

KOCAELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ELEKTRONİK VE HABERLEŞME MÜHENDİSLİĞİ
ANABİLİM DALI

YÜKSEK LİSANS TEZİ

OTONOM ARAÇLAR İÇİN DERİN ÖĞRENME İLE
ÇOKLU ŞERİT ALGILAMA VE ŞERİT TİPİ
SINIFLANDIRMA SİSTEMİ

SİNAN ÖZEN

KOCAELİ 2023

KOCAELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ELEKTRONİK VE HABERLEŞME MÜHENDİSLİĞİ
ANABİLİM DALI

YÜKSEK LİSANS TEZİ

OTONOM ARAÇLAR İÇİN DERİN ÖĞRENME İLE ÇOKLU
ŞERİT ALGILAMA VE ŞERİT TİPİ SINIFLANDIRMA
SİSTEMİ

SİNAN ÖZEN

Doç. Dr. Aysun TAŞYAPI ÇELEBİ

Danışman, Kocaeli Üniversitesi

.....

Doç. Dr. Orhan AKBULUT

Jüri Üyesi, Kocaeli Üniversitesi

.....

Doç. Dr. Ali Can KARACA

Jüri Üyesi, Yıldız Teknik Üniversitesi

.....

Tezin Savunulduğu Tarih: 13.06.2023

ETİK BEYAN VE ARAŞTIRMA FONU DESTEĞİ

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

- Bu tezin bana ait, özgün bir çalışma olduğunu,
- Çalışmamın hazırlık, veri toplama, analiz ve bilgilerin sunumu olmak üzere tüm aşamalarında bilimsel etik ilke ve kurallara uygun davrandığımı,
- Bu çalışma kapsamında elde edilen tüm veri ve bilgiler için kaynak gösterdiğimi ve bu kaynaklara kaynak gösterdiğimi ve bu kaynaklara kaynakçada yer verdiğimi,
- Bu çalışmanın Kocaeli Üniversitesi'nin abone olduğu intihal yazılım programı kullanılarak Fen Bilimleri Enstitüsü'nün belirlemiş olduğu ölçütlere uygun olduğunu,
- Kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- Tezin/Projenin herhangi bir bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı,

beyan ederim.

☒ Bu tez/proje çalışmasının herhangi bir aşaması hiçbir kurum/kuruluş tarafından maddi/alt yapı desteği ile desteklenmemiştir.

☐ Bu tez/proje çalışması kapsamında üretilen veri ve bilgiler tarafından no'lu proje kapsamında maddi/alt yapı desteği alınarak gerçekleştirilmiştir.

Herhangi bir zamanda, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun saptanması durumunda, ortaya çıkacak tüm ahlaki ve hukuki sonuçları kabul ettiğimi bildiririm.

.....

Sinan ÖZEN

YAYINLAMA VE FİKRİ MÜLKİYET HAKLARI

Fen Bilimleri Enstitüsü tarafından onaylanan lisansüstü tezimin/projemin tamamını veya herhangi bir kısmını, basılı ve elektronik formatta arşivleme ve aşağıda belirtilen koşullarla kullanıma açma izninin Kocaeli Üniversitesi'ne verdiğimi beyan ederim. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet haklarımı bende kalacak, tezimin/projemin tamamının ya da bir bölümünün gelecekteki çalışmalarda (makale, kitap, lisans ve patent vb.) kullanımı bana ait olacaktır.

Tezin/projenin kendi özgün çalışmam olduğunu, başkalarının haklarını ihlal etmediğimi ve tezimin/projenin tek yetkili sahibi olduğumu beyan ve taahhüt ederim. Tezimde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanılması zorunlu metinlerin yazılı izin alarak kullandığımı ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederim.

Yüksek öğretim kurulu tarafından yayınlanan **“Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge”** kapsamında tezim aşağıda belirtilen koşullar haricinde YÖK Ulusal Tez Merkezi/ Kocaeli Üniversitesi Kütüphaneleri Açık Erişim Sisteminde erişime açılır.

- ☐ Enstitü yönetim kurulu kararı ile tezimin/projemin erişime açılması mezuniyet tarihinden itibaren 2 yıl ertelenmiştir.
- ☐ Enstitü yönetim kurulu gerekçeli kararı ile tezimin/projemin erişime açılması mezuniyet tarihinden itibaren 6 ay ertelenmiştir.
- ☒ Tezim/projem ile ilgili gizlilik kararı verilmemiştir.

.....

Sinan ÖZEN

ÖNSÖZ VE TEŞEKKÜR

Bu tez çalışmasında, otonom araçlar için çoklu şerit algılama ve şerit tiplerinin sınıflandırılması başarılı bir şekilde gerçekleştirilmiştir. Yaptığımız bu çalışmanın bu alandaki çalışan ve çalışacak olan arkadaşlara yararlı olmasını ümit ediyorum.

Çalışmalarım süresince bana her zaman destek olan ve yol gösteren değerli hocam Doç. Dr. Aysun TAŞYAPI ÇELEBİ'ye çok teşekkür ederim.

Tüm eğitim hayatım boyunca benden desteğini esirgemeyen sevgili annem Hasibe ÖZEN, babam İsmet ÖZEN, abim Kenan ÖZEN'e çok teşekkür ederim.

Lisans ve yüksek lisans eğitimim boyunca bana her zaman destek olan, kendisinden çok şey öğrendiğim değerli hocam Prof. Dr. Oğuzhan URHAN'a çok teşekkür ederim.

Mayıs – 2023

Sinan ÖZEN

İÇİNDEKİLER

ETİK BEYAN VE ARAŞTIRMA FONU DESTEĞİ.....	i
YAYINLAMA VE FİKRİ MÜLKİYET HAKLARI.....	ii
ÖNSÖZ VE TEŞEKKÜR.....	iii
İÇİNDEKİLER.....	iv
ŞEKİLLER DİZİNİ.....	vi
TABLOLAR DİZİNİ.....	vii
SİMGELER VE KISALTMALAR DİZİNİ.....	viii
ÖZET.....	ix
ABSTRACT.....	x
1. GİRİŞ.....	1
2. ÇOKLU ŞERİT ALGILAMA.....	3
3. LİTERATÜRDEKİ ÇALIŞMALAR.....	6
3.1. Şerit Algılamayla İlgili Görü Temelli Çalışmalar.....	7
3.1.1. Kenar Tabanlı Yöntemler.....	7
3.1.2. Özelliğe Dayalı Yöntemler.....	9
3.1.3. Uyarlanabilir Yöntemler.....	10
3.1.4. Model Tabanlı Yöntemler.....	10
3.1.4.1. Segmentasyon Modelleri.....	11
3.2. Şerit Tipi Sınıflandırmak İçin Derin Öğrenme Temelli Çalışmalar.....	17
3.2.1. Evrişimli Sinir Ağları Tabanlı Sınıflandırma Modelleri.....	17
4. ÖNERİLEN ÇOKLU ŞERİT ALGILAMA SİSTEMİ.....	24
4.1. Veri Seti.....	24
4.2. Çoklu Şerit Algılama Sistemi.....	25
4.2.1. Önerilen Birinci Sistem.....	25
4.2.1.1. Segmentasyon Modelinin Uygulanması.....	26
4.2.1.2. Bağlantılı Bileşen Analizi.....	27
4.2.1.3. DBSCAN İşleminin Uygulanması.....	27
4.2.1.4. Sistemin Performansı.....	28
4.2.2. Önerilen İkinci Sistem.....	28
4.2.2.1. Perspektif Dönüşümün Uygulanması.....	29
4.2.2.2. Aracın Bulunduğu Şeritlerin Tanımlanması.....	30
4.2.2.3. Şerit Kontur Bilgilerinin Çıkarılması.....	33
4.2.2.4. Sistemin Performansı.....	34
4.3. Modellerin Eğitimi.....	35
4.4. Modellerin Karşılaştırılması.....	35
4.5. Model Seçimi.....	36
4.6. Deneysel Sonuçlar.....	36
5. ÖNERİLEN ŞERİT TİPİ SINIFLANDIRMA SİSTEMİ.....	38
5.1. Veri Seti.....	38
5.2. Şerit Tipi Sınıflandırılması Sistemi.....	38
5.2.1. Çoklu Şerit Algılama Sistemi ile Şerit Bilgilerinin Çıkarımı.....	39
5.2.2. Perspektif Görüntüden İlgili Bölgesi Çıkarımı.....	39
5.2.3. Sınıflandırma Modelinin Uygulanması.....	41
5.3. Modellerin Eğitimi.....	41
5.4. Modellerin Karşılaştırılması.....	41
5.5. Model Seçimi.....	42

6. SONUÇLAR VE ÖNERİLER.....	43
KAYNAKLAR.....	44
KİŞİSEL YAYIN VE ESERLER	48
ÖZGEÇMİŞ.....	49



ŞEKİLLER DİZİNİ

Şekil 2.1.	Kuzey Marmara Otoyolu.....	3
Şekil 2.2.	Gölgeli şeritler.....	4
Şekil 2.3.	Güneş ışığının aşırı parlaması	4
Şekil 2.4.	Çoklu şerit üzerinde araç gösterimi.....	5
Şekil 3.1.	Sensör türleri	6
Şekil 3.2.	Şerit algılama yöntemleri	7
Şekil 3.3.	ShuffleNetV1 mimarisi	18
Şekil 3.4.	ImageNet için örnek ağ mimarileri	19
Şekil 3.5.	SqueezeNet mimarsinin makro mimari görünümü	20
Şekil 3.6.	MBConv ve Fused-MBConv'un Yapısı	21
Şekil 3.7.	MobileNetV2 Modülü	22
Şekil 3.8.	MobileNetV3 Modülü	22
Şekil 3.9.	Sigmoid ve swish doğrusalsızlıkları ve "sert" karşılıkları.....	23
Şekil 4.1.	TuSimple veri kümesi	24
Şekil 4.2.	TuSimple JSON formatı.....	25
Şekil 4.3.	Önerilen birinci çoklu şerit algılama sistemi.....	25
Şekil 4.4.	Yerleşik araç kamerasından alınan ham görüntü	26
Şekil 4.5.	Segmentasyon işlemi sonrası oluşan ikili görüntü	26
Şekil 4.6.	CCA işlemi öncesi ve sonrası oluşan ikili görüntü	27
Şekil 4.7.	DBSCAN işlemi öncesi ve sonrası oluşan kümelenmiş şeritler	27
Şekil 4.8.	Çoklu şerit algılama sistemi	28
Şekil 4.9.	Giriş görüntüsü üzerindeki 4 nokta	29
Şekil 4.10.	Perspektif dönüşüm uygulanmış ikili görüntü	30
Şekil 4.11.	Perspektif dönüşüm uygulanmış giriş görüntüsü	30
Şekil 4.12.	Sol ve sağ şeritlerin görselleştirilmesi.....	31
Şekil 4.13.	Eğri uydurma işlemi	32
Şekil 4.14.	Şeritlerin konturlarının sınırlayıcı kutu gösterimi	34
Şekil 4.15.	Bulutlu hava test görüntüsü sonuçları	37
Şekil 4.16.	Güneşli hava test görüntüsü sonuçları.....	37
Şekil 5.1.	Şerit tipi veri kümesi. Sol: kesikli, orta: şerit yok, sağ: düz şerit.....	38
Şekil 5.2.	Şerit Tipi Sınıflandırma Sistemi.....	39
Şekil 5.3.	Sol ve sağ şeritlerin kırılmış görüntüleri	40

TABLÖLAR DİZİNİ

Tablo 4.1.	Önerilen birinci sistemin performans sonuçları	28
Tablo 4.2.	MOBILEYE cihazındaki parametrelerin anlamı	33
Tablo 4.3.	Sol şerit için kat sayı örneği	33
Tablo 4.4.	Sağ şerit için kat sayı örneği	33
Tablo 4.5.	Önerilen ikinci sistemin performans sonuçları	34
Tablo 4.6.	Modellerin eğitim ve test sonuçları.....	35
Tablo 4.7.	Modellerin performans sonuçları	35
Tablo 4.8.	Segmentasyon modellerinin FPS sonuçları.....	36
Tablo 5.1.	Modellerin eğitim ve test sonuçları.....	41
Tablo 5.2.	Modellerin test sonuçları.....	42
Tablo 5.3.	Modellerin FPS, parametre sayısı ve dosya boyutu	42



SİMGELER VE KISALTMALAR DİZİNİ

cm	: Santimetre, (m)
m	: Metre, (m)

Kısaltmalar

ACO	: Ant Colony Optimization (Karıncı Kolonisi Optimizasyonu)
ADAS	: Advanced Driver Assistance Systems (Gelişmiş Sürücü Destek Sistemleri)
CCA	: Connected Component Analysis (Bağlantılı Bileşen Analizi)
CNN	: Convolutional Neural Network (Evrişimli Sinir Ağı)
CT	: Computed Tomography (Bilgisayarlı Tomografi)
DBSCAN	: Density-Based Spatial Clustering of Applications with Noise (Gürültülü Uygulamaların Yoğunluk Tabanlı Uzamsal Kümelenmesi)
DNN	: Deep Neural Network (Derin Sinir Ağı)
FCN	: Fully Convolutional Network (Tamamen Evrişimli Ağ)
FPS	: Frame Per Second (Saniye Başına Çerçeve)
GCN	: Graph Convolutional Network (Grafiği Evrişimli Ağ)
IoU	: Intersection over Union (Kesişim Üzeri Birleşim)
JSON	: JavaScript Object Notation (JavaScript Nesnesi Gösterimi)
MIoU	: Mean Intersection Over Union (Ortalama Kesişim-Üzeri-Birim)
MRG	: Magnetic Resonance Imaging (Manyetik Rezonans Görüntüleme)
NAS	: Neural Architecture Search (Sinir Mimarisi Arama)
RANSAC	: RANdom SAMple Consensus (Rastgele Örnek Onaylaşımı)
RNN	: Recurrent Neural Network (Tekrarlayan Sinir Ağı)
ROI	: Region of Interest (İlgi Alanı)
SVM	: Support Vector Machine (Destek Vektör Makinesi)

OTONOM ARAÇLAR İÇİN DERİN ÖĞRENME İLE ÇOKLU ŞERİT ALGILAMA VE ŞERİT TİPİ SINIFLANDIRMA SİSTEMİ

ÖZET

Şerit algılama ve şerit tipi sınıflandırması, otonom araçların kazaların önlenmesine ve yolcu güvenliğinin sağlanmasına önemli ölçüde katkıda bulunan çok önemli bileşenlerdir. Otonom araç üretimi ve kullanımına olan talep sürekli olarak artıyor ve en son teknolojik gelişmeler bu sistemlerin doğruluğunu ve etkinliğini önemli ölçüde artırdı. Yoldaki şeritlerin doğru bir şekilde tanımlanması ve sınıflandırılması, özellikle karmaşık sürüş senaryolarında güvenli navigasyonun sağlanması için çok önemlidir. Bu tez çalışmasında, şerit algılama ve şerit tipi sınıflandırma sistemini derin öğrenme yöntemleri ile gerçekleştirilir. Bu tez çalışmasının amacı, derin öğrenme yöntemleri kullanılarak şerit algılama ve şerit tipi sınıflandırma sistemini yüksek bir performans ile gerçekleştirmektir. Bu çalışma kapsamında şerit algılama için 4 farklı model, şerit tipi sınıflandırma için 5 farklı model eğitilmiş ve performansları test verileri üzerinde hesaplanır. Her iki model içinde en başarılı modeller seçilir. Önerilen yöntemde otonom araç üzerinde başarılı bir şekilde uygulanır. Deneysel sonuçlar, tez kapsamında geliştirilen şerit algılama ve şerit tipi sınıflandırma sisteminin başarılı bir şekilde geliştirilmiş olduğunu gösterir. Geliştirilen gerçek zamanlı sistem otonom araç üzerinde güvenli bir şekilde entegre edilir.

Anahtar Kelimeler: Derin Öğrenme, İmge İşleme, Otonom Araç, Şerit Tespiti, Şerit Tipi Sınıflandırma.

MULTI-LANE DETECTION AND LANE TYPE CLASSIFICATION SYSTEM WITH DEEP LEARNING FOR AUTONOMOUS VEHICLES

ABSTRACT

Lane detection and lane type classification are very important components of autonomous vehicles that contribute significantly to preventing accidents and ensuring passenger safety. The demand for autonomous vehicle production and use is constantly increasing, and the latest technological advances have significantly increased the accuracy and effectiveness of these systems. Accurate identification and classification of lanes on the road is essential to ensure safe navigation, especially in complex driving scenarios. In this thesis, lane detection and lane type classification system are implemented by deep learning methods. The aim of this thesis is to realize lane detection and lane type classification system with high performance by using deep learning methods. In this study, 4 different models for lane detection and 5 different models for lane type classification are trained and their performances are calculated on the test data. Among both models, the most successful models are selected. In the proposed method, it is successfully applied on an autonomous vehicle. Experimental results show that the lane detection and lane type classification system developed in the thesis has been successfully developed. The developed real-time system is securely integrated on the autonomous vehicle.

Keywords: Deep Learning, Image Processing, Autonomous Vehicle, Lane Detection, Lane Type Classification.

