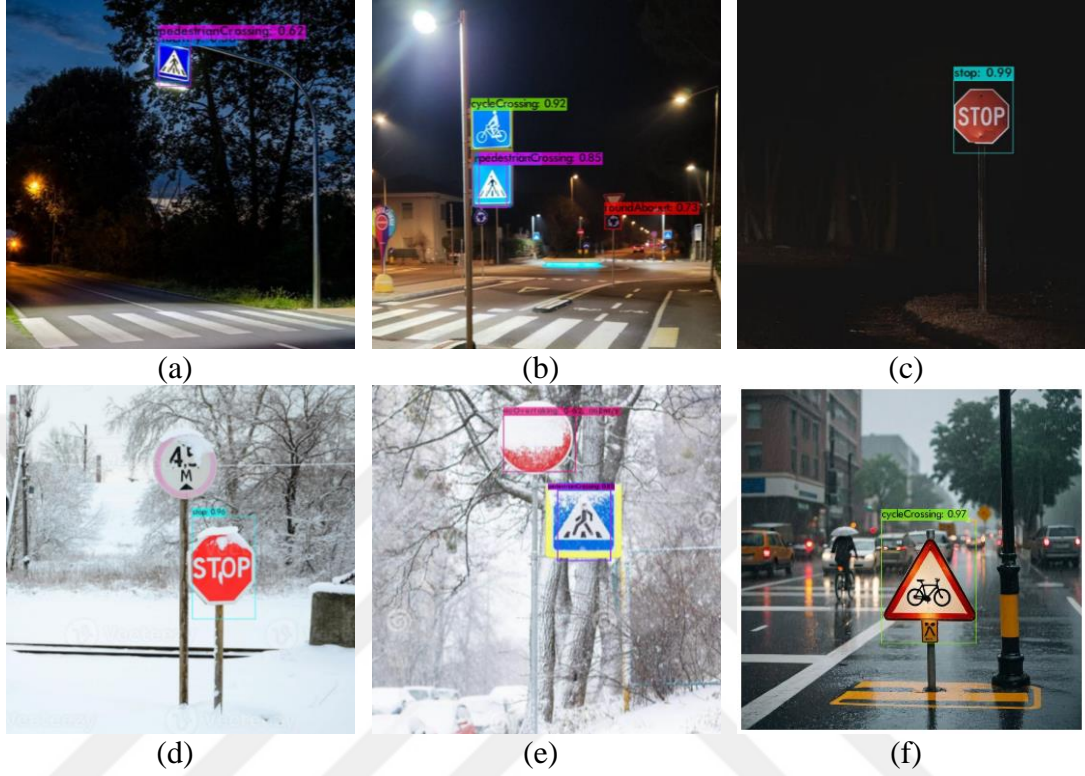
Şekil 4.5’de bulunan görüntüler incelendiğinde a, b, d, f, g, h ve i görüntülerinde bulunan üçgen şeklindeki çift yönü ifade eden trafik işareti, model tarafından tespit edilememiştir. İlgili trafik işareti, ne eğitim ne de test veri setlerinde yer aldığından ve çalışma kapsamında tanımlanan trafik işareti sınıflarına dahil olmadığından model tarafından tespit edilememiştir. “NoOvertaking” trafik işareti ise a, b, c görüntülerinde sırası ile 0.92, 0.97, 1.00 oranları ile model tarafından doğru tespit edilip sınıflandırılmıştır. e, f, h ve i görüntülerinde ise sırası ile 0.31, 0.65, 0.90, 0.94 oranla doğru tespit edilip sınıflandırılmıştır.



Şekil 4.5 : Farklı açı ve mesafelerde artan mAP değeri ile elde edilen test sonuçları.

Şekil 4.6, a, b ve c görüntülerinin “Pedestrian Crossing”, “Cycle Crossing”, “RoundAbout” ve “Stop” trafik işaretleri içerdiğini göstermektedir. Söz konusu trafik işaretlerinin yer aldığı tüm görseller, farklı ışıklandırma koşullarını yansıtmaktadır. Ortam ışığının yetersiz olduğu, sahne yapısının karmaşıklaştığı ya da geceye ait düşük çözünürlüklü görüntülere rağmen, modelin çıktıları gündüz koşullarındaki sonuçlara oldukça yaklaşmıştır. Trafik işaretleri sırasıyla 0.62, 0.92, 0.85, 0.73 ve 0.99 oranlarında doğru tespit edilmiş ve tanımlanmıştır. Şekil 4.6’da yer alan d, e ve f görselleri farklı hava durumlarını yansıtmaktadır ve bazı trafik işaretleri olumsuz hava şartları nedeniyle görünürlüğünü kaybetmiştir. Bu görüntüler üzerinde gerçekleştirilen testlerde, modelin karlı ve yağmurlu hava koşullarında trafik işaretlerini sırasıyla 0.96, 0.62, 0.85 ve 0.97 oranlarında başarılı bir şekilde tespit edip sınıflandırabildiği gözlemlenmiştir. Burada, yağmurlu havadaki “Cycle Crossing” trafik işaretinin bulunduğu görüntünün çözünürlüğünün yüksek olması ve karlı havadaki görüntülerde

trafik işaretlerinin üzerlerinde kar birikmesi nedeniyle kısmen kapanması doğru tespit oranını etkilemiştir.



