ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

DEEP LEARNING BASED ROAD SEGMENTATION FROM MULTI-SOURCE AND MULTI-SCALE DATA

Ph.D. THESIS

Ozan ÖZTÜRK

Department of Geomatics Engineering

Geomatics Engineering Programme

MAY 2023

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

DEEP LEARNING BASED ROAD SEGMENTATION FROM MULTI-SOURCE AND MULTI-SCALE DATA

Ph.D. THESIS

Ozan ÖZTÜRK
(501162611)

Department of Geomatics Engineering

Geomatics Engineering Programme

Thesis Advisor: Prof. Dr. Dursun Zafer ŞEKER

MAY 2023

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

ÇOK KAYNAKLI VE ÇOK ÖLÇEKLİ VERİYLE DERİN ÖĞRENME TABANLI YOL BÖLÜTLENMESİ

DOKTORA TEZİ

Ozan ÖZTÜRK
(501162611)

Geomatik Mühendisliği Anabilim Dalı

Geomatik Mühendisliği Programı

Tez Danışmanı: Prof. Dr. Dursun Zafer Şeker

MAYIS 2022

Ozan ÖZTÜRK, a Ph.D. student of ITU Graduate School student ID 501162611 successfully defended the thesis entitled “DEEP LEARNING BASED ROAD SEGMENTATION FROM MULTI-SOURCE AND MULTI-SCALE DATA”, which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor : **Prof. Dr. Dursun Zafer ŞEKER** ....
Istanbul Technical University

Jury Members : **Prof. Dr. Zaide DURAN** ....
Istanbul Technical University

Prof. Dr. Bülent BAYRAM ....
Yıldız Technical University

Doç. Dr. Uğur ALGANCI ....
Istanbul Technical University

Prof. Dr. Cem GAZİOĞLU ....
İstanbul University

Date of Submission : 7 April 2023
Date of Defense : 12 May 2023

FOREWORD

First of all, I would like to express my deepest gratitude to Prof. Dr. Dursun Zafer ŞEKER, for his invaluable guidance, insightful comments, constant support, and never-ending patience throughout my Ph.D. studies and my academic life. I also would like to thank my thesis steering committee members, Prof. Dr. Zaide DURAN and Prof. Dr. Bülent BAYRAM, for their constructive comments and contributions during these years. I would like thanks to Prof. Dr. Martin KADA for his invaluable advice and support.

It has been both a pleasure and a privilege to work alongside close friends in the scientific community. I am indebted to Mustafa Serkan IŞIK for his tireless efforts and unwavering support in my personal and academic life. I am grateful to have the pleasure of working with my three dearest friends, Batuhan SARITÜRK, Mehmet Furkan ÇELİK, and Ali TUNÇ. Their enthusiasm and dedication to helping me succeed were unwavering. I would like to extend my thanks warmly to my late friend Ömer AKIN who was a remarkable individual, whose warmth, intelligence, and humour brightened the lives of everyone who knew him. Although Ömer is no longer with us, he will always hold a special place in my heart and mind.

I would like to thank to Sefa Sefa BİLGİLİOĞLU, Osman ORHAN, Burhan Baha BİLGİLİOĞLU, Cemil GEZGİN, and Ahmet Tarik TORUN for their support and encouragement during this journey.

I would like to express my sincerest gratitude to my mother Nevin ÖZTÜRK and my brother Kaan ÖZTÜRK for their constant support and encouragement throughout my academic journey. During the most challenging times, they have always believed in me.

To my wife, Elif Rüya ÖZTÜRK, who has been my constant support and encouragement, thank you for your unwavering love, patience, and understanding during this challenging time. I am truly fortunate to have her by my side.

I would like to dedicate this thesis to the memory of my father Galib ÖZTÜRK, who passed away unexpectedly during the course of my studies. He was my biggest supporter and his continuous unwavering love and encouragement helped me overcome every obstacle I encountered.

The support received through Application Number 1059B142000410 from the Scientific and Technological Council of Turkey (TÜBİTAK) via The International Research Fellowship Programme for Ph.D. Students (2214-A) is gratefully acknowledged.

May 2023

Ozan ÖZTÜRK
(Geomatics Engineer, M.Sc.)

TABLE OF CONTENTS

ABBREVIATIONS

API	: Application Programming Interface
AI	: Artificial intelligence
CPU	: Central Processing Unit
DL	: Deep Leaning
DSM	: Digital Surface Model
EGC	: Enhanced Global Context-Aware Network
FCN	: Fully Convolutional Network
FP	: False Positive
FN	: False Negative
GAN	: Generative Adversarial Network
GBC-NET	: Graph Convolutional Broad Network
GPS	: Global Positioning System
GIS	: Geographic Information System
KD-Tree	: K-Dimensional Tree
K-NN	: K-Nearest Neighbor
KITTI	: Karlsruhe Institute of Technology and Toyota Technological Institute
LiDAR	: Light Detection and Ranging
LinkNet	: Light Deep Neural Network
NDSM	: Normalized Digital Surface Model
NDVI	: Normalized Difference Vegetation Index
RAM	: Random Access Memory
ReLU	: Rectified Linear Unit
ResNet	: Residual Neural Network
ResUnet	: Residual U-Shaped Network
Sat2Map	: Satellite to Map
SegNet	: Semantic Pixel-Wise Segmentation
SEResNeXt	: Squeeze-and-Excitation Residual Neural Network
SPOT	: Satellite Pour l'Observation de la Terre
SRCNN	: Super-Resolution Convolutional Neural Network
JSON	: Java Script Object Notation
IRRG	: Infrared-Red-Green
ISPRS	: International Society for Photogrammetry and Remote Sensing
IoU	: Intersection over Union
TP	: True Positive
TN	: True Negative
US	: United States
USGS	: United States Geological Survey
U-Net	: U-Shaped Network
V-Net	: V-Shaped Network
WGS84	: World Geodetic System-84
WSGAN	: Weakly-supervised Generative Adversarial Network

SYMBOLS

$\boldsymbol{L_0}$	: Zero level regularization
λ	: Longitude
φ	: Latitude
$\boldsymbol{dx}$	: Pixel resolution along longitude
$\boldsymbol{dy}$	: Pixel resolution along latitude
$\boldsymbol{w}$	: Eigenvalues
$\boldsymbol{v}$	: Eigenvectors
$\lambda_{\boldsymbol{1}}$	: First eigenvalue
$\lambda_{\boldsymbol{2}}$	: Second eigenvalue
$\lambda_{\boldsymbol{3}}$	: Third eigenvalue
$\boldsymbol{k}$	: Number of neighbour
$\boldsymbol{Z}$	: Height of a point
$\boldsymbol{L}_{\lambda}$	: Linearity
$\boldsymbol{P}_{\lambda}$	: Planarity
$\boldsymbol{S}_{\lambda}$	: Sphericity
$\boldsymbol{O}_{\lambda}$	: Omnivariance
$\boldsymbol{A}_{\lambda}$	: Anisotropy
$\boldsymbol{E}_{\lambda}$	: Eigenentropy
$\sum_{\lambda}$	: Sum of eigenvalues
$\boldsymbol{C}_{\lambda}$	: Change of curvature
$\rho_{\boldsymbol{3D}}$	: Local point density
$r_{\text{kNN,3D}}$	: Radius of k nearest neighbors
$\boldsymbol{\Delta H}_{\text{kNN,3D}}$	: Height difference
$\boldsymbol{\sigma H}_{\text{kNN,3D}}$	: Standard deviation of height values

LIST OF TABLES

LIST OF FIGURES

DEEP LEARNING BASED ROAD SEGMENTATION FROM MULTI-SOURCE AND MULTI-SCALE DATA

SUMMARY

Roads are geographical objects that have been the subject of many application areas, such as city planning, traffic management, disaster management, and military interventions. The success of these applications depends on the speed and accuracy of obtaining road information. Researchers have mostly used satellite and/or aerial photographs as data sources in these studies and focused on the automatic acquisition of road information. Although successful results have been obtained with Artificial intelligence (AI)-based approaches, that are widely used recently, automatic segmentation of roads from remote sensing data is still considered a difficult and important problem due to its complex and irregular structure.

AI has been developed to enable computers to realize human abilities such as reasoning, perception, and problem-solving. The most basic expectation is that AI can overcome the problems in which the traditional approaches are insufficient. As a recent trend of AI, deep learning (DL) methods establish a more complex relationship with the data and distinguish the hidden features of the data more accurately. DL is data-driven, and the quality, number, and variety of training data directly impact the performance of the models. For this purpose, comprehensive data sets such as MNIST, COCO, and ImageNet were published. However, the number of datasets containing geographic details is limited compared to others. In addition, datasets containing geographic details can represent only the characteristics of the regions where they were created. Therefore, the models trained with these data sets can only have the capacity to distinguish details at the level that they can only learn from these limited data. It is extremely difficult for these models to effectively predict roads in regions characterized by complex road networks, such as Istanbul.

In this thesis, it is aimed to overcome the data gap in road segmentation studies with DL algorithms, to produce datasets representative of the study region, and finally to use data obtained from different sources together to overcome the problems encountered in existing research using only optical images. This thesis is divided into five main parts. The introduction provides a general overview of the subject matter, including comprehensive information on current studies and the motivation of this thesis.

In the second part, a fast, accurate, and comprehensive road dataset production infrastructure was created using a web map service to overcome data-related problems. For this purpose, it was found appropriate to utilize service providers where maps can be edited based on user requests. Using the Static API feature of the Google Maps Platform, a data generation program was developed in Python programming language. In this program, the properties of the mask images corresponding to the satellite images were defined with a JavaScript code. An automatic static map style was created for road segmentation. In addition, using this program, the desired number of images can be generated randomly or as a sequence at fixed image sizes and within the boundaries of specified test regions. Furthermore, the Google Maps Platform does not provide geographic information about the images. In order to overcome this deficiency, the geo-referencing of these satellite images and corresponding masks was added to the program.

In the third part of the thesis, it is aimed to create an Istanbul road dataset due to the necessity of producing a dataset that represents the characteristics of the region being tested in the road segmentation studies. Istanbul's road network is in a state of development with an ever-increasing population. As it contains different road types and land use details, it is capable of meeting the data diversity required by DL applications. The changing and evolving structure of Istanbul makes it one of the most important regions to be constantly observed and analyzed. In order to examine the contributions of different resolutions of satellite images and different generalization levels of masks in road segmentation studies, the images at zoom levels 14, 15, 16, and 17 from Google Maps were generated in this thesis. Consequently, 10000 optical images and road mask images were produced for each zoom level in the test regions in Istanbul. In order to test the performance of the generated dataset in DL models, the deep residual U-Net architecture was used. When the training metrics of the models' predictions are examined, it was found that the Istanbul dataset achieved successful results in terms of segmenting road pixels at each zoom level separately. In addition, DeepGlobe and Massachusetts datasets, which are widely preferred in road segmentation studies, were included in the analysis to test the prediction performance of the models trained with these datasets generated outside the study region. When these models were tested with the Istanbul dataset, very low prediction statistics were obtained. This situation revealed the necessity of a dataset that reflects the local characteristics of the road network in the study area.

A comprehensive analysis of the Istanbul dataset indicates that the complexity of pixels and background structures in images produced at zoom level 17 adversely affects prediction accuracy. In this context, two alternative models were proposed. In the first approach, training was performed at zoom level 14 without initial weights, and the weights obtained from this training were included as initial weights in the training at zoom level 15. With this approach, weights were transferred from the previous zoom level to the next zoom level and weights were carried up to zoom level 17. In the second approach, L_0 normalization was applied to the images at zoom level 17. According to the results, the consecutive weighting approach improved the prediction accuracy by up to 2% and the L_0 approach improved the prediction accuracy by up to 3%. Furthermore, a new dataset was created with an equal number of images at all zoom levels. The performance of the DL model trained with this new dataset was tested with images of each zoom level, separately. It was concluded that regardless of the zoom level, the model showed successful predictions and can be used in road segmentation studies.

The performance of DL models is affected by the use of only optical imagery in road segmentation studies due to the shadows cast by buildings and objects in urban

areas, and obstacles on the roads. In addition, especially in rural areas, trees covering the roads are the main source of error. A variety of strategies, including new DL architectures, are being investigated in order to resolve these challenges and enhance the quality of road segmentation. By incorporating additional data sources, it is possible to improve the accuracy of road segmentation.

In the fourth chapter of this thesis, the combination of optical imagery and LiDAR point cloud is proposed to overcome the challenges encountered when using only optical images. A feature-level fusion strategy is proposed to conduct 2D and 3D information. In this strategy, the features of the optical images are extracted from DL-based models, while the features of the LiDAR point cloud are calculated by creating a 3D geometric relationship. For each point, the neighborhoods were determined using the k-nearest neighbor (KNN) algorithm. Based on these neighborhoods, a total of 13 LiDAR features were calculated, including 8 eigenvalue-based and 5 geometric 3D-based features. In the end-to-end deep residual U-Net model, high-level features from optical images and calculated LiDAR features are combined at the feature level before the final convolution layer. Furthermore, a ResNet neural network, capable of deeper feature extraction, was added to the encoder blocks of the U-Net model to extract underlying features. The consistency of the proposed strategy was evaluated using ResNet backbones with a different number of layers. According to the results obtained, it was found that the combination of image and LiDAR performed better in road segmentation by increasing the metric statistics from 1% to 5% in all analyses, regardless of the number of layers in ResNet. Additionally, when the individually generated prediction images were analyzed, it was found that the combination method helped to complete the road geometry in areas where road segmentation is potentially problematic such as woodland and shadowed areas.

As a result, by developing a program that can produce optical images specific to the study area and road mask images corresponding to these satellite images, the production of large amounts of data needed in DL-based road segmentation studies was realized faster. In addition, the development of geo-referenced data generation infrastructure made the integration of these images with data obtained from different sources possible. The performances of the DL models trained with the generated datasets provided successful results for road segmentation studies and met the requirements of the data representing the characteristics of the study area. In addition, the problems in the predictions of DL models due to the obstruction of road pixels in optical images due to reasons such as trees or shadows were successfully overcome with the proposed feature-level fusion strategy of data from different sources.

ÇOK KAYNAKLI VE ÇOK ÖLÇEKLİ VERİYLE DERİN ÖĞRENME TABANLI YOL BÖLÜTLENMESİ

ÖZET

Yollar şehir planlamasından trafik yönetimine, afet yönetiminden askeri müdahalelere kadar birçok araştırma ve uygulama alanına konu olan ve önemini her zaman korumaya devam eden bir coğrafi objelerdir. Bu uygulama ve araştırmaların başarısı, yol bilgilerinin elde edilmesindeki hıza ve doğruluğa bağlıdır. Yoğunlukla veri kaynağı olarak uydu ve/veya hava fotoğraflarından yararlanılan bu çalışmalarda, araştırmacılar yol bilgilerinin otomatik elde edilmesine odaklanmışlardır. Her ne kadar günümüzde yaygın olarak kullanılan yapay zeka temelli yaklaşımlarla başarılı sonuçlar elde edilmiş olsa da, uzaktan algılama verilerinden otomatik olarak yolların bölütlenmesi, karmaşık ve düzensiz yapılarından dolayı hala zorlu ve önemli bir problem olarak ele alınmaktadır.

Yapay zeka, akıl yürütme, algılama ve problem çözme gibi insani yetilerin bilgisayarlar tarafından gerçekleştirilmesi amacıyla geliştirilmiştir. Buradaki en temel beklenti, yapay zekanın insanlardaki düşence olgusunu taklit ederek geleneksel yaklaşımların yetersiz kaldığı problemlerin üstesinden gelebilmesidir. Yapay zeka ilk olarak makine öğrenmesi yaklaşımlarıyla gelişim göstermiştir. Makine öğrenmesinde verinin temsili önceden oluşturulmuş doğrusal ilişkilere dayanmaktadır. Ancak, bu kuralcı çalışma prensibi, yapay zekadan beklenen seviyede insan beynine benzer karar verebilme kapasitesini tam olarak karşılamamıştır. Bu kapsamda, yapay zekanın engel tanımayan gelişimiyle birlikte, makine öğrenmesinden derin öğrenmeye geçiş gerçekleşmiştir. Derin öğrenme, makine öğrenmesinin aksine, veri ile daha karmaşık bir ilişki kurarak verinin gizli kalmış özellikleri daha doğru ayırt edebilmektedir. Derin öğrenme aslında her bir katmanında birden çok temsil düzeyinde özellik öğrenimi gerçekleştirilen çok katmanlı derinleştirilmiş yapay sinir ağlarıdır. Derinlik mimarinin oluşturulduğu yapay sinir ağındaki katman sayısını ifade etmektedir. Verinin temsili anlamlı özelliklerin öğrenilmesi ve bu özelliklerdeki benzer dokuların benzer veri setlerinde aranarak soyutlanması şeklindedir. Derin öğrenme veri odaklı olup, eğitim verilerinin kalitesi, sayısı ve çeşitliliği modellerin performansını doğrudan etkilemektedir. Bu amaç doğrultusunda, MNIST, COCO ve ImageNet gibi kapsamlı veri setleri oluşturulmuştur. Ancak, coğrafi detayları içeren veri setleri, bu veri setlerine kıyasla kısıtlı kalmıştır. Bu kısıtlı veri modellerin oluşturduğu karmaşık ilişkilerinin çözümlerinde detayların doğru biçimde genellemesi gerçekleştirilememektedir. Ayrıca, coğrafi detayları içeren veri setleri oluşturuldukları bölgelerin karakteristik özelliklerini temsil etmektedir. Bu veri setleri ile gerçekleştirilen eğitimler sonucunda üretilen modellerin, sadece bu verilerden öğrenebildiği seviyede detayları ayırt edebilecek kapasiteye sahip olması kaçınılmazdır. Örneğin, derin öğrenme modellerinin yol bölütleme problemlerinde

kullanılabilirliğinin test edilmesi amaçlı yaygın olarak DeepGlobe ve Massachusetts yol veri setleri kullanılmaktadır. Ancak, bu veri setlerin kullanılarak eğitilmiş modellerin, İstanbul gibi karmaşık kent dokusuna sahip bölgelerdeki yolları başarılı bir şekilde tahmin etmesi oldukça zordur.

Bu tez kapsamında, derin öğrenme algoritmalarıyla yol bölütleme çalışmalarındaki veri açığının giderilmesi, çalışma bölgesini temsil eden veri setlerinin üretilmesi ve yalnızca optik görüntülerin kullanımında mevcut araştırmaların karşılaştığı problemlerin üstesinden gelinebilmesi için farklı kaynaklardan elde edilen verilerin birlikte kullanılması hedeflenmiştir. Bu tez çalışması beş ana bölümünden oluşmaktadır. Giriş bölümünde genel olarak konu ile ilgili mevcut çalışmalar ve tez çalışmasıyla ilgili kapsamlı bilgiler verilmiştir. İkinci bölümünde, veri kaynaklı sorunların üstesinden gelinebilmesi amacıyla web harita servisi kullanılarak hızlı, doğru ve kapsamlı yol veri seti üretim alt yapısı oluşturulmuştur. Bu amaçla, kullanıcı isteklerine bağlı haritaların düzenlenebildiği servis sağlayıcılarından yararlanılması uygun bulunmuştur. Google Maps Platformunun Static API özelliği kullanılarak, Python programlama dilinde bir veri üretme programı geliştirilmiştir. Bu programda uydu görüntülerine karşılık gelen maske görüntülerinin özellikleri bir JavaScript koduyla tanımlanmıştır. Yol bölütleme çalışmaları için otomatik bir sabit harita stili oluşturulmuştur. Ayrıca, geliştirilen bu program kullanılarak sabit görüntü boyutlarında ve belirlenen test bölgeleri sınırlarında istenilen sayıda görüntüler rastegele veya dizi halinde üretilebilmektedir. Ayrıca, Google Maps Platform, görüntüler hakkında coğrafi bilgiler sağlamamaktadır. Bu açığı kapatabilmek amacıyla, görüntülerin bilinen merkez pikselinin enlem ve boylamı ile bu görüntülerin jeo-referanslandırma işlemini gerçekleştirebilecek özellikte programa eklenmiştir. Tezin üçüncü bölümünde, yol bülütleme çalışmalarında test edilen bölgenin karakterini ifade eden veri setinin üretilmesi gerekliliğinin ortaya konulması amacıyla, İstanbul yol veri seti oluşturulması hedeflenmiştir. İstanbul'un yol ağı, sürekli artan nüfus miktarıyla birlikte gelişim halindedir. Bununla birlikte, farklı yol tiplerini ve arazi kullanım detaylarını barındırması nedeniyle derin öğrenmenin ihtiyaç duyduğu veri çeşitliliğini karşılayabilecek düzeydedir. İstanbul'un değişen ve gelişen yapısı, onu sürekli gözlemlenmesi ve incelenmesi gereken önemli bölgelerden biri yapmaktadır. Büyük ölçekli yol bölütleme çalışmalarında genellikle yakınlaştırma düzeyi olarak 16 veya 17 tercih edilmektedir. Ancak, farklı yakınlaştırma seviyelerinin yol bölütleme çalışmalarına getireceği katkıların irdelenmesi açısından 14, 15, 16 ve 17 yakınlaştırma seviyeleri çalışmalara dahil edilmiştir. Bu yaklaşımla, İstanbul'da belirlenen test bölgelerinde, her bir yakınlaştırma seviyesi için 10000 adet optik görüntüsü ve bu görüntülerin yol maske görüntüleri üretilmiştir. Üretilen veri setinin derin öğrenme modellerinde performansını test etmek için, görüntü bölütleme çalışmalarında başarılı sonuçlar elde etmiş olan derin artık U-Net mimarisi kullanılmıştır. Eğitimler sonucunda elde edilen modellerin tahmin metrikleri incelendiğinde, İstanbul veri setinin ayrı ayrı her bir yakınlaştırma seviyesinde yol bölütlemesi açısından başarılı sonuçlar elde ettiği tespit edilmiştir. Ayrıca, çalışma bölgesinin dışında üretilen veri setleriyle gerçekleştirilen eğitimlerle üretilen modellerin İstanbul görüntüleriyle tahmin performansını test edebilmek için, yol bölütleme çalışmalarında yaygın olarak tercih edilen DeepGlobe ve Massachusetts veri setleri analizlere dahil edilmiştir. Bu verilerle eğitilen modellerde İstanbul veri seti test edilmiş ve oldukça düşük tahmin doğrulukları elde edilmiştir. Bu durum, çalışma alanındaki yol ağının yerel özelliklerini yansıtan bir veri seti üretilmesi gerekliliğini ortaya koymuştur.

İstanbul veri seti kapsamlı bir şekilde incelendiğinde, 17 yakınlaştırma düzeyinde üretilen görüntülerin diğer yakınlaştırma düzeylerine göre daha karmaşık piksel ve arka plan yapısına sahip olduğu ve tahmin doğruluğunun olumsuz etkilendiği tespit edilmiştir. Bu kapsamda, iki alternatif model önerilmiştir. İlk yaklaşımda başlangıç ağırlıkları olmaksızın 14 yakınlaştırma seviyesinde eğitim gerçekleştirilmiştir ve bu eğitimden elde edilen ağırlıklar 15 yakınlaştırma seviyesindeki eğitime başlangıç ağırlıkları olarak dahil edilmiştir. Bu yaklaşımla bir önceki yakınlaştırma seviyesinden bir sonraki seviyeye ağırlıkları aktararak 17 yakınlaştırma seviyesine kadar ağrılıkların taşınması gerçekleştirilmiştir. İkinci yaklaşımda ise 17 yakınlaştırma seviyesindeki görüntülere L_0 normalizasyon işlemi gerçekleştirilmiştir. Elde edilen sonuçlara göre, ağırlık taşıma yaklaşımında %2'lere ve L_0 yaklaşımıyla tahmin doğruluklarında %3'lere varan artışlar elde edilmiştir. Son olarak, tüm yakınlaştırma seviyelerinde eşit sayıda görüntü içeren yeni bir veri seti oluşturulmuştur. Eğitim sonucunda oluşturulan modelde her bir yakınlaştırma düzeyine ait görüntüler ayrı ayrı test edilmiştir. Yakınlaştırma düzeyi ne olursa olsun modelin amaca uygun olarak otomatik yol bölütleme çalışmalarında kullanılabilecek başarılı bir model olduğu sonucuna varılmıştır. Yol bölütlemesi çalışmaları, sadece optik görüntülerin kullanılması, kentsel alanlardaki binaların ve objelerin oluşturduğu gölgeler ve bu nesnelerin yol üzerinde meydana getirdiği engellemeler nedeniyle olumsuz etkilenmektedir. Bu sorunlar, özellikle yüksek çözünürlüklü uydu görüntüleri kullanıldığında daha çok meydana gelmektedir. Ayrıca, özellikle kırsal alanlarda, ağaçların yolları kaplaması bir diğer hata kaynağıdır. Problemlerin çözümünde derin öğrenme modellerinin geliştirilmesinin yanı sıra, sadece uydu görüntülerinin kullanılması yerine uydu görüntülerinin eksikliklerini giderilmesi gerekmektedir. Bu amaçla, farklı türde veriler çalışmalara dahil edilerek modellerin ihtiyaç duyduğu veri gereksiniminin karşılanması sağlanabilir. Bu verilere örnek olarak ele alınabilecek LiDAR verisi, objeler hakkında detaylı bilgiler sunmaktadır. Ayrıca, optik görüntülerde meydana gelen sorunları barındırmadığı için uydu görüntüleriyle birlikte kullanılmasının gerçekleştirilebilmesi halinde yol bölütleme çalışmalarında daha başarılı modeller üretilebilir.

