UTS UNIVERSITY OF TECHNOLOGY SYDNEY

Automatic updating and verification of road maps using high-resolution remote sensing images based on advanced machine learning methods

By Abolfazl Abdollahi

Thesis submitted in fulfillment of the requirements for
the degree of

Doctor of Philosophy

under the supervision of Distinguished Professor Biswajeet Pradhan
and Dr Nagesh Shukla

University of Technology Sydney
Faculty of Engineering and IT
July 2022

AUTHORSHIP/ORIGINALITY CERTIFICATE

I, Abolfazl Abdollahi, declare that this thesis, is submitted in fulfillment of the requirements for the award of Doctor of Philosophy in the faculty of Engineering and IT (FEIT) at the University of Technology Sydney.

This thesis is wholly my own work unless otherwise referenced or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis. This document has not been submitted for qualifications at any other academic institution.

This research supported by the Australian Government Research Training Program.

Signature: Production Note:
Signature removed prior to publication.

Abolfazl Abdollahi

Date: July 2022

COPYRIGHT

ACKNOWLEDGEMENT

Thank belongs to God, who has given me all the blessings.

I would like to thank the following people, without whom I would not have been able to complete this research and without whom I would not have made it through my Ph.D. degree. First and foremost, I am incredibly grateful to my supervisor, Distinguished Professor Dr. Biswajeet Pradhan, at the School of Civil and Environmental Engineering, Faculty of Engineering and IT, for his invaluable advice, continuous support, and patience during my Ph.D. study. His immense knowledge and plentiful experience have encouraged me in all the time of my academic research and daily life.

I would like to express my gratitude to Dr. Nagesh Shukla and other members of the candidature assessment panel for their feedback and recommendations.

I am thankful to all the members of the department. Their kind help and support have made my study and life in Sydney, Australia, a wonderful time.

I am very much grateful to my friend Ratiranjan Jena and all the lab mates, colleagues, and research team for a cherished time spent together in the lab and social settings.

Finally, I would like to convey my heartiest gratitude to my wife and parents. Without their tremendous understanding and encouragement in the past few years, it would be impossible for me to complete my study.

"Your talent is God's gift to you. What you do with it is your gift back to God."

DEDICATION

This thesis is dedicated to my Wife and Parents.

LIST OF PUBLICATIONS

Published journal papers

1. Abdollahi, A., Pradhan, B., & Alamri, A. (2021). RoadVecNet: a new approach for simultaneous road network segmentation and vectorization from aerial and google earth imagery in a complex urban set-up. *GIScience & Remote Sensing*, 1-24. https://doi.org/10.1080/15481603.2021.1972713.

2. Abdollahi, A., & Pradhan, B. (2022). SC-RoadDeepNet: A new shape and connectivity-preserving road extraction deep learning-based network from remote sensing data. *IEEE Transactions on Geoscience and Remote Sensing*. https://doi.org/10.1109/TGRS.2022.3143855.

3. Abdollahi, A., & Pradhan, B. (2021). Integrated technique of segmentation and classification methods with connected components analysis for road extraction from orthophoto images. *Expert Systems with Applications*, 176, 114908. https://doi.org/10.1016/j.eswa.2021.114908.

4. Abdollahi, A., Pradhan, B., Shukla, N., Chakraborty, S., & Alamri, A. (2020). Deep learning approaches applied to remote sensing datasets for road extraction: A state-of-the-art review. *Remote Sensing*, 12(9), 1444. https://doi.org/10.3390/rs12091444.

5. Abdollahi, A., Pradhan, B., Shukla, N., Chakraborty, S., & Alamri, A. (2021). Multi-Object segmentation in complex urban scenes from high-resolution remote sensing data. *Remote Sensing*, *13*(18), 3710. https://doi.org/10.3390/rs13183710.

6. Abdollahi, A., Pradhan, B., Sharma, G., Maulud, K. N. A., & Alamri, A. (2021). Improving road semantic segmentation using generative adversarial network. *IEEE Access*, 9, 64381-64392. https://doi.org/10.1109/ACCESS.2021.3075951.

7. Abdollahi, A., Pradhan, B., & Alamri, A. (2020). VNet: An end-to-end fully convolutional neural network for road extraction from high-resolution remote sensing data. *IEEE Access*, *8*, 179424-179436. https://doi.org/10.1109/ACCESS.2020.3026658.

8. Abdollahi, A., Pradhan, B., & Shukla, N. (2019). Extraction of road features from UAV images using a novel level set segmentation approach. *International Journal of Urban Sciences*, 23(3), 391-405. https://doi.org/10.1080/12265934.2019.1596040.

9. Abdollahi, A., Pradhan, B., & Shukla, N. (2021). Road extraction from high-resolution orthophoto images using convolutional neural network. *Journal of the Indian Society of Remote Sensing*, 49(3), 569-583. https://doi.org/10.1007/s12524-020-01228-y.