1. GİRİŞ

Günümüzde otoyollardaki araç sayısı her geçen gün artmaktadır. Bu artışla beraber trafik güvenliğinin önemini artırmaktadır. Ülkemizde Karayolları Genel Müdürlüğü'nün Trafik Kazaları Özeti 2021 (T.C. Karayolları Genel Müdürlüğü, 2021) raporuna göre toplam 1.186.353 kaza gerçekleşir. Bu kazalardan 187.963'ü ölümle sonuçlanır. Rapora göre kazaların %87,23 sürücüler tarafından gerçekleşir.

Bu kazaların önlenmesi için otomotiv endüstrisinde sürücü destek sistemleri geliştirmesi hızlanır. Bu sebeple günümüzde birçok sürüş destek sistemi araçlara entegre edilir. Sürüş Güvenliği Destek Sistemleri, sürücülerin güvenli bir şekilde araç kullanmalarına yardımcı olur. Sürücünün görüş alanında olmayan arabaları, motosikletleri ve yayaları algılamak için çeşitli sensörler kullanılır. Şeritte Kalma Sistemi (Lane-Keeping System), istemeden şeritten çıkıp çıkmadığını algılamada kullanılan bir sürüş destek sistemidir. Şeritte Kalma Uyarısı daha sonra direksiyon simidini titreterek ve görsel bir uyarı görüntüleyerek uyarır. Tepki vermekte yavaşsa veya yanıt vermiyorsa, Şeritte Kalma Yardımı güvenli bir şekilde tekrar doğru şeride girmenize yardımcı olmak için direksiyon torku yardımı sağlayabilir.

Karayolları raporuna göre kazaların %11,05'i şerit izleme ve değiştirme kurallarına uymamaktan kaynaklanır. Sürücülerin otoyollarda yorgunluktan dolayı uykuya dalma problemlerini oluşturur. Bu problemin önüne geçilebilmesi durumunda şerit değiştirme gibi durumların önüne geçilmesi öngörülmektedir.

Bu tez çalışması kapsamında otonom araçlar için gerçek zamanlı çoklu şerit algılama ve şerit tipi sınıflandırılma önerilmektedir. Derin öğrenme, bilgisayar görüşü ve nesne algılama alanında önemli bir ilerleme kaydetmiştir ve çoklu şerit algılama ve şerit tipi sınıflandırma için potansiyel bir çözüm sunmaktadır.

Bu çalışma, otomatik sürüş sistemleri ve sürücü destek sistemleri gibi ileri sürüş teknolojilerine önemli bir katkı sağlayabilir. Geliştirilecek çoklu şerit algılama algoritması, otomobillerin doğru ve güvenli bir şekilde şerit değiştirmesini sağlayarak trafik akışını iyileştirebilir ve trafik kazalarını önlemede etkili bir rol oynayabilir. Aynı

zamanda, sürücülerin dikkatini trafik koşullarına odaklanmasına yardımcı olarak sürüş konforunu artırabilir.

Bu tez çalışmasında derin öğrenme tabanlı bir sistem geliştirerek, mevcut yöntemlerin sınırlamalarını aşmayı ve daha doğru sonuçlar elde etmeyi hedeflemektedir. Bu tez çalışmasının literatüre katkıları aşağıda yer almaktadır.

- Segmentasyon tabanlı derin öğrenme modellerinin performans metriklerinin karşılaştırılması ve 136,7 FPS değerinde modelle sistemin gerçekleştirilmesi.
- Otonom aracın bulunduğu şeritleri derin öğrenme modeliyle sınıflandırıp yüksek doğrulukta ve 121,25 FPS değeri ile şerit tipinin sınıflandırılması.
- Otoyoldaki tüm şeritlerin algılanması ve sınıflandırılması ile güvenli sürüş sağlayan sistemin gerçekleştirilmesi.

Tez çalışmasında birinci bölümde bu çalışmanın önemi gösteren istatistiksel veriler gösterilmiştir ve bu alandaki yapılan çalışmalar hakkında genel bilgiler verilir. Aynı bölümde önerilen yöntemin çalışma mantığı ve önerilen yöntemin literatüre katkıları da açıklanır.

Tez çalışmasının ikinci bölümünde çoklu şerit algılama sistemi hakkında bilgiler verilir ve bu sistemlerde karşılaşılan zorluklar özgün veriler ile gösterilir.

Tez çalışmasının üçüncü bölümünde çoklu şerit algılama ve şerit tipi sınıflandırması ile ilgili yapılan literatür taraması detaylı bir şekilde yapılır. Literatürde var olan derin öğrenme yöntemler detaylı bir şekilde açıklanır. Tez çalışmasının dördüncü bölümünde veri seti, çoklu şerit algılama sistemi, segmentasyon modelinin uygulanması, perspektif dönüşümün uygulanması, aracın bulunduğu şeritlerin çıkarılması, modellerin eğitimi, modellerin karşılaştırılması ve model seçimi konuları detaylı bir şekilde açıklanır. Tez çalışmasının beşinci bölümünde veri seti, şerit tipi sınıflandırma sistemi, algılama sistemi, segmentasyon modelinin uygulanması, perspektif dönüşümün uygulanması, aracın bulunduğu şeritlerin çıkarılması, modellerin eğitimi, modellerin karşılaştırılması ve model seçimi konuları detaylı bir şekilde açıklanır. Bu tez çalışmasının son bölümünde ise, tez kapsamında gerçekleştirilen çoklu şerit algılama ve şerit tipi sınıflandırma sisteminin literatürdeki bilimsel katkıları açıklanır. Bu alanda gerçekleştirilecek çalışmalar için önerilere yer verilir.

2. ÇOKLU ŞERİT ALGILAMA

Günümüzde artan otomobil sayısı ile Gelişmiş Sürücü Yardım Sistemlerine (Advanced Driver Assistance Systems-ADAS) ilgi artmaktadır. Gelişmiş sürücü sistemlerinin temel amacı sürücülere araçlarını daha güvenli bir şekilde kullanmalarına yardımcı olmak için tasarlanır. Temel amacı, sürüş esnasında meydana gelebilecek kazaları önlemek ya da etkilerini azaltmak ve sürüş deneyimini geliştirmektir.

Şerit algılama, ADAS'ın önemli bir özelliğidir. Bu özellik, aracın etrafındaki şeritleri algılayarak sürücüyü uyarır ve gerekirse aracın şeridini değiştirmesine yardımcı olur. Şerit algılama sistemi, sürücülerin yorgun ya da dikkatsiz olmaları durumunda kazaları önlemek için önemlidir.

Örneğin, bir sürücü yorgunluktan dolayı aracın şeridini aşırı derecede değiştiriyorsa, şerit algılama sistemi sürücüyü uyarabilir ve aracın şeridini yeniden hizalamasına yardımcı olabilir. Böylece sürücü, olası bir kaza riskini azaltır ve sürüş güvenliğini artırır. Ayrıca, şerit algılama sistemi, bir sürücü diğer araçların yanına yaklaşırken veya araç değiştirirken çevredeki diğer araçları ve engelleri tespit ederek sürücüyü uyarabilir. Bu da sürücülerin araç trafiğinde daha güvenli bir şekilde hareket etmelerine yardımcı olabilir.

Otoyollarda bulunan şeritler örnek vermek amacıyla, Şekil 2.1'de Kuzey Marmara Otoyolundan alınmış bir görüntü gösterilir. Görüntüde görüldüğü gibi sağ ve sol tarafta bulunan kesikli ve devamlı şerit işaretleriyle ayrılmaktadırlar.



Şekil 2.1. Kuzey Marmara Otoyolu

Şerit algılama sistemlerinin çalışması için öncelikle şeritlerin tipleri belirlenmelidir. Şerit tipleri kesikli ve devamlı şerit olarak iki sınıfta ele alınır. Şerit algılama sistemlerinde, sistem çevresel faktörlerden etkilenmektedir. Bu faktörler şeritlerin belirgin olmaması aşınmış olması, güneş ışığının yetersiz olduğu karanlık durumlar, hava şartlarından dolayı kapanan şerit çizgileri gibi durumlardır.



Şekil 2.2. Gölge şeritler

Şekil 2.2’de görüldüğü gibi araç köprü altından geçerken şeritlerin bir bölümü gölge altında kalmıştır. Bu durum şerit algılama sistemlerini olumsuz etkilemektedir.



Şekil 2.3. Güneş ışığının aşırı parlaması

Şekil 2.3’te görüldüğü gibi köprü ve tünel çıkışlarında kameraya yansıyan görüntüdür. Bu durum aşırı ışıklılık olarak değerlendirilmektedir. Burada güneş ışığının kameranın

görüş sistemini olumsuz etkilediği görülür. Bu ve buna benzer birçok durum şerit algılama sistemlerini olumsuz bir şekilde etkilemektedir.

Bu tez kapsamında geliştirilen sistem ile bahsi geçen olumsuz durumların üstesinden gelinir ve gürbüz bir şekilde çalışması sağlanır.

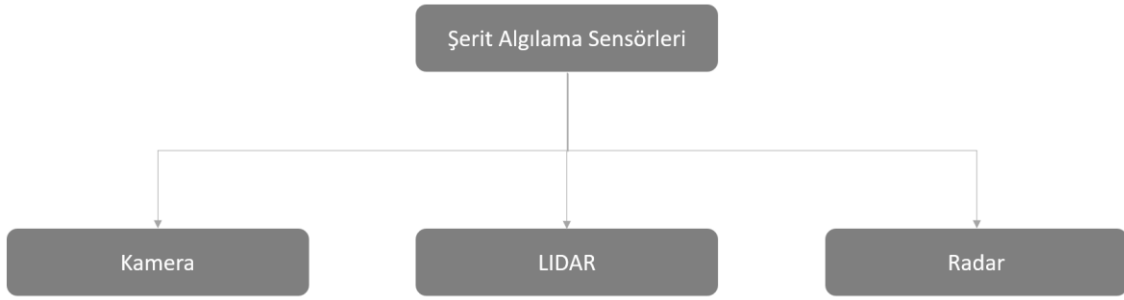
Şekil 2.4'te gösterilen görüntüde bir aracın otoyol üzerindeki halidir. Çoklu şerit algılama sisteminde otoyoldaki aracın yoldaki tüm şeritleri algılaması, şeritlerin tipini algılayıp hangi şerit üzerinde hareket halinde olduğunu tespit etmesi hedeflenir.



Şekil 2.4. Çoklu şerit üzerinde araç gösterimi

3. LİTERATÜRDEKİ ÇALIŞMALAR

Literatürde kameralar, LIDAR (Light Detection and Ranging) ve radar dahil olmak üzere çeşitli sensör türleri kullanılarak gerçekleştirilir. Literatürde şerit tespiti için kullanılan farklı sensör türleri Şekil 3.1’de gösterilir.



Şekil 3.1. Sensör türleri

Kameralar, şerit tespiti için kullanılan en yaygın sensörlerden biridir. Yolun yüksek çözünürlüklü görüntülerini yakalayabilirler ve boyalı çizgiler veya yol kenarları gibi şerit işaretlerini tespit etmek için kullanılabilirler. Kamera görüntülerinden şeritleri algılamak ve izlemek için kenar algılama, Hough dönüşümü ve makine öğrenimi algoritmaları gibi görüntü işleme teknikleri kullanılabilir.

LIDAR sensörleri, şerit işaretleri dahil olmak üzere ortamın doğru üç boyutlu nokta bulutu verilerini sağlayabilir. LIDAR şerit kenarlarını, bordürleri ve yoldaki diğer nesneleri algılayabilir, bu da onu şerit algılama için değerli bir sensör haline getirir. LIDAR tabanlı şerit algılama sistemleri, nokta bulutu verilerinden şeritleri çıkarmak ve izlemek için tipik olarak kümeleme, segmentasyon ve eğri uydurma gibi algoritmalar kullanılır.

Radar sensörleri, öncelikle yol kenarlarını tespit ederek şerit tespiti için de kullanılabilir. Radar tabanlı şerit algılama sistemleri, gölgeler veya zayıf aydınlatma koşulları nedeniyle gizlenenler gibi kamera görüntülerinde görünmeyenler de dahil olmak üzere şerit sınırlarını algılayabilir.

Bu sensörlere ek olarak, literatürdeki bazı çalışmalar, şerit algılama doğruluğunu iyileştirmek için bir sensör kombinasyonunun kullanılması önerilir. Örneğin, kamera ve

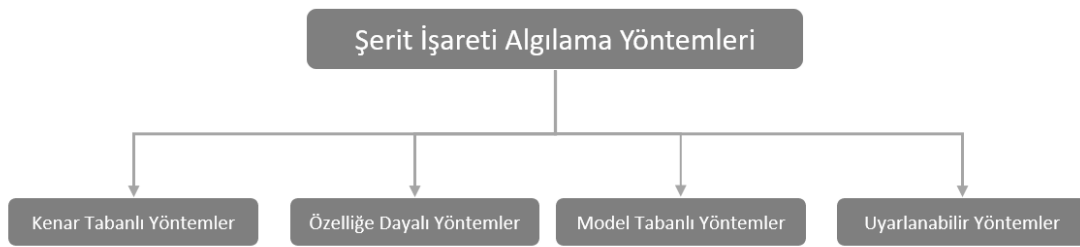
LIDAR verilerini birleştirmek, zorlu aydınlatma ve hava koşullarında daha sağlam şerit algılama sağlayabilir.

Bu çalışma kapsamında sadece kamera tabanlı yapılan görüntü temelli algılama teknikleri üzerinde durulur. Kamera tabanlı çalışmalar literatürde görü olarak adlandırılmaktadır.

3.1. Şerit Algılamayla İlgili Görü Temelli Çalışmalar

Görü temelli çalışmalar, kameralar veya diğer sensörler tarafından yakalanan görüntülerin veya videoların analizini, yorumlanmasını ve anlaşılmasını içeren bilgisayarla görme alanındaki araştırma ve geliştirmeleri ifade eder. Görüntü tabanlı sistemler, nesne algılama ve tanıma, görüntü bölümleme, izleme ve daha fazlasını içeren çok çeşitli uygulamalar için kullanılabilir.

Görüntü tabanlı çalışmalarda, araştırmacılar genellikle görüntü verilerini analiz etmek ve ilgili bilgileri çıkarmak için görüntü işleme tekniklerini ve makine öğrenimi algoritmalarını kullanır. Örneğin, şerit algılamada araştırmacılar, görüntüdeki şerit işaretlerini algılamak ve zaman içinde izlemek için kenar algılama algoritmaları veya makine öğrenimi modelleri kullanabilir. Şekil 3.2’de şerit işareti algılama yöntemlerinin çeşitleri gösterilir.



Şekil 3.2. Şerit algılama yöntemleri

3.1.1. Kenar Tabanlı Yöntemler

Kenar tabanlı yöntemler, görüntüdeki şerit işaretlerini algılamak için Canny kenar algılama (Daigavane ve Bajaj, 2010), Sobel operatörü (Humaidi ve diğ., 2018) veya Laplacian of Gaussian (Low ve diğ., 2014) gibi kenar algılama algoritmalarını kullanır.

Daha sonra şeritleri çıkarmak ve izlemek için Hough dönüşümü (Niu ve diğ., 2016) veya eğri uydurma (Sun ve diğ., 2011) algoritmaları kullanılabilir.

Önerilen Canny kenar algılama (Daigavane ve Bajaj, 2010) algoritması ve karınca kolonisi optimizasyonu (ACO) tekniğini kullanarak yol şeritlerini tespit etmek için bir yöntem sunulur. Bu çalışmada iki popüler tekniği, Canny kenar algılama ve ACO'yu birleştiren yol şeridi algılama sunulur.

Önerilen algoritma (Humaidi ve diğ., 2018) üç ana adımdan oluşur: kenar algılama, şerit işaretleme çıkarma ve şerit izleme. İlk adımda, Sobel operatörü kullanılarak giriş görüntüsündeki kenarlar algılanır. İkinci adımda, şerit işaretlerini kenar haritasından çıkarmak için Hough dönüşümü kullanılır. Üçüncü adımda, tespit edilen şeritleri izlemek ve aracın şeritlere göre konumunu tahmin etmek için bir şerit izleme algoritması kullanılır. Şerit izleme algoritması, şeritlerin sabit bir genişliğe sahip olduğunu ve birbirine paralel olduğunu varsayan bir modele dayanmaktadır. Model, bir sonraki çerçevedeki şeritlerin konumunu tahmin etmek ve geçerli çerçevedeki algılanan şerit işaretlerini kullanarak tahmini düzeltmek için kullanılır.

Önerilen yöntem (Low ve diğ., 2014) üç ana modülden oluşmaktadır: görüntü elde etme, kenar algılama ve şerit algılama. Birinci modülde, giriş görüntüsü araca monte edilmiş bir kamera kullanılarak yakalanır. İkinci modülde, giriş görüntüsünden kenarları çıkarmak için Sobel kenar algılama algoritması kullanılır. Üçüncü modülde, kenar haritasından şerit işaretlerini tespit etmek için Hough dönüşümü kullanılmıştır. Önerilen sistem bir Xilinx Virtex-5 FPGA kartında uygulanır.

Önerilen yöntem (Niu ve diğ., 2016) dört ana adımdan oluşur: görüntü ön işleme, kenar algılama, şerit aday tespiti ve şerit doğrulama. İlk adımda, giriş görüntüsü, kontrastı artırmak ve gürültüyü gidermek için önceden işlenir. İkinci adım, Canny kenar algılama algoritması kullanılarak önceden işlenmiş görüntüdeki kenarların algılanmasını içerir. Üçüncü adımda, önceki adımda tespit edilen kenarlara dayalı olarak potansiyel şerit adayları belirlenir. Son olarak, son adımda, şerit adayları bir dizi buluşsal yöntem kullanılarak doğrulanır ve en olası şerit işaretleri seçilir.