Bu tezin dördüncü bölümünde, tespit edilen problemlerin çözülebilmesi amacıyla optik görüntüler ile LiDAR nokta bulutunun birlikte kullanılabilirliği irdelenmiştir. Bu birleşimin gerçekleştirilmesinde özellik düzeyinde birleşim stratejisinin kullanılması önerilmiştir. Bu stratejide, optik görüntülerin özellikleri derin öğrenme tabanlı modellerden hesaplanırken, LiDAR nokta bulutunun özellikleri 3B geometrik ilişki oluşturularak hesaplanmıştır. Optik görüntüler ve bu görüntülerin yol maskeleri içeren verinin temininde, bu tezin üçüncü bölümünde geliştirilen veri üretim altyapısı kullanılmıştır. Önerilen birleşim stratejisinde, optik görüntülerin özellikleri doğrudan derin öğrenme modelinden hesaplanmıştır. Ancak, 3B nokta bulutu düzensiz bir yapıdadır ve bu yapıdan anlamlı bilgiler çıkartılması için bir dizi ön işleme adımına tabi tutulması gerekmektedir. Noktaların komşularıyla özellik vektörleri oluşturarak bağlamsal bilgi elde edilmektedir. Bu kapsamda, her bir noktanın 3 boyutlu komşuluk ilişkilerinin belirlenmesi ve bu komşuluklara bağlı geometrik özelliklerin hesaplanması gerekmektedir. Her bir nokta için k-en yakın komşuluk (KNN) algoritması kullanılarak komşulukları belirlenmiştir. Bu komşuluklara bağlı olarak 8 adet özdeğer vektörüne ve 5 adet 3B geometrik özelliklere bağlı olmak üzere toplamda 13 adet LiDAR özelliği hesaplanmıştır. Bu hesaplanan özelliklerin, optik görüntü özellikleri ile birleştirilebilmesi için LiDAR özelliklerinin 2B özellik görüntülerine dönüşümü gerçekleştirilmiştir. Optik görüntülerin girdi olarak verildiği derin artık U-Net modelinde son konvolüsyon katmanının öncesinde, yüksek seviye optik görüntü özelikleri ile hesaplanan LiDAR özellikleri, özellik düzeyinde birleştirilmiştir. Ayrıca, U-Net modelinin kodlayıcı bloklarına daha derin özellik çıkarımı gerçekleştirebilen ResNet sinir ağı eklenmiştir. Her biri farklı analiz olacak şekilde entegre edilen ResNet (18, 34, 50 ve 152) mimarileriyle özellik düzeyinde birleşimin performans tutarlılığı irdelenmiştir. Elde edilen sonuçlara göre görüntü ve LiDAR birleşimi, yol bölütleme konusunda tüm analizlerde %1'den %5'e kadar metrik istatistiklerini artırarak daha iyi performans gösterdiği tespit edilmiştir. Bununla birlikte, tek tek oluşturulan tahmin görselleri incelendiğinde, birleşim yönteminin ağaçların olduğu alanlarda yol geometrisinin tamamlanmasına yardımcı olduğu tespit edilmiştir. Bu yöntemle, ağaçlık ve gölge etkilerinin oluştuğu alanlarda derin öğrenme modelinde yalnızca uydu görüntülerinin kullanılmasına nazaran daha iyi bir doğruluk ve kalitede yol piksellerinin bölütlenmesini gerçekleştirildiğini göstermiştir.

Sonuç olarak, çalışma alanına özel optik görüntüler ve bu görüntülere karşılık gelen yol maske görüntüleri üretebilen program geliştirilmesi ile derin öğrenme tabanlı yol bölütlemesi çalışmalarında ihtiyaç duyulan yüksek miktarda verilerin üretiminin daha hızlı gerçekleştirilmesi sağlanmıştır. Ayrıca, jeo-referanslı veri üretim altyapısının geliştirilmesi, üretilen veri setlerinde elde edilen optik görüntülerin, özellikle farklı kaynaklardan elde edilen verilerle kullanılabilirliğini mümkün kılmıştır. Üretilen veri setleri ile eğitilen derin öğrenme modellerinin performansları, hem yol bölütleme çalışmaları için başarılı sonuçlar elde edilmesini sağlamış hem de çalışma alanını karakteristiğini temsil eden verilerin gereksinimini karşılamıştır. Bununla birlikte, optik görüntülerdeki yol piksellerinin ağaç veya gölge gibi sebeplerden engellenmesinden kaynaklı derin öğrenme modellerinin tahminlerindeki problemler, önerilen farklı kaynaklı verilerin özellik düzeyinde birleşim stratejisi ile başarılı bir şekilde giderilmiştir.

1. INTRODUCTION

Roads have always been a status symbol of modern society. In this era, roads are an integral part of sustaining crucial functions such as transportation, traffic management, and navigation facilities. Furthermore, they play a key role in disaster management, emergency response, and military strategies, making them an essential component of economic development and social integration. Therefore, roads remain a prominent area of interest in numerous research studies presently.

The expansion of living spaces has led to the growth of the road network, connecting them with one another. This situation necessitates the real-time updating of road information, which is used in the applications stated above. However, an unavoidable dimensional increase in the data makes the solution to this vital problem difficult by creating a huge data stack. The only solution is to convert road information into meaningful information ready for processing. Traditionally, road information has been acquired through direct survey campaigns or digitization utilizing digital aerial photographs or satellite imagery. However, these techniques require significant time and labor costs. Furthermore, the accuracy of the output of these techniques depends on the proficiency of the person who is performing the digitization process. As a result, it is impossible to provide real-time road data this way, which is a crucial parameter for analysis, monitoring, and planning systems.

With the development of technology, remote sensing data and artificial intelligence are increasingly being used together to overcome challenges associated with acquiring road information. Remote sensing data has proven to be highly effective in various applications, thanks to its numerous opportunities such as access to large-scale and accurate data. The potency has once more been demonstrated in road segmentation studies that extensively utilized satellite images. Various studies achieved promising results by using different types of satellite images. For example, Grinias et al. (2016) employed Quickbird images, while Maboudi et al. (2017) utilized IKONOS satellite images. Then, Abdollahi et al. (2018) utilized Google Earth imagery, and Wu et al. (2019) examined a deep globe dataset based on digital globe imagery. Additionally, Dai et al. (2020) employed Pleiades satellite images. Additionally, there are studies to extract roads using different remote sensing data instead of satellite imagery, such as LiDAR point clouds (Sánchez et al., 2020), SAR images (Wei et al., 2021), and hyperspectral images (Rajamani et al., 2022).

As satellite images continue to expand in size and improve in quality, accurate analysis of such data remains crucial to extracting meaningful information. Conventional image processing techniques and methods are limited for these applications. Nowadays, artificial intelligence-based approaches are the backbone of almost every image processing technique. The evolution of artificial intelligence has progressed from machine learning to deep learning methods, with the goal of bringing a learnable and unsupervised approach to image processing. In the context of road extraction studies, machine learning-based approaches such as Support Vector Machine (Song and Civco, 2004), Kalman Filtering (Movaghati et al., 2010), Conditional Random Forest (Wegner et al., 2013), and Markov Random Field (Grinias et al., 2016) were commonly conducted. However, machine learning methods utilize linear relationships to perform learning. Although these algorithms may achieve results with faster training times, they exhibit lower accuracy compared to deep learning algorithms. Additionally, they may lead to bottlenecks in extracting information from remote sensing data that contain complex information. Consequently, deep learning methods are gradually replacing machine learning techniques, as they enable the extraction of complex information from remote sensing data, efficiently.

Deep learning consists of an input layer, hidden layers, and an output layer as a neural network to imitate the human brain. It has overcome the problems that machine learning cannot solve by making non-linear relations and providing complex correlations. There are many deep learning architectures developed for image analyses, which are pioneered by Convolutional Neural Networks (CNNs). They offered a solution that is able to learn internal representations as a trainable multi-stage architecture (LeCun et al., 2010). They could simplify the data by its hierarchical structure and extract meaningful information. They have shown successful results by creating reliable as well as accurate predictions and have been widely used in various artificial intelligence studies, including road extraction studies (Wang et al., 2015; Liu et al., 2019).

Long et al. (2015) introduced Fully Convolutional Networks (FCNs) by replacing the fully connected layers with convolutional layers. This architecture provided the ability to learn filters at every level as opposed to CNNs for semantic segmentation. Typical CNNs are designed mainly to identify an object within a whole image whereas FCNs are proposed to categorize every pixel of an image according to its corresponding object class. In other words, FCN focuses on segmentation problems in contrast to classification problems. FCNs enabled end-to-end learning and more accurate predictions by providing pixel-level training via back-propagation and dense feed-forward computation in the whole image. After proving the success of the FCN, U-Net was introduced by changing the pooling operators to up-sampling operators (Ronneberger et al., 2015). U-Net provided more precise segmentation with fewer training images. It has taken image segmentation to another level by regenerating the original image size while preserving spatial resolution. These architectures led to significant contributions to road segmentation studies (Buslaev et al., 2018; Abderrahim et al., 2020). However, In U-Net and FCN, as well as in many deep-learning models, there is a degradation problem. He et al. (2015) introduced deep residual learning (ResNet) to avoid this problem, which is easier to optimize and can be able to increase accuracy by extracting underlying features. A novel approach was presented by Zhang et al. (2018a), combining the benefits of both U-Net and ResNet which resulted in improved predictions for road segmentation studies.

In recent years, various deep learning techniques have been proposed to tackle segmentation issues, making it an essential method. However, deep learning is data-oriented and it is essential that the training data set is adequate to represent the variations in reality. There are a few published data sets for road segmentation studies. Nevertheless, these data sets are representative of the attributes of the geographic area in which the data is generated. Models trained with data sets outside the study regions cannot accurately distinguish and predict objects in these areas. Therefore, local road data sets should be created to meet the requirements of applications related to road segmentation studies.

In order to address these challenges, the second chapter of this thesis aims to establish a data generation infrastructure for deep learning models. The infrastructure is built using the Google Maps Platform and its corresponding application programming interfaces (APIs). The Google Maps Platform is a cloud-based mapping service that enables users to create static maps according to their desired style arrangements. A Python-based program has been developed for this purpose, which utilizes the Static Maps API provided by the platform. With this tool, satellite images and their corresponding road mask images can be able to automatically generated with the fixed dimension. The service limits geographic details based on zoom levels, similar to a scale. In this program, users can determine the zoom levels according to the desired image resolution. Additionally, the program can generate randomly or sequentially fixed-sized images within the pre-defined limits set by the user. Furthermore, data sets published in deep learning studies are often in location-independent image formats, such as PNG or JPEG, which can lead to difficulties in their use in different geographical environments. By geo-referencing generated images, it becomes possible to use within geographical information systems.

In the third chapter of this thesis, it was aimed to produce a local road data set within the designated study area of Istanbul city. The efficiency of the data set in terms of road segmentation accuracy was evaluated through the application of deep learning techniques. Furthermore, the significance of creating local road data sets was highlighted by comparing prediction accuracy with those generated by commonly utilized datasets. Satellite images and corresponding mask images of Istanbul city at different zoom levels were generated via the program developed in the second chapter of this thesis. Trained models were produced using Deep Residual U-Net. This study consists of three parts. The first part includes training each zoom level, from 14 to 17, separately. The recall statistics of models were found as 95%, 94%, 92%, 86%, and corresponding F1 - Score statistics were calculated as 95%, 95%, 92%, 86%, respectively. Next, consecutive training with priory weight and L_0 regularization on satellite images were tested to increase the prediction results. The consecutive model and L_0 regularization methods resulted in an increase in both recall and F1-Score by 2% - 3%. In the third part, a new model was generated by combining all images which were produced at different levels. The combined model was found successful for the prediction of roads in all zoom levels. Finally, the performance of these models trained with the Istanbul data set was compared with well-known DeepGlobe and Massachusetts road data sets. Hence, this study illustrated that generating all data sets specifically for the area of interest is crucial.

Countless deep-learning approaches have been developed regarding road segmentation via optical satellite images, yet there are still problems in the prediction of roads caused by the complex structure of the pixels and distortions in the scenery. The principal factor responsible for this problem is the occurrence of shadows coming down onto roads by buildings and trees, as well as the coverage of roads by trees. The success of deep learning depends on how well the training data reflects the road characteristics. Incorporating additional data sources, such as LiDAR data, can overcome these problems. Additionally, in regions where optical images are insufficient to segment roads properly, geometric features presented by LiDAR data can be a valuable input to enhance model performance.

In the fourth chapter of this thesis, a feature-wise fusion strategy was implemented to combine LiDAR data and satellite images. To this purpose, high-resolution optical images of the Google Maps platform were used together with the point cloud data generated from the airborne LiDAR campaign of the U.S. Geological Survey (USGS) in Florida state, USA. LiDAR-derived features, including eigenvalue-based features and geometric 3D properties, were calculated and gridded as a 2D contextual image to incorporate into the fusion strategy. These features were combined with features extracted from satellite images before the final convolution layer in deep residual U-Net architecture. To assess the impact of the fusion strategy on existing problems, different backbones from ResNet 18 to ResNet 152 were tested in the model architecture. The results demonstrated that the proposed fusion strategy statistically improved the road prediction capacity in all ResNet backbones. All models exhibited a 1% to 5% increase in prediction statistics, particularly in woodland and shadowed areas where road segmentation has potential difficulties. In conclusion, the feature-wise fusion strategy of combining 2D high-level features from optical images and 3D geometric features from LiDAR can be useful in areas where there is object blockage in optical images.

This thesis consist of five chapters. Introduction is the first chapter and the forming of an automatic road data set generation infrastructure is described in chapter 2. Chapters 3 and 4 of this thesis are organized based on the two journal papers. The last chapter finalizes the thesis with conclusions and discussions. The journal papers which were published in journals indexed by SCI-Expanded as,

- Chapter 3 is published as a paper entitled *"Generation of Istanbul Road Data Set using Google Map API for Deep Learning-Based Segmentation"* in *International Journal of Remote Sensing*.
- Chapter 4 is published as a paper entitled *"Improving Road Segmentation by Combining Satellite Images and LiDAR Data with a Feature-Wise Fusion Strategy"* in *Applied Sciences*.

2. DATASET CREATION

2.1 Introduction

This chapter describes the development of an automatic road data set generation infrastructure. This infrastructure was designed with careful consideration of several parameters, including the ability to produce a specified number of images, the ability to randomly or grid-wise select production zones, and the geo-referencing of the final product. The implementation of these parameters ensures that the infrastructure is capable of generating high-quality data sets that meet the requirements of various applications such as disaster management, city planning and navigation systems. This development is aimed to provide an efficient and reliable means of generating data sets for road segmentation tasks using deep learning techniques.

2.2 Google Maps Platform

Google Map Static API is a map service that provides access to the Google Maps platform that developers can connect via *Python* language and extract necessary satellite images and their labels. This platform has features that enable the modification of layers of maps on Google Cloud and provides flexibility to alter the styling and management of the interactive maps of Google Maps for many purposes.

Any data layers other than road network information, such as parks, land use types, buildings, and points of interest can be hidden or removed to create the corresponding road network information. Furthermore, some elements of the road class may need to be hidden in the API request, as well. For instance, even though the tunnels are an integral part of the transportation system and road network, these structures cannot be predicted by using optical satellite images only. Road layer labels have been removed and thickness, transparency and color adjustments have been made. The desired output style can be configured as a JSON styling and can be downloaded directly using Python language via Google Maps Static API.

2.3 Zoom Level

The images can be extracted from Google Maps API from zoom level-0 to Zoom level-20 (see Table 2.1). The first zoom level (Level-0) represents the smallest scale possible. At this level, the whole world is presented in a single tile with 512 × 512 pixel size. The zoom levels go up to level-20 in which the image scale is 1:500 (or 0.149 m/pixel) and details such as a mid-size building are visible. From levels-5 to 7, small and large countries can be seen easily. After zoom level 9, large metropolitan areas and cities become noticeable. The details of the city transportation such as the main, side, and crossroads can be seen after Zoom level-14. Buildings can be separated from other structures clearly after zoom level-18. It should be noted that after Zoom level-18, roads cannot be extracted as a whole part, thus it is not recommended to use Zoom levels beyond-17.

In Google Maps, the relation between the pixel coordinate and the word coordinate has a relation of,

$$pixelCoordinate = WorldCoordinate \times 2^{ZoomLevel} \tag{2.1}$$

On the Equator, the resolution of pixels for a chosen image size can be formulated as,

$$\text{Pixel size} = \frac{360^\circ}{2^{zoom\ level} \times \text{Image Size}} \tag{2.2}$$

In this equation, the pixel size is formulated in unit of degrees. To calculate the corresponding metric pixel resolution, the pixel size in degrees can be scaled via the distance between two consecutive longitudes as,

$$\text{Pixel size (meters/pixel)} = \text{Pixel size (degrees)} \times \frac{2\pi R}{360^\circ} \tag{2.3}$$

where R is the radius of Earth in meters. The pixel size along the east-west (latitudinal) direction shortens as one goes away from the Equator to the poles by the factor of the cosine of the latitude.

Table 2.1 : Zoom Levels and Corresponding Scales in Google Maps for an image size 256×256.

Zoom level	Number of tiles	Tile width (° of longitudes)	m/ pixel (on Equator)	Scale
0	1	360	156412	1:500M
1	4	180	78206	1:250M
2	16	90	39103	1:150M
3	64	45	19551	1:70M
4	256	22.5	9776	1:35M
5	1024	11.25	4888	1:15M
6	4096	5.625	2444	1:10M
7	16384	2.813	1222	1 : 4M
8	65536	1.406	610.984	1 : 2M
9	262144	0.703	305.492	1 : 1M
10	1048576	0.352	152.746	1:500k
11	4194304	0.176	76.373	1 : 250k
12	16777216	0.088	38.187	1 : 150k
13	67108864	0.044	19.093	1 : 70k
14	268435456	0.022	9.547	1 : 35k
15	1073741824	0.011	4.773	1:15k
16	4294967296	0.005	2.387	1 : 8k
17	17179869184	0.003	1.193	1 : 4k
18	68719476736	0.001	0.596	1 : 2k
19	274877906944	0.0005	0.298	1 : 1k
20	1099511627776	0.00025	0.149	1:500

2.4 Rectify Images

Google Maps platform allows the extraction of images from API in GIF, JPEG, and PNG formats. In these image formats, the concept of geospatial information cannot be preserved. In cases where these images will be used in combination with other geospatial data types, they must be registered to generate location information.

To rectify these images, the coordinates of the corner pixels and the center pixel are used. Using the coordinate of the center pixel, the corresponding real-world coordinates of corners can be calculated as,

$$\begin{aligned}
\lambda_{\text{NW}} &= \lambda_{\text{Center}} - dx \times \frac{\text{Image size}}{2} \\
\varphi_{\text{NW}} &= \varphi_{\text{Center}} + dy \times \frac{\text{Image size}}{2} \\
\lambda_{\text{NE}} &= \lambda_{\text{Center}} + dx \times \frac{\text{Image size}}{2} \\
\varphi_{\text{NE}} &= \varphi_{\text{Center}} + dy \times \frac{\text{Image size}}{2} \\
\lambda_{\text{SW}} &= \lambda_{\text{Center}} - dx \times \frac{\text{Image size}}{2} \\
\varphi_{\text{SW}} &= \varphi_{\text{Center}} - dy \times \frac{\text{Image size}}{2} \\
\lambda_{\text{SE}} &= \lambda_{\text{Center}} + dx \times \frac{\text{Image size}}{2} \\
\varphi_{\text{SE}} &= \varphi_{\text{Center}} - dy \times \frac{\text{Image size}}{2}
\end{aligned} \tag{2.4}$$

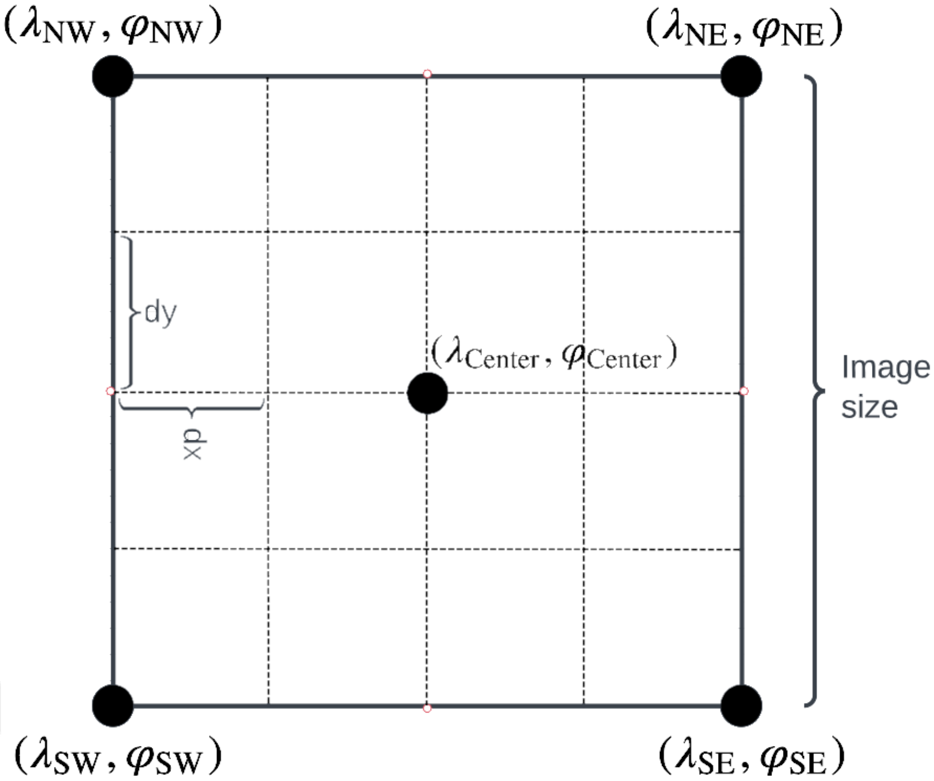
In these equations, $(\lambda_{\text{NW}}, \varphi_{\text{NW}})$, $(\lambda_{\text{NE}}, \varphi_{\text{NE}})$, $(\lambda_{\text{SW}}, \varphi_{\text{SW}})$, and $(\lambda_{\text{SE}}, \varphi_{\text{SE}})$ represent the geographic coordinates of North-West, Nort-East, South-West, and South-East corner pixels, respectively. $((\lambda_{\text{Center}}, \varphi_{\text{Center}}))$ are the longitude and latitude of the center pixel and dx, dy are the pixel resolutions in degree unit along the longitude and latitude directions (see Figure 2.1). These coordinates are used to rectify satellite images and corresponding road masks to generate images in GeoTIFF format via *rasterio* library.