Şekil 4.6 : Farklı ışık ortamlarının ve hava koşullarının trafik işaretlerinin algılanması üzerindeki etkisi.

Otonom araçların trafik işaretlerini otomatik olarak algılama ve tanıma başarımı, farklı senaryolar altında test edilmiştir. Ayrıca, modelin doğruluğu veri artırma yöntemlerinin uygulanmasıyla iyileştirilebilir (örneğin döndürme, yeniden boyutlandırma, yer değiştirme, parlaklık ve kontrast düzenlemeleri) [66]. Veri setinin özgünlüğünü ve çeşitliliğini artırmak adına, veri çeşitlendirme, temiz ve gerçek verilerin toplanması ile farklı kaynaklardan veri elde edilmesi büyük önem taşımaktadır. Ayrıca veri kümesi oluşturulurken verilerin doğruluğunu ve tutarlılığını sağlamak için eksik, hatalı veya tutarsız verilerin çıkarılması gerekmektedir. Verilerdeki anormalliklerin tespit edilmesi ve bunların temizlenmesi modelin yanlış öğrenmesini engeller. Modelin sağlıklı bir şekilde öğrenebilmesi açısından verilerin doğru ve tutarlı şekilde etiketlenmesi büyük bir önem taşımaktadır. Yanlış veya tutarsız etiketler modelin genelleme kabiliyetini azaltabilir [67]. Birden fazla modeli birleştirerek (örn. torbalama, güçlendirme) daha yüksek doğruluk elde etmek

mümkündür. Bu, farklı modellerin güçlü yönlerini birleştirir ve hata oranını azaltır. Daha fazla detay içerdiğinden ve yüksek görüntü kalitesine sahip veri setleri üzerinde eğitilen modelin performansı yüksek olacağından yüksek çözünürlüklü veriler kullanılarak doğruluk artırılabilir [68]. Eğitim verileri çeşitlendirilerek modelin genelleme kabiliyeti artırılabilir. Bununla birlikte, bu durum modelin hafıza ve işlem gücü gereksinimlerini artırabilir. Bu nedenle görüntü işleme ve sonrasındaki işlemler verimli hale getirilmelidir. Örneğin, trafik işareti şekillerinin belirli boyut aralıklarında olacağı varsayımıyla işlem yükü azaltılabilir.

Bu çalışmada geliştirilen model, %98.18 oranında *mAP* başarısına ulaşarak gerçek veri setleri üzerinde yüksek performans sergilemiştir. YOLOv4-Tiny tabanlı model, mobil platformlarda kullanılmaya elverişli olmasının yanı sıra, görüntü boyutundan ve renk sınırlamalarından bağımsız olarak çalışabilmektedir. Ayrıca, düşük maliyetli yapısı ve hızlı işlem yeteneğiyle zaman açısından da önemli avantajlar sunmaktadır. Tespit edilen trafik işaretleri, gerçek zamanlı olarak dijital haritalara entegre edilerek aracın güncel çevresel koşullara anlık uyum sağlaması mümkün hale gelmektedir. Bu işaretlerin yüksek çözünürlüklü haritalara eklenmesi, harita verilerinin ayrıntı düzeyini artırırken, navigasyon sistemlerinin de rotaları daha hassas biçimde güncellemesine olanak tanır. Böylece, sürüş esnasında trafik işaretlerine dayalı olarak daha doğru ve güvenli yönlendirmeler sağlanabilmektedir. Trafik işareti bilgileri, sürücülere görsel ipuçları sağlamak için aracın ekranına (örneğin, baş üstü ekranı veya bilgi-eğlence ekranı) yerleştirilebilir. Algılanan ve sınıflandırılan trafik işaretleri, Gelişmiş Sürücü Destek Sistemleri (ADAS) kapsamında karar verme ve kontrol modüllerinde belirli eylemleri tetikleyebilmektedir. Örneğin, bir hız sınırı işaretinin tespiti, sistemin hız kontrol mekanizmasını devreye sokarak aracın hızını mevcut trafik kurallarına uygun şekilde ayarlamasını sağlayabilir.