Önerilen yöntemde (Sun ve diğ., 2011) iki ana aşamadan oluşur: özellik çıkarma ve eğri uydurma. İlk aşamada, giriş görüntüsünden iki tür öznitelik çıkarılır: kenar öznitelikleri ve yoğunluk öznitelikleri. Kenar özellikleri Canny kenar algılama algoritması kullanılarak elde edilirken, yoğunluk özellikleri gri tonlamalı görüntünün eşiklenmesiyle elde edilir. Bu iki tür özelliğin kombinasyonu, potansiyel şerit işaretlerini vurgulayan ikili bir görüntü oluşturmak için kullanılır. İkinci aşamada, birinci aşamada oluşturulan ikili görüntüye bir dizi eğri sığdırmak için bir eğri uydurma algoritması kullanılır. Eğri uydurma algoritması, verilerdeki aykırı değerleri ve gürültüyü işleyebilen sağlam bir tahmin yöntemi olan RANSAC (Random Sample Consensus) algoritmasına dayanır. Bu algoritma, şeritlerin konumu ve eğriliği gibi şerit işaretlemelerinin parametrelerini çıkarmak için kullanılır.

3.1.2. Özelliğe Dayalı Yöntemler

Özelliğe dayalı yöntemler, görüntüdeki şerit işaretlerini algılamak için doku analizi, köşe algılama veya ölçekle değişmeyen özellik dönüşümü (SIFT) (Zhang ve diğ., 2005) gibi özellik çıkarma tekniklerini kullanır. Önerilen yöntemde, mobil haritalama sistemlerinden gelen görüntü dizilerinde otomatik trafik şeridi tespiti için bir algoritma önerilmiştir. Bu çalışmada iki özellik sunulmuştur. İlk olarak, şerit işaretleme özellik çıkarımı için Maksimum Kararlı Ekstrem Bölge (MSER) dedektörünü ve bölge şekillerini tanımlamak için afin değişmez bir şekilde yeni bir şekil tanımlayıcısı ve özellik görünüm özelliklerini yakalamak için değiştirilmiş bir SIFT tanımlayıcısı türetilir.

Destek vektör makineleri (SVM) (Cheng ve diğ., 2022) veya derin sinir ağları (DNN) (Li ve diğ., 2013) gibi makine öğrenimi algoritmaları, şeritleri sınıflandırmak ve izlemek için kullanılmıştır. Önerilen yöntemde (Cheng ve diğ., 2022), bir desteğe dayalı bir şerit algılama algoritması önerilir. SVM sınıflandırıcısı, giriş görüntüsünden özellikleri çıkarmak için renk ve kenar bilgilerini kullanır ve bunlar daha sonra her pikselin şeride ait olup olmadığını belirlemek için bir SVM sınıflandırıcı ile sınıflandırılır.

Önerilen yöntemde (Li ve diğ., 2013) iki boyutludan üç boyutlu şerit rekonstrüksiyonu sürecinin altındaki geometri yapısından yararlanarak monoküler üç boyutlu şerit

algılama sorununu hedeflemede gelişmiş bir yaklaşım önerilir. Bu çalışmada geometri ön bilgilerini DNN tabanlı bir yaklaşım ile elde ediyor ve uzun mesafedeki şeritleri algılamaya yardımcı olduğunu gösterilir.

3.1.3. Uyarlanabilir Yöntemler

Uyarlanabilir yöntemler, değişen aydınlatma, hava veya yol koşullarında şerit algılama doğruluğunu iyileştirmek için uyarlanabilir eşikleme, ilgi alanı (ROI) seçimi veya diğer uyarlanabilir teknikleri kullanılır. Önerilen yöntemde (Ding ve diğ., 2013), yol ve potansiyel şerit işaretlerini içeren bir girdi görüntüsünde ROI belirlemek için bir algoritma önerirler.

ROI belirleme algoritması uyarlanabilirdir ve aracın yönünü, yol eğriliğini ve görüntüdeki ufku konumunu hesaba katarlar. Önerilen yöntemlerinde daha sonra ROI içindeki şerit işaretlerini tespit etmek için bir Hough dönüşümü ve çizgi uydurma algoritması da uygulanır.

3.1.4. Model Tabanlı Yöntemler

Model tabanlı yöntemler, şerit işaretlerinin konumlarını tahmin etmek için bir şerit modeli kullanır ve şeritleri tespit etmek ve izlemek için bunları gerçek görüntü verileriyle karşılaştırılır. Model, şerit çizgilerinin matematiksel veya geometrik bir temsili veya DNN gibi veriye dayalı bir model olabilir. Model tabanlı çalışmalara örnek olarak segmentasyon tabanlı yöntemler gösterilebilir.

Segmentasyon model tabanlı şerit algılama sistemi, yoldaki şeritleri gerçek zamanlı olarak algılamak için derin öğrenme tekniklerini kullanan bir bilgisayarla görme algoritmasıdır. Sistem, yolu farklı bölgelere ayırarak ve ardından bu bölgelerdeki şerit işaretlerini belirleyerek çalışır.

Segmentasyon modeli, şerit işaretlemelerinin konumu ile etiketlenmiş geniş bir yol görüntüleri veri kümesi üzerinde eğitilir. Model, şerit işaretlemelerinin rengi, genişliği ve konumu gibi benzersiz özelliklerini tanımlamayı öğrenir ve bu bilgiyi görüntüdeki pikselleri şeride ait olup olmadıklarını sınıflandırmak için kullanır. Segmentasyon

modeli şerit işaretlerini belirledikten sonra, şerit işaretlerine bir eğri uydurmak ve aracın şerit içindeki konumunu tahmin etmek için bir şerit algılama algoritması kullanılabilir.

Segmentasyon modeline dayalı şerit algılama sistemleri, gerçek dünya senaryolarında umut verici sonuçlar göstermiştir, ancak aydınlatma koşulları, hava durumu ve yol yüzeyi kalitesindeki değişikliklere duyarlı olabilirler. Bu nedenle, bu sistemlerin sağlamlığını ve güvenilirliğini artırmak için devam eden araştırmalar yürütülür.

3.1.4.1. Segmentasyon Modelleri

Derin öğrenme mimarisi olan tamamen evrişimli ağ (FCN) (Long ve diğ., 2015) görüntüler üzerinde anlamsal bölümlere gerçekleştiren bir yöntemdir. Yöntem genellikle daha iyi segmentasyon doğruluğu veya verimliliği elde etmek için mevcut FCN modellerinde yeni bir yaklaşım veya iyileştirme sunulur. FCN'ler, her giriş görüntüsü için piksel bazında sınıflandırma yapmak üzere eğitilmiş sinir ağlarıdır. Görüntü segmentasyon görevleri için özel olarak tasarlanmışlardır ve bir görüntü içindeki nesneleri tanımlamak ve sınıflandırmak için kullanılabilirler. Anlamsal bölümlerde amaç, bir görüntüdeki her piksele belirli bir nesneye veya sınıfa karşılık gelen bir etiket atamaktır. Makale tipik olarak, katman sayısı, evrişimli katmanların türü ve hiper parametreler dahil olmak üzere FCN mimarisinin ayrıntılarını sunar. Ayrıca, büyük ölçekli veri kümelerinin kullanımı, veri artırma teknikleri ve kayıp fonksiyonları dahil olmak üzere eğitim sürecini de açıklar. Yöntem, tipik olarak FCN modelinin kıyaslama veri kümelerindeki performansını değerlendirir ve birleşim üzerinden ortalama kesişim (mIoU) ve piksel doğruluğu gibi metrikleri bildirir.

Önerilen yöntem (Zhou ve diğ., 2018), tıbbi görüntüleri bölümlere ayırmak amacıyla derin öğrenme tekniklerinin, özellikle evrişimli sinir ağlarının (CNN) uygulanmasını ifade eder. Tıbbi görüntü segmentasyonu, teşhise, tedavi planlamasına ve araştırmaya yardımcı olmak için tıbbi görüntülerde tümörler, kan damarları veya organlar gibi belirli ilgi alanlarını tanımlamayı ve ayırmayı içerir. U-Net, biyomedikal görüntü bölümlere görevleri için tasarlanmış özel bir CNN mimarisi türüdür. Segmentasyon gerçekleştirmek için bir daralma yolu ve bir genişleme yolu kullanan tamamen evrişimli bir ağıdır. Büzülme yolu, giriş görüntüsünün uzamsal çözünürlüğünü azaltan tekrarlanan evrişim uygulamalarından ve maksimum havuzlama işlemlerinden oluşur. Genişleyen

yol, orijinal görüntü çözünürlüğünü geri kazanmak için büzülen yoldan karşılık gelen yüksek çözünürlüklü özelliklerle yer değiştiren evrişimlerin ve birleştirmenin tekrarlanan uygulamalarından oluşur. U-Net, beyin MRG taramalarında tümörlerin tanımlanması, retinal görüntülerde kan damarlarının bölümlere ayrılması ve cilt kanseri görüntülerinde lezyonların saptanması gibi çeşitli tıbbi görüntü segmentasyon uygulamalarında kullanılmıştır. Biyomedikal görüntü bölümleme için U-Net ve diğer CNN mimarileri, geleneksel görüntü bölümleme yöntemlerine göre doğruluk ve verimlilikte önemli gelişmeler göstermiştir ve tıp alanındaki klinik uygulama ve araştırmalara büyük fayda sağlama potansiyeline sahiptir.

Önerilen yöntem (Ronneberger ve diğ., 2015), orijinal U-Net mimarisine dayalı olarak tıbbi görüntüleri bölümlere ayırmak için kullanılan bir derin öğrenme modelidir. Bu model, orijinal U-Net mimarisini ek iç içe U-Net modülleri ile genişlettiği için UNet++ olarak da anılır. Yuvalanmış U-Net mimarisi, girdi görüntüsünü alt örnekleme ve düşük seviyeli özellikleri yakalamak için bir dizi evrişimli ve havuzlama katmanı kullanır. Daha sonra, devrik konvolüsyonel katmanları kullanarak görüntüyü yukarı örnekler ve onu, iç içe geçmiş bir U-Net modülünün daralma yolundan karşılık gelen özellik haritasıyla birleştirir. Bu süreç, her biri farklı ölçeklerde özellikleri yakalayan iç içe geçmiş U-Net modüllerinin bir hiyerarşisini oluşturmak için yinelemeli olarak tekrarlanır. Nested U-Net mimarisi, akciğer nodülü tespiti, karaciğer segmentasyonu ve beyin tümörü segmentasyonu dahil olmak üzere çeşitli tıbbi görüntü segmentasyon görevlerinde kullanılır. Özellikleri birden çok ölçekte yakalama yeteneğinin, özellikle daha küçük nesneler ve ince ayrıntılar için segmentasyon doğruluğunu iyileştirdiği gösterilir.

Önerilen yöntem (Fan ve diğ., 2020), özellikle bilgisayarlı tomografi (CT) taramaları olmak üzere tıbbi görüntülerde karaciğer ve tümörlerin segmentasyonu için derin bir öğrenme mimarisidir. MA-Net, CT taramalarında düşük kontrast, şekil ve boyuttaki değişkenlik ve gürültü varlığı gibi karaciğer ve tümörleri segmentlere ayırmanın zorluklarını ele almak için tasarlanır. MA-Net mimarisi, birden çok ölçekte birleştirilmiş dikkat mekanizmaları ile çok ölçekli bir kodlayıcı-kod çözücü yapısından oluşur. Kodlayıcı, giriş CT görüntüsünden birden fazla ölçekte özellikler çıkaran ve daha sonra seçici olarak önemli özelliklere odaklanan dikkat modüllerinden geçirilen bir dizi

evrişimli katmandan oluşur. Kod çözücü daha sonra özellik haritalarını yukarı örnekler ve bunları, segmentasyonu iyileştirmek için dikkat mekanizmalarını kullanarak kodlayıcıdan gelen karşılık gelen özellik haritalarıyla birleştirilir.

Önerilen yöntemde (Chaurasia ve Culurciello, 2020), görüntülerin verimli semantik segmentasyonu için tasarlanmış bir derin öğrenme mimarisidir. U-Net'e benzer kodlayıcı-kod çözücü mimarisine dayanmaktadır, ancak bellek kullanımı ve eğitim süresi açısından daha verimli hale getirmek için bazı değişiklikler yapılır. LinkNet'teki temel yenilik, ağı kodlayıcı ve kod çözücü kısımlarını birbirine bağlayan ve kod çözücünün kodlayıcı tarafından oluşturulan özellik haritalarına erişmesine ve yeniden kullanmasına izin veren bir "bağlantı" modülünün kullanılmasıdır. Bu yaklaşım, ağı kod çözücü kısmında ihtiyaç duyulan parametre sayısını azaltır, bu da bellek kullanımını ve eğitim süresini azaltır. LinkNet'in bir diğer önemli yönü, ağı daha karmaşık özellikleri öğrenmesine olanak tanıyan ve segmentasyonun doğruluğunu artıran kodlayıcı ve kod çözücü modülleri arasındaki artık bağlantıların kullanılmasıdır. LinkNet, uydu görüntülerinde binaların bölümlenmesi ve mikroskobik görüntülerde hücrelerin bölümlenmesi dahil olmak üzere çeşitli semantik bölümlendirme görevlerinde uygulanır. Genel olarak LinkNet, bellek kullanımını ve eğitim süresini azaltmak için kodlayıcı temsillerinden yararlanmaya odaklanan anlamsal bölümlendirme için verimli ve etkili bir derin öğrenme mimarisidir.

Önerilen yöntem (Zhao ve diğ., 2017), Piramit Sahne Ayrıştırma Ağı (PSPNet) doğru ve verimli semantik görüntü segmentasyonu için tasarlanmış bir derin öğrenme mimarisidir. Nesne ölçeklerinde büyük varyasyonlara sahip görüntülerin işlenmesinde mevcut semantik bölümlendirme ağlarının sınırlamalarının ele alınması önerildi. PSPNet mimarisi, standart FCN mimarisi üzerine kuruludur, ancak girdi görüntüsünün farklı bölgelerinden özellikleri birden çok ölçekte çıkaran bir piramit havuzlama modülü ekler. Modül, giriş görüntüsünü örtüşmeyen bölgelere bölerek ve her bölgeye maksimum havuzlama uygulayarak çalışır ve bu da farklı ölçeklerde bir dizi özellik sağlar. Bu özellikler daha sonra birleştirilir ve nihai segmentasyon çıktısını oluşturmak için bir dizi evrişimli katmandan geçirilir. PSPNet mimarisi ayrıca, özellik haritalarını kodlayıcıdan karşılık gelen kod çözücü katmanlarına bağlayan ve ağı görüntüdeki hem düşük seviyeli hem de yüksek seviyeli özellikleri yakalamasına izin veren bir atlama

mimarisi içerir. Otonom sürüş, robotik ve tıbbi görüntü analizi gibi bir dizi uygulamada kullanılır. Genel olarak, PSPNet, piramit havuzundan yararlanan ve birden çok ölçekteki özellikleri yakalamak için bağlantıları atlayarak, nesne ölçeklerinde büyük varyasyonlarla görüntülerin doğru ve verimli bir şekilde bölümlenmesini sağlayan, semantik bölümlendirme için güçlü bir derin öğrenme mimarisidir.

Önerilen yöntem Semantik Bölümlendirme için Piramit Dikkat Ağı (PANet) (Li ve diğ., 2018) giriş görüntüsündeki daha ayrıntılı özellikleri yakalayarak anlamsal bölümlendirmenin doğruluğunu artırmak için tasarlanmış bir derin öğrenme mimarisidir. PANet, standart kodlayıcı-kod çözücü mimarisini temel alır, ancak özellikleri birden çok ölçekte yakalayan ve bunları bölümlendirme görevindeki önemlerine göre seçici olarak toplayan bir piramit dikkat modülü (Pyramid Attention Module) ekler. Piramit dikkat modülü, her biri bir dizi evrişimli katman kullanarak giriş özelliği haritalarını farklı bir ölçekte işleyen birden çok paralel daldan oluşur. Daha sonra küresel bir dikkat toplama katmanı, her bir özellik haritasının önemini ağırlıklandırır ve girdi görüntüsünün birleşik bir temsili oluşturmak için bunları bir araya getirir. Bu piramit dikkat mekanizması, ağırlık özellikleri birden çok ölçekte yakalamasına ve bölümlendirme görevi için en ilgili özelliklere odaklanmasına olanak tanır. PANet, piramit dikkat modülüne ek olarak, bir dizi yanal bağlantı aracılığıyla kodlayıcıyı (encoder) kod çözücüye (decoder) bağlayan aşağıdan yukarıya bir yol da içerir. Bu, ağırlık kodlayıcıdan yüksek seviyeli özellikler yakalamasına ve bunları kod çözücüden düşük seviyeli özelliklerle birleştirerek segmentasyon doğruluğunu iyileştirmesine olanak tanır. Genel olarak PANet, bir piramit dikkat modülü ve birden fazla ölçekten ayrıntılı özellikleri yakalayan aşağıdan yukarıya bir yol kullanarak semantik segmentasyonun doğruluğunu artıran güçlü bir derin öğrenme mimarisidir.