2.5 Selection of Image Area

The images can be generated via random sampling inside a given area bounded by the geographic coordinates of north-west and south-east corners of the rectangular region. In Figure 2.2, random image samples, taken between 26-45 E and 36-43 N region, are shown in an interactive map created by *Folium* in Python language.

Aside from the random generation of these images, the tool can be used to extract grid-wise images with an overlapping area between neighboring images. In this case, the center coordinates of the generated images are shifted with constant separation along latitude and longitude to create a grid of images. Figure 2.3 shows an example of a grid-wise image generation in the same region.

This feature of the tool can be exploited when the study area needs to be covered completely or there is a need to overlap different data sources. Another example can



$(\lambda_{NW}, \varphi_{NW})$
$(\lambda_{NE}, \varphi_{NE})$
dy
dx
$(\lambda_{Center}, \varphi_{Center})$
Image size
$(\lambda_{SW}, \varphi_{SW})$
$(\lambda_{SE}, \varphi_{SE})$

Figure 2.1 : Center and corner coordinates of an image.

be seen in Figure 2.4 to visualize how these satellite and corresponding mask images can be integrated with airborne LiDAR point cloud tiles, located in Florida USA. In this figure, the satellite images are intentionally left un-overlapped to be able to show the frames separately. The red frames on top of the satellite images represent the airborne LiDAR point cloud frames.

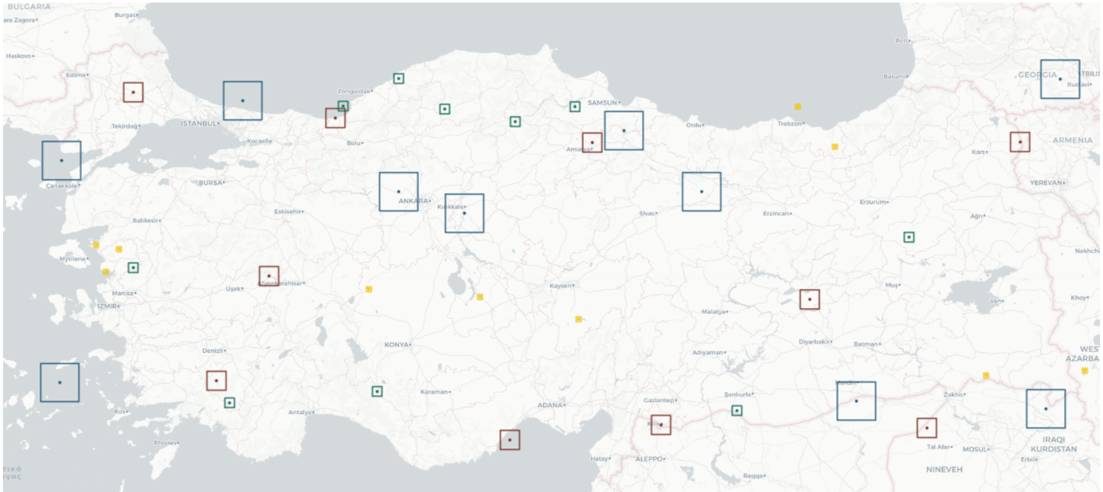


Figure 2.2 : Random images in the selected study area.

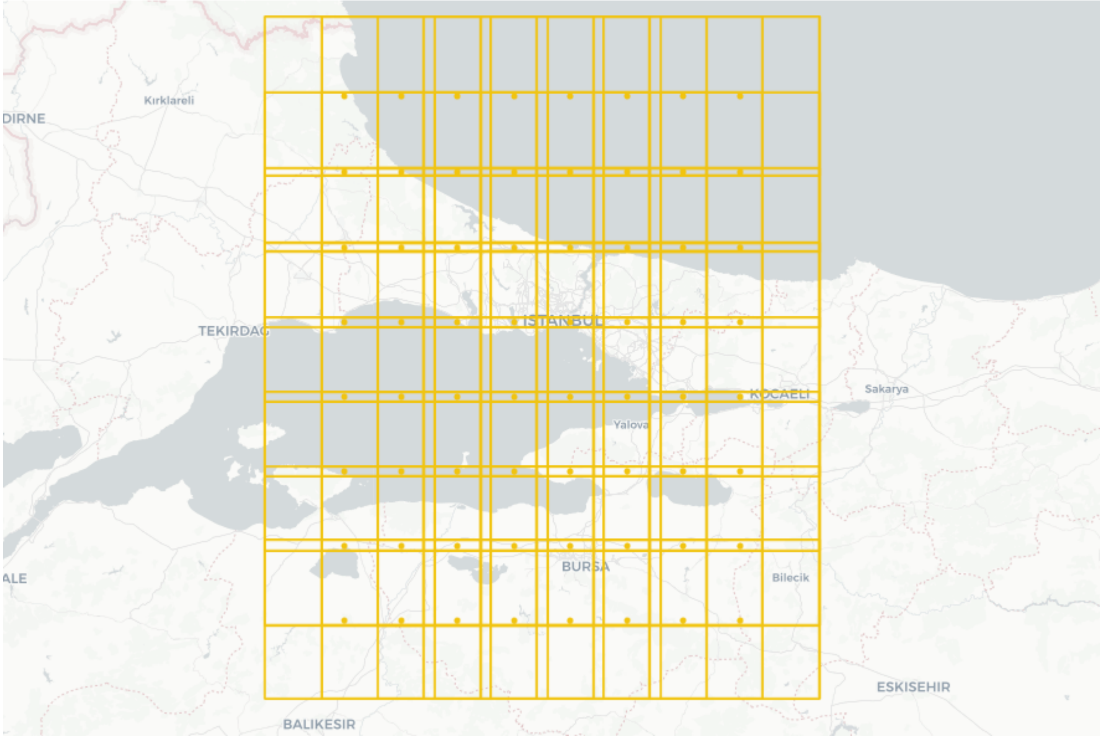


Figure 2.3 : Grid-wise images in the selected study area.

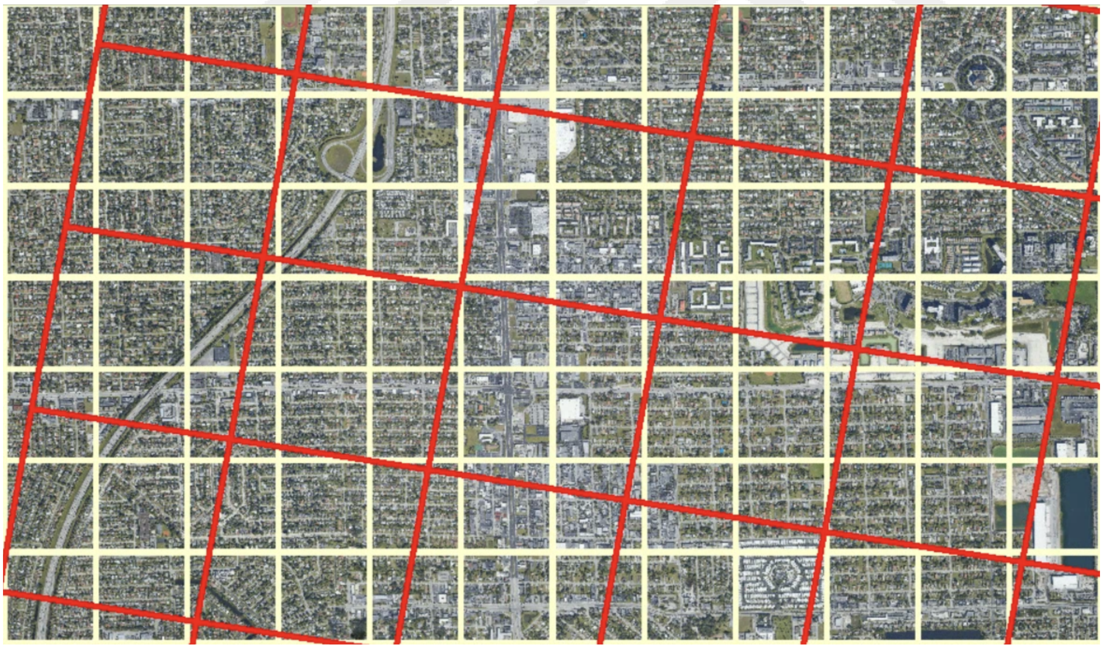


Figure 2.4 : Satellite images and airborne LiDAR data overlapped.

3. GENERATION OF ISTANBUL ROAD DATA SET USING GOOGLE MAP API FOR DEEP LEARNING-BASED SEGMENTATION[1]

3.1 Introduction

Roads are the primary standard data source in transportation network design, navigation, emergency planning in disasters, response and evacuation, urban planning, military exercises, GIS data updates, and many applications. In today's cutting-edge era, continuous monitoring and updating of such data with high accuracy are indispensable. Accurate and reliable information regarding the road network enables accessibility information from local to global applications.

Traditionally, road data is produced with the conventional digitization method. Although this method seems sufficient in terms of accuracy, it requires both manpower and time to create country-wide seamless road network data. Furthermore, the dynamic structure of cities makes this information outdated quickly. This problem has led the roads to be a subject of many remote sensing studies, from black and white aerial images to high-resolution satellite images. The earlier studies that used remote sensing data have suffered a lot from the complex structure of pixel content in road segmentation. The size of the roads was too small to be detected from satellite imageries with lower spatial resolutions, which had a negative impact on these segmentation studies. The noise in the background, the blockage and shadow effects of buildings and trees in the scenery, were too high and could not be eliminated by the used methods proposed in the previous studies. That is why the determination of the main roads was better than the intermediate roads considering the resolution of the image and the size of the main roads, and it was quite difficult to distinguish

[1] This chapter is based on: Özturk, O., Işık, M. S., Sarıturk, B., Şeker, D. Z. (2022). Generation of Istanbul Road Data Set using Google Map API for Deep Learning-Based Segmentation. International Journal of Remote Sensing, 43(8), 2793-2812. https://doi.org/10.1080/01431161.2022.2068989

sub-classes of road type such as asphalt, soil, etc. (Gruen and Li, 1995; Bacher and Mayer, 2004, 2005; Amini, 2009).

The segmentation of roads has become more detectable and interpretable through the detailed texture information of the high-resolution images with the advancement in the field of satellite technologies in the recent years. There have been many studies that uses high-resolution optical satellites, such as Spot, Ikonos and QuickBird imageries (Mokhtarzade and Zoej, 2007; Christophe and Inglada, 2007; Movaghati et al., 2010; Miao et al., 2015; Maboudi et al., 2017; Ghandorh et al., 2022). It is important to note that while satellite images from non-commercial satellites are insufficient for these studies in terms of spatial resolution, the use of high-resolution images from commercial satellites can be costly and unaffordable for developing countries.

Remote sensing image analyses are speeding up by using newly developed deep learning-based modeling approaches. This can be observed in the recent segmentation studies such as Liu et al. (2020); Zhao et al. (2021); Boonpook et al. (2021); Xia et al. (2021). In these studies, deep learning architectures such as U-Net, LinkNet, SegNet, and FCN became prominent models for the segmentation of roads. Among these neural networks, U-Net architecture has been used for the segmentation of road networks from remotely sensed data sets (Abdollahi et al., 2018; Zhang et al., 2018a; Abderrahim et al., 2020). Abdollahi et al. (2018) inspired by the performance of U-Net architecture for the segmentation of biomedical images, and successfully implemented the same architecture for the road segmentation problem. they showed that U-Net model outperformed the other models built using FCN, SRCNN and SegNet architectures. Zhang et al. (2018a) proposed the use of U-Net architecture together with deep residual blocks, so called Deep ResUnet, that replaces the classical convolutional block with a residual block. Cira et al. (2020) evaluated the effect of backbone integration and the production of hybrid deep learning models for road segmentation from high-resolution aerial images, in addition to analyzing the original design of U-Net, LinkNet, and Feature Pyramid Network to create a large-scale road segmentation system in Spain. They found that the model created using U-Net architecture with SEResNeXt50 backbone showed the best performance among other models. Abdollahi et al. (2020a) proposed a new deep learning-based model V-Net, a similar model to U-net, for creating a road segmentation map. The model is successfully tested with Massachusetts road data set, and another one in Ottowa generated using Google Earth images. They indicated that V-Net architecture provides convergence over classical U-Net architecture with no residual block. However, the instability of convergence is not reported for residual U-Net. Zhang and Hu (2021) took a different look at the extraction of roads in forested areas. It has successfully extracted principal and secondary roads by combining multivariate Gaussian and Laplacian of Gaussian (LoG) filters and VGG 16. As they mentioned in their study, the number of image samples they posses is not enough to train a successful model using U-Net architecture, but the architecture is further recommended in the availability of large number of image samples. In addition, Wang et al. (2021b) proposed segmenting Unmanned Aerial Vehicle images and Massachusetts road information using a deep convolutional neural network. In comparison with previous deep learning models such as FCN, U-Net, SegNet, DeepLab, D-LinkNet, and EGC-Net, this technique called global context-aware network has achieved more successful outcomes. A more recent deep learning approach called generative adversarial network (GAN) provided comparative segmentation results with the state-of-art deep learning models using a less training data set with more efficient computation time (Zhang et al., 2019). GAN models have been successfully applied in the extraction of the road network in literature (Shi et al., 2018; Varia et al., 2018; Abdollahi et al., 2020b; Yang and Wang, 2020; Abdollahi et al., 2021). The majority of the studies that applied GAN algorithm for the road segmentation has reported the use of U-net based architecture for the *"generative model"* part. In light of these recent studies that conclusively proved the success of U-Net based architectures for the segmentation of road network, the model used in this study was chosen as a modified version of U-net architecture, as well.

Several road datasets have been published on web services such as Kaggle, Github, Amazon Web Services and Institutional Web Sites. The leading ones are; Deep Globe road dataset and the Massachusetts road dataset. DeepGlobe road dataset was published as a challenge and is expected to overcome complex road structure difficulties in developing countries (Demir et al., 2018). Massachusetts Road Dataset is a more regular data structure with pre-processes applied compared to the Deep

Globe Dataset (Mnih, 2013). Another dataset, the KITTI Vision Benchmark Suite, was produced with a 2D Bird's Eye View unlike other datasets mentioned above. KITTI, produced mainly for the automated driving studies, is open access dataset and consists of 600 annotations (Fritsch et al., 2013).

Each published data set represents the characteristics of its region. These characteristics vary considerably from country to country, and even between cities in the same country. Even if the road data sets would be generated in detail with high resolution, the accurate and reliable segmentation of the road information is challenging for the regions outside the data coverage. Data dependency highly affects the prediction capacity of the model taught in deep learning studies. It is critical to developing data sets that represent the study area to include data with different characteristics serving the same purpose in the model training of the model. Due to the lack of diversity in public domain data sets and the large volume of data is required for the training of the models, researchers frequently use data augmentation. The primary reason for this is to avoid overfitting in training. However, rather than using the data augmentation approach, the inclusion of a large number of randomly generated data that do not overlap in the training of the model will provide more realistic estimation results because the model will constantly see and learn different information with similar characteristics in the same region. For this purpose, unique data sets should be produced for each study, and it is essential to perform analyses in line with these data. In this regard, online map platforms come to the forefront for generating up- to- date road data set.

In this study, the road data set that serves the purpose of automated road segmentation for Istanbul city was produced. Study areas were selected from different regions in the city. Static Map API services of the Google Maps platform (Google Map Static API) was used to generate the Istanbul data set. Random images from satellite images and their corresponding mask images in different zoom levels were produced in the study areas. The utility of Istanbul road data set was analyzed using U-Net segmentation model with ResNet50 backbone. Four different models were trained using images with the same zoom level to assess the predictability of models in varying levels. The performance of these models was compared with a combined model trained with the images using all zoom levels. Models trained with Deep Globe road data set and Massachusetts road data set were tested with Istanbul road data set to clarify the necessity of producing a local data set. The use of different zoom levels of satellite images in road segmentation studies is tested to guide the studies that can benefit from the road network data at a different scale.

This study begins by presenting the data set generated for this research and others that were used for comparison. It is followed by the theoretical explanation of the deep learning model based on the U-Net algorithm with the ResNet50 backbone. The details of the pre-processing and the implementation of the deep learning model were explained together with the numerical results of the performance of trained models.

3.2 Materials and Methods

3.2.1 Data sets

In this research, three data set were used: publicly available satellite image-based Deepglobe Challenge Road Extraction Dataset, the Massachusetts Roads Datasets, and Istanbul Road Dataset that was generated within this study.

3.2.1.1 DeepGlobe road data set

DeepGlobe Road Challenge is a project developed to obtain road network data in developing countries for disaster management. For this purpose, the satellite images and overlapping mask images of the test areas were opened to online access under the name of DeepGlobe road data set (hereafter will be named as DeepGlobe data set). This dataset consists of DeepGlobe satellite imageries, and covers urban and rural sites of Thailand, Indonesia and India. The main challenge expected to be overcome using this data set is to be able to accurately predict the road information from high resolution scenery with complex structures. This data set consists of 8570 images, covering an area of 2200 km^2, generated from 190 DeepGlobe Satellite images with 0.5 m resolution and 1024×1024 pixel size. Example images from this data set are given in Figure 3.1 The computation parameters and numerical results of some of the deep learning studies conducted with this data set are presented in table (Table 3.1).

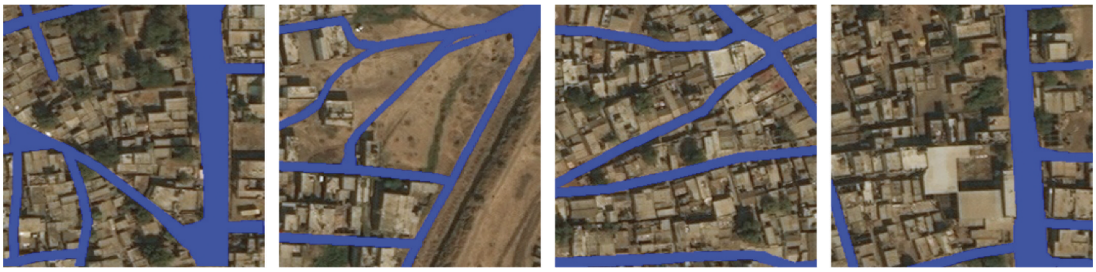


Figure 3.1 : Example images from DeepGlobe road data set.

3.2.1.2 Massachusetts road data set

Massachusetts road data set (hereafter will be named as Massachusetts data set) was generated by processing the aerial images of Massachusetts State. While creating the mask of the road data, 7 pixels width thickness was used no smoothing was applied in the rasterization of the vector data from the OpenStreetMap project. In total, 1172 images with a resolution of 1500 × 1500 pixel size and 1.2 m resolution have been made publicly available for an area of 2.25 km^2. Figure 3.2 shows several samples of Massachusetts data set. The deep learning studies conducted with this data set were summarized with the computation parameters and numerical results in Table 3.1.

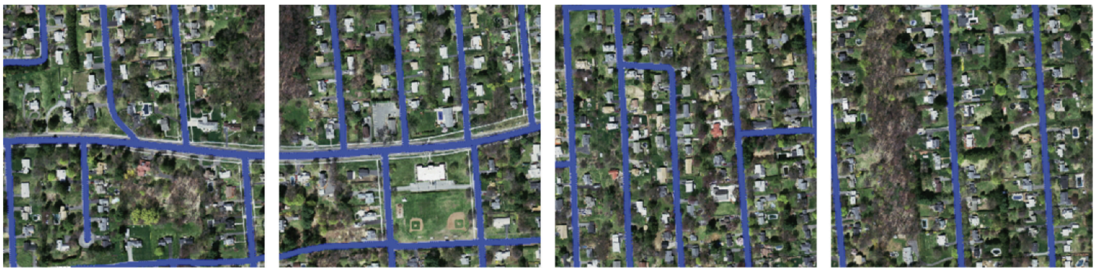


Figure 3.2 : Example images from Massachusetts road data set.

3.2.1.3 Istanbul road data set

Because of the convergence of the continents of Europe and Asia, having a rich historical and cultural heritage, Istanbul is located in an important transportation and commerce center, with a population of more than 15 million people. It is one of the largest metropolitans both in Europe and in the World. The road network expands in parallel with the speed of overpopulation. Thus, in this study creating an accurate

Table 3.1 : Some results of Using DeepGlobe and Massachusetts data set.

Data Set	Study	Model	Backbone	Data Aug.	Precison	Recall	F1 Score	IoU
DeepGlobe	Zhou et al. (2018)	D-LinkNet	ResNet-34	Yes	-	-	-	0.64
	Wu et al. (2019)	Attention Dilation-LinkNet	ResNet-50	Yes	-	-	-	0.65
	Zhu et al. (2021)	Global Context-Aware and Batch-Independent Network (GCB-Net)	ResNet-34	Yes	-	-	0.82	0.71
	Wei and Ji (2022)	Scribble-Based Weakly Supervised Network	-	No	0.80	0.71	0.72	0.58
Massachusetts	Singh and Dash (2019)	Two Convolution Neural	-	No	0.88	0.89	0.89	-
	He et al. (2019)	Integrated Encoder-Decoder Network	-	Yes	0.87	0.81	0.84	-
	Wu et al. (2021)	Dense-Global-Residual Network	Residual Blocks	No	-	0.72	0.77	0.63
	Abdollahi et al. (2020a)	V-NET	-	Yes	-	-	0.91	0.84
	Abderrahim et al. (2020)	U-Net	-	Yes	0.87	0.89	0.88	-
	Wang et al. (2021a)	Inner Convolution Integrated Encoder-Decoder Network	-	Yes	0.87	0.82	0.85	-

Istanbul road data set (hereafter will be named as Istanbul data set) was aimed to be used for deep learning applications. The satellite and corresponding mask images were generated from four different regions. Figure 3.3 shows the regions with urban and rural details constituting the study area.

Istanbul road data set, generated within the scope of this study, is created using Static Map API services of the Google Map platform. Similar studies that use Google Map Static API for deep learning studies appear in literature. Isola et al. (2017) published a data set called sat2map for the translation of satellite images into map view. The same data set was used by Srinivasan et al. (2020) and Spiegl (2021) for image translation using deep learning. Xu and Zhao (2018) and Ganguli et al. (2019) created their satellite and mask data for the translation of satellite images into maps. Li et al. (2019a) used the same approach for the segmentation of buildings by generating their satellite images from Google Map Static API and their corresponding building labels from not only Google Map Static API, but also from OpenStreetMap for compensating the information loss in the Google Map data set.

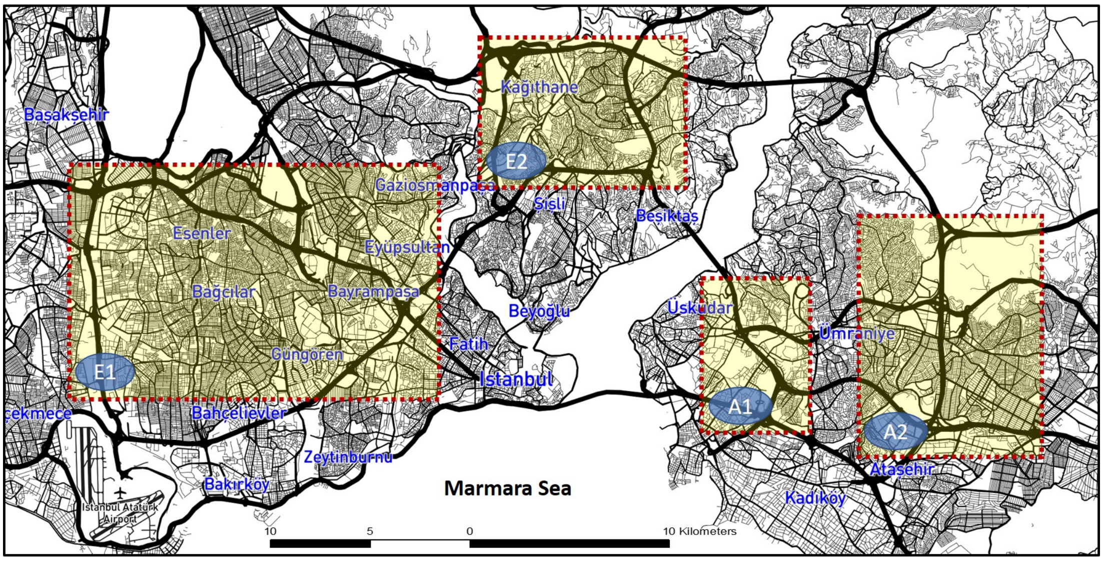


Figure 3.3 : Selected study areas (E: Europe, A: Asia).