5. SONUÇ VE ÖNERİLER

Bu çalışmada, otonom sürüş için önemli yol envanterlerinden biri olan trafik işaretlerinin tespiti ve sınıflandırılması işlemi, performansı ve mobil araçlar için daha uygun olması nedeniyle YOLOv4-Tiny algoritması kullanılarak gerçekleştirilmiştir. Bu amaçla belirlenen bir karayolu güzergahında test alanında tam donanımlı bir mobil haritalama sistemi ile saha ölçümleri yapılmıştır. Karadeniz bölgesinde 6 km'lik bir otoyol güzergahında yapılan ölçümlerden elde edilen veriler kullanılarak trafik levhasının tespit edilmesi ve tanımlanması gerçekleştirilmiştir. Bu çalışma kapsamında, YOLOv4-Tiny modelinin güncel derin öğrenme tabanlı karmaşık ağ yapılarıyla kıyaslandığında tespit doğruluğu ve işlem hızı açısından önemli avantajlar sunduğu ortaya konulmuştur. Bu çalışmanın bir parçası olarak, Türk trafik işaretlerinin görüntüleri MHS ile toplanmış, bu veriler incelenerek uygun olan ve işaret içeren görüntüler belirlenmiş ve Türk Trafik İşareti veri seti oluşturulmuştur. Bu çalışma kapsamında belirlenen 10 farklı trafik işareti sınıfının tespit ve tanımlama işlemleri hem görüntü tabanlı testlerde hem de gerçek zamanlı test uygulamalarında hızlı ve doğru sonuçlar vermiştir. Elde edilen bulgular, modelin gerçek trafik sahnelerinde daha yüksek tespit doğruluğuna ulaştığını göstermektedir. Önerilen yöntem ortalama olarak yaklaşık %90 oranında doğru tahminler yapmıştır. Tespiti yapılan trafik levhalarına bakıldığında geometrik şekli üçgen, daire ve altıgen şeklinde olup üzerinde yazı olanların, şekil olanlara göre tespit oranlarının daha yüksek olduğu görülmüştür. Üzerinde yazı olanların tamamının %100 doğru tespit edilebildiği görülmüştür.

Önerilen metodoloji, yol envanterlerinin oluşturulmasında trafik işareti bilgilerinin toplanmasına yönelik daha hızlı, güvenli, düşük maliyetli ve otomatik bir çözüm sunmaktadır. Bu tür verilerin toplanmasına yönelik geleneksel yöntemler genellikle yoğun el emeği gerektirmekte, bu da hem işletme maliyetlerini artırmakta hem de sahada çalışan personel için güvenlik riskleri oluşturmaktadır. Gelişmiş bilgisayarla görme teknikleri ve derin öğrenme tabanlı nesne algılama modellerinden yararlanan bu metodoloji, insan müdahalesini azaltırken trafik işareti tespiti ve tanımlanmasının verimliliğini ve doğruluğunu önemli ölçüde arttırmaktadır. Dahası, bu yaklaşım

yalnızca trafik işareti algılama ile sınırlı değildir; yol bakımı ve altyapı yönetimi için gerekli olan diğer yol envanteri unsurlarına da genişletilebilir. Yol kenarındaki bariyerler, şerit işaretleri, yaya geçitleri, trafik ışıkları ve diğer kritik varlıklar da benzer metodolojiler kullanılarak otomatik olarak tespit edilebilir ve sınıflandırılabilir. Yol envanteri veri toplama otomasyonları, akıllı ulaşım sistemlerinin geliştirilmesine katkıda bulunarak daha etkili trafik yönetimi sağlar ve genel yol güvenliğini artırır. Bu açıdan yol bakım uygulamalarına ek olarak, önerilen metodolojinin dijitalleşme süreçleri için de önemli etkileri bulunmaktadır. Mevcut altyapının doğru ve güncel dijital modellerinin oluşturulabilmesi, şehir planlaması, otonom araç navigasyonu ve akıllı şehir girişimleri için çok önemlidir. Bu yaklaşım, yüksek çözünürlüklü görüntü ve mekânsal veri toplamak için mobil haritalama sistemlerini kullanarak yol ağlarının kapsamlı dijital ikizlerinin oluşturulmasını da kolaylaştırmaktadır. Bu dijital modeller daha ileri analizler, simülasyonlar ve öngörücü bakım stratejileri için temel veri kümeleri olarak kullanılabilir.

Artan şehir nüfusu, teknolojik gelişmeler, akıllı şehir konseptlerinin evrimi ve büyük verinin verimli bir şekilde yönetilmesi ihtiyacı, daha fazla coğrafi verinin hızlı, ekonomik ve hassas bir şekilde toplanmasını ve sunulmasını gerekli hale getirmiştir. Bu nedenle MHS, daha büyük hacimli verilerin hızlı, doğru ve uygun maliyetli bir şekilde üretilmesi için önemli bir çözüm olarak öne çıkmaktadır.