Önerilen modelde (Chen ve diğ., 2017) genişletilmiş konvolüsyon olarak da bilinen atrous konvolüsyonu kullanarak semantik segmentasyon için yeni bir yöntem öneren bir araştırma makalesidir. Atrous konvolüsyon, herhangi bir yeni parametre eklemeyen CNN alıcı alanının genişlemesine izin veren konvolüsyonel bir işlemdir ve bu da onu semantik bölümlendirme için verimli bir yöntem haline getirir. Atrous Spatial Pyramid Pooling (ASPP) adlı yeni bir rahatsız edici evrişim yöntemi önerir. ASPP, giriş görüntüsünden çok ölçekli bağlamsal bilgileri yakalayarak anlamsal bölümlendirme

performansını geliřtirmek için tasarlanır. ASPP'nin farklı ölçeklerdeki özellikleri yakalayabildiğini ve çeřitli boyutlardaki nesnelerin doğru segmentasyonuna izin verdiğini göstermektedir. Ayrıca, ASPP'nin semantik bölümlleme performansını daha da geliřtirmek için atlama bağlantıları gibi diğler yöntemlerle birleřtirilebileceğini de gösterir. Genel olarak makale, mevcut yöntemlerin sınırlamalarını ele alan atrous evriřim kullanarak anlamsal bölümllemeye yeni bir yaklařım sunar.

Önerilen yöntemde (Chen ve diğ., 2018), birbirinden ayrılabilir evriřimli bir kodlayıcı-kod çözücü mimarisine dayalı yeni bir semantik görüntü bölümlleme yaklařımı önerir. Önerilen mimari bir kodlayıcı ağı ve bir kod çözücü ağından oluřmaktadır. Kodlayıcı ağı, bir girdi görüntüsü alır ve ayrılabilir filtrelerle sahip yoğun konvolüsyonlar kullanarak birden çok ölçekteki özellikleri çıkarır. Atrous konvolüsyonlar, parametre sayısını arttırmadan daha geniř bir alıcı alana izin veren bir konvolüsyon türüdür. Ayrılabilir filtreler ise, evriřimi iki ayrı iřleme ayırır ve bu da hesaplama maliyetini azaltır. Kod çözücü ağı, kodlayıcı ağı tarafından çıkarılan özellikleri alır ve giriř görüntüsüyle aynı uzamsal çözünürlüğe sahip bir bölümlleme haritası üretir. Kod çözücü ağı, özelliklerin çözünürlüğünü artırmak için çift doğrusal üst örnekleme kullanır ve ardından bunları, atlamalı bağlantıları kullanarak kodlayıcı ağındaki karřılık gelen özelliklerle birleřtirir. Atlanan bağlantılar, ince taneli ayrıntıların korunmasına ve segmentasyonun doğruluğunun iyileřtirilmesine yardımcı olur.

Önerilen yöntemde (Neven ve diğ., 2018) iki ana bileřenden oluřur: bir omurga ağı ve bir řerit örnek bölümlleme ağıdır. Omurga ağı, giriř görüntüsünden özellikleri çıkaran tamamen evriřimli bir ağıdır. řerit örneđi bölümlleme ağı, özellikleri omurga ağından alır ve her řerit için örnek bölümlendirme maskelerini tahmin eder. Yazarlar, řerit örneđi bölümlleme ağı için Mask R-CNN mimarisinin deđiřtirilmiř bir sürümünü kullanıyor. Mask R-CNN mimarisi, Faster R-CNN nesne algılama çerçevesini bir segmentasyon dalı ile geniřleten popöler bir örnek segmentasyon çerçevesidir. Önerilen yaklařım, geniř ölçekli bir řerit iřaretleme veri seti üzerinde uçtan uca eđitilmiř ve birkaç kıyaslama veri seti üzerinde deđerlendirilir. Deneysel sonuçlar, önerilen yaklařımın hem düz hem de virajlı řerit algılama görevlerinde en geliřmiř performansı elde ettiğini göstermektedir.

Önerilen yöntem (Hur ve diğ., 2013) Koşullu Rastgele Alanlar (CRF) kullanarak şehir içi yollarda çok şeritli algılama yöntemi öneren bir yöntemdir. Önerilen yöntem, önce derin CNN kullanarak bir girdi görüntüsünden özellikleri çıkarır. Özellikler daha sonra şeritler arasındaki uzamsal ve bağlamsal ilişkileri hesaba katan bir CRF modeli oluşturmak için kullanılır. CRF modeli, bir giriş görüntüsü verildiğinde bir şerit konfigürasyonunun olasılığını öğrenmek için etiketli veri kümeleri üzerinde eğitilir. Çıkarım sırasında, şerit konfigürasyonlarının bir olasılık haritasını oluşturmak için CRF modeli kullanılır ve bu daha sonra nihai şerit işaretlerini elde etmek için sonradan işlenir. Son işleme, şerit işaretlerinin ikili bir haritasını elde etmek için olasılık haritasının eşiklenmesini ve ardından nihai şerit sınırlarını elde etmek için şerit işaretlerinin kümelenmesini ve uydurulmasını içerir.

Önerilen yöntemde (Chao ve diğ., 2019) özellik çıkarıcı olarak VGG-16 CNN mimarisinin değiştirilmiş bir sürümünü kullanıyor ve onu, açıklamalı görüntülerden oluşan büyük ölçekli bir veri kümesi üzerinde eğitiyor. Bu çalışmada, çıkarılan özelliklerden şerit konumlarını tahmin etmek için bir dizi evrişimli katman, havuzlama katmanı ve tamamen bağlı katmanlardan oluşan bir şerit algılama ağı kullanır. Ayrıca, geometrik kısıtlamalara ve şerit işaretlerinin sürekliliğine dayalı olarak tahmin edilen şerit konumlarını iyileştiren bir son işleme adımı önerir. Son işleme adımı, kesikli çizgiler, düz çizgiler ve oklar dahil olmak üzere çeşitli şerit işaretlerini işliyor ve değişen genişlik ve boşluklara sahip şerit işaretlerini elde ediyor.

Önerilen yöntemde (Abualsaud ve diğ., 2021), şehir içi sürüş ortamlarında çok şeritli algılama için bir yöntem önerilir. Önerilen yöntem, bir görüntüdeki pikseller arasındaki ikili ilişkilerin öğrenilmiş temsilleri olan yakınlık alanları kavramına dayanmaktadır. LaneAF yöntemi, önce bir CNN kullanarak şerit işaretleri için bir olasılık haritası oluşturur. Olasılık haritası daha sonra görüntünün bir grafik temsilini oluşturmak için kullanılır; burada düğümler piksellere karşılık gelir ve kenarlar, pikseller arasındaki ikili ilişkilere karşılık gelir. Benzeşim alanları daha sonra bir grafik evrişimli ağ (GCN) kullanılarak bu grafik temsilde öğrenilir. Öğrenilen yakınlık alanları, aynı şeride ait iki pikselin olasılığını temsil eder. Son olarak, şerit sınırları, piksellerin yakınlık değerlerine göre kümelenmesiyle elde edilir.

Önerilen yöntem (Sun ve diğ., 2022), örnek bölümlendirmeyi kullanarak yollarda birden çok şeridi algılamak için önerilen bir yöntemdir. Önerilen yöntem, farklı aydınlatma koşullarına, yol yüzeylerine ve şerit işaretlerine dayanıklı olacak şekilde tasarlanır. Yöntem, önce bir CNN kullanarak bir giriş görüntüsünden özellikleri çıkarır.

Özellikler daha sonra şerit işaretlerinin bir olasılık haritasını oluşturmak için kullanılır. Ayrıca, şerit algılamının sağlamlığını artırmak için yerel görüntü özelliklerine dayanan yeni bir uyarlamalı eşikleme tekniği önerdiler. Daha sonra, şerit işaretleri için örnek bölümleme maskeleri oluşturmak için olasılık haritası kullanılır. Örnek segmentasyon maskeleri, daha sonra nihai şerit sınırlarını elde etmek için birleştirilen bireysel şerit sınırlarını çıkarmak için kullanılır.

3.2. Şerit Tipi Sınıflandırmak İçin Derin Öğrenme Temelli Çalışmalar

Şerit türlerini sınıflandırmak için yapılan derin öğrenme tabanlı çalışmalar, yoldaki farklı şerit türlerini kesikli, kesintisiz ve çift çizgi gibi otomatik olarak sınıflandırılan modeller geliştirmeyi amaçlar. Bu modeller, yoldaki farklı şerit türlerini doğru bir şekilde algılamak ve sınıflandırmak için şerit tutma ve şeritten ayrılma uyarı sistemleri gibi çeşitli otonom sürüş uygulamalarında kullanılır.

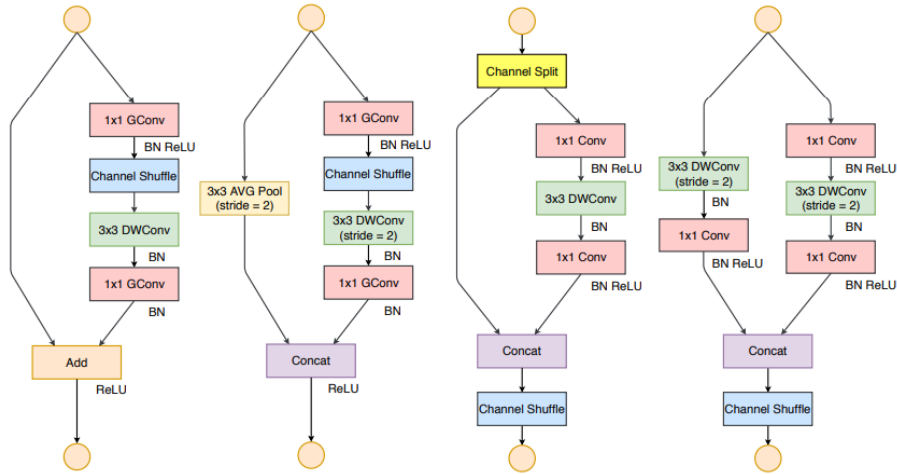
Şerit türü sınıflandırması için derin öğrenmeye dayalı yaklaşımlardan bazıları, giriş görüntülerinden özellikleri çıkarmak için CNN kullanır ve ardından şerit türünü tahmin etmek için bir sınıflandırıcı kullanır. Diğerleri, şerit işaretleri arasındaki zamansal ve mekânsal bağımlılıkları yakalamak ve daha doğru tahminler yapmak için tekrarlayan sinir ağları (RNN) veya dikkat tabanlı modeller (attention-based models) gibi daha karmaşık modeller kullanır. Bu modellerin performansı, doğruluk, kesinlik, hatırlama ve F1 puanı gibi metrikler kullanılarak çeşitli veri kümelerinde değerlendirilir. Deneysel sonuçlar, derin öğrenmeye dayalı yaklaşımların şerit türlerini sınıflandırmada yüksek doğruluk elde edebildiğini ve geleneksel bilgisayarla görme yöntemlerinden daha iyi performans gösterir.

3.2.1. Evrişimli Sinir Ağları Tabanlı Sınıflandırma Modelleri

ShuffleNet V2 (Ma ve diğ., 2018) mimarisi, grup konvolüsyonu ve kanal karıştırma fikrine dayanmaktadır. Grup evrişiminde, giriş ve çıkış kanalları birkaç gruba ayrılır ve

her grup ayrı ayrı evrilir. Bu, ağı zengin özellik temsillerini öğrenmesine izin verirken evrimlerin hesaplama karmaşıklığını azaltır. Kanal karıştırma (channel shuffling), ağı doğruluğunu artırmaya yardımcı olan, farklı gruplardan özellik haritalarını karıştıran bir tekniktir. Ayrıca bu model, verimli CNN mimarileri tasarlamak için parametre sayısını azaltmak, darboğaz yapıları kullanmak ve atlamalı bağlantıları dahil etmek gibi çeşitli pratik yönergeler sağlar. Bu yönergeler, farklı mimariler ve hiper parametrelerle yapılan kapsamlı deneylere dayanmaktadır.

Şekil 3.3'te ShuffleNet V1'in (Zhang ve diğ., 2018) mimarisi gösterilmiştir. Şekil 3.3'te soldan birinci ve ikinci bölümlerde temel birim ve uzamsal alt örnekleme birimi gösterilmiştir. Şekil 3.3'te birinci ve ikinci bölümlerde ShuffleNet V2'deki temel birim ve uzamsal alt örnekleme birimi gösterilmiştir. Şekil 3.3'te sağdan ikinci bölümde kanal bölme adı verilen basit bir operatör gösterir.



Şekil 3.3. ShuffleNetV1 mimarisi

Önerilen çalışmada görüntü tanıma görevleri için ResNet (He ve diğ., 2016) adlı derin bir öğrenme mimarisini tanıttı. ResNet, istenen eşlemeyi doğrudan öğrenmek yerine ağı girdiye eklenen artık eşlemeleri öğrenmesine izin veren artık blokları (residual blocks) kullanır. Bu yaklaşım, ağı çok daha derin temsilleri öğrenmesini sağladı ve çok derin sinir ağlarında ortaya çıkan yok olan gradyan sorununu çözmeye yardımcı olur.

Şekil 3.4'te VGG-19, 34 katmanlı düz sinir ağı ve 34 katmanlı derin artık sinir ağının karşılaştırılması gösterilir. Noktalı çizgiler, görüntünün boyutunda bir artık bloktan (residual block) diğerine değişiklik olduğunu gösterir ve 1x1 çekirdekler kullanılarak

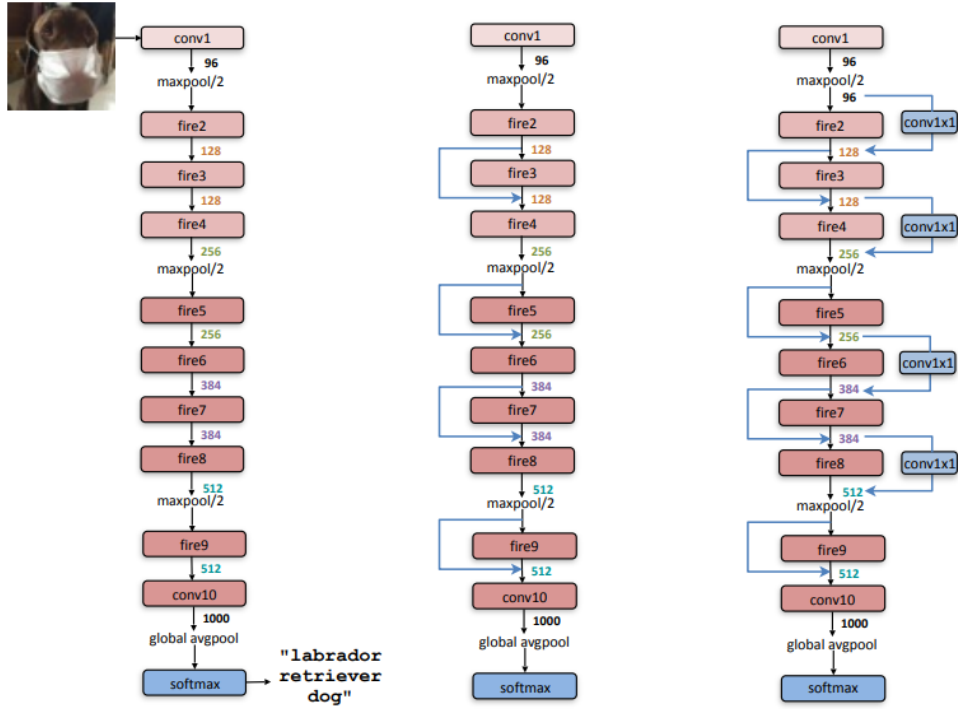
gerçekleştirilebilen doğrusal projeksiyondur. Artık blokların birden fazla avantajı vardır. Optimize edilmesi kolaydırlar. Katmanları istiflenmeye devam edildikçe düz sinir ağlarında olduğu gibi sinir ağının derinliği arttıkça eğitim hatası artmaz. Kimlik eşlemenin (identity mapping) eklenmesi herhangi bir ekstra parametre getirmez bu sebeple hesaplama karmaşıklığı (computational complexity) artmaz. Şekil 3.4'te soldan birinci model referans olarak VGG-19 (Simonyan ve Zisserman., 2014) modelidir. Ortadaki model 34 parametre katmanına sahip sade bir ağ. Sağdaki model 34 parametre katmanına sahip bir modeldir.



Şekil 3.4. ImageNet için örnek ağ mimarileri

SqueezeNet (Iandola ve diğ., 2016) modelinde ilk olarak, hesaplama açısından verimli olan ve modeldeki parametre sayısını azaltmaya yardımcı olan 1x1 evrişimli filtreler kullanılır. İkinci olarak, çoklu özellik haritalarını daha az sayıda kanalda birleştiren

"sıkıştır" (squeeze) katmanları kullanılır, bu da sonraki evrişimli katmanların hesaplama maliyetini azaltır. Son olarak, 1x1 ve 3x3 evrişimli filtrelerin birleşiminden oluşan "yangın" (fire) modüllerini hem alt düzey hem de üst düzey özellikleri verimli bir şekilde öğrenmek için kullanılır. Şekilde 3.5'te SqueezeNet'in mimarisi gösterilir. Şekilde 3.5'teki soldaki model SqueezeNet, ortadaki basit baypaslı SqueezeNet, sağdaki karmaşık baypaslı SqueezeNet'tir.