In order to obtain images via the Google Map platform, one of the most important parameters, the zoom level, must be determined priory. While zoom level 0 (1:500 Million scale) covers the whole world in a single frame, at the zoom level 20 (1:500 scale), which is the maximum zoom level, resolution increases up to $\sim$ 0.15 m. Zoom level 17 (1/4000 scale) is generally recommended for large scale studies. However, zoom levels 14 (1:35000 scale), 15 (1:15000 scale), 16 (1:8000 scale) are also included in the study, as different zoom levels can contribute to road studies at different scales. These levels correspond to $\sim$ 9.55, $\sim$ 4.77, $\sim$ 2.39 and $\sim$ 1.19 m/pix. image resolutions, respectively. The areas covered by these levels from zoom level 14 to 17 on Earth (from left to right) are shown in Figure 3.4.

Each zoom level includes 10000 pairs of images and their corresponding mask images, generated in random areas that fall within the regions shown in Figure 3.5. The size of the images created is 256×256. Sample images and corresponding masks are shown in Figure 3.5.

3.2.2 Preprocessing

Deep Globe data set was directly clipped from 1024 x 1024 to 256 x 256 pixel size sub-images. Massachusetts data set was first resized from 1500 x 1500 pixel size to 1024x1024 pixel size, and then clipped into same size sub-images. Finally, in all

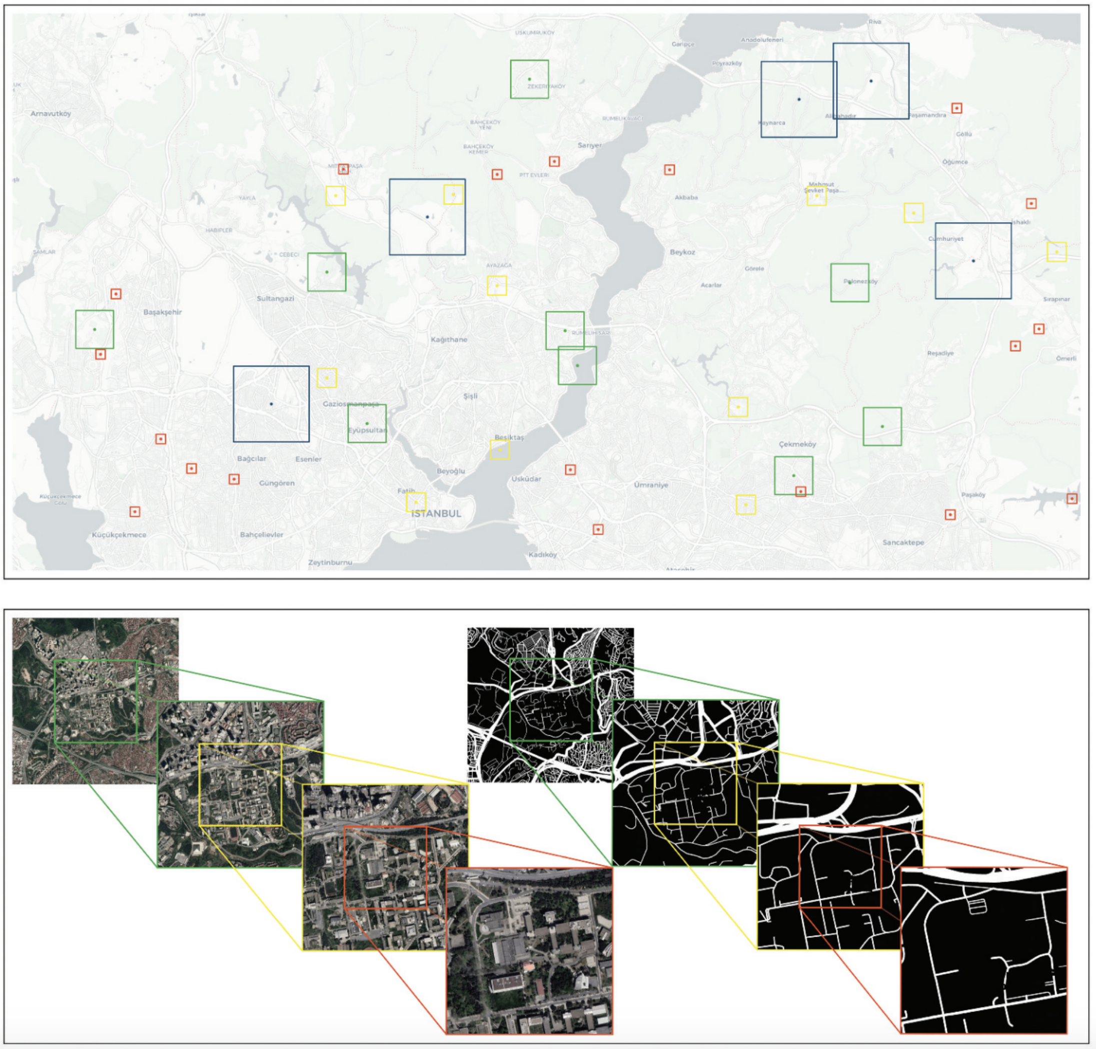


Figure 3.4 : The ground coverage of randomly generated images at different zoom levels on the upper part. The area covered by satellite images (on the left), and corresponding mask images (on the right) in zoom levels from 14 to 17 on the lower part.

data sets including all zoom levels of the Istanbul data set, image and mask files of 10000 images were created. All the preprocessing, performed were applied to the mask images, and finally, the mask images were converted into binary images.

When the Istanbul data set is extensively examined, zoom 17 has more complex pixel structures compared to other zoom levels. It is expected that using L_0 regularization will contribute to the generalization of this complexity. In L_0 regularization, the scattered data are sparse with the least assumptions. With this process, training and segmentation processing time can be accelerated and the generalization of data can be increased. L_0 regularization is defined as;

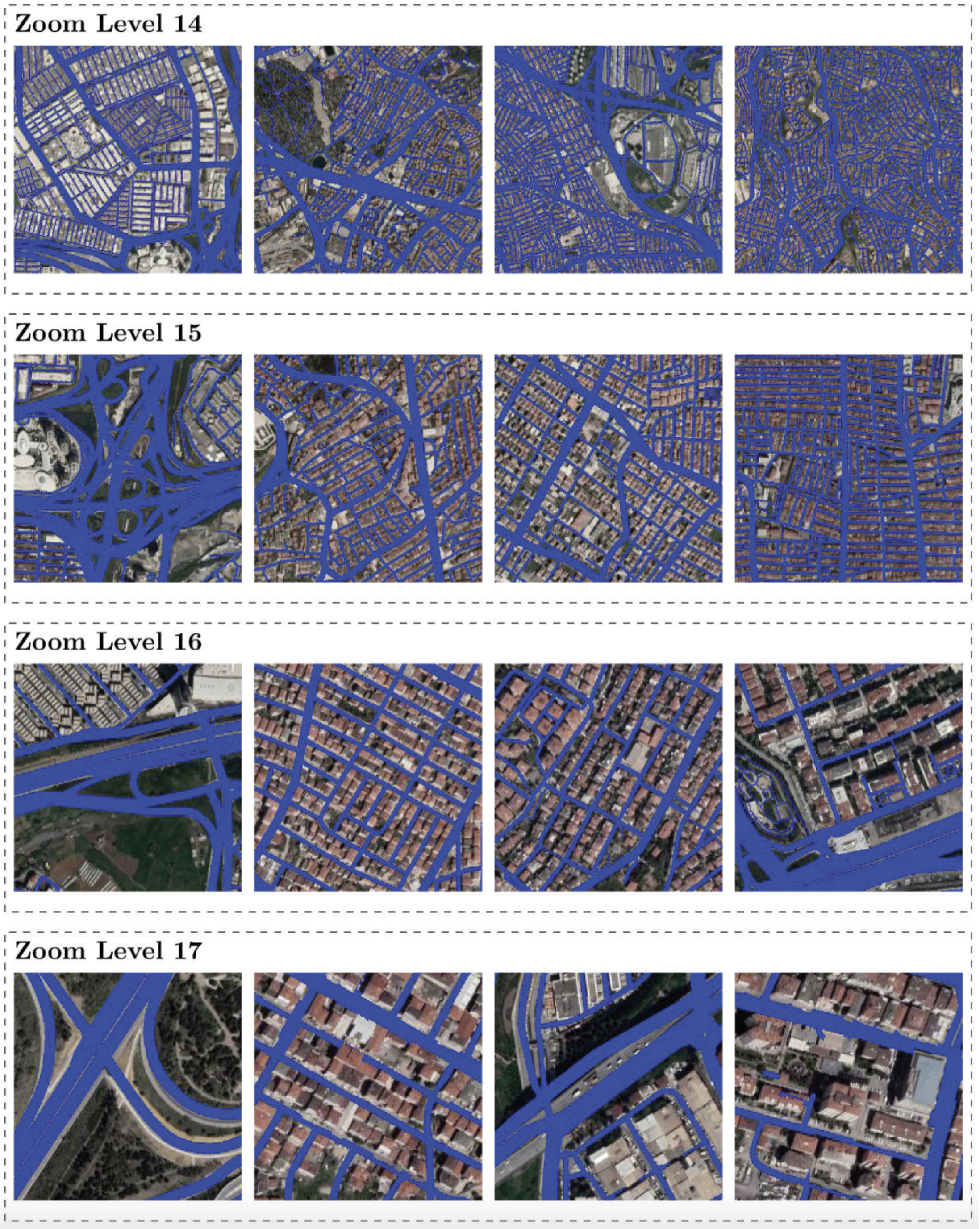


Figure 3.5 : Example images from Istanbul road data set.

$$\mathscr{R}(\boldsymbol{\theta}) = \frac{1}{N}\left(\sum_{i=1}^{N} \mathscr{L}\left(h\left(\mathbf{x}_i;\boldsymbol{\theta}\right),\mathbf{y}_i\right)\right) + \lambda\|\boldsymbol{\theta}\|_0 \tag{3.1}$$

where N is the number of input parameters, $|\theta|$ is the dimensionality of the parameters, weighting factor λ is used for the regularization and $\mathscr{L}\left(h\left(\mathbf{x}_i;\boldsymbol{\theta}\right),\mathbf{y}_i\right)$ represents a loss function for each x_i, y_i input and output pairs (Louizos et al., 2018, Eq.1). In this study, the regularization parameter λ is taken as 0.02.

3.2.3 Deep Residual U-Net approach

U-Net was originally published and successfully implemented for segmentation of biomedical images. Due to its structure, U-Net can produce the output image with the same size as the input image, which makes this architecture well suited for image segmentation studies. The structure of the architecture consists of two parts; one of which is called the contracting path that corresponds to the encoder, and the other is the expansive path that represents the decoder. These two parts yield a U-shape structure that created the so-called symmetric model (Ronneberger et al., 2015). The encoder block of the architecture consists of two consecutive 3×3 convolution layers, a rectified linear unit (ReLU) for the activation function, and a 2×2 max-pooling operation. The decoder block has a 2×2 up-convolution, a concatenation with the corresponding feature map of the encoder, two 3×3 convolution layers and a ReLU activation layer. The consecutive operations in encoder and decoder blocks are repeated four times.

The studies that deepen the model depth by adding more layers to the deep learning model reported a decrease in training accuracy to achieve better accuracy by (Srivastava et al., 2015). To overcome the degradation problem that occurred while adding more layers into the model architecture, He et al. (2015) proposed a deep residual learning framework that fits the stacked layers into residual mapping rather than the desired mapping itself. The detailed structure of the residual block is presented in Figure 3.6.

In this study, a model that consists of 5 encoder blocks, 1 bridge block, and 5 decoder blocks was utilized. The python codes of the U-Net architecture provided by Yakubovskiy (2019) were adopted using the ResNet50 backbone. 10 residual blocks in the encoder path and 2 residual blocks in the bridge block were implemented. The decoder path did not contain any residual block. The feature maps from the encoder path were concatenated with the output of decoder blocks to achieve precise localization. In the final layer of the decoder path, the final binary segmentation output

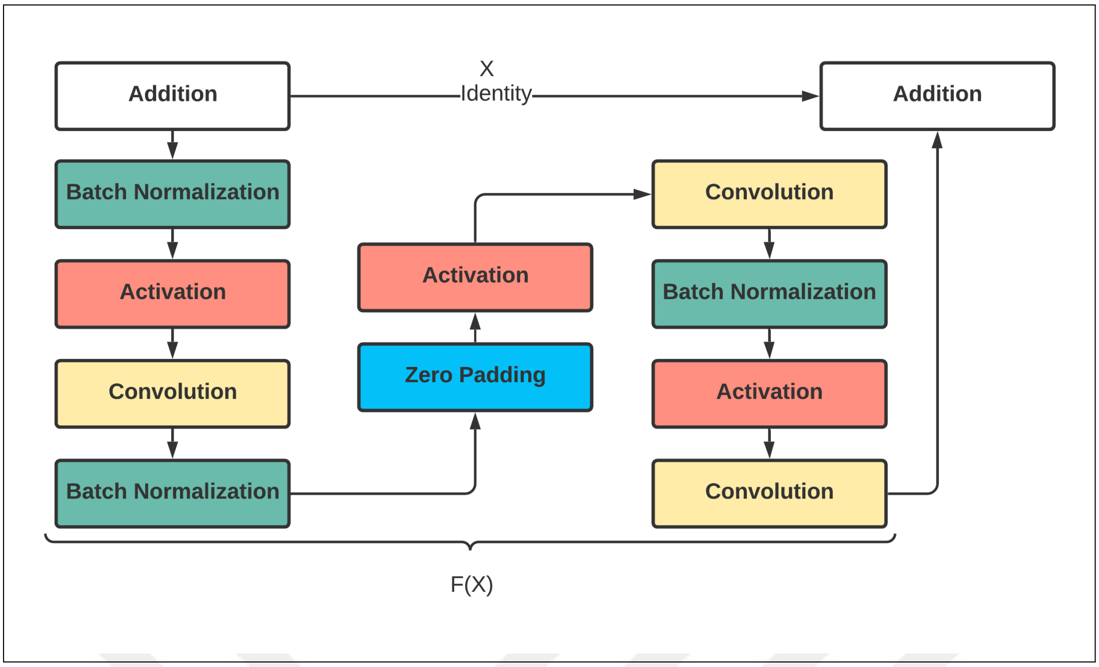


Figure 3.6 : Residual block used in the study.

was generated using a 1×1 convolution and sigmoid activation function. The final model had a total of 32,561,114 trainable parameters. The complete architecture of the utilized model is shown in Figure 3.7.

3.2.4 Training and testing process

In the analyses, training, validation and test files were not created separately, and the data sets were randomly split into 60% for training, 20% for validation and 20% for testing. Data augmentation techniques were not applied due to the sufficient number of images. On a personal computer with an NVIDIA RTX 3060 graphics card with 6 GB RAM and 32 GB CPU RAM, each training process took about 4 hours to complete. In the training, *Adam optimizer* Kingma and Ba (2017) was used with a learning rate of 0.001. *Binary Cross Entropy* was used as the loss function. The models were trained for 100 epochs with a batch size of 4. For the segmentation part, a threshold value of 0.5 was selected.

To evaluate the trained models; precision, recall, F1-Score and Intersection over Union (IoU) were used as evaluation metrics. These metrics cannot be used solely to evaluate the performance of a trained model. Each of them represents the probability of a

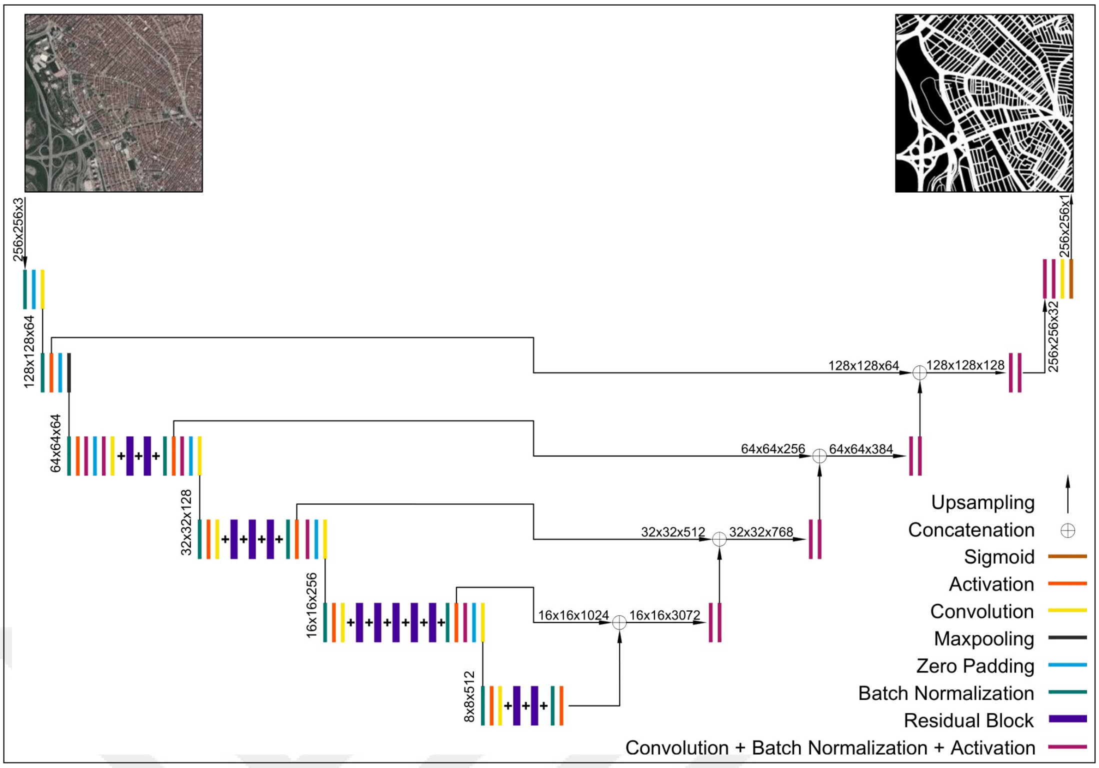


Figure 3.7 : U-Net with ResNet50 backbone architecture used in the study.

particular case. In the case of road segmentation, precision is the ratio of correctly predicted roads to all road predictions. The recall is the ratio of the correctly predicted roads to all actual roads. F1-Score is the harmonic mean of precision, and recall is an important criterion to analyze complex data sets. IoU is the ratio of the overlapped area to the size of the total area of prediction and the ground truth. This metric can be used as an important evaluation criteria in road segmentation studies. In the calculation of these metrics; TP represents the true positive, FP represents false positive and FN represents the false negative. The metrics are defined as:

$$Precision = \frac{TP}{TP+FP} \tag{3.2}$$

$$Recall = \frac{TP}{TP+FN} \tag{3.3}$$

$$F1-Score = \frac{2TP}{2TP+FP+FN} \tag{3.4}$$

$$IoU = \frac{TP}{TP+FP+FN} \tag{3.5}$$

3.3 Results and Discussion

The analyses were separated into three parts. In the first part, Istanbul, DeepGlobe and Massachusetts data sets were trained separately, and the model statistics were calculated with their own testing data sets. To clarify the performance of these models in Istanbul city, these models were evaluated with the Istanbul data set. In the second part, training by taking the starting weights of each training weight from zoom 14 to zoom 17 in the next training procedure was performed to improve the accuracy of the zoom 17 level which is the highest level of resolution selected within the scope of this study. In addition, we investigated the contribution of L_0 regularization to the prediction of model performance trained by using zoom level 17 images. In the third part, we generated a combined model trained with images from all zoom levels simultaneously.

In the first part of the study, the performance of each model trained using Istanbul, DeepGlobe, and Massachusetts data sets were evaluated based on their metrics. The assessment results of models trained using DeepGlobe and Massachusetts data sets were found to be close to each other (Table 4.2). Considering the statistics results of our model performance in prediction of DeepGlobe data set, it was found that Deep Residual U-Net architecture showed better IoU statistics, which is common statistics criteria for all studies, from 3% to 16% than other studies listed in Table 3.1. D-LinkNet with ResNet34 backbone resulted in 10% lower IoU than our model. Later, D-LinkNet is expanded to Attention D-LinkNet with ResNet50 backbone by Wu et al. (2019) and the model showed almost no improvement in terms IoU statistics. Compared to the F1-Score, our model exceed the GCB-Net and Scribble-Based Weakly Supervised Network models reported by Zhu et al. (2021) and Wei and Ji (2022) by 2% and 12%, respectively. Additionally, precision and recall values are 6% and 10% better than Wei and Ji (2022). On the other hand, the performance of our model for Massachusetts data set is lower than the studies listed in Table 3.1. The model showed comparable results against He et al. (2019) and Wang et al. (2021a). The dense and global residual network, proposed by Wu et al. (2021), showed worse performance compared to our model by 8% in recall, 4% in F1-Score, 9% in IoU statistics. However, the models proposed by Singh and Dash (2019), Abdollahi et al. (2020a), and Abderrahim et al. (2020) showed better performance than our model, specifically in F1-Score by 8%, 10% and 7%, respectively. The shortage of number of samples in Massachusetts data set has been overcome by data augmentation methods in these studies, but this is not the case in our study. It must be noted that these studies Singh and Dash (2019) separated the data set into 1000 images for training (93%), 20 images for validation (2%) and 50 images for testing (5%). Even though Abderrahim et al. (2020) and Abdollahi et al. (2020a) used different data size, similar data split ratio exists in these studies, as well. The reason behind this split ratio is to keep the training, validation and testing splitting of original Massachusetts data set given by Mnih (2013). The size of validation and testing images for these studies are too little to fairly compare with our test results which were calculated with 20% of whole data set. The study conducted by Wu et al. (2021) has reported 25% testing split which is acceptable split ratio to compare with our testing results.

The models trained with different zoom levels of the Istanbul data set have put forth successful results with their testing images. While the zoom level increases from 14 to 17, a decrease in the performances was observed (Table 3.2). Nevertheless, the test results of DeepGlobe and Massachusetts data sets for the prediction of Istanbul data set were found to be unacceptable than all zoom levels of the Istanbul data set. The performance of zoom level 17 is closer to DeepGlobe and Massachusetts data sets considering the spatial resolution of zoom level 17 (1.19) which was closer to those of DeepGlobe (0.50 m) and Massachusetts (1.20 m) data sets.

Considering the characteristics of the DeepGlobe data set, it is expected to achieve a better performance using the Massachusetts data set. The area covered by the DeepGlobe data set is part of a developing country that includes dirty roads which produce a complex scene in the images. Furthermore, the mask data set is missing label information for dirt roads; thus, some of these secondary roads can result in misleading predictions. In the pre-processing step of data sets, the number of images generated from the DeepGlobe data set in 256x256 pixel size was significantly higher than the other data sets. To be able to keep the same number of training and testing images from all data sets, most of the images from DeepGlobe were eliminated manually.

Table 3.2 : The prediction statistics of Istanbul, DeepGlobe and Massachusetts models.