Özellikle güzergâh boyunca LiDAR aracılığıyla hızlı toplanan verilerden, nesnelerin koordinatları 3D nokta bulutu şeklinde elde edildiğinden, yol envanterlerinin konumlarını da sembolleri ile beraber çıkartmak mümkündür. MHS ile elde edilen veriler, coğrafi olarak referanslandırılabilen ve buna göre sunulabilen bina görünümünün oluşturulmasına da olanak tanımaktadır. Sonuç olarak, MHS yöntemiyle elde edilen kent modelleri, bina cephe bilgileriyle birlikte kentsel alanlar için çeşitli analizlere olanak sağlamaktadır. Bu analizler arasında kent peyzaj özelliklerine ilişkin değerlendirmeler, sürdürülebilirlik çalışmaları, kültürel miras değerlendirmeleri, kent sosyolojisine ilişkin mekânsal analizler, görünürlük değerlendirmeleri ve ekolojik çalışmalar sayılabilir. Ayrıca, bu veriler kullanılarak bina cephe elemanı çıkarımı, sel riski değerlendirmeleri ve diğer afet analizleri yapılabilir.

Ülkemizde yakın zamanda kullanılmaya başlanan MHS sisteminin, doğrusal güzergahların gelişmiş haritalarının üretilmesinde etkili olduğu kanıtlanmıştır.

Kadastro çalışmaları için gerekli doğruluk standartlarını tam olarak karşılamasa da sistem tarafından çekilen panoramik görüntüler kapsamlı görsel veriler sunmakta ve toplanan bilgilerin yeniden doğrulanmasına olanak sağlamaktadır. MHS, ulaşım envanterlerinin çıkarılması ve varlık yönetim sistemlerine entegre edilmesi için uygun maliyetli, hızlı, son derece hassas ve yenilikçi bir hibrit model çözümü sunmaktadır. Ancak bu önemli avantajlara rağmen MHS'lerin kayda değer dezavantajları da bulunmaktadır. Yüksek ilk yatırım maliyetleri, toplanan büyük miktarda veriyi depolamak ve işlemek için yüksek kapasiteli donanım gereksinimi ve verileri analiz etmek ve yorumlamak için uzman teknik personel gerekliliği, sistemin temel zorlukları ve sınırlamaları arasındadır.

Bu çalışmada geliştirilen prototip tespit ve tanımlama sisteminin ileride yapılacak yapay zeka araştırmalarıyla daha da geliştirilerek akıllı ulaşım envanter yönetiminde stratejik bir araç haline gelmesi beklenmektedir. Özetle sadece etkili çözümler sunmakla kalmayacak, aynı zamanda ülkemiz için önemli ekonomik faydalar sağlayacaktır.

Yapay zeka uygulamalarının daha geniş bir yelpazedeki trafik işaret ve işaretçilerini tanıma yeteneğini geliştirmek için kapsamlı ve çeşitli veri kümelerinden yararlanmak çok önemlidir. Bu yaklaşım, sistemin çeşitli koşullardaki performansını artıracaktır. Bu nedenle farklı senaryolar ile yapılan testler de bu çalışmada sunulmuştur. Trafik işaretleri hava koşulları veya ışık farklılıkları gibi faktörler sebebiyle tam ve net gözükmeyebilir. Yağmurlu veya karlı gibi değişik hava koşullarında farklı trafik işaretleri üzerinde yapılan testler de bu çalışmada önerilen yöntemin tespit ve tanımlama başarısı görülmüştür. Benzer şekilde farklı ışık koşullarından karanlık gece trafik işareti içeren görüntüler de test edilmiştir ve yine önerilen yöntemin yüksek doğruluklu ve hızlı sonuçlar verdiği görülmüştür.

Sınıflara ait örnek görüntü sayısının artırılması, tahmin doğruluğunu yükseltmeye katkı sağlayabilir. Bununla birlikte, sınıf sayısındaki artış, çalışmanın kapsamını genişleterek modelin daha çeşitli trafik işaretlerini tanıyabilmesine olanak tanıyacaktır. Endüstri ortaklıklarının teşvik edilmesi, mobil haritalama ve yapay zeka tabanlı uygulamaların genişlemesini ve çeşitlenmesini destekleyecek ve nihayetinde bu teknolojilerin gerçek dünya senaryolarında yaygın olarak benimsenmesini teşvik edecektir.