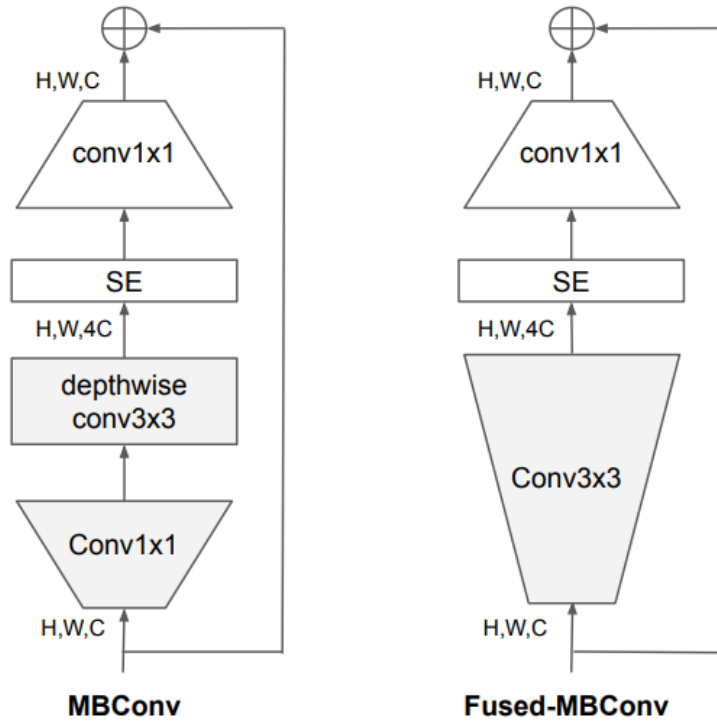
Şekil 3.5. SqueezeNet mimarsinin makro mimari görünümü

Bu mimarideki sıkma modülü (squeeze module) yalnızca 1x1 filtre içerir, yani aynı konumdaki özellik noktalarında çalışan tamamen bağlı bir katman gibi çalışır. Yani mekânsal soyutlama yeteneğine sahip değildir. Adından da anlaşılacağı gibi, avantajlarından biri özellik haritasının derinliğini azaltmaktır. Derinliği azaltmak, genişletme katmanındaki 3x3 filtrelerin daha az hesaplama yapması anlamına gelir. 3x3'lük bir filtrenin ihtiyaç duyduğu hızı, 1x1'lik bir filtrenin hesaplamasının 9 katı kadar artırır. Sezgisel olarak, çok fazla sıkıştırma bilgi akışını sınırlar; çok az sayıda 3x3 filtre, alan çözünürlüğünü sınırlar.

EfficientNetV2 (Tan ve Le, 2021) modelinde, EfficientNetV1 (Tan ve Le, 2019) mimarisinin verimliliğini daha da artırmak için birkaç yeni teknik sunulur. İlk olarak,

modelin derinliğini, genişliğini ve çözünürlüğünü dengeli bir şekilde ayarlayarak kaynakların daha verimli kullanılmasını sağlayan yeni bir bileşik ölçeklendirme yöntemi sunulur. İkincisi, eğitim sırasında bilgiyi daha büyük, daha doğru bir modelden daha küçük, daha verimli bir modele aktarmak için "bilgi damıtma ile eğitim" (training with knowledge distillation) adı verilen yeni bir eğitim şeması kullanır. Bu, daha küçük modelin daha hızlı yakınsaması ve geliştirilmiş doğruluğu olarak adlandırılır.

Şekil 3.6'de MBConv (Gupta ve Tan, 2019) ve Fused-MBConv'un yapısı gösterilmiştir. Fused-MBConv, MBConv'daki derinlemesine 3×3 konvolüsyon ve genişletme 1×1 konvolüsyon tek normal 3×3 konvolüsyon ile değiştirir.



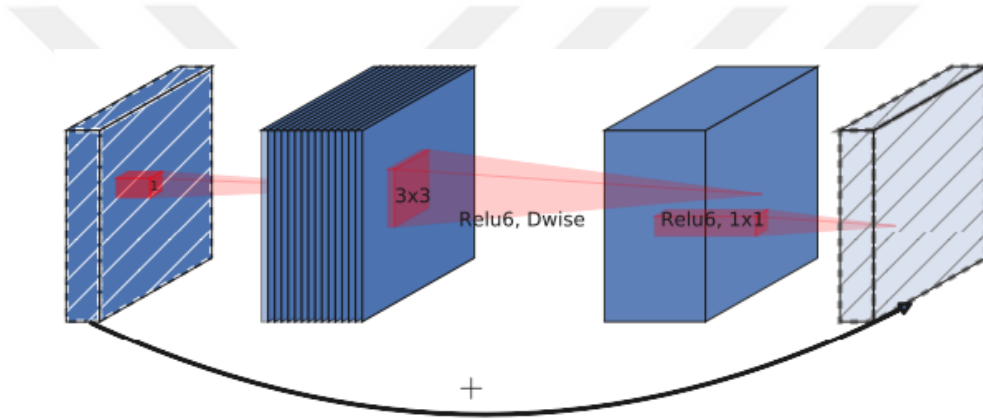
Şekil 3.6. MBConv ve Fused-MBConv'un Yapısı

Erken aşama (stage) 1-3'te uygulandığında, Fused-MBConv, parametreler ve FLOP'lar üzerinde küçük bir ek yük ile eğitim hızını artırabilir ancak tüm bloklar Fused-MBConv aşama 1-7 kullanıyorsa, eğitimi yavaşlatırken aynı zamanda parametreleri ve FLOP'ları önemli ölçüde artırır.

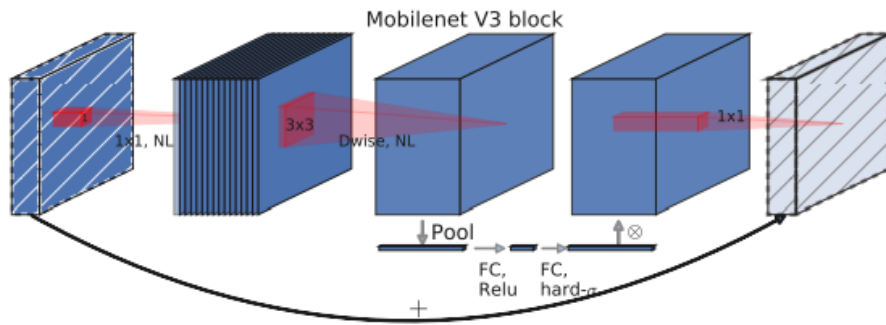
MobileNetV3 (Howard ve diğ., 2019) modelinde farklı konfigürasyonlara ve mimarilere sahip geniş bir sinir ağıları tasarım alanını keşfetmek için bir sinir mimarisi arama

(NAS) yaklaşımı kullanılır. Sınırlı bilgi işlem kaynaklarına sahip mobil cihazlar için kritik olan doğruluk ve çıkarım hızı arasındaki dengeyi optimize etmeye özellikle odaklanılır. NAS süreci, çok sayıda aday modeli eğitmeyi ve değerlendirmeyi ve en iyi performans gösteren mimariyi seçmeyi içerir. Arama sürecini hızlandırmak için yazarlar, küçük bir ağla başlayan ve yeni bloklar ve katmanlar ekleyerek karmaşıklığı kademeli olarak artıran, aşamalı arama adı verilen bir teknik kullanılır.

Şekil 3.7’de MobileNetV2 (Sandler ve diğ., 2018) modülü Şekil 3.8’de MobileNetV3 modülü gösterilmiştir. MobileNetV2 ile karşılaştırıldığında MobileNetV3, SENet’ten (Hu ve diğ., 2018) kaynaklanan Squeeze and Excitation (SE) modülü eklenir.

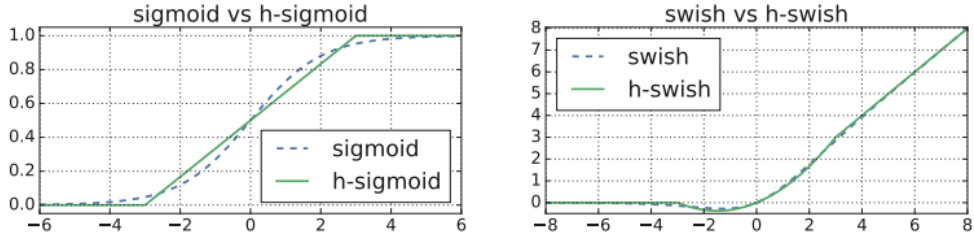


Şekil 3.7. MobileNetV2 Modülü



Şekil 3.8. MobileNetV3 Modülü

Verimli hesaplama için SE modülünde sigmoid yerine sert-sigmoid (hard-sigmoid) kullanılır. Şekil 3.9’de sigmoid ve sert-sigmoid gösterilir.



Şekil 3.9. Sigmoid ve swish doğrusalsızlıkları ve "sert" karşılıkları

Swish, doğrusal olmamayı (nonlinearity) iyileştirmek için literatürde ReLU'nun yerini almak için kullanılır. Denklem (3.1)'de swish fonksiyonu gösterilmiştir. Burada σ işareti sigmoid fonksiyonunu ifade etmektedir.

$$swish = x\sigma(x) \quad (3.1)$$

Denklem (3.2)'de h-swish fonksiyonu gösterilmiştir. H-Swish fonksiyonu, monoton olmayan bir şekle (non-monotonic shape) sahip olması bakımından Swish işlevine benzer, yani bir sinir ağının girdisi ve çıktısı arasındaki karmaşık ilişkileri yakalayabilir. Bununla birlikte, H-Swish'in eğrisinde daha belirgin bir doğrusal bölge vardır, bu da hesaplamayı kolaylaştırır ve aktivasyon işlevinin düzleştiği ve eğitim için yararlı gradyanlar sağlayamadığı derin öğrenme ağlarında yaygın bir sorun olan doygunluğa daha az eğilimlidir.

$$h-swish[x] = x \left(\frac{ReLU_6(x+3)}{6} \right) \quad (3.2)$$

MobileNetV3 donanımına uyumlu olabilmesi için MnasNet (Tan ve diğ., 2019) kullanılır. MnasNet kullanarak ağı aradıktan sonra, basitçe model oluşturmak için NetAdapt (Yang ve diğ., 2018) kullanılır.

4. ÖNERİLEN ÇOKLU ŞERİT ALGILAMA SİSTEMİ

Bu bölümde çoklu şerit algılama için kullanılan veri seti ve önerilen yöntem detaylı bir şekilde ele alınacaktır. Bu tez kapsamında önerilen sistemde yüksek hızda koşan çoklu şerit algılama sistemi gerçekleştirmek hedeflenir.

4.1. Veri Seti

Gerçekleştirilen çoklu şerit algılama sisteminde yapay sinir ağlarının eğitilmesi gerekmektedir. Bu eğitim işlemi için TuSimple firmasının şerit algılama çalışmaları için hazırlamış olduğu veri kümesi kullanılır. Şekil 4.1’de gösterilen görüntüler TuSimple veri kümesine aittir (URL-1, 02.02.2023).



Şekil 4.1. TuSimple veri kümesi

TuSimple veri kümesindeki görüntüler farklı günlerde toplanmış ve farklı hava koşullarını içerir. Veri kümesindeki görüntüler güneşli, bulutlu ve parçalı bulutlu şeklindedir. Bu veri kümesi özellikle 2, 3, ve 4 şeride sahip otoyollarda görüntülerden oluşur. Veri kümesi oluşturulurken saniyede 20 görüntü alınır. Toplamda 3626 adet eğitim, 2782 adet test görüntüsü bulunur. TuSimple firması bu veri kümesini paylaşıırken ham görüntüleri ve bu ham görüntüler üzerindeki şeritlerin bir çizgi halinde olduğunu ifade eden bir JSON formatında dosya ile paylaşılır. Bu JSON dosyası içindeki koordinat bilgilerini kullanarak ikili görüntü halini kullanıcılar kendileri

oluşturur. Şekil 4.2’de JSON formatı gösterilir. Bu JSON formatı görüntü yüksekliği boyunca 10 piksel adımda bir beyaz şerit pikseli var veya yok bilgisini içerir.

```
{
  "lanes": [
    [-2, -2, -2, -2, 632, 625, 617, 609, 601, 594, 586, 578, 570, 563, 555, 547, 539, 532, 524, 516, 508, 501, 493,
    [-2, -2, -2, -2, 719, 734, 748, 762, 777, 791, 805, 820, 834, 848, 863, 877, 891, 906, 920, 934, 949, 963, 978,
    [-2, -2, -2, -2, -2, 532, 503, 474, 445, 416, 387, 358, 329, 300, 271, 241, 212, 183, 154, 125, 96, 67, 38, 9,
    [-2, -2, -2, 781, 822, 862, 903, 944, 984, 1025, 1066, 1107, 1147, 1188, 1229, 1269, -2, -2, -2, -2, -2, -2, -2
  ],
  "h_samples": [240, 250, 260, 270, 280, 290, 300, 310, 320, 330, 340, 350, 360, 370, 380, 390, 400, 410, 420, 430, 440
  "raw_file": "path_to_clip"
}
```

Şekil 4.2. TuSimple JSON formatı

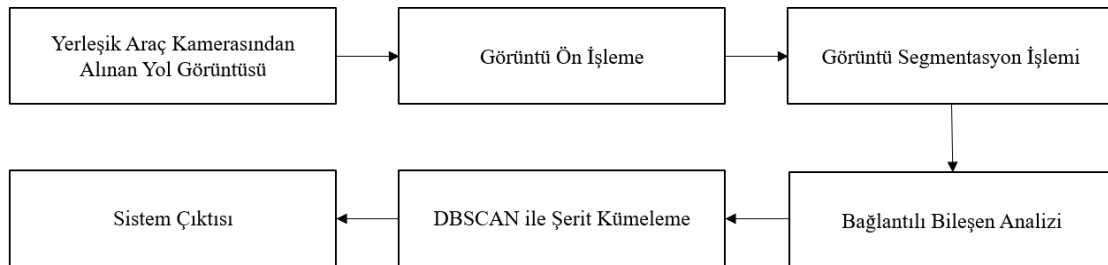
Burada “lanes” kısmındaki “-2” ifadesi belirtilen koordinatta beyaz pikselin olmadığına karşılık gelmektedir. Şekil 4.2’deki JSON formatında bir görüntü üzerinde 4 adet şeritin olduğu görülür.

4.2. Çoklu Şerit Algılama Sistemi

Bu bölümde tez kapsamında gerçekleştirilen çoklu şerit algılama sistemleri detaylı bir şekilde açıklanır. Bu tezde iki farklı sistem gerçekleştirilir. Bu sistemleri önerilen sistem bir ve önerilen sistem iki başlıkları altında detaylı bir şekilde açıklanacaktır.

4.2.1. Önerilen Birinci Sistem

Bu bölümde tez kapsamında önerilen birinci çoklu şerit algılama sistemi detaylı bir şekilde açıklanır. Şekil 4.3’te önerilen birinci çoklu şerit algılama sisteminin hangi işlemlere sahip olduğu gösterilir.



Şekil 4.3. Önerilen birinci çoklu şerit algılama sistemi

Şekil 4.3.'de gösterildiği gibi ilk olarak yerleşik araç kamerasından yol görüntüsü alınır. Alınan bu ham yol görüntüsü 1280x720 çözünürlüğe sahiptir. Şekil 4.4'te alınan bu ham görüntü gösterilir.

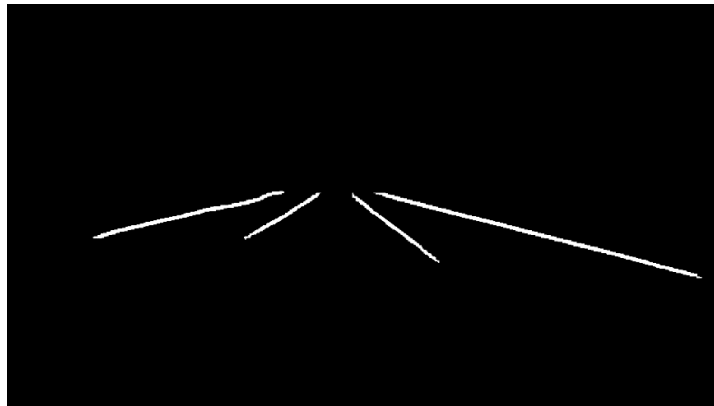


Şekil 4.4. Yerleşik araç kamerasından alınan ham görüntü

Bu görüntü üzerinde görüntü ön işleme işlemleri gerçekleştirilir ve görüntünün yeni çözünürlüğü 640x480 olarak düzenlenir.