Published conference articles

1. Abdollahi, A., & Pradhan, B. (2021). Road extraction from open source remotely sensing dataset based on the modified deep convolutional autoencoders model. *43rd COSPAR Scientific Assembly*. Held 28 January-4 February, 43, 115. https://ui.adsabs.harvard.edu/abs/2021cosp...43E.115A/abstract.

All of the publications mentioned above were published during my Ph.D. candidature.

PAPERS ADDED IN THE THESIS

Publication citation – incorporated within chapters in a conventional form

Abdollahi, A., Pradhan, B., Shukla, N., Chakraborty, S., & Alamri, A. (2020). Deep learning approaches applied to remote sensing datasets for road extraction: A state-of-the-art review. *Remote Sensing*, 12(9), 1444. https://doi.org/10.3390/rs12091444.

Contributors	Contribution statement	Thesis chapters
Abolfazl Abdollahi	Literature review and analysis (100%) Writing the manuscript and directing the research (75%)	Chapters 1 and 2
Biswajeet Pradhan	Reviewing the manuscript and directing the research (15%)	
Nagesh Shukla	Reviewing the manuscript and directing the research (5%)	
Other authors	Reviewing the manuscript and directing the research (5%)	

Abdollahi, A., Pradhan, B., & Shukla, N. (2019). Extraction of road features from UAV images using a novel level set segmentation approach. *International Journal of Urban Sciences*, 23(3), 391-405. https://doi.org/10.1080/12265934.2019.1596040.

Abdollahi, A., & Pradhan, B. (2021). Integrated technique of segmentation and classification methods with connected components analysis for road extraction from orthophoto images. *Expert Systems with Applications*, 176, 114908. https://doi.org/10.1016/j.eswa.2021.114908.

Contributors	Contribution statement	Thesis chapters
Abolfazl Abdollahi	Literature review and analysis (100%) Writing the manuscript and directing the research (75%)	Chapters 3, 4 and 5
Biswajeet Pradhan	Reviewing the manuscript and directing the research (20%)	
Nagesh Shukla	Reviewing the manuscript and directing the research (5%)	

Abdollahi, A., & Pradhan, B. (2021). Road extraction from open source remotely sensing dataset based on the modified deep convolutional autoencoders model. *43rd COSPAR Scientific Assembly*. Held 28 January-4 February, 43, 115. https://ui.adsabs.harvard.edu/abs/2021cosp...43E.115A/abstract.

Abdollahi, A., Pradhan, B., Sharma, G., Maulud, K. N. A., & Alamri, A. (2021). Improving road semantic segmentation using generative adversarial network. *IEEE Access*, 9, 64381-64392. https://doi.org/10.1109/ACCESS.2021.3075951.

Abdollahi, A., Pradhan, B., & Alamri, A. (2020). VNet: An end-to-end fully convolutional neural network for road extraction from high-resolution remote sensing data. *IEEE Access*, *8*, 179424-179436. https://doi.org/10.1109/ACCESS.2020.3026658.

Abdollahi, A., Pradhan, B., Shukla, N., Chakraborty, S., & Alamri, A. (2021). Multi-Object segmentation in complex urban scenes from high-resolution remote sensing data. *Remote Sensing*, *13*(18), 3710. https://doi.org/10.3390/rs13183710.

Abdollahi, A., & Pradhan, B. (2022). SC-RoadDeepNet: A new shape and connectivity-preserving road extraction deep learning-based network from remote sensing data. *IEEE Transactions on Geoscience and Remote Sensing*. https://doi.org/10.1109/TGRS.2022.3143855.

Abdollahi, A., Pradhan, B., & Alamri, A. (2021). RoadVecNet: a new approach for simultaneous road network segmentation and vectorization from aerial and google earth imagery in a complex urban set-up. *GIScience & Remote Sensing*, 1-24. https://doi.org/10.1080/15481603.2021.1972713.

Contributors	Contribution statement	Thesis chapters
Abolfazl Abdollahi	Literature review and analysis (100%) Writing the manuscript and directing the research (75%)	Abstract Chapters 3, 4 and 5
Biswajeet Pradhan	Reviewing the manuscript and directing the research (15%)	
Nagesh Shukla	Reviewing the manuscript and directing the research (5%)	
Other authors	Reviewing the manuscript and directing the research (5%)	