KAYNAKLAR

- [1] **T.C. Çevre, Şehircilik ve İklim Değişikliği Bakanlığı.** (2020). *Akıllı Şehir Terminolojisi*.<https://cevresehirklimkutuphanesi.csb.gov.tr/SourcePDF/7a4952cd-1c5c-462a-8d0a-bccf0b49b44a>
- [2] **T. C. Çevre, Şehircilik ve İklim Değişikliği Bakanlığı.** (2020). *2024-2030 Ulusal Akıllı Şehirler Stratejisi ve Eylem Planı*.
<https://www.akillisehirler.gov.tr/>
- [3] **Singh, S.** (2015). Critical Reasons for Crashes Investigated in the National Motor Vehicle Crash Causation Survey (Report No: DOT HS 812 115). Washington, DC: National Highway Traffic Safety Administration.
- [4] **Karayolları Genel Müdürlüğü.** (2025). Satış Cinsine Göre Yol Ağı (km.). 01.01.2025.<https://www.kgm.gov.tr/SiteCollectionDocuments/KGMdocuments/Istatistikler/DevletIIYolEnvanter/SatihYolAgiUzunlugu.pdf>
- [5] **Türkiye İstatistik Kurumu.** (2023). *TUIK Trafik Kazaları 2022 Verileri*.
<https://www.tuik.gov.tr/>
- [6] **McGee, H. W.** (2010). Maintenance of Signs and Sign Supports : A Guide for Local Highway and Street Maintenance Personnel (Report No: FHWA-SA-09-025). United States. Federal Highway Administration. Office of Safety. <https://rosap.nhtl.gov/view/dot/24522>
- [7] **Osichenko, D. & Spielhofer, R.** (2018). Monitoring and Inventory of Road Signs and Road Markings State of the Art – A Review of Existing Methods and Systems. In *Proceedings of 7th Transport Research Arena*, (pp. 1-9). Vienna, Austria.
<https://zenodo.org/record/1320920#.X8DYaShKguU>
- [8] **Ellahyani, A., El Ansari, M. & El Jaafari, I.** (2016). Traffic sign detection and recognition based on random forests, *Appl. Soft Comput. J.*, 46, 805–815. doi: 10.1016/j.asoc.2015.12.041.
- [9] **Saadna, Y. & Behloul, A.** (2017). An overview of traffic sign detection and classification methods, *Int. J. Multimed. Inf. Retr.*, 6 (3), 193–210. doi: 10.1007/s13735-017-0129-8.
- [10] **Stallkamp, J., Schlipsing, M., Salmen, J. & Igel, C.** (2011). The German Traffic Sign Recognition Benchmark: A Multi-Class Classification Competition. In *Proceedings of the International Joint Conference on Neural Networks*, (pp. 1453–1460). California, USA: IEEE. doi: 10.1109/IJCNN.2011.6033395.

- [11] **El Baz, M., Zaki, T. & Douzi, H.** (2021). An Improved Method for Red Segmentation Based Traffic Sign Detection. In *2021 IEEE 9th International Conference on Information, Communication and Networks (ICICN)*, (pp. 490–494). Xi'an, China, November 25-28. doi: 10.1109/ICICN52636.2021.9673847.
- [12] **Sheikh, M. A. A., Kole, A. & Maity, T.** (2016). Traffic Sign Detection and Classification Using Colour Feature and Neural Network. In *2016 International Conference on Intelligent Control Power and Instrumentation (ICICPI)*, IEEE, (pp. 307–311). Kolkata, India, October 21-23. doi: 10.1109/ICICPI.2016.7859723.
- [13] **Bahlmann, C., Zhu, Y., Ramesh, V., Pelkofer, M. & Koehler, T.** (2005). A System for Traffic Sign Detection, Tracking, and Recognition Using Color, Shape, and Motion Information. In *IEEE Proceedings. Intelligent Vehicles Symposium*, (pp. 255-260). Las Vegas, NV, USA, June 6-8. doi: 10.1109/IVS.2005.1505111.
- [14] **Redmon, J. & Farhadi, A.** (2017). YOLO9000: Better, Faster, Stronger. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (pp. 7263-7271). http://www.worldscientific.com/doi/abs/10.1142/9789812771728_0012
- [15] **Yuan, X., Hao, X., Chen, H. & Wei, X.** (2014). Robust traffic sign recognition based on color global and local oriented edge magnitude patterns. *IEEE Trans. Intell. Transp. Syst.*, *15* (4), 1466–1474. doi: 10.1109/TITS.2014.2298912.
- [16] **Saadna, Y., Behloul, A. & Mezzoudj, S.** (2019). Speed limit sign detection and recognition system using SVM and MNIST datasets, *Neural Comput. Appl.*, *31* (9), 5005-5015. doi: 10.1007/s00521-018-03994-w.
- [17] **Chen, T. & Lu, S.** (2016). Accurate and efficient traffic sign detection using discriminative adaboost and support vector regression, *IEEE Trans. Veh. Technol.*, *65* (6), 4006-4015. doi: 10.1109/TVT.2015.2500275.
- [18] **Zeng, Y., Xu, X., Shen, D., Fang, Y. & Xiao, Z.** (2017). Traffic sign recognition using kernel extreme learning machines with deep perceptual features, *IEEE Trans. Intell. Transp. Syst.*, *18* (6), 1647-1653. doi: 10.1109/TITS.2016.2614916.
- [19] **Redmon, J., Divvala, S., Girshick, R. & Farhadi, A.** (2016). You Only Look Once: Unified, Real-Time Object Detection. In *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, (pp. 779–788). Las Vegas, NV, USA, June 27-30. doi: 10.1109/CVPR.2016.91.
- [20] **Wang, C. Y. & Cheng-Yue, R.** (2012). Traffic Sign Detection using You Only Look Once Framework. Stanford, Technical Report.
- [21] **Hasegawa, R., Iwamoto, Y. & Chen, Y. W.** (2019). Robust Detection and Recognition of Japanese Traffic Sign in the Complex Scenes Based on Deep Learning. In *2019 IEEE 8th Global Conference on Consumer Electronics (GCCE)*, (pp. 575-578). Osaka, Japan, October 15-18. doi: 10.1109/GCCE46687.2019.9015419.