4.2.1.1. Segmentasyon Modelinin Uygulanması

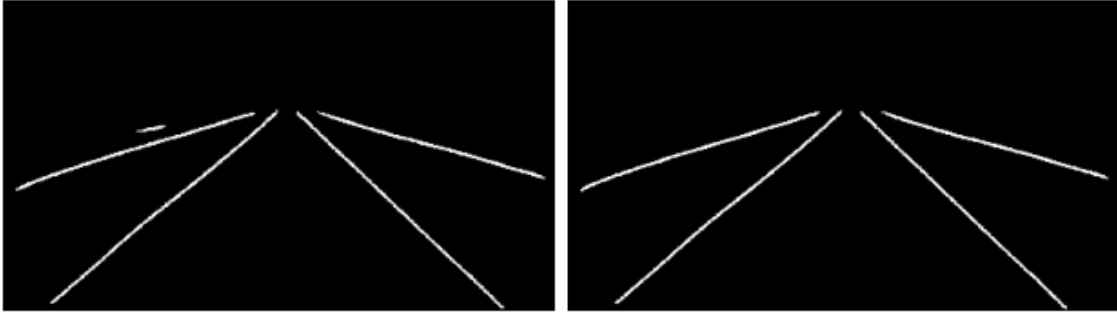
Bu tez kapsamında segmentasyon modellerinden UNet, UNet++, DeepLabv3, DeepLabv3+ modelleri eğitilmiştir ve performansları karşılaştırılır. Elde edilen deneysel sonuçlar doğrultusunda DeepLabv3+ modeli ile çalışmaya karar verilir. DeepLabv3+ modelinin neden seçildiği modellerin karşılaştırılması bölümünde detaylı bir şekilde ele alınır. Kameradan alınan ham görüntüye segmentasyon işlemi uygulanır ve ikili görüntü elde edilir. Şekil 4.5'te segmentasyon işleminden sonra elde edilen ikili görüntü gösterilir. Şekil 4.5'teki görüntü boyutu giriş görüntüsü ile aynı boyutlardadır.



Şekil 4.5. Segmentasyon işlemi sonrası oluşan ikili görüntü

4.2.1.2. Baęlantılı Bileşen Analizi

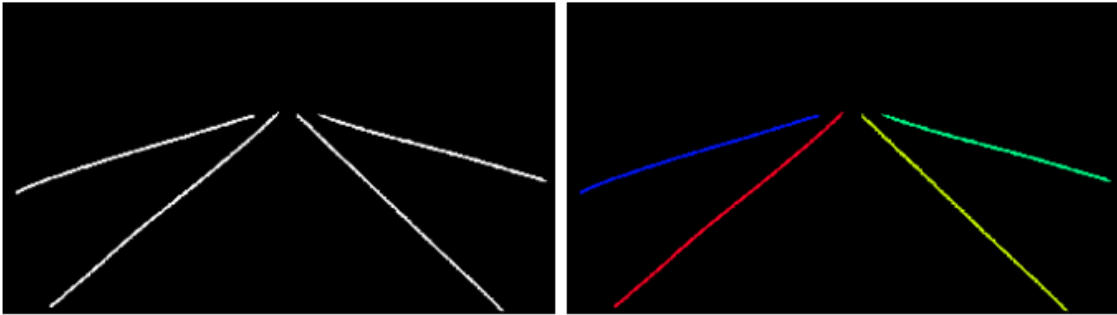
Bu bölümde Baęlantılı Bileşen Analizi (CCA) (Samet ve Tamminen, 1986) yöntemi uygulanır. CCA, bir son işleme algoritmasıdır. İkili (yani eşiklenmiş) bir görüntünün baęlı bileşenlerini analiz etmek ve özelliklerine göre filtreleme gibi işlemleri gerçekleştirir. Bu özellikler genişlik yükseklik ve alan gibi bilgilerdir. Bu yöntem ile istenmeyen beyaz piksel kümeleri görüntü üzerinden kaldırılır. Şekil 4.6’da CCA işlemi ile istenmeyen beyaz piksel kümesinin kaldırıldığı görüntü işlemi gösterilmiştir.



Şekil 4.6. CCA işlemi öncesi ve sonrası oluşan ikili görüntü

4.2.1.3. DBSCAN İşleminin Uygulanması

Bu bölümde gürültülü uygulamaların yoğunluk tabanlı uzamsal kümelenmesi (DBSCAN) (Hinneburg, 1996) işlemi gerçekleştirilir. Bu işlem ile çoklu şeritler kümelenmiş hale getirilir. Şekil 4.7’de DBSCAN işlemi uygulandıktan sonra görüntü üzerindeki tüm şeritlerin kümелendięi gösterilir.



Şekil 4.7. DBSCAN işlemi öncesi ve sonrası oluşan kümelenmiş şeritler

4.2.1.4. Sistemin Performansı

Bu bölümde önerilen birinci sistemin performansı detaylı bir şekilde açıklanır. Sistem performansı değerlendirilmesi için saniyedeki çerçeve sayısı (FPS) kullanılır. Sistem performans testlerini NVIDIA GeForce RTX 3090 ekran kartı ve 10 merkezi işlem birimine sahip olan Linux işletim sistemi üzerinde gerçekleştirilir. Tablo 4.1’de önerilen birinci sistemde uygulanan her işlem sonrasında elde edilen FPS değerleri gösterilir. FPS değerleri ekran kartı üzerinde elde edilir.

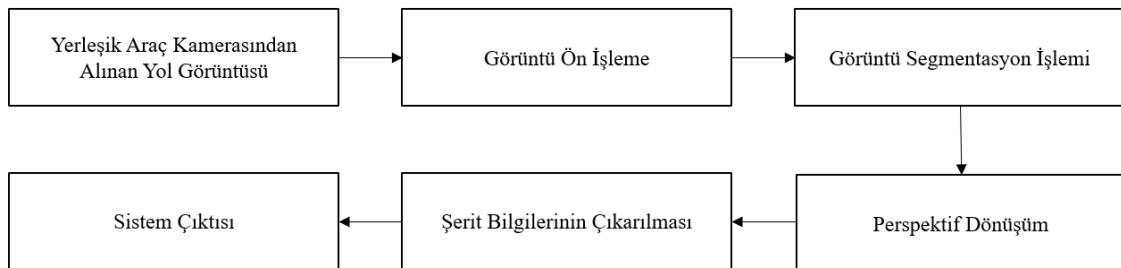
Tablo 4.1. Önerilen birinci sistemin performans sonuçları

Uygulanan İşlemlerin Adları	FPS Sonuçları
DeepLabv3+	136,7
DeepLabv3+ + CCA	92,4
DeepLabv3+ + CCA + DBSCAN	16,8

Tablo 4.1’de gösterildiği gibi segmentasyon modeli 136,7 FPS değerine sahiptir. Segmentasyon işlemi sonrasında CCA işlemi uygulandıktan sonra FPS değeri 92,4 olur. CCA işleminden sonra DBSCAN işlemi uygulanır ve bu işlem sonrasında sistemin FPS değeri 16,8 olur. DBSCAN işlemi sistemin performansını aşırı derecede yavaşlatmış olduğu tespit edilir. Bu sebeple ikinci bir sistem tasarımı gerçekleştirilir.

4.2.2. Önerilen İkinci Sistem

Bu bölümde tez kapsamında önerilen ikinci çoklu şerit algılama sistemi detaylı bir şekilde açıklanır. Şekil 4.8’de önerilen ikinci çoklu şerit algılama sisteminin hangi işlemlere sahip olduğu gösterilir. Bu sistemde görüntü segmentasyonu işlemine kadar olan tüm bölümler birinci önerilen sistemdeki ile aynıdır. Bu kısımda aynı olan bölümler tekrar anlatılmaz, sadece aynı olmayanlar kısımlar detaylı bir şekilde açıklanır.



Şekil 4.8. Çoklu şerit algılama sistemi

4.2.2.1. Perspektif Dönüşümün Uygulanması

Bu bölümde elde edilen ikili görüntü üzerinde perspektif dönüşüm yöntemi uygulanır. Perspektif dönüşüm, görüntü işleme ve bilgisayarla görmede yaygın olarak kullanılan bir tekniktir. Görüntüdeki bir nesnenin bakış açısını veya yönünü değiştirmek gibi bir görüntünün bir perspektiften diğerine dönüştürülmesine olanak tanır.

Perspektif dönüşümünün teknik detayları, üç boyutlu uzayda iki düzlem arasındaki yansıtılmalı bir dönüşümü temsil eden homografi matrislerinin kullanımını içerir. Görüntü işleme bağlamında, bu düzlemler tipik olarak görüntü düzlemini ve bir referans düzlemi (gerçek dünyadaki düz bir yüzey gibi) temsil eder.

Perspektif dönüşüm işlemini uygulamak için giriş görüntüsü üzerinde, gerçek dünyada bir dikdörtgenin veya dörtgenin dört köşesine karşılık gelen dört nokta seçilir. Giriş ve çıkış görüntüleri için bu noktalar tanımlanır. Giriş görüntüsünü çıkış görüntüsüne eşlemek için homografi matrisi hesaplanır. Çıkış görüntüsünü elde etmek için bu homografi matrisi giriş görüntüsüne uygulanır. Şekil 4.9’da giriş görüntüsü üzerindeki 4 adet nokta gösterilir.

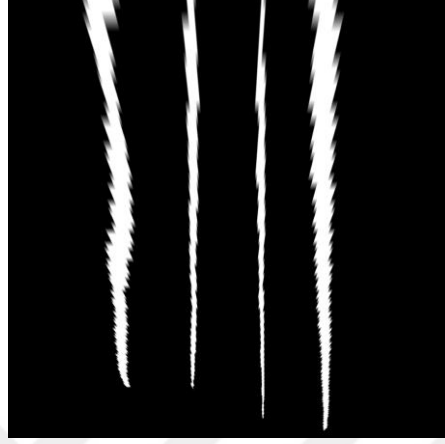


Şekil 4.9. Giriş görüntüsü üzerindeki 4 nokta

Şekil 4.9’daki 4 nokta kullanılarak homografi matrisi elde edilir. Denklem (4.1)’de hesaplanan homografi matrisi gösterilir.

$$H = \begin{bmatrix} -2.18829263e - 01 & -2.03013876e + 00 & 7.77469538e + 02 \\ 1.26318709e - 14 & -4.49594669e + 00 & 1.53009400e + 03 \\ 1.31469079e - 17 & -3.13323718e - 03 & 1.00000000e + 00 \end{bmatrix} \quad (4.1)$$

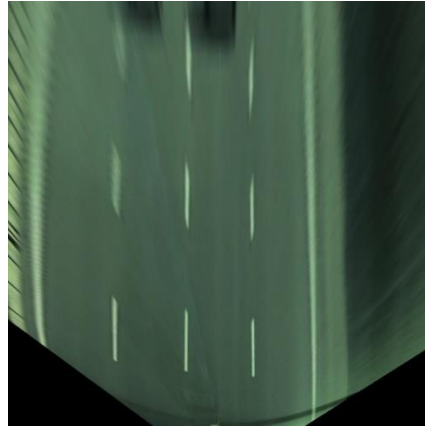
Denklem (4.1)'deki homografi matrisinin ikili görüntü üzerinde uygulanması ile Şekil 4.10'daki ikili görüntü elde edilir. Elde edilen görüntünün boyutları 1280x1280 çözünürlüğüne ayarlanır. Bu çözünürlük ile tüm şeritlerin daha kolay ayırt edilebilir olması sağlanır.



Şekil 4.10. Perspektif dönüşüm uygulanmış ikili görüntü

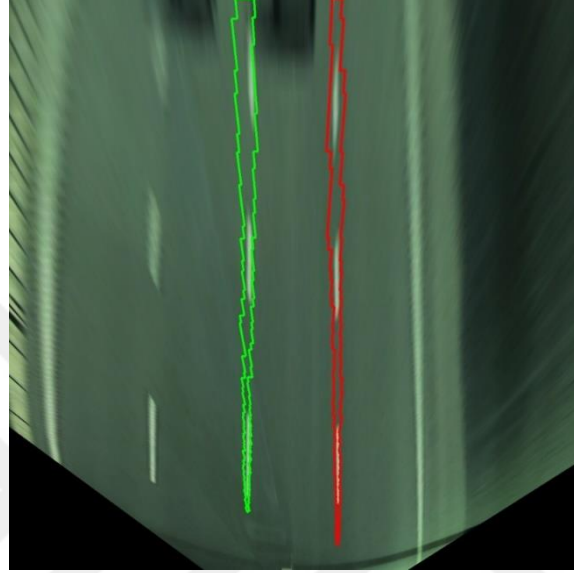
4.2.2.2. Aracın Bulunduğu Şeritlerin Tanımlanması

Şekil 4.10'daki perspektif dönüşüm uygulanmış ikili görüntü üzerinden beyaz pikselleri içeren konturlar, OpenCV kütüphanesinde bulunan kontur bulma fonksiyonları ile bulunur. Araç üzerindeki yerleşik kamera aracın tam orta noktasına yerleştirildiği için kamera görüntüsündeki orta noktayı da aracın merkez noktası olarak değerlendirilir. Kuzey Marmara Otoyolu'ndaki sol ve sağ şeritlerin genişliği 3,75 metredir. Sol ve sağ şeritlerin doğru bir şekilde tanımlanması için aracın merkez noktasının sol ve sağ tarafında 1,875 m uzaklık içinde birer şerit olması gerekmektedir.



Şekil 4.11. Perspektif dönüşüm uygulanmış giriş görüntüsü

Şekil 4.11’de giriş görüntüsüne uygulanmış olan perspektif dönüşümün çıktısı gösterilir. Şekil 4.10’daki görüntü üzerinden aracın bulunduğu şeritleri buluyoruz ve Şekil 4.11’deki görüntü üzerinde çizdirilir. Şekil 4.12’deki görüntü sol ve sağ şeritlerin konturların renkli olarak gösterilir.



Şekil 4.12. Sol ve sağ şeritlerin görselleştirilmesi

Şekil 4.12’deki yeşil renkli nokta bulut sol şerit, kırmızı renkli nokta bulutu sağ şerit olarak gösterilir. Bu nokta bulutlarının değerleri görüntü üzerindeki değerleri yansıtmaktadır. Bu değerleri görüntü uzayından gerçek dünya uzayına dönüştürülmesi gerekir.

Görüntü üzerindeki sol ve sağ şeritlerinin birbirine olan uzaklıkları gerçek dünyada bilinir. Görüntü üzerinde birbirine olan uzaklık piksel tabanlı hesaplanır. Bu işlem şeritlerin araca en yakın noktası yani görüntüdeki en alt kısımdan yapılır. Ölçüm hesapları Kuzey Marmara Otoyolu’nda yapılırken şerit genişlikleri 3,75 m olduğu bilinir. Görüntü üzerindeki 100 piksel 3,75 m’ye denk gelmektedir.

Bu durumda görüntünün çözünürlüğünün 1 pikselde yaklaşık olarak 3,75 cm olduğu hesaplanır. Aynı şekilde y ekseninde çözünürlük 3,33 cm olarak hesaplanır. Gerçek dünya değerlerini elde etmek için 3,33 cm çözünürlük değeri ile piksel değerleri çarpılır. İşlem sonucunda gerçek dünya uzunlukları bulunur.

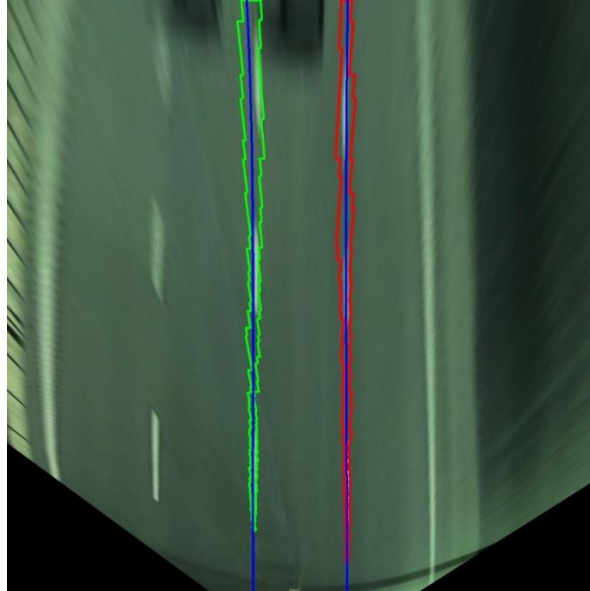
Şekil 4.9'daki yeşil ve kırmızı nokta bulutlarını kullanarak şeritlerin 3. dereceden en küçük kareler polinom uyumu işlemi uygulanır. Denklem (4.2)'de polinom denklemi gösterilir. Bu denklemde üçüncü derecedendir ve deg değeri 3'tür.

$$p(x) = p[0] * x^{deg} + \dots + p[deg] \quad (4.2)$$

Denklem (4.2), deg, deg-1 ve 0 sıralamasında hatanın karesini en aza indiren p katsayılarının bir vektörünü döndürür. Bu işlem kare hatasını en aza indirmek için Denklem (4.3)'i kullanır.

$$E = \sum_{j=0}^k |p(x_j) - y_j|^2 \quad (4.3)$$

Şeritlere uygulanan eğri uydurma (curve fitting) (Neven ve diğ., 2018) işleminin çıktısı Şekil 4.13'te gösterilir. Görüntü üzerindeki mavi renkli çizgiler eğri uydurma işleminin çıktısıdır.



Şekil 4.13. Eğri uydurma işlemi

3. dereceden eğri uydurma sonrası elde edilen denklemdeki her bir katsayının bir anlamı vardır. Birçok otomotiv firmasının kullandığı MOBILEYE (Li ve diğ., 2022) cihazları

da bu denklem sonucu oluşan katsayıları kullanır. Bu katsayılar kullanılarak aracın şeride uzaklığı, şerit ile yaptığı açı, şeridin ilerideki eğrili ve bu eğrilik oranı bulunur.

Tablo 4.2’de MOBILEYE ürününün sisteminde kullanılan parametre ve katsayılar gösterilir. Her bir parametrenin tanımı tablo içinde gösterilir.