Training Data Set	Testing Data Set	Precision	Recall	F1-Score	IoU
Istanbul Zoom 14	Istanbul Zoom 14	0.95	0.95	0.95	0.91
Istanbul Zoom 15	Istanbul Zoom 15	0.96	0.94	0.95	0.91
Istanbul Zoom 16	Istanbul Zoom 16	0.93	0.92	0.92	0.86
Istanbul Zoom 17	Istanbul Zoom 17	0.88	0.86	0.86	0.78
DeepGlobe	DeepGlobe	0.86	0.81	0.82	0.74
	Istanbul Zoom 14	0.75	0.06	0.11	0.06
	Istanbul Zoom 15	0.82	0.13	0.22	0.13
	Istanbul Zoom 16	0.77	0.22	0.33	0.20
	Istanbul Zoom 17	0.71	0.41	0.50	0.34
Massachusetts	Massachusetts	0.84	0.80	0.81	0.72
	Istanbul Zoom 14	0.79	0.02	0.04	0.02
	Istanbul Zoom 15	0.83	0.07	0.13	0.07
	Istanbul Zoom 16	0.81	0.12	0.21	0.12
	Istanbul Zoom 17	0.69	0.09	0.15	0.09

Though this might affect the fair comparison of statistics results between DeepGlobe and Massachusetts data sets, it is excluded from this study to find the superiority of DeepGlobe and Massachusetts data sets over each other. It was rather aimed to show the prediction capability of models trained using these data sets in Istanbul. In Table 3.2, the testing results of DeepGlobe and Massachusetts models against the Istanbul data set were given. The results were found quite poor compared to the original statistics of models trained using the Istanbul data set. This indicates the necessity of producing a data set that reflects the local characteristics of the road network in the study area. The testing results of the DeepGlobe model increases from zoom level 14 to 17, considering the increase in the zoom level up to 17 makes the data set characteristics closer to the images of the DeepGlobe data set. Nevertheless, the best statistics of DeepGlobe found for zoom 17 is still not efficient enough to predict the road network. The model trained using the Massachusetts data set showed significantly poor prediction results. It is worth noting that the prediction of both

DeepGlobe and Massachusetts models revealed comparable precision scores, meaning that the predicted roads are true, but the models cannot predict the whole network. The prediction results of Istanbul models are presented in Figure 3.8. In these maps, white, red, blue and green colors represent TP, FP, TN, FN, respectively.

The statistics of zoom 14 and zoom 15 levels are quite high, as shown in Table 4.2, all roads were accurately predicted compared to the ground truth data. Additionally, the noise level in the prediction images was quite low. The road boundaries were extracted with quite a high level of smoothing, as in reality. Accordingly, crossroads could be distinguished. In addition, since the gaps between the roads were clearly defined, it was possible to distinguish between different roads. However, there were few noises at Zoom 15 compared to Zoom 14. Some of the pedestrian roads were not located in the mask image at zoom 16 level (see Figure 3.8j and Figure 3.8k), the statistics were lowering because there was no ground truth data that the model could compare these roads, causing false TN values in the map illustration given in Figure 3.8l. As in the zoom 14 and zoom 15 levels, the separability of the roads was high and the smoothness of the roads was clear. The number of FP values which was mostly seen at the road boundaries was lower than the other zoom levels. Noises have been detected in zoom 17, although it did not greatly distort the estimate compared to other zoom levels. Problems have been identified due to the insensitivity of smoothness of the road borders observed in the ground-truth data. There were problems in the predictions of the intersections of the roads, especially in the oval curves. However, a reduction in roadside FPs was observed at other zoom levels.

Generally, the main source of error was caused by not overlapping the satellite image and corresponding mask image. The width of the road in the label data did not cover the width of the road on the ground completely, hence false TN pixels in the prediction images of all zoom levels were seen. Apart from this error source, the predictions made for all zoom levels were quite successful. Furthermore, the pedestrian roads that were large enough to be seen in the satellite images could be predicted by the trained models, however; the Google Map platform did not classify these as primary or secondary roads; hence these roads were not available in the mask images (Figure 3.8). This situation caused an increase in the number of FPs.

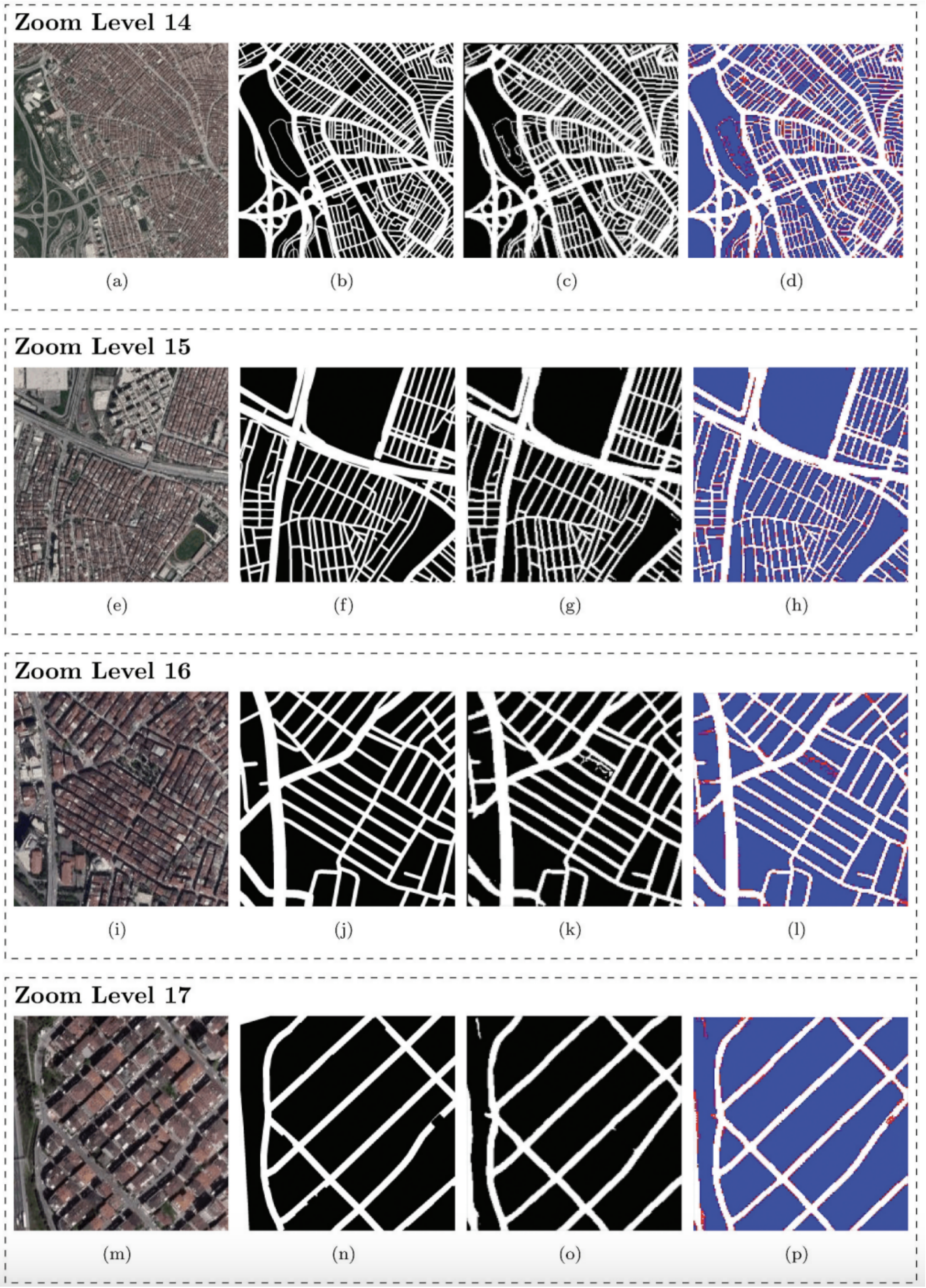


Figure 3.8 : Prediction results of Istanbul data set. The first column is satellite images, the second column is ground truth, the third column is predicted images, and the final column is a map illustrating TP/TN/FP/FN.

Table 3.3 : The prediction statistics of the Istanbul model at zoom level 17.

Model	Precision	Recall	F1-Score	IoU
Zoom level 17	0.88	0.86	0.86	0.78
Consequtive weighted model	0.89	0.88	0.88	0.80
Zoom level 17 with L_0 regularization	0.89	0.88	0.89	0.81

In the second part of the study, two alternative models were generated using the Istanbul data set to improve the prediction results of zoom level 17, which had the lowest statistics results among all zoom levels. The first of these models was generated consecutively from zoom levels 14 to 17 in four training steps. With initial training carried out using zoom level 14 images with no weights, the next training steps were carried out by taking the weights obtained from previous zoom level training from zoom 14 to zoom 17. The second model was generated using images of zoom level 17 with regularization applied to reduce the pixel complexity in the satellite images. The prediction results of these alternative models are given in Table 3.3. The increase in the precision, recall, F1-score and IoU statistics for the first model are 1%, 2%, 2%, and 2% respectively, while those of the second model are 1%, 2%, 3%, and 3% improvement, respectively.

In the final part of the study, a new data set was created that included an equal number of images at all zoom levels. The main purpose here was to generate a single model in which all levels were contributed and tested. In the model formed as a result of the training, the images of each zoom level were tested separately and the metrics of these predictions are presented in Table 3.4. Considering the calculated statistics, the inclusion of all zoom levels in the training part resulted in an apparent degrading in the performance of the prediction of each zoom level separately, causing a decrease in the statistic results which are additionally given in Table 3.4. However, it should be noted that decreases in the ratio of prediction statistics of the model are acceptable and the model can be used in automated road segmentation studies, regardless of the zoom level, according to the purpose of the study.

In general, the models trained with level 14 and level 15 showed very successful testing statistics compared to the reported results of various model architectures together with

Table 3.4 : The prediction statistics of the combined Istanbul model.

Training Data Set	Zoom Level	Precision	Recall	F1-Score	IoU
Combined all zoom level	14	0.91(-4%)	0.91(-%4)	0.90(-%8)	0.83(-8%)
	15	0.91(-5%)	0.89(-5%)	0.90(-5%)	0.81(-10%)
	16	0.86(-7%)	0.82(-10%)	0.84(-8%)	0.73(-13%)
	17	0.82(-6%)	0.79(-7%)	0.79(-7%)	0.68(-10%)

other available data sets such as DeepGlobe and Massachusetts. The precision, recall, F1-score and IoU statistics are all over 90% for these models. Considering that these levels have lower spatial resolution in terms of pixel size compared to DeepGlobe and Massachusetts data sets, the simplicity in the scene complexity of these zoom levels enhances the prediction results of road segmentation. For level 16, though IoU is below 90%, the rest of the statistics are still as high as level 14 and 15. The segmentation capability of model trained with level 17 is of better importance for the comparison of results with other data sets since this the scale of this level is closest to them. The statistics of this level's testing score is equivalent to those reported for Massachusetts data set (see Table 3.1). The statistics of Istanbul road data set is better compared to the statistics of DeepGlobe data set. Though the model architectures of studies conducted by DeepGlobe are not the same as in this study, this cannot be solely attributed to the benefit of model architecture, but it is rather varying data characteristics in the images in terms of differences of road network structures between Istanbul city and urban and rural sites of Thailand, Indonesia and India regions, and the similarities between Istanbul and Massachusetts regions.

As a result of the study, the necessity and feasibility of local data production in deep learning studies have been revealed. With the help of the tool (codes are available) developed within the scope of this study, satellite images at different scale levels and corresponding mask images were produced as desired in a selected study area, and a preliminary data set preparation study that could be used in similar studies was carried out. Considering the segmentation of the road network, which is considered within the scope of the study, up-to-date local data sets can be produced with the help of Google

Map Static API, together with Python codes provided in this study, and it answers a very important requirement to meet the data sets needed in regional solutions. The developed code allows users to not only create new data sets but also provides rectified version of satellite and mask images in WGS84 datum. These data will be able to be merged in spatial databases with different geospatial data sets, if necessary, allowing them to be used in a variety of different fields.

3.4 Conclusions

An accurate local road data set is vital for the segmentation of road networks using deep learning techniques in remote sensing was generated. Firstly, we investigated the use of Google Map Static API at different zoom levels for generating road data set in Istanbul, Turkey. In addition, the generated datasets were tested on models trained with well-known DeepGlobe and Massachusetts road data sets. As the deep learning-based method, we used U-Net with ResNet50 backbone called Deep Residual U-Net, an efficient technique that gained importance with skip connections within a residual unit and between low levels. The first investigation provides insight into the performance of different zoom levels generated data sets while the second test clarifies the necessity of producing a local data set. In conclusion, promising results for generating road data set in Istanbul by using the Google Map Static API were shown. These findings will be interesting to scientists who work on computer vision, object detection and extraction, image segmentation, classification, and image processing studies using machine and deep learning with Earth observation data, especially high-resolution aerial images. The generated data set used in the study was shared with the scientific community who are going to study on a similar subject.

4. IMPROVING ROAD SEGMENTATION BY COMBINING SATELLITE IMAGES AND LiDAR DATA WITH A FEATURE-WISE FUSION STRATEGY[1]

4.1 Introduction

Road segmentation is the process of predicting and classifying road pixels in an image to aid the generation of accurate road network information. Road segmentation using satellite images is a critical tool for efficient traffic management and planning. Proactive measures can be taken to identify potentially dangerous intersections and congestion areas, thereby reducing the risk of traffic-related accidents and improving the overall safety of a road network. Hence, authorities can use road segmentation to monitor changes over time and detect any obstacles or hazardous conditions that could make a route less safe, allowing for constructive action to be taken.

Recently, advances in the field of artificial intelligence have resulted in various machine learning and deep learning approaches being applied to improve the quality of road segmentation from remote sensing data (Song and Civco, 2004; Wegner et al., 2013; Grinias et al., 2016; He et al., 2022). Naturally, a proper solution of road segmentation using remote sensing images is a highly non-linear problem, hence deep learning-based semantic segmentation solutions are more favorable in this matter (Zhang et al., 2018a; Henry et al., 2018; Li et al., 2019b; Zhang et al., 2022). Deep learning model development is seen as an epochal development. However, it is strongly data-driven, which sometimes makes it difficult to make accurate predictions. The performance of deep learning models is greatly influenced by both the quality and quantity of the data utilized in their training process. (Cira et al., 2022; Sariturk and Seker, 2022). If the data is not representative of the real-world scenario, it leads to poor performance. A major challenge is the shadow effect,

[1] This chapter is based on: Ozturk, O., Isik, M. S., Kada, M., Seker, D.Z. (2023). Improving Road Segmentation by Combining Satellite Images and LiDAR Data with a Feature-Wise Fusion Strategy. Applied Sciences, 13(10), 6161. https://doi.org/10.3390/app13106161

which makes it difficult to segment the road pixels (Abdollahi et al., 2020c). The resolution and spectral capability are highly interrelated, and then they result in false perceptions at the boundary points of the road (Hu et al., 2021). The resolution of the satellite image should give high spatial information for extraction. However, accessing high-resolution images can be costly or impossible to perform qualified road segmentation. There are also a lot of problems with woodlands because of the lack of information at roads. The missing information can be completed from the LiDAR point cloud. Therefore, it may be possible to overcome the limitations of satellite images in areas where the road information is problematic and improve the accuracy of road segmentation by exploiting the additional geometric relations by integrating LiDAR data into deep learning models (Sánchez et al., 2020; Abdollahi et al., 2021).

LiDAR is seen as a valuable data source that provides useful information that cannot be extracted by optical images. There are some substantial research on road segmentation using only LiDAR data. In Li et al. (2015), using a grid index structure to detect roads in LiDAR point clouds, a morphological gradient was applied on ground points which were filtered by local intensity distribution. Li et al. (2016) conducted a study based on similar parameters to the method of Li et al. (2015). After identifying the differences in shape, reflectance, and road width, road centrelines were extracted with local principle component analysis. Then, road networks were extracted using the global primitive grouping method. Hui et al. (2016) introduced a novel approach consisting of skewness balancing, rotating neighbourhoods, and hierarchical fusion and optimization to extract roads from point cloud data. Tejenaki et al. (2019) implemented a hierarchical method that refined intensity with mean shift segmentation and extraction of road centerlines with a Voronoi diagram. The proposed method improved both road extraction results and water surface detection. Sánchez et al. (2020) aimed to distinguish between road and ground points based on intensity constraints. In their proposed method, an improved skewness balancing algorithm was used for the calculation of the intensity threshold.

As a deep learning technique that takes the point cloud as an input, PointNet (Qi et al., 2016) was proposed for object classification and part segmentation without rasterization. In contrast to rasterization and state-of-the-art techniques, PointNet can segment each point into classes, resulting in different representations on the point cloud. Following the gradual development of PointNet, PointNet++ was published for labeling the road points in the point cloud data (Ma et al., 2022). There are, however, still issues to be addressed when relying solely on LiDAR data. For instance, there is no spectral information in LiDAR data which can lead to mixing between objects with similar density and texture such as parking lots. Therefore, the combination of satellite images and LiDAR data can close such deficiencies, leading to more accurate segmentation of road network (Li et al., 2016; Zhou et al., 2019; Sun et al., 2022).

There have been numerous studies conducted on the potential benefits of remote sensing data in segmenting images, however, the combination and integration of different data sources are yet to reach the same level of attention. Audebert et al. (2018) conducted thorough research on integrating multiple sources and models and, proposed an approach that combined LiDAR and multispectral images using different fusion strategies. They used an Infrared-Red-Green (IRRG) image and a combination of the Normalized Digital Surface Model (NDSM), Digital Surface Model (DSM), and Normalized Difference Vegetation Index (NDVI) obtained from LiDAR data. In a similar study, Zhang et al. (2018b) suggested using high-resolution images and nDSM derived from LiDAR. Their proposed method involves segmenting fused data, classifying image objects, generating a road network, and extracting the road centerline network using a multistage approach, including morphology thinning, harris corner detection, and least square fitting. Zhou et al. (2021) introduced a novel method named the FuNet (Fusion Network) that integrates satellite images with binary road images generated from GPS data collection. This approach, which depends on a multi-scale feature fusion strategy, has been found to be more effective in resolving the road connectivity issue than using solely satellite imagery. Torun and Yuksel (2021), on the other hand, proposed a technique that involves combining hyperspectral images with LiDAR data for unsupervised segmentation. They applied a Gaussian filter to the point cloud while performing principle component analysis on the images, which yielded results used to create an affinity matrix. In another study, Gao et al. (2021) presented multi-scale feature extraction for 3D road segmentation, combining characteristic features from high-resolution images and LiDAR data.

Although there are findings suggesting the benefits of integrating diverse data sources for multiple applications, existing research indicates that there are still challenges that need to be addressed in this area. The combination of different types of remotely sensed data can be challenging, yet resourceful, for segmentation studies, considering the complexity of the solution. In this study, high-resolution optical satellite images obtained from the Google Map platform were combined with airborne LiDAR data to increase the performance of deep learning-based semantic road segmentation. The combination of the two data sets was carried out feature-wise in a deep residual U-Net-based deep learning model. The features generated by different ResNet backbones using only optical satellite images were fused with the geometric features calculated from the LiDAR point cloud before the final convolution layer of the model. The improvement brought by the LiDAR data was outlined together with the statistical results of the models and the prediction performance of the fused model was evaluated in areas where the road segmentation is challenging.

4.2 Data and Methods

4.2.1 Study area and data collection

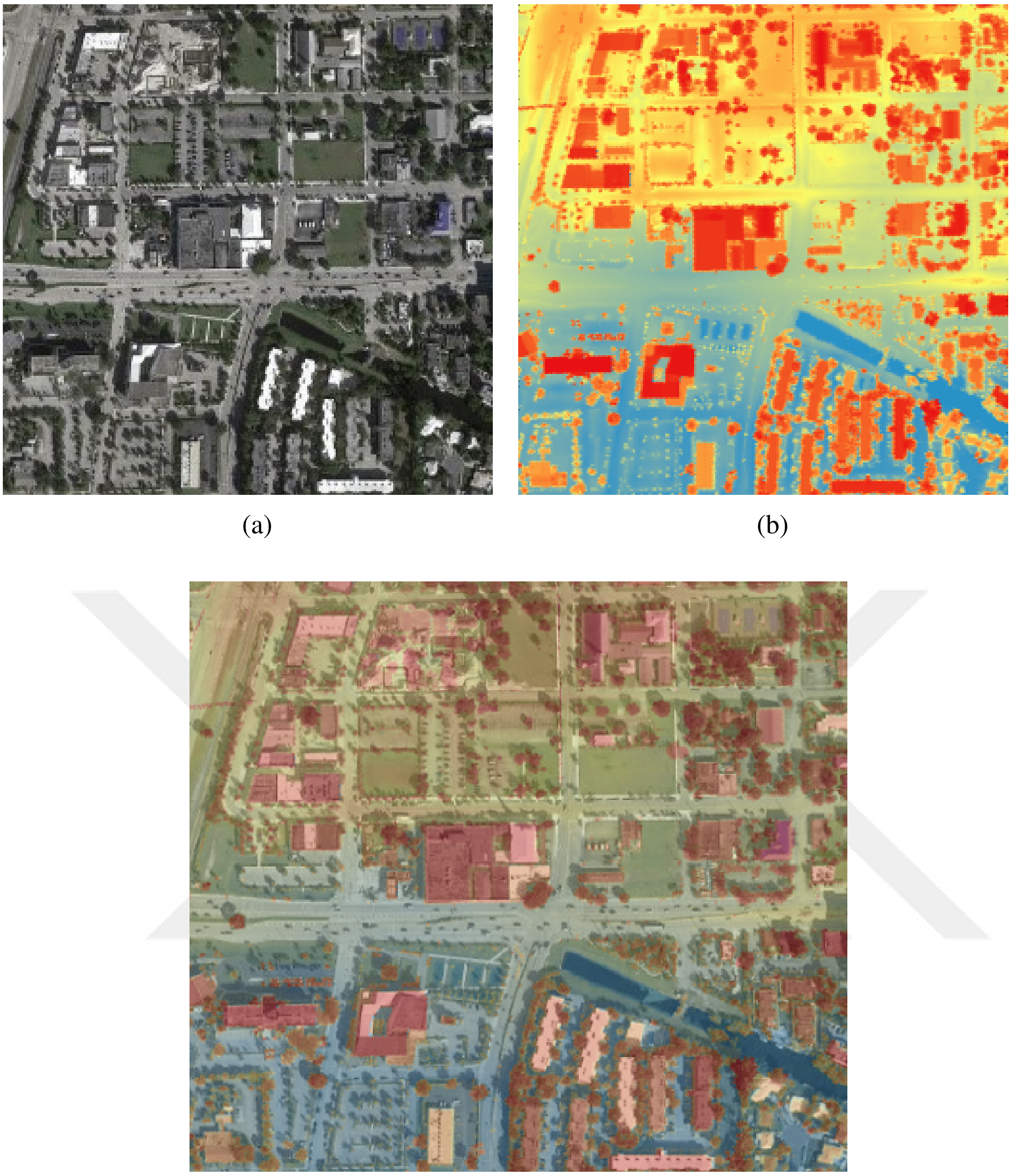
In this study, open-source LiDAR data collected by the U.S. Geological Survey was used together with high-resolution optical satellite imageries of the Google Maps platform. While the Google Map API service is available to gather satellite images everywhere on Earth, airborne LiDAR is a valuable data source, as well as it is expensive. U.S. Geological Survey initiated the National Geospatial Program with airborne LiDAR campaigns that cover the United States to improve and deliver topographic information. This program contains different quality completed products from quality Level-0 to Level-3. In order to test the performance of the feature-wise fusion of optical images and LiDAR, only level-1 quality data that are located over major cities are taken into account. In this context, the Florida southeast project that falls within the Florida counties of Broward, Collier, Hendry, Miami-Dade, Monroe, and Palm Beach was chosen as the study area (U.S. Geological Survey, 2019).

This LiDAR data was collected on June 2018 and published by U.S. Geological Survey with the name of Florida Southeast LiDAR − Block 1. The point cloud was generated at level-1 quality with a source DEM of 0.5 meters/pixel. It has seven classes which are namely ground, low noise, water, bridge decks, high noise, ignored ground, and unclassified. The point density is 14.87 $points/m^2$. The dataset consists of 526 tiles, each of them covering a $1km\ x\ 1km$ area. Concerning data size, only 393 tiles, which cover the majority of the road network in the area, were used in the study.

After the LiDAR data was obtained, the required satellite images were generated using the Google Map Static API-based tool. This tool was created by (Ozturk et al., 2022) and it generates satellite and corresponding mask images randomly or in sequence in the defined region based on latitude and longitude. It also produces a metadata file to be able to rectify these satellite images, if required. In this way, the registration of the LiDAR data and the satellite images could be performed. The Google Maps Static API provides images at various levels which correspond to different scales and resolutions on the Earth's surface. To generate images, zoom level 17 was chosen because of its ideal coverage area and pixel resolution. The satellite and mask images were extracted with a dimension of 512 × 512 pixels. This resulted in an image with a spatial resolution of 1.07 m. × 0.96 m. Consequently, a total of 1426 images were generated which cover the LiDAR data set and contain roads. The overlapping of the LiDAR data and the satellite images are shown in Figure 4.1.