- [22] **Redmon, J. & Farhadi, A.** (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.
- [23] **Liu, C., Tao, Y., Liang, J., Li, K., & Chen, Y.** (2018, December). Object detection based on YOLO network. In *2018 IEEE 4th information technology and mechatronics engineering conference (ITOEC)* (pp. 799-803). IEEE. doi: 10.1109/ITOEC.2018.8740604.
- [24] **Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M.** (2020). Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934. doi: <https://doi.org/10.48550/arXiv.2004.10934>.
- [25] **Gezgin, H., & Alkan, R. M.** (2024). Traffic sign detection and recognition based on MMS data using YOLOv4-Tiny algorithm, *Neural Computing and Applications*, 36(33), 20633-20651. doi: 10.1007/s00521-024-10279-y.
- [26] **Olsen, M. J.** (2013). Guidelines for the use of mobile LIDAR in transportation applications (Vol. 748). Transportation Research Board.
- [27] **Findley, D. J., Cunningham, C. M., & Hummer, J. E.** (2011). Comparison of mobile and manual data collection for roadway components, *Transportation Research Part C: Emerging Technologies*, 19(3), 521-540. doi: 10.1016/j.trc.2010.08.002.
- [28] **Elhashash, M., Albanwan, H., & Qin, R.** (2022). A review of mobile mapping systems: From sensors to applications, *Sensors*, 22(11), 4262. doi: 10.3390/s22114262.
- [29] **Khattak, A. J., Hummer, J. E., & Karimi, H. A.** (2001). Evaluation of roadway infrastructure data collection technologies, *Public Works Management & Policy*, 6(1), 18-31. doi: 10.1177/1087724X0161003.
- [30] **Lam, J., Kusevic, K., Mrstik, P., Harrap, R., & Greenspan, M.** (2010, May). Urban scene extraction from mobile ground based lidar data. In *Proceedings of International Symposium on 3D Data Processing Visualization and Transmission* (pp. 1-8). Munchen, Germany.
- [31] **Vasic, D., & Davidovic, M.** (2020). Gathering road infrastructure inventory with Mobile Mapping Systems for road administration.
- [32] **Soilán, M., Riveiro, B., Martínez-Sánchez, J., & Arias, P.** (2016). Automatic road sign inventory using mobile mapping systems, *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 41, 717-723. doi: 10.5194/isprsarchives-XLI-B3-717-2016.
- [33] **Hienonen, P., Lensu, L., Melander, M., & Kälviäinen, H.** (2017). Towards condition analysis for machine vision based traffic sign inventory. In *Advanced Concepts for Intelligent Vision Systems: 18th International Conference, ACIVS 2017*, (pp. 212-224). Antwerp, Belgium, September 18-21. doi: 10.1007/978-3-319-70353-4_10.
- [34] **Wang, J., Chen, Y., Dong, Z., & Gao, M.** (2023). Improved YOLOv5 network for real-time multi-scale traffic sign detection, *Neural Computing and Applications*, 35(10), 7853-7865. doi: 10.1007/s00521-022-08077-5.