Tablo 4.2. MOBILEYE cihazındaki parametrelerin anlamı

Parametreler	Tanımlar
$p0$: Şerit Pozisyonu	Ana aracın merkez hattından sol/sağ şerit işaretlemesine olan yanıl mesafe
$p1$: Yön Açısı	Şerit istikametine göre araç istikameti
$p2$: Eğrilik	İlerideki şeridin eğriliği
$p3$: Eğrilik Türevi	Eğrilik oranı

Tablo 4.3’te sol şeritlere denk gelen şeridin kat sayıları örnek olarak gösterilir. Tablo 4.4’te sağ şeritlere denk gelen şeridin kat sayıları örnek olarak gösterilir.

Tablo 4.3. Sol şerit için kat sayı örneği

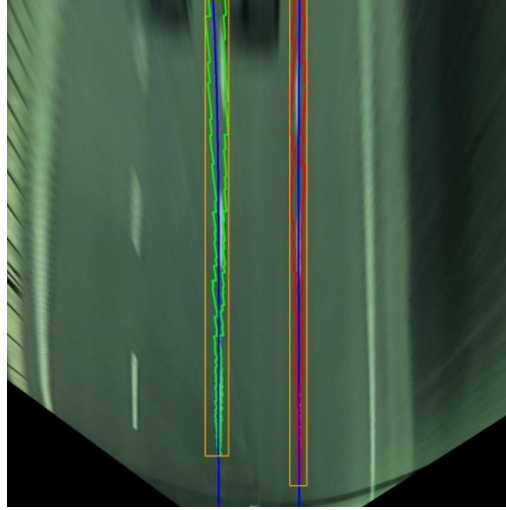
Kat Sayılar	Kat Sayı Değerleri
$p0$	-1,959303805
$p1$	0,001233617
$p2$	-0,000177809
$p3$	8,63e-06

Tablo 4.4. Sağ şerit için kat sayı örneği

Kat Sayılar	Kat Sayı Değerleri
$p0$	1,816926029
$p1$	-0,001316298
$p2$	-0,000268133
$p3$	1,33e-05

4.2.2.3. Şerit Kontur Bilgilerinin Çıkarılması

Bu bölümde aracın bulunduğu şeritlerin konturlarını bulduktan sonra bu bilgileri OpenCV kütüphanesinde yer alan sınırlayıcı kutu (bounding box) bilgilerini bulunur. Bu bilgi içinde şeridin kapladığı genişlik, yükseklik, sol üst x ve y koordinat bilgilerini içerir. Şekil 4.14’te gösterilen görüntüde turuncu dikdörtgen kısımlar bu şeritlerin genişlik ve yükseklik bilgilerini kullanılarak gösterilir.



Şekil 4.14. Şeritlerin konturlarının sınırlayıcı kutu gösterimi

4.2.2.4. Sistemin Performansı

Bu bölümde önerilen ikinci sistemin performansı detaylı bir şekilde açıklanır. Sistem performansı değerlendirilmesi için FPS kullanılır. Sistem performans testlerini NVIDIA GeForce RTX 3090 ekran kartı ve 10 merkezi işlem birimine sahip olan Linux işletim sistemi üzerinde gerçekleştirilir.

Tablo 4.5'te önerilen ikinci sistemde uygulanan her işlem sonrasında elde edilen FPS değerleri gösterilir. FPS değerleri ekran kartı üzerinde elde edilir. İkinci sistemin segmentasyon işleminden sonra uygulanan perspektif dönüşüm işlemi FPS değerini 136,7 değerinden 70,70 değerine düşürür. Perspektif dönüşüm sonrasında uygulanan şerit bilgilerinin çıkarılması kısmındaki tüm işlemler sistemin FPS değerini 40,53 değerine düşürür. Önerilen ikinci sistemin FPS değeri için 40,53 değeri elde edilir. Performans sonuçlarına bakıldığında önerilen ikinci sistem önerilen birinci sistemden 23,73 FPS değeri daha iyi olduğu bulunur.

Tablo 4.5. Önerilen ikinci sistemin performans sonuçları

Uygulanan İşlemlerin Adları	FPS Sonuçları
DeepLabv3+	136,7
DeepLabv3+ + Perspektif Dönüşüm	70,70
DeepLabv3+ + Perspektif Dönüşüm + Şerit Bilgilerinin Çıkarılması	40,53

4.3. Modellerin Eğitimi

Bu bölümde literatür araştırması sonrası literatürde var olan segmentasyon modellerinin veri kümesi üzerinde eğitilmesini içerir. Bu bölümde literatürdeki güncel segmentasyon modelleri eğitilir. Bu modeller U-Net, U-Net++, DeepLabv3 ve DeepLabv3++'dır.

Modellerin eğitimi sırasında kullanılan çözücü (encoder) olarak ResNet34 kullanılır. Aktivasyon fonksiyonu olarak Softmax aktivasyon (Bridle, 1989) fonksiyonu kullanılır. Optimizasyon fonksiyonu olarak Adam optimizasyon fonksiyonu (Kingma ve Ba, 2014) kullanılır. Öğrenme oranı (learning rate) olarak 0,0001 değeri kullanılır. Modeller 40 epoch (devir) ile eğitilir. Tablo 4.6'de modellerin eğitim ve test sonuçları gösterilir.

Tablo 4.6. Modellerin eğitim ve test sonuçları

Modellerin Eğitim ve Doğrulama Sonuçları	Başarım ve Kaybın Teknik Sonuçları			
	Eğitim Başarımı	Eğitim Kaybı	Test Başarımı	Test Kaybı
U-Net	98,90%	98,39%	0,55%	0,81%
U-Net++	98,89%	98,39%	0,55%	0,81%
DeepLabv3	96,61%	96,63%	1,72%	1,71%
DeepLabv3+	98,97%	98,48%	0,52%	0,77%

4.4. Modellerin Karşılaştırılması

Eğitilen modeller test verileri üzerinde test edilir. Test verileri, eğitim sırasında modelin öğrenmediği TuSimple veri kümesinden seçilir. Tablo 4.7'da modellerin performans değerleri 4 farklı performans metrik ile gösterilir.

Tablo 4.7. Modellerin performans sonuçları

Modellerin Adı	Modellerin Performans Sonuçları			
	Kesişim Üzeri Birleşim (IoU)	Kesinlik	Duyarlılık	F1 Skoru
U-Net	58,08%	88,87%	81,50%	84,79%
U-Net++	34,37%	82,03%	51,50%	62,46%
DeepLabv3	59,57%	85,59%	86,74%	86,03%
DeepLabv3+	62,93%	89,72%	85,87%	87,60%

Eğitilen modeller test verileri üzerinde test edilir. Test verileri, eğitim sırasında modelin öğrenmediği TuSimple veri kümesinden seçilir.

Keşişim üzeri birleşim (IoU) performans sonuçlarında en iyi modelin DeepLabv3+ modeli olduğu görülür. Kesinlik (precision) performans sonuçlarında DeepLabv3+ daha başarılı olduğu görülür. Duyarlılık (recall) performans sonuçlarına bakıldığında DeepLabv3 modelin 0,87 ile DeepLabv3+ modelinden daha başarılı olduğu görülür. F1 skorundaki performanslarına bakıldığında DeepLabv3+ modelinin diğerlerinden daha iyi olduğu bulunur. Buradaki performans metriklerinde 4 tanesinden 3'ünde DeepLabv3+ modeli daha başarılı olduğu görülür.

4.5. Model Seçimi

Eğitilen segmentasyon modellerinin bir saniyedeki işlenen kare sayısının yüksek olması sistemin hızlı çalışması için önemli olacaktır. Bu modellerin FPS sonuçları Tablo 4.8'de gösterilir.

Tablo 4.8'de DeepLabv3+ modelinin FPS değerinin diğer modellerden oldukça daha iyi olduğu görülür. Teknik performans sonuçlarını baz alındığında modeller arasındaki en mantıklı modelin DeepLabv3+ olduğu görülür. Bu tez kapsamında DeepLabv3+ modeli kullanılmaya karar verilir.

Tablo 4.8. Segmentasyon modellerinin FPS sonuçları

Modellerin Adı	FPS Sonuçları
U-Net	115,47
U-Net++	64,67
DeepLabv3	68,45
DeepLabv3+	136,7

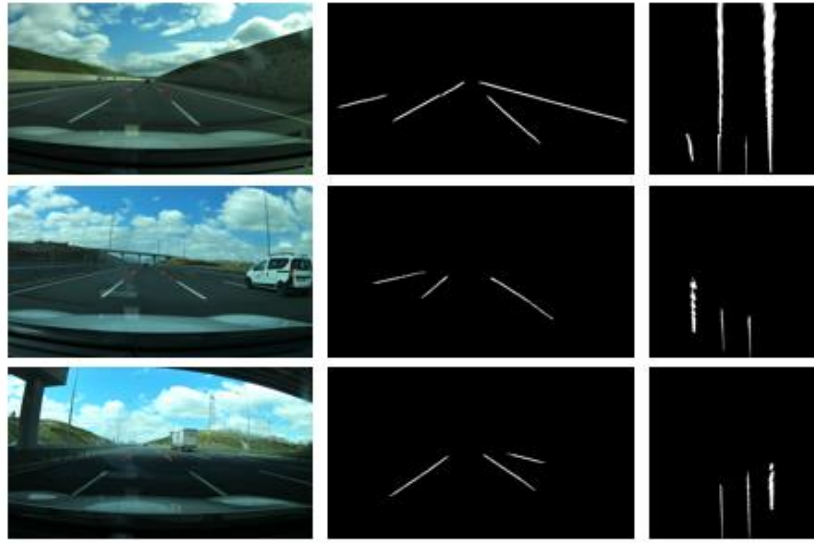
4.6. DeneySEL Sonuçlar

Bu bölümde, tez kapsamında seçtiğimiz DeepLabv3+ modelinin Kuzey Marmara Otoyolu'nda performansı testi detaylı bir şekilde değerlendirilir. Test görüntüleri farklı gün, saat ve hava koşullarından oluşur.

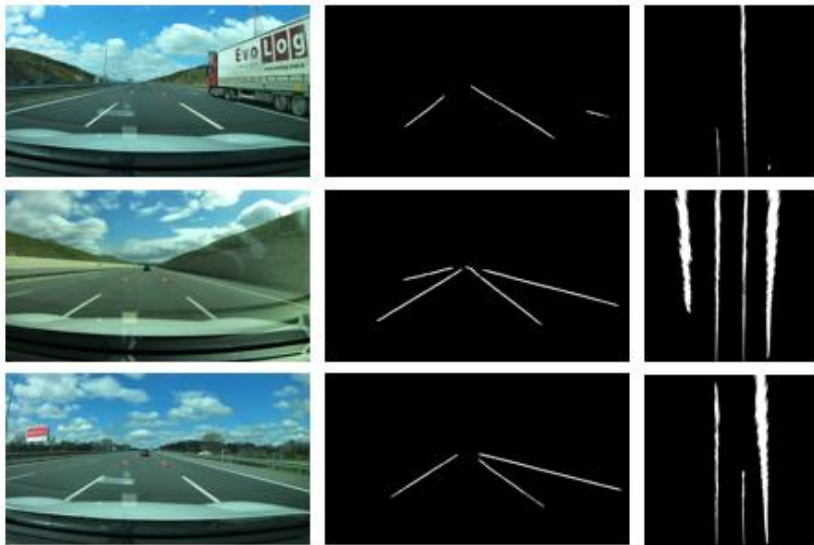
Şekil 4.15'te bulutlu hava ve akşam saatlerinde toplanmış görüntüleri ve model çıktılarını gösterir. Bu görüntülerin sonuçlarından anlaşıldığı üzere model bulutlu

havalarda kesikli şeritleri algılamadaki başarısı düz çizgideki başarısından daha az olduğu görülür. Model bulutlu havalarda düz çizgileri başarılı bir şekilde algılar.

Şekil 4.16’de güneşli havada toplanmış görüntüler gösterilir. Bu görüntülerden anlaşıldığı üzere model kapalı havadaki sonuçlardan daha başarılıdır. Ufukdaki şeritlerin algılanması da burada oldukça başarılıdır. Eğitim veri kümesindeki görüntüler ile Kuzey Marmara Otoyolu’ndaki test görüntüleri birbirlerinden oldukça farklı olmasına rağmen şeritlerin algılanması oldukça başarılıdır. Eğitim verilerine test görüntüsündeki özgün görüntüler de eklendiğinde sistemin doğruluğu oldukça artacaktır.



Şekil 4.15. Bulutlu hava test görüntüsü sonuçları



Şekil 4.16. Güneşli hava test görüntüsü sonuçları

5. ÖNERİLEN ŞERİT TİPİ SINIFLANDIRMA SİSTEMİ

Bu bölümde aracın bulunduğu şeritlerin tipinin algılanması için kullanılan veri seti ve yöntem detaylı bir şekilde ele alınacaktır. Önerilen sistemde yüksek hızda şerit tipi sınıflandırma sistemi gerçekleştirmek hedeflenir.

5.1. Veri Seti

Gerçekleştirilen şerit tipi sınıflandırma modelinde yapay sinir ağlarının eğitilmesi gerekmektedir. Bu eğitim işlemi için Türkiye otoyollarından topladığımız veri kümesi kullanılır. Şekil 5.1’de kullanılan bu veri kümesi gösterilir.

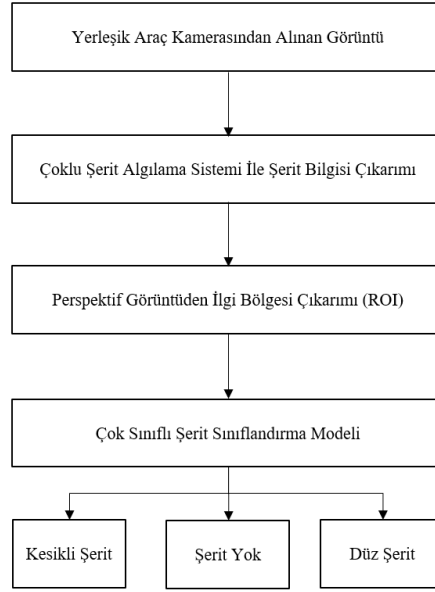


Şekil 5.1. Şerit tipi veri kümesi. Sol: kesikli, orta: şerit yok, sağ: düz şerit

Şekil 5.1’de 3 farklı sınıfta şerit tipi gösterilir. Bu veriler farklı günlerde, farklı otoyollarda ve farklı hava koşullarında toplanır. Bu veri kümesinde toplam 18.000 adet veri vardır. Bu verilerin %70’i eğitim verisi, %15 test verisi, %15 de doğrulama verisi olarak kullanılır.

5.2. Şerit Tipi Sınıflandırılması Sistemi

Bu bölümde tez kapsamında gerçekleştirilen şerit tipi sınıflandırma sistemi detaylı bir şekilde açıklanır. Şekil 5.2’de sistemin genel mimarisi gösterilmiştir.



Şekil 5.2. Şerit Tipi Sınıflandırma Sistemi

Şekil 5.2’de görüldüğü gibi araç kamerasından alınan görüntü çoklu şerit algılama sistemine girer ve burada şeritlere ait olan bilgilerin elde edilmesini sağlar. Şeritlere ait bilgiler kullanılarak perspektif görüntü üzerinden bu şerit görüntüsünü içeren kısımlar kırılır ve çok sınıflı şerit sınıflandırma modeli kullanılarak sınıflandırılır. Sınıflandırma sonucunda 3 sınıftan birisine ait olan bilgi elde edilir.

5.2.1. Çoklu Şerit Algılama Sistemi ile Şerit Bilgilerinin Çıkarımı

Bu bölümde şerit bilgilerinin çoklu şerit algılama sisteminden alınması detaylı bir şekilde açıklanmıştır. İlk olarak araçtan alınan görüntü çoklu şerit algılama sistemine girer ve çıktı olarak şeritlerin perspektif görüntü üzerindeki x eksen bilgisi ile aynı şeridin kontur genişliği bilgisi elde edilir. Bu bilgiler kullanılarak hedef şeridin görüntüsü perspektif görüntü üzerinden kırılarak sisteme vermek üzere hazırlanır.

5.2.2. Perspektif Görüntüden İlgili Bölgesi Çıkarımı

Bu bölümde perspektif dönüşüm uygulanmış görüntü üzerinden ilgili şerit bilgileri kullanarak şerit bilgilerini içeren ilgi bölgesi çıkarımı işlemi detaylı bir şekilde açıklanır.

Çoklu şerit algılama sisteminden gelen şeritin bulunduğu x koordinat değeri ile şerit genişliği bilgisi kullanılarak 1280x1280 çözünürlüklü perspektif görüntü üzerinden ilgili

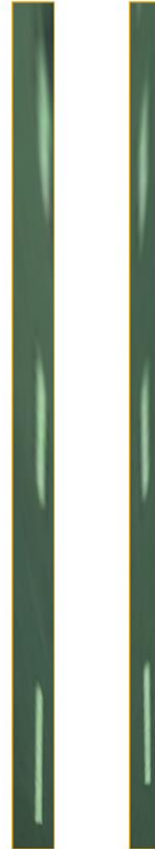
şeridin görüntüsü kırılır. Bu şerit görüntüsü alınırken ilgi bölgesi görüntüsünün yükseklik değeri 1280'e eşittir. İlgi bölgesi görüntüsünün genişlik değerinin hesaplanması için Denklem (5.1) kullanılır.

$$\text{Başlangıç } x \text{ noktası} = x - \left(\frac{\text{Görüntü genişliği}}{2} \right) \quad (5.1)$$

Bu denklemdeki başlangıç x noktası ilgi bölgesinin kırılması için kullanılan değeridir. Kırılan ilgi bölgesi görüntüsünün boyutları Denklem (5.2)'de gösterilmiştir.

$$(x, y) = ((\text{Başlangıç } x \text{ noktası}) + \text{genişlik}, y) \quad (5.2)$$

Şekil 5.3'te aracın bulunduğu şeritlerin ilgi bölgelerinin çıkarılması gösterilir.