4.2.2 LiDAR feature extraction

LiDAR is an active sensing system that operates by measuring the round-trip timing of a laser beam from an object in order to determine the distance from the sensor to the target. By analyzing the laser time range in combination with the scan angle and spatial coordinates of the laser scanner, the spatial coordinates of an object as (X, Y, Z) are obtained. Along with spatial data, intensity values, number of returns, point classification values, and GPS times are recorded. However, the complete geometric properties of the targetted objects in the point cloud cannot be solely represented by their spatial information (X, Y, and Z), but they can be characterized by their geometric features. In order to avoid the heavy computational load of using complex point cloud



(a)

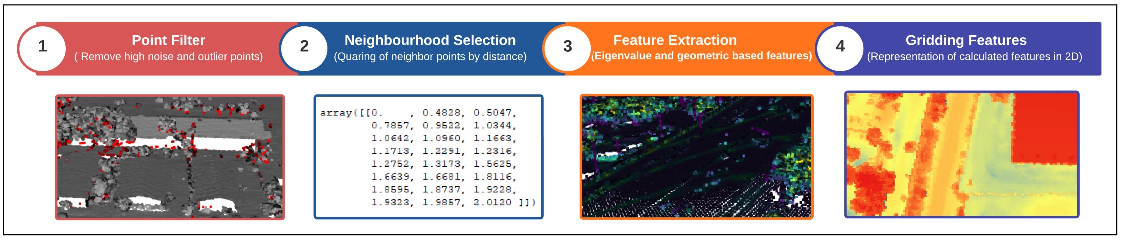
(b)

(c)

Figure 4.1 : Satellite image (a), rendering of point cloud (b) and their overlapping image composite (c).

data, semantic information, extracted by generating geometric features, can be used instead.

As it is presented in Figure 4.2, feature extraction was carried out in four steps. In the first step, points whose noise levels are high are eliminated from the LiDAR point cloud. After noise removal, outlier detection was performed by removing the points whose average distance to their neighbouring points exceeds a given threshold value. In the next step, the statistical relationship of the points was formed by creating a new data structure based on neighbouring points, similar to the second step. But this time, the neighbouring points were determined using a clean point cloud by applying a k-dimensional tree (*k*-d tree) algorithm in 3D. In the final step, eigenvalues and the 3D geometric features were calculated.


1
Point Filter
(Remove high noise and outlier points)
2
Neighbourhood Selection
(Quaring of neighbor points by distance)
3
Feature Extraction
(Eigenvalue and geometric based features)
4
Gridding Features
(Representation of calculated features in 2D)

```
array([[0.    , 0.4828, 0.5047,
        0.7857, 0.9522, 1.0344,
        1.0642, 1.0960, 1.1663,
        1.1713, 1.2291, 1.2316,
        1.2752, 1.3173, 1.5625,
        1.6639, 1.6681, 1.8116,
        1.8595, 1.8737, 1.9228,
        1.9323, 1.9857, 2.0120 ]])
```

Figure 4.2 : Feature extraction pipeline.

There are various methods available to determine the neighborhood of points in LiDAR data. The neighborhood of points can be constructed using a spherical radius or parameterized based on the number of closest neighbours in 2D or 3D space (i.e. k-NN methods). In this study, k-NN based neighbourhood selection was constructed via the well-known *k*-d tree algorithm in which the data is partitioned using a binary tree structure (Bentley, 1975). As a result of the neighbourhood selection, each query point ($P(i)$) and its neighbors were indexed. The distance between the query point and its k neighbours is determined and indexed in order from nearest to farthest.

After determining the neighbourhood, the covariance matrix is calculated and its eigenvalues (w) and eigenvectors (v) are determined. The direction along which the data set has the maximum variation is indicated by the eigenvector with the largest eigenvalue. These will be used for eigenvalues-based feature extraction. First, the eigenvalues, and correspondingly the eigenvectors, are sorted in ascending order as $\lambda_1 > \lambda_2 > \lambda_3$. In order to perform accurate feature extraction, it must be ensured that the vectors are normalized between 0 and 1 and that the eigenvalues are greater than 0 (Weinmann et al., 2015).

The calculated eiganvalue based features are linearity L_λ, planarity P_λ, sphericity S_λ, omnivariance O_λ, anisotropy A_λ, eigenentropy E_λ, sum of eigenvalues $\sum_\lambda$ and change of curvature C_λ. Other features can be calculated by using the height of each query point (absolute height), the height difference between the query point and its neighbouring points, the standard deviation of absolute height within neighbouring points, the local point density in ρ_{3D}, and local 3D neighbourhood (radius of k nearest neighbours). The features are calculated using the definitions provided in Table 4.1.

Table 4.1 : Selected LiDAR-derived features and their formulas (Weinmann et al., 2015).

	Features	Definition
Eigenvalue-Based	Linearity	$L_\lambda = \frac{\lambda_1 - \lambda_2}{\lambda_1}$
	Planarity	$P_\lambda = \frac{\lambda_2 - \lambda_3}{\lambda_1}$
	Scattering	$S_\lambda = \frac{\lambda_3}{\lambda_1}$
	Omnivariance	$O_\lambda = \sqrt[3]{\lambda_1 \lambda_2 \lambda_3}$
	Anisotropy	$A_\lambda = \frac{\lambda_1 - \lambda_3}{\lambda_1}$
	Eigenentropy	$E_\lambda = -\sum_{i=1}^{3} \lambda_i \ln(\lambda_i)$
	Sum of eigenvalues	$\Sigma_\lambda = \lambda_1 + \lambda_2 + \lambda_3$
	Change of curvature	$C_\lambda = \frac{\lambda_3}{\lambda_1 + \lambda_2 + \lambda_3}$
3D - Geometric Proporties-Based	Radius of k nearest neighbors	$r_{\mathrm{kNN,3D}}$
	Local point density	$\rho_{\mathrm{3D}} = \frac{k+1}{\frac{4}{3}\pi r^3_{\mathrm{kNN,3D}}}$
	Absolute height	$H = \lvert Z \rvert$
	Height difference	$\Delta H_{\mathrm{kNN,3D}}$
	Standard deviation of height values	$\sigma H_{\mathrm{kNN,3D}}$

The 3D features calculated for each point $P(i)$ are reduced to the gridded horizontal plane in order to be combined with the features calculated from the satellite images. Each tile covered by LiDAR point clouds was divided into 1m $\times$ 1m cells in (x, y). Due to one or more $P(i)$ points falling within the same cell, the feature value of each cell is represented by the average value of these points. All gridded LiDAR features can be used directly, but the absolute height requires post-processing to be done. The elevations derived from LiDAR data represent geographic elevations as absolute heights, in contrast to other features that are characterized by statistical relationships. To determine the distance between objects and the ground in this study, the digital elevation model was subtracted from the height values of points in the point cloud. Finally, the LiDAR feature extraction was completed by clipping the features based on satellite images.

Figure 4.3 illustrates the rendering of the generated features in blue and red colour for low and high-impact feature values. In general, these features provide insight into the geometry of objects in the point cloud and their relation with the surrounding objects. For instance, anisotropy represents the uniformity of a point cloud, while linearity can be defined as a measure of linear attributes. Furthermore, absolute height provides a distinction between the roads and other objects above (Lai et al., 2019). Together, these features complete the geometric relation between the road and its neighbouring object which is not properly handled in the satellite-only segmentation solutions.

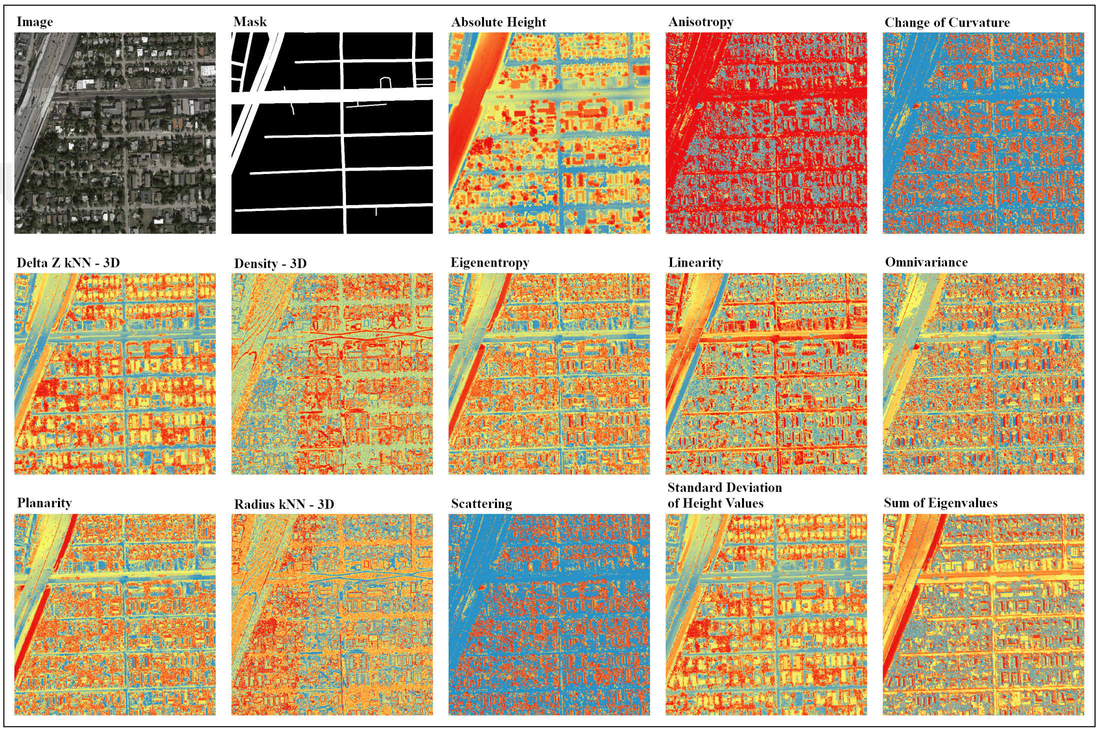


Figure 4.3 : Illustration of gridded feature samples from LiDAR data.

4.2.3 U-Net model structure

The U-Net was initially proposed for segmenting biomedical images (Ronneberger et al., 2015). It is formed as the encoder (contracting), bridge, and decoder (expansive) blocks. As its name implies, it is a U-shaped architecture in which the high-resolution features are extracted by down-sampling the input image, followed by up-sampling them to recover the original spatial resolution. These features are concatenated with the up-sampled output to achieve precise localization. High-dimensional feature spaces, created by up-sampling, allow the architecture to feed the context information into higher-resolution layers. In order to achieve a precise output, high-resolution features from the encoder path are concatenated with the up-sampled output. Consequently, pixel-wise predictions can be made.

As with many deep learning models, U-Net is susceptible to vanishing gradient issues. Although deepening the networks is intended to extract the complex information that cannot be obtained in the background, it also tends to reveal the problem of vanishing gradients. The reason for this is that in the back-propagation stage, the gradient of the loss values updates very little to the previous layers, or even does not update at all. Additionally, as the depth of the network increases, it becomes more difficult to optimize and reaches a kind of accuracy saturation, which leads to higher training error rates (He and Sun, 2014). Accordingly, He et al. (2015) introduced residual learning, called ResNet, that can extract underlying features by increasing the depth. Basically, it is a deep-learning algorithm used to classify images. By integrating shortcut connections into the plain network and copying identity mapping from a shallower model, residual learning will prevent the training error of a deeper model from increasing. The ResNet architecture includes residual connections, which enable the gradients to be carried forward and work out the vanishing gradient problem by going deeper. A ResNet model is identified by the number of layers it contains. There are five different ResNet models published with 18, 34, 50, 101, and 152 layers. There is an increase in model depth from ResNet 18 to ResNet 152. In order to resolve a highly non-linear problem, important features can be extracted by deepening the model architecture. However, deeper models require high computation costs and hyperparameter tuning is more challenging compared to shallow models. By exploiting ResNet's ability to provide identity mapping between layers, these training problems can be overcome by extracting high-resolution complex features.

Zhang et al. (2018a) introduced a novel method for segmenting roads by combining the strengths of U-Net and deep residual learning. This cooperation solves the vanishing gradient problem and performs a powerful segmentation. In the case of ResNet, the network can be trained more easily, while in the case of U-Net, the intricacies of the algorithm will be greatly reduced. As part of this approach, called deep residual U-Net architecture, residual blocks are incorporated into the encoder stages. Thus encoders of ResNet are responsible for extracting difficult-to-obtain features from images, while decoders of U-Net are responsible for generating segmentation masks with precise localization.

In this study, the deep residual U-Net, described as an end-to-end network with a ResNet backbone was used. To implement feature-wise fusion for segmentation, the training was carried out in a sequential model where the satellite-based features and LiDAR-based features were both integrated. This was done by forwarding images through the encoder and decoder segments of the model to extract high-resolution features from satellite images. Before the final convolution block of the model, these features were concatenated with the geometric features derived from the LiDAR point cloud along the channel dimension to form a multi-layer feature map. This combined feature map passed to the final convolution layer and an output layer with *Sigmoid* as an activation function, to predict road pixels (see Figure 4.4).

4.2.4 Model setup

The deep residual U-Net architecture consists of five down-sample blocks, each with decreasing filter sizes of 256, 128, 64, 32, and 16. The convolution blocks consist of a sequence of convolution, batch normalization, activation, and zero padding layers repeated two times for each block. The convolution layers use a 3×3 kernel size and the ReLU activation function. There is a 4×4 transpose convolution with stride 2×2 in the upsampling stage, which is consistent with the size of the filter. In this stage, the dropout layer was added to each convolution block flow.

The model was trained with 80% of the dataset, randomly split from the entire optical images, LiDAR, and corresponding mask images. The training data were further divided into 75% training and 25% validation data. The optuna framework (Akiba et al., 2019) was used to optimize the hyperparameters, such as the optimizer, learning rate, and dropout rate. *Adam* was selected as an optimizer and *Binary cross entropy* was used as a loss function used in the model. The learning rate was tested between 10^{-1} and 10^{-5}. The Dropout rate was tested from 10% to 50%. The training process was stopped after 10 epochs if no improvement was observed in the validation accuracy.

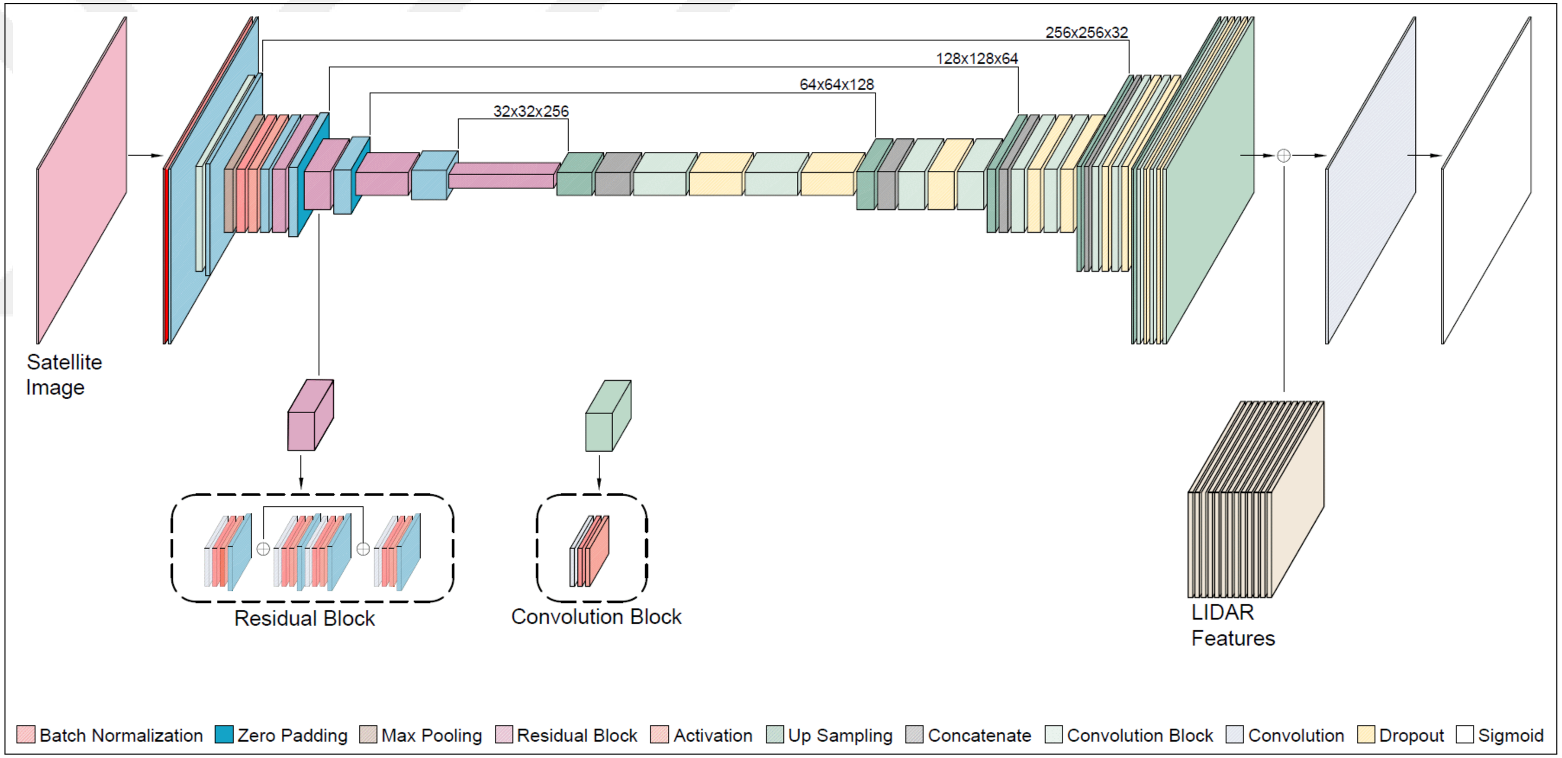


Figure 4.4 : Proposed model structure.

4.3 Results and Discussion

In this section, the numerical results of the road segmentation carried out via feature-wise fusion of optical images and LiDAR are presented together with the visual comparison of enhancements.

4.3.1 Experimental results

The contribution of LiDAR features was clarified by using two training scenarios. In the first scenario, the deep residual U-Net model was trained using satellite images only, while the second scenario includes the feature-wise fusion of optical and LiDAR data sets. In both scenarios, the U-Net architecture is integrated with ResNet-18, ResNet-34, ResNet-50, and ResNet-152 backbones for the feature extraction, leading to eight different model training setups. The results obtained from the analyses are given with the models' prediction metrics in Table 4.2. The model with the best results for each scenario is indicated in bold.

Table 4.2 : The prediction statistics of all models.

Model No	Input Data	Backbone	Metrics			
			Precision	Recall	F1-Score	IoU
1	Image	ResNet18	0.828	0.817	0.820	0.711
2	Image + LiDAR		**0.842**	**0.820**	**0.824**	**0.739**
3	Image	ResNet34	0.825	0.825	0.823	0.717
4	Image + LiDAR		**0.839**	**0.855**	**0.845**	**0.748**
5	Image	ResNet50	0.854	0.82	0.834	0.735
6	Image + LiDAR		**0.865**	**0.829**	**0.845**	**0.750**
7	Image	ResNet152	0.862	0.809	0.830	0.732
8	Image + LiDAR		**0.880**	**0.852**	**0.863**	**0.781**

The results clearly showed that the integration of optical satellite imagery with LiDAR features enhances the performance of road segmentation by 1% to 5% in all models. For each backbone configuration, the scenario with the LiDAR data performed better than satellite image-only training. On the other hand, the ResNet-152 backbone performed better for feature extraction in U-Net architectures, indicating that increasing the depth of the model increased the performance of segmentation as well.

4.3.2 Discussions

The contribution of the fusion strategy can be readily observed by examining individual results and visualized road predictions, as shown in Figure 4.5. It can be seen that the fusion helped to complete the road geometry in areas where there are trees. The model is able to further segment the road pixels in the presence of tree cover, better than using only satellite images in the deep learning model. The precision value for the individual predictions, where the image-only model performs worse, is increased up to 5% by using the fusion strategy. This indicates that adding LiDAR features increased the number of true positives obtained. Using the ResNet-152, the highest F1-Score (93.7%) was achieved and the lowest (77.3%) when applying the ResNet-18. Road boundary continuity was achieved for the woodlands shown in Figure 5 and was able to increase up to 3% of recall values. Moreover, LiDAR-image fusion increased the performance of prediction in areas where there are shadow effects in the satellite images. The recall and the F1-score values are significantly higher in these images. Additionally, the shadow factor can be seen to be too complex when relying only on satellite data. The proposed approach is effective in providing reinforcing information. Another problem related to road integrity at circular intersections has been resolved. This study demonstrates that they were perfectly executed in both terms of quality and quantity of road extraction.

Compared to the recall and IoU values, improved GAN approaches, such as those introduced by Zhang et al. (2019b) and WSGAN Hu et al. (2021), have demonstrated similar recall and IoU results using only satellite images. Studies utilizing the Vaihingen and Potsdam data sets published by the International Society for Photogrammetry and Remote Sensing (ISPRS) are particularly prominent when studying multiple data sources. These data sets consist of Digital Aerial Images and Digital Surface Models, which classify various features such as surfaces and buildings. To benchmark their studies against previous studies, the researchers evaluated the accuracy of estimating impermeable surfaces formed by roads in these data sets. Their

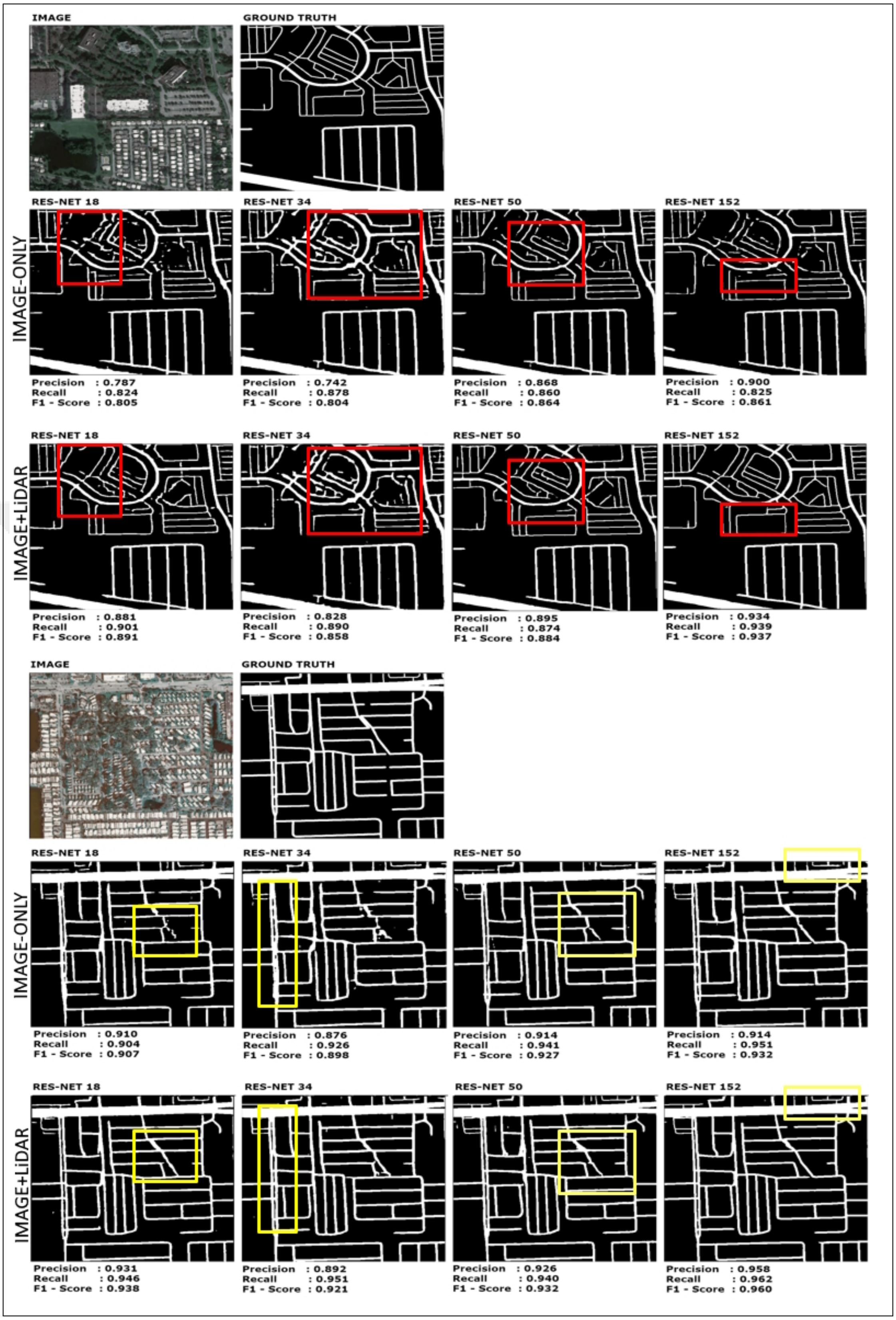


Figure 4.5 : Example of individual prediction samples.

findings indicate that the use of fusion techniques led to an increase in the F1-Score statistics by 1% in the Vaihingen dataset and 2% in the Potsdam dataset when compared to the original U-Net model with ResNet 50 integration, as reported in previous studies by Audebert et al. (2018) and Sun et al. (2022). The study utilized models 5-6, which yielded similar results. Furthermore, it was found that integrating GPS data into satellite images improved relative accuracy by 2% Zhou et al. (2021), and analysis of the tested models supports this observation in this study. In addition, this study revealed that roads were more completely represented in individual image analysis, a significant finding previously reported in other studies, particularly in woodland and shadow areas. As noted by Zhang et al. (2018b), while this study has achieved road connectivity completion even on roads with curves, post-processing techniques can be employed to further enhance the quality of results on such roads.