- [35] **Sirigineedi, M., Kumaravel, T., Natesan, P., Shruthi, V. K., Kowsalya, M., & Malarkodi, M. S.** (2023). Deep learning approaches for autonomous driving to detect traffic Signs. In *2023 International Conference on Sustainable Communication Networks and Application (ICSCNA)* (pp. 1276-1282). Theni, India. November 15-17. doi: 10.1109/ICSCNA58489.2023.10370617.
- [36] **Cui, Y., Guo, D., Yuan, H., Gu, H., & Tang, H.** (2024). Enhanced YOLO network for improving the efficiency of traffic sign detection, *Applied Sciences*, 14(2), 555. doi: 10.3390/app14020555.
- [37] **Yang, J.** (2023). Traffic sign detection based on classic visual recognition models. In *Journal of Physics: Conference Series (Vol. 2646, No. 1, p. 012024)*, Oxford, UK: December. doi: 10.1088/1742-6596/2646/1/012024.
- [38] **Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y.** (2016). Deep learning (Vol. 1, No. 2). Cambridge: MIT press.
- [39] **Bengio, Y., Courville, A., & Vincent, P.** (2013). Representation learning: A review and new perspectives, *IEEE transactions on pattern analysis and machine intelligence*, 35(8), 1798-1828. doi: 10.1109/TPAMI.2013.50.
- [40] **Dean, J., Corrado, G., Monga, R., Chen, K., Devin, M., Mao, M., ... & Ng, A.** (2012). Large scale distributed deep networks. *Advances in neural information processing systems*, 25.
- [41] **Paszke, A.** (2019). Pytorch: An imperative style, high-performance deep learning library. arXiv preprint arXiv:1912.01703.
- [42] **McCulloch, W. S., & Pitts, W.** (1990). A logical calculus of the ideas immanent in nervous activity, *Bulletin of mathematical biology*, 52, 99-115.
- [43] **Bengio, Y., Goodfellow, I., & Courville, A.** (2017). Deep learning (Vol. 1, pp. 23-24). Cambridge, MA, USA: MIT press.
- [44] **LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P.** (1998). Gradient-based learning applied to document recognition, *Proceedings of the IEEE*, 86(11), 2278-2324.
- [45] **Albawi, S., Bayat, O., Al-Azawi, S., & Ucan, O. N.** (2018). Social touch gesture recognition using convolutional neural network, *Computational Intelligence and Neuroscience*, 2018(1), 6973103. doi: 10.1109/ICEngTechnol.2017.8308186.
- [46] **Krizhevsky, A., Sutskever, I., & Hinton, G. E.** (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.
- [47] **Li, F., Johnson, J. & Yeung, S.** (2017). CS231n: Convolutional Neural Networks for Visual Recognition. <https://cs231n.stanford.edu/2017/>
- [48] **Nielsen, M. A.** (2015). *Neural networks and deep learning* (Vol. 25, pp. 15-24). San Francisco, CA, USA: Determination press.
- [49] **Raschka, S., & Mirjalili, V.** (2019). *Python machine learning: Machine learning and deep learning with Python, scikit-learn, and TensorFlow 2*. Packt publishing ltd.

- [50] **Kingma, D. P., & Ba, J.** (2015). Adam: A method for stochastic optimization. In *3rd International Conference for Learning Representations*, San Diego, CA, USA, May 7-9.
- [51] **Bishop, C. M., & Nasrabadi, N. M.** (2006). Pattern recognition and machine learning (Vol. 4, No. 4, p. 738). New York: springer.
- [52] **Szeliski, R.** (2022). Computer vision: algorithms and applications. Springer Nature. doi: 10.5860/choice.48-5140.
- [53] **Matthes, E.** (2023). Python crash course: A hands-on, project-based introduction to programming. No Starch Press.
- [54] **Sweigart, A.** (2019). Automate the boring stuff with Python: practical programming for total beginners. No Starch Press.
- [55] **Ramalho, L.** (2015). Fluent Python: Clear, concise, and effective programming. " O'Reilly Media, Inc."
- [56] **Numpy.** (2024). The fundamental package for scientific computing with Python. <https://numpy.org/>
- [57] **Keras.** A superpower for ML developers. <https://keras.io/>
- [58] **Matplotlib.** Comprehensive library for creating static, animated, and interactive visualizations. <https://matplotlib.org/stable/>
- [59] **OpenCV.** Open source computer vision and machine learning software library. <https://opencv.org/about/>
- [60] **Diez-Tomillo, J., Martinez-Alpiste, I., Golcarenarenji, G., Wang, Q., & Alcaraz-Calero, J. M.** (2024). Efficient CNN-based low-resolution facial detection from UAVs, *Neural Computing and Applications*, 36(11), 5847-5860. doi: 10.1007/s00521-023-09401-3.
- [61] **Al-Owais, A., Sharif, M. E., Ghali, S., Abu Serdaneh, M., Belal, O., & Fernini, I.** (2023). Meteor detection and localization using YOLOv3 and YOLOv4, *Neural Computing and Applications*, 35(21), 15709-15720. doi: 10.1007/s00521-023-08575-0.
- [62] **Gündüz, M. Ş., & Işık, G.** (2023). A new YOLO-based method for social distancing from real-time videos, *Neural Computing and Applications*, 35(21), 15261-15271. doi: 10.1007/s00521-023-08556-3.
- [63] **Talaat, F. M., & ZainEldin, H.** (2023). An improved fire detection approach based on YOLO-v8 for smart cities, *Neural Computing and Applications*, 35(28), 20939-20954. doi: 10.1007/s00521-023-08809-1.
- [64] **Lopez-Montiel, M., Orozco-Rosas, U., Sánchez-Adame, M., Picos, K., & Ross, O. H. M.** (2021). Evaluation method of deep learning-based embedded systems for traffic sign detection, *IEEE Access*, 9, 101217-101238. doi: 10.1109/ACCESS.2021.3097969.
- [65] **Chu, J., Zhang, C., Yan, M., Zhang, H., & Ge, T.** (2023). TRD-YOLO: a real-time, high-performance small traffic sign detection algorithm, *Sensors*, 23(8), 3871. doi: 10.3390/s23083871.