Şekil 5.3. Sol ve sağ şeritlerin kırılmış görüntüleri

5.2.3. Sınıflandırma Modelinin Uygulanması

Bu bölümde kırılmış sol ve sağ şeritlerin tipinin algılanmasını detaylı bir şekilde açıklanır. Bu bölümde sınıflandırma algoritması için MobileNetV3 modeli kullanılır. Sınıflandırma modeline girecek giriş görüntülerin boyutları başlangıçta farklı sabit değildir. Giriş görüntüleri 224x224 sabit boyuta çevrilir ve modele verilir. Sistem “kesikli şerit”, “şerit yok” ve “düz şerit” olmak üzere 3 sınıf olarak geliştirilir ve model çıktısı olarak da “kesikli şerit”, “şerit yok” ve “düz şerit” sınıflarından bir tanesi elde edilir.

5.3. Modellerin Eğitimi

Bu bölümde literatür araştırması sonrası literatürde var olan sınıflandırma modellerinin veri kümesi üzerinde eğitilmesini içerir. Bu bölümde literatürdeki güncel sınıflandırma modelleri eğitilir. Bu modeller ShuffleNetV2, ResNet18, SqueezeNet, EfficientNetV2 ve MobileNetV3’tür. Modellerin eğitimi sırasında model giriş boyutu 224x224’e ayarlanır. Optimizasyon fonksiyonu olarak stokastik gradyan inişi (stochastic gradient descent) (Ketkar, 2017) kullanılır. Öğrenme oranı olarak 0,001 kullanılır. Toplu boyutu (batch size) olarak 16’şarlı görüntüler halinde eğitime verilir. Modeller eğitimde 50 epoch (devir) ile eğitilir. Tablo 5.1’de modellerin eğitimi ve test sonuçları gösterilir.

Tablo 5.1. Modellerin eğitim ve test sonuçları

Başarım ve Kayıp Oranları	Başarım ve Kaybın Teknik Sonuçları			
	Eğitim Başarımı	Eğitim Kaybı	Test Başarımı	Test Kaybı
ShuffleNetV2	98,60%	4,00%	99,93%	0,29%
ResNet18	98,72%	3,66%	99,81%	0,30%
SqueezeNet	97,76%	6,79%	99,89%	0,26%
EfficientNetV2	98,68%	3,70%	99,93%	0,15%
MobileNetV3	98,50%	4,33%	99,96%	0,12%

5.4. Modellerin Karşılaştırılması

Eğitilen modeller test verileri üzerinde test edilir. Test verileri, eğitim sırasında modelin öğrenmediği özgün veri kümesinden seçilir. Tablo 5.2’de modellerin performans değerleri 3 farklı performans metrik ile gösterilir.

Tablo 5.2. Modellerin test sonuçları

Modellerin Adı	Çıkarım Süresinin Teknik Sonuçları		
	Kesinlik	Duyarlılık	F1 Skoru
ShuffleNetV2	99,93%	99,93%	99,92%
ResNet18	99,81%	99,81%	99,81%
SqueezeNet	99,87%	99,89%	99,89%
EfficientNetV2	99,93%	99,93%	99,93%
MobileNetV3	99,96%	99,96%	99,96%

Eğitilen modeller test verileri üzerinde test edilir. Test verileri, eğitim sırasında modelin öğrenmediği özgün veri kümesinden seçilir. Tablo 5.2’de modellerin performans değerleri 3 farklı performans metrik ile gösterilir. Kesinlik (precision) performans sonuçlarında MobileNetV3 daha başarılı olduğu görülür. Duyarlılık (recall) performans sonuçlarına bakıldığında MobileNetV3 modelin daha başarılı olduğu görülür. F1 skorundaki performanslarına bakıldığında MobileNetV3 modelinin diğerlerinden daha iyi olduğu bulunur. Buradaki performans metriklerinde 3 tanesinde en başarılı modelin MobileNetV3 olduğu görülür.

5.5. Model Seçimi

Eğitilen sınıflandırma modellerinin bir saniyedeki işlenen çerçeve sayısının yüksek olması sistemin hızlı çalışması için önemli olur. Bu modellerin FPS sonuçları, parametre sayısı ve dosya boyutu Tablo 5.3’te gösterilir.

Tablo 5.3. Modellerin FPS, parametre sayısı ve dosya boyutu

Model adları	FPS Sonuçları	Parametre Sayısı	Dosya Boyutu (MB)
ShuffleNetV2	109,7	1366792	5,3
ResNet18	171,73	11689512	44,7
SqueezeNet	171,7	1248424	4,8
EfficientNetV2	45,98	21458488	82,7
MobileNetV3	121,25	2542856	9,8

Tablo 5.3’teki bilgilere bakıldığında ResNet18 modelinin en yüksek FPS değerine sahip olduğu görülür fakat parametre sayısı ve dosya boyutundan dolayı bu modelin seçilmesi uygun görülür. ResNet18 modelinden sonra en yüksek FPS değeri MobileNetV3 modelidir. EfficientNetV2 modelinin FPS değeri sistem için yetersiz kalır. MobileNetV3 modeli bu modeller arasında en hızlı ikinci model ve diğerlerinden daha başarılı olduğu için sisteme en uygun model olarak seçilir.

6. SONUÇLAR VE ÖNERİLER

Bu tez çalışmasının amacı otonom araçlar için çoklu şerit algılama ve şerit tipi sınıflandırma sistemini gerçekleştirmek. Bu sistemde çoklu şerit algılama sistemi için birçok segmentasyon modeli eğitilir ve performansları karşılaştırılır. Bu karşılaştırma sonucunda TuSimple veri kümesi üzerinde DeepLabv3+ modelinin en iyi sonuç verdiği bulunur. Bu tez kapsamında gerçekleştirilen sistemde de segmentasyon işlemi için DeepLabv3+ modeli kullanılır. Segmente edilmiş görüntüye perspektif dönüşüm uygulayarak yoldaki çoklu şeritlerin kontur bilgileri bulunur. Kontur bilgileri ile kullanılarak şeritleri içeren ilgi bölgesi görüntüleri kırılır ve yeni şerit görüntüleri oluşur.

Sistemdeki şerit tipi sınıflandırma bölümü için sınıflandırma modelleri eğitilmiş ve performansları karşılaştırılır. Bu karşılaştırma özgün veri kümesi üzerinde gerçekleştirilir. Bu karşılaştırma sonucunda MobileNetV3 sınıflandırma modelinin en iyi sonuç verdiği bulunur.

Şerit bilgilerini içeren ilgi bölgesi görüntüleri şerit tipi sınıflandırma modeli tarafından sınıflandırılır. Sınıflandırma modeli “kesikli şerit”, “şerit yok” ve “düz şerit” sınıflarından bir tanesini çıktı olarak verir. Deneysel sonuçlara göre, bu tez kapsamında önerilen yöntem, gerçek zamanlı çalışmadan ödün vermeden çeşitli durumlarda şerit işaretlerini tespit etmede ve sınıflandırmada oldukça başarılıdır.

Önerilen yöntemde derin öğrenme temelli yöntemlerin kullanılmasından kaynaklı olarak bütün senaryoların model üzerinde eğitilmesi şu an için mümkün değildir. Bu yüzden bu sorunu çözmek için bir pekiştirmeli sistem ile sorunun çözülebileceği öngörülmektedir. Ancak otonom araçlar üzerinde koşacak bu sistemin gerçek zamanlı olarak pekiştirmeli öğrenme metodu ile gerçekleştirilmesi hala büyük risklere sebep olabileceği unutulmamalıdır.

KAYNAKLAR

- Abualsaud, H., Liu, S., Lu, D. B., Situ, K., Rangesh, A., Trivedi, M. M. (2021). Laneaf: Robust multi-lane detection with affinity fields. *IEEE Robotics and Automation Letters*, 6(4), 7477–7484.
- Bridle, J. (1989). Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters. *Advances in Neural Information Processing Systems*, 2.
- Chao, F., Yu-Pei, S., & Ya-Jie, J. (2019). Multi-lane detection based on deep convolutional neural network. *IEEE Access*, 7, 150833–150841.
- Chaurasia, A., & Culurciello, E. (2017). Linknet: Exploiting encoder representations for efficient semantic segmentation. In *2017 IEEE visual communications and image processing (VCIP)*, 1-4.
- Chen, L.-C., Papandreou, G., Schroff, F., Adam, H. (2017). Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*.
- Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. *Proceedings of the European Conference on Computer Vision (ECCV)*, 801–818.
- Cheng, G., Li, C., Shi, J., Wang, Y. (2022). Reconstruct from top view: A 3d lane detection approach based on geometry structure prior. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4370–4379.
- Daigavane, P. M., Bajaj, P. R. (2010). Road lane detection with improved canny edges using ant colony optimization. *2010 3rd International Conference on Emerging Trends in Engineering and Technology*, 76–80.
- Ding, D., Lee, C., & Lee, K. Y. (2013). An adaptive road ROI determination algorithm for lane detection. In *2013 IEEE International Conference of IEEE Region 10 (TENCON 2013)*, 1-4.
- Fan, T., Li, Y., Wang, G., Wang, H. (2020). Ma-net: A multi-scale attention network for liver and tumor segmentation. *IEEE Access*, 8, 179656–179665.
- Gupta, S., Tan, M. (2019). EfficientNet-EdgeTPU: Creating accelerator-optimized neural networks with AutoML. *Google AI Blog*, 2(1).
- He, K., Ren, S., Sun, J., Zhang, X. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.

- He, J., Rong, H., Gong, J., & Huang, W. (2010). A lane detection method for lane departure warning system. *2010 International Conference on Optoelectronics and Image Processing*, 1, 28–31.
- Hinneburg, A. (1996). A density based algorithm for discovering clusters in large spatial databases with noise. *KDD Conference*, 1996.
- Howard, A., Sandler, M., Chu, G., Chen, L. C., Chen, B., Tan, M., Adam, H. (2019). Searching for mobilenetv3. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1314–1324.
- Hu, J., Shen, L., Sun, G. (2018). Squeeze-and-excitation networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7132–7141.
- Humaidi, A. J., Hassan, S., Fadhel, M. A. (2018). FPGA-based lane-detection architecture for autonomous vehicles: A real-time design and development. *The Asian International Journal of Life Sciences*, 16(1), 223–237.
- Hur, J., Kang, S. N., & Seo, S. W. (2013). Multi-lane detection in urban driving environments using conditional random fields. *2013 IEEE Intelligent Vehicles Symposium*, 1297–1302.
- Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., Keutzer, K. (2016). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. *arXiv preprint arXiv:1602.07360*.
- Ketkar, N. (2017). Stochastic gradient descent. *Deep learning with Python: A hands-on introduction*, 113-132.
- Kingma, D. P., Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Li, C., Shi, J., Wang, Y., & Cheng, G. (2022). Reconstruct from top view: A 3d lane detection approach based on geometry structure prior. *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4370-4379.
- Li, H., Tarik, K., Arefnezhad, S., Magosi, Z. F., Wellershaus, C., Babic, D., Babic, D., Tihanyi, V., Eichberger, A., Baunach, M. C. (2022). *Phenomenological Modelling of Camera Performance for Road Marking Detection*. *Energies*, 15(1). DOI: 10.3390/en15010194
- Li, H., Xiong, P., An, J., Wang, L. (2018). Pyramid attention network for semantic segmentation. *arXiv preprint arXiv:1805.10180*.
- Long, J., Shelhamer, E., Darrell, T. (2015). Fully convolutional networks for semantic segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3431–3440.

- Low, C. Y., Zamzuri, H., Mazlan, S. A. (2014). Simple robust road lane detection algorithm. *2014 5th International Conference on Intelligent and Advanced Systems (ICIAS)*, 1–4.
- Ma, N., Zhang, X., Zheng, H. T., Sun, J. (2018). Shufflenet v2: Practical guidelines for efficient cnn architecture design. *Proceedings of the European Conference on Computer Vision (ECCV)*, 116–131.
- Neven, D., De Brabandere, B., Georgoulis, S., Proesmans, M., Van Gool, L. (2018). Towards end-to-end lane detection: an instance segmentation approach. *2018 IEEE Intelligent Vehicles Symposium (IV)*, 286–291.
- Niu, J., Lu, J., Xu, M., Lv, P., Zhao, X. (2016). Robust lane detection using two-stage feature extraction with curve fitting. *Pattern Recognition*, 59, 225–233.
- Ronneberger, O., Fischer, P., Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III 18*, 234–241.
- Samet, H., & Tamminen, M. (1986). An improved approach to connected component labeling of images. *International Conference on Computer Vision And Pattern Recognition*, 318, 312.
- Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L. C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4510–4520.
- Simonyan, K., Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- Sun, Y., Li, J., Xu, X., Shi, Y. (2022). Adaptive multi-lane detection based on robust instance segmentation for intelligent vehicles. *IEEE Transactions on Intelligent Vehicles*, 8(1), 888–899.
- Sun, H., Wang, C., El-Sheimy, N. (2011). Automatic traffic lane detection for mobile mapping systems. *2011 International Workshop on Multi-Platform/Multi-Sensor Remote Sensing and Mapping*.
- Tan, M., Chen, B., Pang, R., Vasudevan, V., Sandler, M., Howard, A., Le, Q. V. (2019). Mnasnet: Platform-aware neural architecture search for mobile. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2820–2828.
- Tan, M., Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. *International Conference on Machine Learning*, 6105–6114.

- Tan, M., Le, Q. (2021). Efficientnetv2: Smaller models and faster training. *International Conference on Machine Learning*, 10096–10106.
- T.C. Karayolları Genel Müdürlüğü. (2021). *Trafik Kazaları Özeti 2021*. Ankara.
- URL-1: <https://github.com/TuSimple/tusimple-benchmark>, (Ziyaret Tarihi: 2 Şubat 2023).
- Yang, T. J., Howard, A., Chen, B., Zhang, X., Go, A., Sandler, M., Adam, H. (2018). Netadapt: Platform-aware neural network adaptation for mobile applications. *Proceedings of the European Conference on Computer Vision (ECCV)*, 285–300.
- Zhang, H., Hou, D., Zhou, Z. (2005). A novel lane detection algorithm based on support vector machine. *Progress In Electromagnetics Research Symposium*, 22–26.
- Zhang, X., Zhou, X., Lin, M., Sun, J. (2018). Shufflenet: An extremely efficient convolutional neural network for mobile devices. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6848–6856.
- Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J. (2017). Pyramid scene parsing network. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2881–2890.
- Zhou, Z., Rahman Siddiquee, M. M., Tajbakhsh, N., Liang, J. (2018). Unet++: A nested u-net architecture for medical image segmentation. *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings 4*, 3–11.

KİŞİSEL YAYIN VE ESERLER

- Çelik, A., **Özen, S.**, & Öz, B. (2022). Visual Feeding of Magnetic Cores for Winding Automation. *In Recent Advances in Intelligent Manufacturing and Service Systems: Select Proceedings of IMSS 2021*, 1-8.
- Öz, B., Sevim, B., Akpak, G., Işık, S., Karakullukçu, S., & **Özen, S.** (2020). Kapasitör Dizgisine Yönelik Görü Tabanlı Bir Algoritma. *2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, 1-4.
- Özdemir, V., Çalışkan, A., Özgen, M., Baytürk, E. & **Özen, S.** (2021). The performance comparison of optical character recognition and object detection methods on rubber products. *International Journal of Current Innovations in Interdisciplinary Scientific Studies*. 5(1), 01-15.
- Özen, S.**, Çelik, A., Öz, B., & Çelebi, A. T. (2022). PCB Solder Pad Defect Detection Using GLCM Features. *In Recent Advances in Intelligent Manufacturing and Service Systems: Select Proceedings of IMSS 2021*.
- Özen, S.**, Kaya, U., Semiz, A., & Çelebi, A. T. (2023). Multi-Lane Detection System Based on Segmentation Model for Autonomous Vehicle. *2023 10th International Conference on Electrical and Electronics Engineering (ICEEE)*.

ÖZGEÇMİŞ

İlk, orta ve lise öğrenimi İstanbul'da tamamladı. 2014 yılında girdiği Kocaeli Üniversitesi Elektronik ve Haberleşme Mühendisliği Bölümü'nden 2019 yılında Elektronik ve Haberleşme Mühendisi olarak mezun oldu. 2020 yılında, Kocaeli Üniversitesi Fen Bilimleri Enstitüsü, Elektronik ve Haberleşme Mühendisliği Anabilim Dalı'nda Yüksek Lisans öğrenimine başladı. 2019 yılından beri özel sektörde çalışmaktadır. Şu anda Kıdemli Otonom Sürüş Geliştirme Mühendisi olarak AVL Türkiye'de görev yapmaktadır.