The effectiveness of the fusion strategy approach can be demonstrated by calculating feature importance. In Figure 4.6, the relative importance of each feature is presented. The optical satellite images were found to be one of the most relevant features, as expected, with 23% importance. The model architecture seems to benefit the most from the change in absolute heights within neighbourhood, i.e. height difference, and the linearity out of 13 LiDAR features. These two features, individually, have importance as much as the optical images, and together they represent 47% of the model's predictions. The rest of the feature importance is represented by the sum of eigenvalues, standard deviation of height values, planarity, entropy, anisotropy, and absolute height features, each of which has importance less than 10%. Scattering, radius of k nearest neighbours, omnivariance, and change of curvature features did not contribute to the model's performance, at all. 1D and 2D geometric features computed with the eigenvalues, such as linearity and planarity, were found to be effective, while 3D geometric feature, scattering (or sphericity) was found insignificant. The change in absolute heights, such as height difference and standard deviation of absolute heights, were found to be dominant predictors compared to absolute heights which indicates the physical properties of road geometry are represented better by the statistical relations of physical height values, rather than the height values.

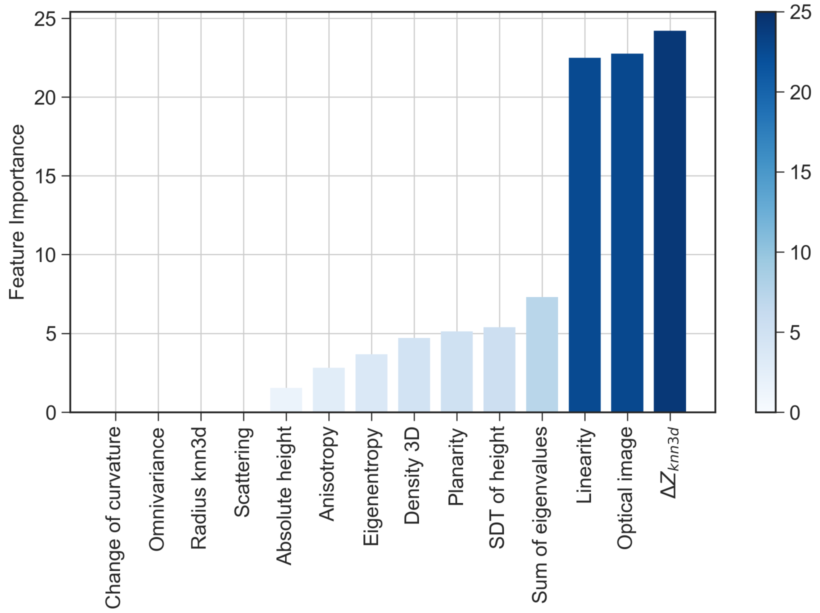


Figure 4.6 : Importance of optical image and LiDAR features.

4.4 Conclusions

In this study, a feature-wise fusion strategy of optical images and point cloud was performed to enhance the road segmentation performance of a deep learning model based on Deep Residual U-Net architecture. In order to compensate for the missing information in optical satellite images that stems from the obstacles and shadow effects in the scenery, we proposed to combine 2D and 3D information to compensate for the absence of depth information in satellite images. For this purpose, high-resolution satellite images and their corresponding road masks, generated over Florida state, were combined with 3D geometric features computed from airborne LiDAR data. In the proposed fusion technique, the optical satellite images were fed into U-Net-based model architecture to generate deep features. Before the final convolution layer, these high-level features were concatenated with the geometric features of the point cloud.

The experimental results validated the effectiveness of the proposed fusion approach for enhancing road segmentation. The combination of 2D and 3D information from satellite images and LiDAR point cloud showed superior results in areas where the satellite image-only models could not predict the road pixels accurately. In challenging areas where the road pixels precluded observation, such as wooded areas and shadows from objects in scenery, the deep learning models trained with the fusion approach predicted the road pixels and the continuity of the road network better than deep learning models trained with satellite images only. The study provided new insight into the relationship between the 2D and 3D features of satellite images and LiDAR data. Moreover, the findings of this study showed the combination of data from various sources can be promising for enhancing the quality of road segmentation. The importance of the features showed that while the optical images have a significant impact on the prediction of road pixels, the contribution of LiDAR features, specifically linearity and height difference within neighbouring points, can lead to a more effective segmentation model development for the extraction of the road network. These 3D features and the geometric relation between the neighbouring points are proven to be significant as the optical images and indicate the importance of contextual information among the point cloud data. It is necessary to further investigate the optimal integration of the most important LiDAR features with the optical satellite images so as to not only capture the geometric properties of the road network efficiently but also to minimize computation costs. It is worth noting that the advancement in the remote sensing data collection techniques and mobility of measuring platforms for LiDAR and laser scanning can ease the generation of accurate and reliable point cloud data. Hence, with these technological developments, the combination of 2D and 3D information from the road networks to increase the performance of deep learning models can be a feasible solution for road extraction studies over challenging areas. The aim of this study is not to exceed the performance of all existing models, but rather to show the combination of optical images and LiDAR can exceed the performance of satellite-only segmentation models. The improved model showed superiority over the problematic areas. In future studies, multi-model fusion strategies will be analyzed to exploit the contribution of LiDAR features that were found to be most effective in road network extraction.

5. CONCLUSIONS

In this thesis, a tool that can automatically generate optical images and corresponding road mask images has been produced to be used in road segmentation studies as given in Chapter 2 in detail. This tool automates the production of optical image patches from optical imageries and corresponding mask images. Google Maps Platform was used as a map service provider in this tool. On the other hand, the Google Maps Platform does not provide geo-information on satellite images, posing challenges in exporting these images to different location-based platforms. To overcome this, the tool includes image rectification capabilities, enabling the export of optical images and road mask images to different environments. Additionally, this facilitates overlaying data from diverse sources, as demonstrated in the fourth chapter of this thesis, overcoming challenges in working with multi-sourced data. Finally, the development of a program capable of producing satellite images designed for the study area, along with corresponding road mask images, has enabled faster production of large amounts and variety of data required for deep learning-based road segmentation studies. Additionally, the development of a geo-referenced data generation infrastructure has made it possible to integrate images with data obtained from various sources. The program allows for random and grid-based image acquisition, enabling dataset customization based on field conditions and research objectives. This tool has made significant contributions to the generation of road datasets and the utilization of satellite images and LiDAR point cloud data in the third and fourth chapters of this thesis.

In the third chapter of this thesis, the segmentation of the road network was conducted for Istanbul city using varying zoom levels of satellite images and their corresponding mask images by exploiting the Google Map Static API. The motivation for using Google Map Static API specifically in Istanbul city was to generate a data set that reflects the road characteristics of the area of interest, considering the dependency of deep learning models on the images they trained with. The produced data set was used for training deep learning models with U-Net architecture with the ResNet50 backbone. The prediction results were compared with DeepGlobe and Massachusetts road data sets which are commonly used publicly available data sets for released for the road segmentation studies. The performances of the deep learning models trained with the generated datasets provided successful results for road segmentation studies and met the requirements of the data representing the characteristics of the study area. The main outcome of this chapter demonstrated the effectiveness of generating accurate and reliable road datasets.

The results showed that the models trained using the Istanbul data set were successful in the estimation of the road network at all zoom levels of images. The best performance was achieved at zoom levels 14 and 15 with all metric scores higher than 90%, though the prediction results of zoom levels 16 and 17 were also acceptable. The failure of DeepGlobe and Massachusetts models for the estimation of road networks in Istanbul justifies the necessity of creating such data sets for deep learning studies. The data set produced from Google Map Static API has limitations that affect the prediction performance of models. The label data, which is originally a vector layer that represents the road network, includes inconsistencies with the satellite image. The width of the road in label data may not cover the real width of the road in the ground truth image. This could indicate possible vector errors in Google Map data either in digitization or data collection. Another possible explanation could be that the dynamic structure of the road network is not up-to-date in the database of the Google Map Platform, which is understandable considering the speed of change in the structure and design of roads in Istanbul city. In different studies, the required different zoom levels should be considered to get higher accuracy from the imageries and bring the studies to a higher technical level.

Promising results for generating road data set in Istanbul by using the Google Map Static API are proven. This generation can be extended to any region if required. The deep learning model that adopted multi-scale data showed promising prediction performance that can be used in the segmentation of road networks from medium-to-high-resolution satellite images. The performance of models that used the road datasets at varying zoom levels provided valuable insights into the suitability of different scales for geospatial applications. This thesis focuses on the significance of road segmentation and its application in studies utilizing images captured at various zoom levels. Instead of limiting road segmentation studies to specific zoom levels, such as the commonly used zoom level 17, this research recognizes the importance of considering images obtained at different zoom levels. For instance, zoom level 14 can provide valuable data for disaster management and logistics, while zoom level 17 is essential for navigation and traffic management studies.

In the fourth chapter of this thesis, a feature-based fusion strategy was proposed to perform more qualified road segmentation. Experimental results validated the effectiveness of the proposed approach against conventional methods, which only use satellite images. It also clarified how more challenging issues, such as the problems in a wooded area and the presence of shadow in optical images, can be overcome for road segmentation. The experiment provides new insight into the relationship between satellite images and LiDAR data. Moreover, the findings of this study showed the combination of data from various sources can be promising to enhance the quality of road segmentation. The contribution of LiDAR features can lead to a more effective segmentation model development for the extraction of road networks. Obtained results demonstrate the potential of the fusion strategy in developing more reliable road segmentation models that can accurately detect road pixels even in challenging areas. The flexibility of the fusion strategy makes it a valuable tool for advancing the field of deep learning and developing more accurate and effective road segmentation models. Finally, the potential of the feature-wise fusion strategy of 2D and 3D information for segmenting roads more accurately and reliably which can be used in various real-world scenarios was highlighted.

In future studies, the generation of geospatial data set from the Google Maps Platform can be extended to different land-use applications by proper modifications in the features of mask images. Moreover, it is necessary to investigate further the optimal integration of the most important LiDAR features with the optical satellite images so as to not only capture the geometric properties of the road network efficiently but also to minimize computation costs. For this purpose, multi-model fusion strategies will be analyzed to exploit the contribution of LiDAR features that were found to be most effective in road network extraction. Multi-modal training strategy offers applicable options for combining heterogeneous data. It is relatively under-explored, but one promising method involves feeding the same data into two parallel deep-learning models, extracting features from each, and then combining the features to make a single prediction. This approach enables end-to-end feature extraction and fusion by reducing computational costs.

The advancement in the remote sensing data collection techniques and mobility of measuring platforms for LiDAR and laser scanning can ease the generation of accurate and reliable point cloud data. Additionally, point cloud data which is produced by photogrammetry can easily be used in similar studies. Hence, with these technological developments, the combination of 2D and 3D information from the road networks to increase the performance of deep learning models can be a feasible solution for road extraction studies over challenging areas.

REFERENCES

Abderrahim, N. Y. Q., Abderrahim, S., and Rida, A. (2020). Road segmentation using u-net architecture. In *2020 IEEE International conference of Moroccan Geomatics (Morgeo)*, pages 1–4.

Abdollahi, A., Bakhtiari, H. R. R., and Nejad, M. P. (2018). Investigation of SVM and Level Set Interactive Methods for Road Extraction from Google Earth Images. *Journal of the Indian Society of Remote Sensing*, *46*(3):423–430.

Abdollahi, A., Pradhan, B., and Alamri, A. (2020a). VNet: An End-to-End Fully Convolutional Neural Network for Road Extraction From High-Resolution Remote Sensing Data. *IEEE Access*, *8*:179424–179436.

Abdollahi, A., Pradhan, B., Sharma, G., Maulud, K. N. A., and Alamri, A. (2021). Improving road semantic segmentation using generative adversarial network. *IEEE Access*, *9*:64381–64392.

Abdollahi, A., Pradhan, B., Shukla, N., Chakraborty, S., and Alamri, A. (2020b). Deep Learning Approaches Applied to Remote Sensing Datasets for Road Extraction: A State-Of-The-Art Review. *Remote Sensing*, *12*(9):1444.

Abdollahi, A., Pradhan, B., Shukla, N., Chakraborty, S., and Alamri, A. (2020c). Deep learning approaches applied to remote sensing datasets for road extraction: A state-of-the-art review. *Remote Sensing*, *12*.

Akiba, T., Sano, S., Yanase, T., Ohta, T., and Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. *arXiv, preprint*(1907.10902).

Amini, J. (2009). Road Extraction from Satellite Images using a Fuzzy-Snake Model. *The Cartographic Journal*, *46*(2):164–172.

Audebert, N., Le Saux, B., and Lefèvre, S. (2018). Beyond RGB: Very high resolution urban remote sensing with multimodal deep networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, *140*:20–32.

Bacher, U. and Mayer, H. (2004). Automatic road extraction from IRS satellite images in agricultural and desert areas. In *Proceedings of The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pages 1055–1060.

Bacher, U. and Mayer, H. (2005). Automatic road extraction from multispectral high resolution satellite images. In *Proceedings of CMRT05*, volume 36, page 3.

Bentley, J. L. (1975). Multidimensional binary search trees used for associative searching. *Communications of the ACM*, *18*(9):509–517.

Boonpook, W., Tan, Y., and Xu, B. (2021). Deep learning-based multi-feature semantic segmentation in building extraction from images of UAV photogrammetry. *International Journal of Remote Sensing*, *42*(1):1–19.

Buslaev, A., Seferbekov, S., Iglovikov, V., and Shvets, A. (2018). Fully convolutional network for automatic road extraction from satellite imagery. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 197–1973.

Christophe, E. and Inglada, J. (2007). Robust Road Extraction for High Resolution Satellite Images. In *2007 IEEE International Conference on Image Processing*, volume 5, pages 437– 440. IEEE.

Cira, C.-I., Alcarria, R., Manso-Callejo, M.-Á., and Serradilla, F. (2020). A Deep Learning-Based Solution for Large-Scale Extraction of the Secondary Road Network from High-Resolution Aerial Orthoimagery. *Applied Sciences*, *10*(20):7272.

Cira, C. I., Kada, M., Ángel Manso-Callejo, M., Alcarria, R., and Sanchez, B. B. (2022). Improving road surface area extraction via semantic segmentation with conditional generative learning for deep inpainting operations. *ISPRS International Journal of Geo-Information*, *11*.

Dai, J., Zhu, T., Wang, Y., Ma, R., and Fang, X. (2020). Road extraction from high-resolution satellite images based on multiple descriptors. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *13*:227–240.

Demir, I., Koperski, K., Lindenbaum, D., Pang, G., Huang, J., Basu, S., Hughes, F., Tuia, D., and Raska, R. (2018). DeepGlobe 2018: A challenge to parse the earth through satellite images. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, volume 2018-June, pages 172–181.

Fritsch, J., Kuhnl, T., and Geiger, A. (2013). A new performance measure and evaluation benchmark for road detection algorithms. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 1693–1700. IEEE, IEEE.

Ganguli, S., Garzon, P., and Glaser, N. (2019). Geogan: A conditional GAN with reconstruction and style loss to generate standard layer of maps from satellite images. *CoRR*, *abs/1902.05611*.

Gao, L., Shi, W., Zhu, J., Shao, P., Sun, S., Li, Y., Wang, F., and Gao, F. (2021). Novel framework for 3D road extraction based on airborne LiDAR and high-resolution remote sensing imagery. *Remote Sensing*, *13*(23).

Ghandorh, H., Boulila, W., Masood, S., Koubaa, A., Ahmed, F., and Ahmad, J. (2022). Semantic segmentation and edge detection&mdash;approach to road detection in very high resolution satellite images. *Remote Sensing*, *14*(3).

Grinias, I., Panagiotakis, C., and Tziritas, G. (2016). Mrf-based segmentation and unsupervised classification for building and road detection in peri-urban areas of high-resolution satellite images. *ISPRS Journal of Photogrammetry and Remote Sensing*, *122*:145–166.

Gruen, A. and Li, H. (1995). Road extraction from aerial and satellite images by dynamic programming. *ISPRS Journal of Photogrammetry and Remote Sensing*, *50*(4):11–20.

He, H., Yang, D., Wang, S., Wang, S., and Li, Y. (2019). Road extraction by using atrous spatial pyramid pooling integrated encoder-decoder network and structural similarity loss. *Remote Sensing*, *11*(9).

He, K. and Sun, J. (2014). Convolutional neural networks at constrained time cost. *arXiv*, *preprint*(1412.1710).

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition. *arXiv*, *preprint*(abs/1512.03385).

He, L., Peng, B., Tang, D., and Li, Y. (2022). Road extraction based on improved convolutional neural networks with satellite images. *Applied Sciences*, *12*(21).

Henry, C., Azimi, S. M., and Merkle, N. (2018). Road segmentation in sar satellite images with deep fully convolutional neural networks. *IEEE Geoscience and Remote Sensing Letters*, *15*(12):1867–1871.

Hu, A., Chen, S., Wu, L., Xie, Z., Qiu, Q., and Xu, Y. (2021). Wsgan: An improved generative adversarial network for remote sensing image road network extraction by weakly supervised processing. *Remote Sensing*, *13*:2506.

Hui, Z., Hu, Y., Jin, S., and Yevenyo, Y. Z. (2016). Road centerline extraction from airborne lidar point cloud based on hierarchical fusion and optimization. *ISPRS Journal of Photogrammetry and Remote Sensing*, *118*:22–36.

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134.

Kingma, D. P. and Ba, J. (2017). Adam: A method for stochastic optimization. *arXiv, preprint*(1412.6980).

Lai, X., Yang, J., Li, Y., and Wang, M. (2019). A building extraction approach based on the fusion of lidar point cloud and elevation map texture features. *Remote Sensing*, *11*(14).

LeCun, Y., Kavukcuoglu, K., and Farabet, C. (2010). Convolutional networks and applications in vision. In *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, pages 253–256.

Li, W., He, C., Fang, J., Zheng, J., Fu, H., and Yu, L. (2019a). Semantic Segmentation-Based Building Footprint Extraction Using Very High-Resolution Satellite Images and Multi-Source GIS Data. *Remote Sensing*, *11*(4):403.

Li, Y., Hu, X., Guan, H., and Liu, P. (2016). An efficient method for automatic road extraction based on multiple features from lidar data. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, *41*:289–293.

Li, Y., Peng, B., He, L., Fan, K., and Tong, L. (2019b). Road segmentation of unmanned aerial vehicle remote sensing images using adversarial network with multiscale context aggregation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *12*:2279–2287.

Li, Y., Yong, B., Wu, H., An, R., and Xu, H. (2015). Road detection from airborne lidar point clouds adaptive for variability of intensity data. *Optik*, *126*:4292–4298.

Liu, J., Wang, S., Hou, X., and Song, W. (2020). A deep residual learning serial segmentation network for extracting buildings from remote sensing imagery. *International Journal of Remote Sensing*, *41*(14):5573–5587.

Liu, Y., Yao, J., Lu, X., Xia, M., Wang, X., and Liu, Y. (2019). Roadnet: Learning to comprehensively analyze road networks in complex urban scenes from high-resolution remotely sensed images. *IEEE Transactions on Geoscience and Remote Sensing*, *57*(4):2043–2056.

Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440.

Louizos, C., Welling, M., and Kingma, D. P. (2018). Learning sparse neural networks through l_0 regularization. *arXiv*, *preprint*(1712.01312).

Ma, H., Ma, H., Zhang, L., Liu, K., and Luo, W. (2022). Extracting urban road footprints from airborne lidar point clouds with pointnet++ and two-step post-processing. *Remote Sensing*, *14*.

Maboudi, M., Amini, J., Hahn, M., and Saati, M. (2017). Object-based road extraction from satellite images using ant colony optimization. *International Journal of Remote Sensing*, *38*(1):179–198.

Miao, Z., Shi, W., Gamba, P., and Li, Z. (2015). An object-based method for road network extraction in vhr satellite images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *8*(10):4853–4862.

Mnih, V. (2013). *Machine learning for aerial image labeling*. PhD Thesis, University of Toronto (Canada).

Mokhtarzade, M. and Zoej, M. V. (2007). Road detection from high-resolution satellite images using artificial neural networks. *International Journal of Applied Earth Observation and Geoinformation*, *9*(1):32–40.

Movaghati, S., Moghaddamjoo, A., and Tavakoli, A. (2010). Road extraction from satellite images using particle filtering and extended kalman filtering. *IEEE Transactions on Geoscience and Remote Sensing*, *48*:2807–2817.

Ozturk, O., Isik, M. S., Sariturk, B., and Seker, D. Z. (2022). Generation of istanbul road data set using google map api for deep learning-based segmentation. *International Journal of Remote Sensing*, *43*:2793–2812.

Qi, C. R., Su, H., Mo, K., and Guibas, L. J. (2016). Pointnet: Deep learning on point sets for 3d classification and segmentation. *CoRR*, *abs/1612.00593*.

Rajamani, T., Sevugan, P., and Ragupathi, S. (2022). Automatic building footprint extraction and road detection from hyperspectral imagery. *Journal of Electronic Imaging*, *32*(1):011005.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. *CoRR*, *abs/1505.0*:234–241.

Sariturk, B. and Seker, D. Z. (2022). A residual-inception u-net (riu-net) approach and comparisons with u-shaped cnn and transformer models for building segmentation from high-resolution satellite images. *Sensors*, *22*(19).

Shi, Q., Liu, X., and Li, X. (2018). Road Detection From Remote Sensing Images by Generative Adversarial Networks. *IEEE Access*, *6*:25486–25494.

Singh, P. and Dash, R. (2019). A Two-Step Deep Convolution Neural Network for Road Extraction from Aerial Images. In *2019 6th International Conference on Signal Processing and Integrated Networks (SPIN)*, pages 660–664. IEEE, IEEE.

Song, M. and Civco, D. (2004). Road extraction using svm and image segmentation. *Photogrammetric Engineering & Remote Sensing*, *70*:1365–1371.

Spiegl, B. (2021). Contrastive unpaired translation using focal loss for patch classification. *arXiv*, *preprint*(arXiv:2109.12431).

Srinivasan, V., Müller, K.-R., Samek, W., and Nakajima, S. (2020). Langevin cooling for domain translation. *arXiv*, *preprint*(2008.13723).