- [66] **Wang, J., & Perez, L.** (2017). The effectiveness of data augmentation in image classification using deep learning, *Convolutional Neural Networks Vis. Recognit*, 11(2017), 1-8.
- [67] **Kotsiantis, S. B., Kanellopoulos, D., & Pintelas, P. E.** (2006). Data preprocessing for supervised learning, *International journal of computer science*, 1(2), 111-117. doi: 10.1080/02331931003692557.
- [68] **Geiger, A., Lenz, P., Stiller, C., & Urtasun, R.** (2013). Vision meets robotics: The kitti dataset, *The international journal of robotics research*, 32(11), 1231-1237. doi: 10.1177/0278364913491297.



ÖZGEÇMİŞ

Ad-Soyad : Hilal GEZGİN

ÖĞRENİM DURUMU:

- **Lisans** : 2009, Süleyman Demirel Üniversitesi, Teknik Eğitim Fakültesi, Bilgisayar ve Kontrol Öğretmenliği
- **Lisans** : 2016, Trakya Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği
- **Yükseklisans** : 2018, İstanbul Teknik Üniversitesi, Bilişim Enstitüsü, Bilişim Uygulamaları

MESLEKİ DENEYİM VE ÖDÜLLER:

- 2009-2010 yılları arasında Altınay Koleji'nde Bilişim Teknolojileri Öğretmeni olarak çalıştı.
- 2010-2011 yılları arasında Atılım Koleji'nde Bilişim Teknolojileri Öğretmeni olarak çalıştı.
- 2011-2012 yılları arasında Bilkey Mesleki Eğitim Kursları'nda Bilişim Teknolojileri Öğretmeni olarak çalıştı.
- 2011-2012 yılları arasında Aydos Cebir Koleji'nde Bilişim Teknolojileri Öğretmeni olarak çalıştı.
- 2012-2013 yılları arasında Avrasya Üniversitesi'nde Bilgi İşlem Müdürü olarak çalıştı.
- 2012-2013 yılları arasında Avrasya Üniversitesi Bilgisayar Programcılığı Programı'nda Öğretim Görevlisi (Bölüm başkanı) olarak çalıştı.
- 2013-2014 yılları arasında Era Teknik Koleji'nde Bilişim Teknolojileri Öğretmeni olarak çalıştı.
- 2014-2020 yılları arasında İstanbul Topkapı Üniversitesi Plato Meslek Yüksekokulu'nda Öğretim Görevlisi (Program başkanı) olarak çalıştı.
- 2021-2022 yılları arasında Çanakkale Onsekiz Mart Üniversitesi Bilgi İşlem Daire Başkanlığı'nda Öğretim Görevlisi olarak çalıştı.
- 2018- İstanbul Teknik Üniversitesi Bilimsel Araştırma Projeleri'nde araştırmacı olarak çalıştı.
- 2022- Bursa Uludağ Üniversitesi Bilgi-İşlem Daire Başkanlığı'nda Öğretim Görevlisi olarak çalışıyor.

DOKTORA TEZİNDEN TÜRETİLEN YAYINLAR, SUNUMLAR VE PATENTLER:

- **Gezgin, H., & Alkan, R. M.** (2024). Traffic sign detection and recognition based on MMS data using YOLOv4-Tiny algorithm, *Neural Computing and Applications*, 36(33), 20633-20651. doi: 10.1007/s00521-024-10279-y.
- **Gezgin, H., & Alkan, R. M.** (2024). Detection and Recognition of Traffic Signs from Data Collected by the Mobile Mapping System. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 48, 183-188.

DİĞER YAYINLAR, SUNUMLAR VE PATENTLER:

- **Türk, H., & Töreym, B. U.** (2018). Soil moisture estimation with inverse distance weighting for home agriculture applications. In *26th Signal Processing and Communications Applications Conference (SIU)* (pp. 1-4). İzmir, Turkey, May 2-5.
- **Türk, H., & Alkan, R. M.** (2019). Smart City Applications. *XXIX International Symposium on: "Modern Technologies, Education and Professional Practice in Geodesy and Related Fields"* (with a special session for young scientists and students), Istanbul, Turkey.
- **Türk H.** (2018). Sağlık Alanında Mobil Uygulamaların Kullanımı. *Sağlık Mesleki Eğitim Kongresi*, Istanbul, Türkiye.