Srivastava, R. K., Greff, K., and Schmidhuber, J. (2015). Training Very Deep Networks. *arXiv*, *abs/1505.0*(1507.06228).

Sun, Y., Fu, Z., Sun, C., Hu, Y., and Zhang, S. (2022). Deep multimodal fusion network for semantic segmentation using remote sensing image and lidar data. *IEEE Transactions on Geoscience and Remote Sensing*, *60*:1–18.

Sánchez, J. M., Rivera, F. F., Domínguez, J. C. C., Vilariño, D. L., and Pena, T. F. (2020). Automatic extraction of road points from airborne lidar based on bidirectional skewness balancing. *Remote Sensing*, *12*.

Tejenaki, S. A. K., Ebadi, H., and Mohammadzadeh, A. (2019). A new hierarchical method for automatic road centerline extraction in urban areas using lidar data. *Advances in Space Research*, *64*:1792–1806.

Torun, O. and Yuksel, S. E. (2021). Unsupervised segmentation of LiDAR fused hyperspectral imagery using pointwise mutual information. *International Journal of Remote Sensing*, *42*(17):6465–6480.

U.S. Geological Survey (2019). 3D elevation program lidar point cloud. Available online: https://apps.nationalmap.gov/lidar-explorer/, Accessed on 07 April 2023).

Varia, N., Dokania, A., and Senthilnath, J. (2018). DeepExt: A Convolution Neural Network for Road Extraction using RGB images captured by UAV. In *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1890–1895. IEEE.

Wang, J., Song, J., Chen, M., and Yang, Z. (2015). Road network extraction: a neural-dynamic framework based on deep learning and a finite state machine. *International Journal of Remote Sensing*, *36*(12):3144–3169.

Wang, S., Mu, X., Yang, D., He, H., and Zhao, P. (2021a). Road Extraction from Remote Sensing Images Using the Inner Convolution Integrated Encoder-Decoder Network and Directional Conditional Random Fields. *Remote Sensing*, *13*(3):465.

Wang, X., Ye, H., Liao, X., and Yue, H. (2021b). An enhanced global context-aware network for road extraction from high-resolution imagery. *Remote Sensing Letters*, *12*(9):859–868.

Wegner, J. D., Montoya-Zegarra, J. A., and Schindler, K. (2013). A higher-order crf model for road network extraction. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1698–1705.

Wei, X., Lv, X., and Zhang, K. (2021). Road extraction in sar images using ordinal regression and road-topology loss. *Remote Sensing*, *13*(11).

Wei, Y. and Ji, S. (2022). Scribble-Based Weakly Supervised Deep Learning for Road Surface Extraction From Remote Sensing Images. *IEEE Transactions on Geoscience and Remote Sensing*, *60*:1–12.

Weinmann, M., Jutzi, B., Hinz, S., and Mallet, C. (2015). Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers. *ISPRS Journal of Photogrammetry and Remote Sensing*, *105*:286–304.

Wu, M., Zhang, C., Liu, J., Zhou, L., and Li, X. (2019). Towards Accurate High Resolution Satellite Image Semantic Segmentation. *IEEE Access*, *7*:55609–55619.

Wu, Q., Luo, F., Wu, P., Wang, B., Yang, H., and Wu, Y. (2021). Automatic Road Extraction from High-Resolution Remote Sensing Images Using a Method Based on Densely Connected Spatial Feature-Enhanced Pyramid. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *14*:3–17.

Xia, M., Cui, Y., Zhang, Y., Xu, Y., Liu, J., and Xu, Y. (2021). DAU-Net: a novel water areas segmentation structure for remote sensing image. *International Journal of Remote Sensing*, *42*(7):2594–2621.

Xu, C. and Zhao, B. (2018). Satellite image spoofing: Creating remote sensing dataset with generative adversarial networks (short paper). In *10th International conference on geographic information science (GIScience 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Yakubovskiy, P. (2019). Segmentation Models. Available from GitHub repository: https://github.com/qubvel/segmentation_models, Accessed on 15 January 2023.

Yang, C. and Wang, Z. (2020). An Ensemble Wasserstein Generative Adversarial Network Method for Road Extraction From High Resolution Remote Sensing Images in Rural Areas. *IEEE Access*, *8*:174317–174324.

Zhang, W. and Hu, B. (2021). Forest roads extraction through a convolution neural network aided method. *International Journal of Remote Sensing*, *42*(7):2706–2721.

Zhang, X., Han, X., Li, C., Tang, X., Zhou, H., and Jiao, L. (2019). Aerial image road extraction based on an improved generative adversarial network. *Remote Sensing*, *11*:930.

Zhang, Z., Liu, Q., and Wang, Y. (2018a). Road extraction by deep residual u-net. *IEEE Geoscience and Remote Sensing Letters*, *15*:749–753.

Zhang, Z., Miao, C., Liu, C., and Tian, Q. (2022). Dcs-transupernet: Road segmentation network based on cswin transformer with dual resolution. *Applied Sciences*, *12*(7).

Zhang, Z., Zhang, X., Sun, Y., and Zhang, P. (2018b). Road centerline extraction from very-high-resolution aerial image and LiDAR data based on road connectivity. *Remote Sensing*, *10*(8).

Zhao, T., Xu, J., Chen, R., and Ma, X. (2021). Remote sensing image segmentation based on the fuzzy deep convolutional neural network. *International Journal of Remote Sensing*, *42*(16):6264–6283.

Zhou, K., Ming, D., Lv, X., Fang, J., and Wang, M. (2019). Cnn-based land cover classification combining stratified segmentation and fusion of point cloud and very high-spatial resolution remote sensing image data. *Remote Sensing*, *11*.

Zhou, K., Xie, Y., Gao, Z., Miao, F., and Zhang, L. (2021). Funet: A novel road extraction network with fusion of location data and remote sensing imagery. *ISPRS International Journal of Geo-Information*, *10*(1).

Zhou, L., Zhang, C., and Wu, M. (2018). D-linknet: Linknet with pretrained encoder and dilated convolution for high resolution satellite imagery road extraction. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 192–1924.

Zhu, Q., Zhang, Y., Wang, L., Zhong, Y., Guan, Q., Lu, X., Zhang, L., and Li, D. (2021). A Global Context-aware and Batch-independent Network for road extraction from VHR satellite imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, *175*:353–365.

APPENDICES

APPENDIX A : Google Maps API tool

APPENDIX A : Google Maps API tool

```python
import numpy as np
from pathlib import Path
from PIL import Image
from tqdm import tqdm
from urllib.request import urlopen
import folium
import base64
from IPython.display import HTML, display, clear_output
# Rasterio is not a default library, may require installation
try:
    import rasterio
except ModuleNotFoundError:
    !pip install rasterio
    import rasterio

# Create project folders

# Project working directory and sub-directories
project_name = r"googlemap"

# Zoom levels and number of images to be downloaded
metadata = [
    (14, 10, "#1A5276"),  # Zoom level 14: 10 images
    (15, 10, "#78281F"),  # Zoom level 15: 10 images
    (16, 10, "#0E6655"),  # Zoom level 16: 10 images
    (17, 10, "#F1C40F"),  # Zoom level 17: 10 images
]
zoom_levels = [level[0] for level in metadata]

pwd = Path(project_name)
satellite_dir = pwd.joinpath("sat")
mask_dir = pwd.joinpath("mask")

# Google Map Static API KEY (specific to each user)
API_key = "INSERT YOUR API KEY HERE"

# Boundaries of the area of interest (in degrees)
minLat, maxLat, minLon, maxLon = 41.0, 41.2, 28.75, 29.5

# Pixel size of satellite and mask images
# Note: Google Map does not allow an image size greater than 640X640
# though it does not give an error about it.
image_size = 512  # 512x512 image size

# Rectifying satellite images
# True  : produces GeoTiff file of Google Map images
# False : will not produce GeoTiff files, though log file
# contains coordinate information anyway
rectify = True

```

```
# ------------------------------------------------------------------------------
# Create project working directory

if pwd.is_dir() == True:
    print("Project working directory exists...")
    if satellite_dir.is_dir() == False:
        satellite_dir.mkdir()
    if mask_dir.is_dir() == False:
        mask_dir.mkdir()
    for zoom in zoom_levels:
        satellite_zoom_dir = satellite_dir.joinpath(str(zoom))
        mask_zoom_dir = mask_dir.joinpath(str(zoom))
        if satellite_zoom_dir.is_dir() == False:
            satellite_zoom_dir.mkdir()
        if mask_zoom_dir.is_dir() == False:
            mask_zoom_dir.mkdir()
else:
    pwd.mkdir()
    print("Project working directory is created | PWD:", pwd.absolute())
    satellite_dir.mkdir()
    mask_dir.mkdir()
    for zoom in zoom_levels:
        satellite_zoom_dir = satellite_dir.joinpath(str(zoom))
        mask_zoom_dir = mask_dir.joinpath(str(zoom))
        satellite_zoom_dir.mkdir()
        mask_zoom_dir.mkdir()
# ------------------------------------------------------------------------------
# Log file of project
log = open(project_name + "/log.txt", "wt")
print("Project Name:", project_name, file=log)
print("Zoom levels selected: ", zoom_levels, file=log)
print("Area of interest:", (minLat, maxLat, minLon, maxLon), file=log)
print("Image size:", image_size, file=log)

# Google Map API JSON Styling
url = [
    "https://maps.googleapis.com/maps/api/staticmap?center={},{}",
    "zoom={}",
    "size=640x640",
    "scale=1",
    "maptype={}",
    "style=element:geometry%7Ccolor:0x000000",
    "style=element:labels.icon%7Cvisibility:off",
    "style=element:labels.text.fill%7Ccolor:0x000000",
    "style=element:labels.text.stroke%7Ccolor:0x000000",
    "style=feature:administrative%7Cvisibility:off",
    "style=feature:administrative%7Celement:geometry%7Ccolor:0x000000",
    "style=feature:administrative.country%7Celement:labels
    .text.fill%7Ccolor:0x000000",
    "style=feature:administrative.land_parcel%7Cvisibility:off",
    "style=feature:administrative.locality%7Celement:labels
    .text.fill%7Ccolor:0x000000",
    "style=feature:landscape%7Celement:geometry.fill%7Ccolor:0x000000",
    "style=feature:poi%7Celement:geometry.fill%7Ccolor:0x000000",
    "style=feature:poi%7Celement:labels.text.fill%7Ccolor:0x000000",
    "style=feature:poi.park%7Cvisibility:off",
    "style=feature:poi.park%7Celement:geometry%7Ccolor:0x000000",
    "style=feature:poi.park%7Celement:labels
    .text.fill%7Ccolor:0x000000",
    "style=feature:poi.park%7Celement:
    labels.text.stroke%7Ccolor:0x000000",
    "style=feature:road%7Celement:geometry.fill%7Ccolor:0xffffff",
    "style=feature:road%7Celement:
    labels%7Ccolor:0x000000%7Cvisibility:off",
    "style=feature:road%7Celement:labels.text.fill%7Ccolor:0xffffff",
    "style=feature:road.arterial%7Celement:geometry%7Ccolor:0xffffff",
    "style=feature:road.highway%7Celement:geometry%7Ccolor:0xffffff",
    "style=feature:road.highway.controlled_access%7Celement:
    geometry%7Ccolor:0xffffff",
    "style=feature:road.local%7Celement:
    labels.text.fill%7Ccolor:0xffffff",
    "style=feature:road.local.trail%7Celement:
    geometry%7Ccolor:0x000000"
    "style=feature:transit%7Cvisibility:off",
    "style=feature:transit%7Celement:labels
    .text.fill%7Ccolor:0x000000",
    "style=feature:water%7Cvisibility:off",
    "style=feature:water%7Celement:geometry%7Ccolor:0x000000",
    "style=feature:water%7Celement:labels
    .text.fill%7Ccolor:0x000000",
    "key={}",
]

# Function for calculating corner coordinates
def get_corners(latitude, longitude, image_size, zoom):
    """
    Function that calculates the geographic coordinates
    of corner pixels in WGS84

    input:
        latitude    : Latitude of central pixel in degrees
        longitude   : Longitude of central pixel in degrees
        image_size  : image size (ex: 512 for 512x512 image)
        zoom        : zoom level in Google Map API

    return:
        latitude and longitude of NE, NW, SW
        and SE corners respectively
    """
    corner_pixel_coordinates = np.array(
        [
            [image_size, 0],  # North-East
            [0, 0],  # North-West
            [0, image_size],  # South-West
            [image_size, image_size],  # South-East
        ]
    )
    latitude_scale = np.cos(np.deg2rad(latitude))
    dx_degrees = 360 / 2 ** (zoom + 8)
    dy_degrees = 360 / 2 ** (zoom + 8) * latitude_scale
    lon = longitude +
    dx_degrees * (corner_pixel_coordinates[:, 0] - image_size / 2)
    lat = latitude -
    dy_degrees * (corner_pixel_coordinates[:, 1] - image_size / 2)
    return (lat, lon)


folium_map = folium.Map(
    location=[(minLat + maxLat) / 2, (minLon + maxLon) / 2],
    zoom_start=10,
    tiles="cartodbpositron",
)  #  CartoDB dark_matter

starting_pixel = (640 - image_size) // 2
ending_pixel = starting_pixel + image_size

print(file=log)
print("-" * 20 * 13, file=log)
print(
    ("{:>20}" * 13).format(
        "Image ID",
        "Satellite Image",
        "Mask Image",
        "Center Latitude",
        "Center Longitude",
        "NE Latitude",
        "NE Longitude",
        "NW Latitude",
        "NW Longitude",
        "SW Latitude",
        "SW Longitude",
        "SE Latitude",
        "SE Longitude",
    ),
    file=log,
)
print("-" * 20 * 13, file=log)
for index in metadata:
    zoom = index[0]
    num_images = index[1]
    zoom_color = index[2]
    satellite_zoom_dir = satellite_dir.joinpath(str(zoom))
    mask_zoom_dir = mask_dir.joinpath(str(zoom))
    for i in tqdm(
        range(0, num_images), desc="| Zoom level:"
        + str(zoom), position=0, leave=True
    ):
        # Generate random coordinates inside the boundaries
        randomLat = np.random.uniform(low=minLat, high=maxLat)
        randomLon = np.random.uniform(low=minLon, high=maxLon)

        # Satellite view
        satellite_filename = "sat_{}_zoom{}.png".format(i, zoom)
        satellite_view = Image.new("RGB", (image_size, image_size))
        url_formatted = "&".join(url).format(
            randomLat, randomLon, zoom, "satellite", API_key
        )
        imx = Image.open(urlopen(url_formatted)).convert("RGB")
        satellite_view = imx.crop(
            (starting_pixel, starting_pixel,
            ending_pixel, ending_pixel)
        )
        satellite_view.save(
        satellite_zoom_dir.joinpath(satellite_filename))

        # Roadmap view
        roadmap_filename = "mask_{}_zoom{}.png".format(i, zoom)
        roadmap_view = Image.new("RGB", (image_size, image_size))
        url_formatted = "&".join(url).format(
            randomLat, randomLon, zoom, "roadmap", API_key
        )
        imx = Image.open(urlopen(url_formatted)).convert("RGB")
        roadmap_view = imx.crop(
            (starting_pixel, starting_pixel,
            ending_pixel, ending_pixel)
        )
        roadmap_view.save(mask_zoom_dir.joinpath(roadmap_filename))

        encoded = base64.b64encode(
            open(
                satellite_zoom_dir.joinpath(
                "sat_0_zoom" + str(zoom) + ".png"), "rb"
            ).read()
        )
        html = "".format
        iframe = folium.IFrame(html(
        encoded.decode("UTF-8")), width=256, height=256)
        marker = folium.CircleMarker(
            location=[randomLat, randomLon],
            radius=1,
            popup=folium.Popup(iframe,
            max_width=256),  # "Image_" + str(i),
            color=zoom_color,
            fill=True,
            fill_color=zoom_color,
        )
        marker.add_to(folium_map)

        # Calculate coordinates of corner pixels
        corners = get_corners(randomLat, randomLon, image_size, zoom)
        coordinates = [
            [corners[0][0], corners[1][0]],
            [corners[0][1], corners[1][1]],
            [corners[0][2], corners[1][2]],
            [corners[0][3], corners[1][3]],
            [corners[0][0], corners[1][0]],
        ]
        borders = folium.PolyLine(locations=coordinates,
        weight=2, color=zoom_color)
        folium_map.add_child(borders)

        print(
            ("{:>20}" * 3 + "{:20.8f}" * 10).format(
                str(i),
                satellite_filename,
                roadmap_filename,
                randomLat,
                randomLon,
                corners[0][0],
                corners[1][0],
                corners[0][1],
                corners[1][1],
                corners[0][2],
                corners[1][2],
                corners[0][3],
                corners[1][3],
            ),
            file=log,
        )
        if rectify == True:
            minlon, minlat, maxlon, maxlat = (
                np.sort(corners[1])[0],
                np.sort(corners[0])[0],
                np.sort(corners[1])[-1],
                np.sort(corners[0])[-1],
            )
            # Rectify Images Using Rasterio
            satellite_data_temp = rasterio.open(
                satellite_zoom_dir.joinpath(satellite_filename), "r"
            )
            satellite_data = satellite_data_temp.read()
            transform = rasterio.transform.from_bounds(
                minlon,
                minlat,
                maxlon,
                maxlat,
                satellite_data.shape[1],
                satellite_data.shape[2],
            )

            with rasterio.open(
                satellite_zoom_dir.joinpath(
                satellite_filename.split(".")[0] + ".tiff"),
                "w",
                driver="GTiff",
                width=satellite_data.shape[1],
                height=satellite_data.shape[2],
                count=3,
                dtype=satellite_data.dtype,
                nodata=0,
                transform=transform,
            ) as dst:
                dst.write(satellite_data)

            mask_data_temp = rasterio.open(
                mask_zoom_dir.joinpath(roadmap_filename), "r"
            )
            mask_data = mask_data_temp.read()
            transform = rasterio.transform.from_bounds(
                minlon, minlat, maxlon, maxlat,
                mask_data.shape[1], mask_data.shape[2]
            )
            with rasterio.open(
                mask_zoom_dir.joinpath(
                roadmap_filename.split(".")[0] + ".tiff"),
                "w",
                driver="GTiff",
                width=mask_data.shape[1],
                height=mask_data.shape[2],
                count=3,
                dtype=mask_data.dtype,
                nodata=0,
                transform=transform,
            ) as dst:
                dst.write(mask_data)
        # Display ground coverage
        if (i + 1) % 10 == 0:
            clear_output(wait=True)
            display(HTML(folium_map._repr_html_()))
clear_output(wait=True)
display(HTML(folium_map._repr_html_()))
folium_map.save(project_name + "/ground_coverage.html")
print("-" * 20 * 13, file=log)
log.close()
```

CURRICULUM VITAE

Name Surname: Ozan ÖZTÜRK

EDUCATION :

- **B.Sc.** : 2009, Aksaray University, Faculty of Engineering, Department of Geomatics Engineering
- **M.Sc.** : 2016, Aksaray University, Institute of Science and Technology Sciences, Department of Geomatics Engineering

PUBLICATIONS AND PRESENTATIONS ON THE THESIS:

- **Ozturk, O.**, Işık, M. S., Kada, M., Şeker, D. Z. (2023). *Improving Road Segmentation by Combining Satellite Images and LiDAR Data with A Feature-Wise Fusion Strategy*, Applied Sciences, 13(10), 6161, `https://doi.org/10.3390/app13106161`.

- **Ozturk, O.**, Işık, M. S., Sarıturk, B., Şeker, D. Z. (2022). *Generation of Istanbul Road Data Set using Google Map API for Deep Learning-Based Segmentation*, International Journal of Remote Sensing, 43(8), 2793-2812, `https://doi.org/10.1080/01431161.2022.2068989`.

- **Ozturk, O.**, Sariturk, B., Seker, D. Z. (2022). *Contributions of Open Source Tools For Road Data Production in Deep Learning Applications*, Fresenius Environmental Bulletin, 31, 8548-8552.

- **Ozturk, O.**, Sarıtürk, B., Seker, D. Z. (2020). *Comparison of Fully Convolutional Networks (FCN) and U-Net for Road Segmentation from High Resolution Imageries*, International Journal of Environment and Geoinformatics, 7 (3), 272-279, `https://doi.org/10.30897/ijegeo.737993`.

- **Ozturk, O.**, Sariturk, B. Seker, D. Z. (2020). Generating Road Data Set for Deep Learning Applications. Presented at 20th International Symposium on Environmental Pollution and its Impact on Life in the Mediterranean Region (MESAEP-2020), 26-27 October, Virtual. (Poster Presentation)

- **Ozturk, O.**, Sariturk, B. Seker, D. Z. (2019). Comparison of FCN and U-Net for Image Segmentation using High Resolution Images and Aerial Orthophoto Maps. Presented at International Symposium on Applied Geoinformatics (ISAG-2019), 7-8 November, İstanbul, Türkiye. (Oral Presentation)

OTHER PUBLICATIONS AND PRESENTATIONS:

- Sariturk, B., Seker, D. Z., **Ozturk, O.**, Bayram, B. (2022). *Performance evaluation of shallow and deep CNN architectures on building segmentation from high-resolution images.*, Earth Science Informatics, 15(3), 1801-1823, https://doi.org/10.1007/s12145-022-00840-5.

- **Ozturk, O.**, Sariturk, B., Seker, D. Z. (2022). *Contributions of Open Source Tools For Road Data Production in Deep Learning Applications*, Fresenius Environmental Bulletin, 31, 8548-8552.

- Sariturk, B., **Ozturk, O.**, Vardar, E., Seker, D. Z. (2022). *Investigation of Potential Usage Of UAV Based Photogrammetry and Lidar for Forest Inventory*, Fresenius Environmental Bulletin, 31, 8475-8481.

- Atik, M.E., **Ozturk, O.**, Duran, Z., Seker, D. Z. (2020). *An automatic image matching algorithm based on thin plate splines*, Earth Science Informatics, 13, 869–882, https://doi.org/10.1007/s12145-020-00476-3.

- Atik, M.E., İncekara, A.H., Sarıtürk, B., **Ozturk, O.**, Duran, Z., Şeker, D.Z. (2019). *Object Recognition with Keypoint Based Algorithms* International Journal of Environment and Geoinformatics, 6 (1), 139-142, https://doi.org/10.30897/ijegeo.551747.

- Bilgilioglu, B.B., **Ozturk, O.**, Sariturk, B., Seker, D. Z. (2019). *Object Based Classification of Unmanned Aerial Vehicle (UAV) Imagery for Forest Fires Monitoring*, Fresenius Environmental Bulletin, 28(2), 1011-1017. https://doi.org/10.1007/s12145-020-00476-3.

- **Ozturk, O.**, Seker, D. Z., Kaya, S. (2018). Using Terrestrial and Unmanned Aerial Vehicle (UAV) Based Photographs for 3D Documentation of a Medieval Castle in Turkey. Presented at the 2018 American Geophysical Union, Fall Meeting 2018, 10-18 December, Washington, DC, USA.

- **Ozturk O.**, Sariturk B., Incekara A.H., Seker D.Z., Bayram B., Duran Z. (2018). Contribution of smartphones for documentation of cultural heritage: A case study of Zilkale. Presented at the 39th Asian Conference on Remote Sensing (ACRS 2018), 15-19 October, Kuala Lumpur, Malaysia.

- Albeni, Y., Sariturk, B., Atik, M. E., **Ozturk, O.**, Duran, Z., Seker, D. Z., Bayram, B., Incekara, A.H., Celik, M. F. (2018). Combination of UAV Based Digital Surface Model with 3D Model of ITU Bicycle House Produced by Close-Range Photogrammetry. Presented at the 2018 ICOMOS Inter-ISC Meeting & Colloquium, 10-13 July, Kastamonu, Turkey.

- **Öztürk, O.**, Bilgilioğlu, B. B. , Çelik, M. F. , Bilgilioğlu, S. S., Uluğ, R. (2017). *İnsanız Hava Aracı (İHA) Görüntüleri İle Ortofoto Üretiminde Yükseklik Ve Kamera Açısının Doğruluğa Etkisinin Araştırılması*, Geomatik , 2 (3) , 135-142, `https://doi.org/10.29128/geomatik.327049`.