TABLE OF CONTENTS

LIST OF TABLES

Tables **Page**

LIST OF FIGURES

Figures **Page**

LIST OF ABBREVIATIONS

ADAM	Adaptive Moment Estimation
AEML UNets	Adaboost-Like End-To-End Multiple Lightweight UNets
AI	Artificial Intelligence
ANN	Artificial Neural Network
ASPP	Atrous Spatial Pyramid Pooling
BAL	Boundary-Aware Loss
BCL-UNet	Bi-directional ConvLSTM UNet
BConvLSTM	Bi-directional ConvLSTM
BL	Boundary Learning
BN	Batch Normalization
CasNet	Cascaded End-To-End
CDG	Coord Dense Global
CE	Cross Entropy
CEDL	Cross Entropy Dice Loss
cGAN	Conditional Generative Adversarial Network
CNNs	Convolutional Neural Networks
CP_clDice	Connectivity-Preserving Centerline Dice
CRFs	Conditional Random Fields
CycleGAN	Cycle Generative Adversarial Network
DA-CapsUNet	Dual-Attention Capsule UNet
DA-RoadNet	Densely Connected Blocks Called Dual-attention Network
DCCs	Densely Connected Convolutions
DCNNs	Deep Convolutional Neural Networks
DDSPP	Dense Dilated Spatial Pyramid Pooling
DEM	Digital Elevation Model
DenseNet	Densely Connected Convolutional Network
DH-GAN	Dual-Hot Generative Adversarial Networks
DL	Deep Learning
DLF	Dice Loss Function
DMM	Dirichlet Mixture Model
DNNs	Deconvolutional Neural Networks
DT	Decision Trees

ELU	Exponential Linear Unit
FCN	Fully Convolutional Network
FN	False Negative
FP	False Positive
FRN	Filter Response Normalization
FSM	Finite State Machine
FuNet	Fusion Network
GAN	Generative Adversarial Network
GAP	Global Average Pooling
GCA	Global Context-aware
GCB-Net	Global Context-Aware and Batch-Independent Network
GCPs	Ground Control Points
GIS	Geospatial Information Systems
HRSI	High-resolution Remote Sensing Images
HsgNet	High-Order Spatial Information Global Perception Network
ICN-DCRF	Inner Convolution Integrated Network and Directional CRFs
ILSVRC	ImageNet Large Scale Visual Recognition Challenge
IOU	Intersection Over Union
ITS	Intelligent Transportation Systems
KNN	K-nearest Neighbors
LiDAR	Laser Scanning of Light Detection and Ranging
LLF	Local Laplacian Filtering
LMs	Landscape Metrics
LP	Laplacian Pyramids
LSTM	Long Short-Term Memory
MCC	Matthews Correlation Coefficient
McGAN	Multi-Conditional Generative Adversarial Network
MCG-UNet	Multi-Level Context Gating UNet
MFB_FL	Focal Loss Weighted by Median Frequency Balancing
MIOU	Mean Intersection Over Union
ML	Machine Learning
MRENet	Multitask Road-Related Extraction Network
MRFs	Markov Random Fields

MsGAN	Multi-Supervised Generative Adversarial Network
MUNet	Modified UNet
OA	Overall Accuracy
OBIA	Object-Based Image Analysis
PCA	Principal Component Analysis
PRelu	Parametric Rectified Linear Unit
RBMs	Restricted Boltzmann Machine
RCFs	Richer Convolutional Features
RDRCNN	Refined Deep Residual CNN
ReLU	Rectified Linear Unit
RF	Random Forest
RMSE	Root Mean Square Error
RoadVecNet	Road Vectorization Network
RRCLs	Recurrent Residual Convolutional Layers
RRCNN	Recurrent Residual CNN
RSRCNN	Road Structure-Refined CNN
SC-RoadDeepNet	Shape and Connectivity-Preserving Road Identification Deep Learning Network
SDF	Signed Distance Function
SE	Squeeze and Excitation
SEEDS	Super-Pixels Extracted via Energy-Driven Sampling
SGD	Stochastic Gradient Descent
SVM	Support Vector Machines
THEOS	Thailand Earth Observation System
TN	True Negative
TP	True Positive
TWS	Trainable Weka Segmentation
UAV	Unmanned Aerial Vehicle
UFCN	U-Shaped Fully Convolutional Network
UIteR	Universal Iteration Reinforcement
WGAN-GP	Wasserstein Generative Adversarial Network with Gradient Penalty

AUTOMATIC UPDATING AND VERIFICATION OF ROAD MAPS USING HIGH-RESOLUTION REMOTE SENSING IMAGES BASED ON ADVANCED MACHINE LEARNING METHODS

By

ABOLFAZL ABDOLLAHI

July 2022

Supervisor: Professor Biswajeet Pradhan,

Abstract

One of the significant objects among urban features is the road network. Automatic road network extraction and vectorization from high-resolution remote sensing imagery (HRSI) is a major application in the field of remote sensing and geospatial information systems (GIS), which has a significant role in various purposes such as GIS maps updating, urban cover change detection, emergency tasks, navigation and so on. Nowadays, obtaining accurate information of road networks using various supervised and unsupervised segmentation and classification approaches from HRSI is a challenging task as they are changing very swiftly. In addition, various types of barriers like vehicles, trees, shadows, building roofs exist in the images with having the same spectral values and transparency as the class of road. Moreover, the structure of the road network is complicated and irregular. Traditional and manual methods for road network segmentation and vectorization that human operators manage are time-consuming and expensive. Recently, deep learning (DL) techniques have obtained efficient performance in the field of remote sensing images processing and features semantic segmentation. Therefore, in this research, the state-of-the-art deep convolutional neural networks (DCNNs) are applied for automatic and simultaneous road network surface segmentation and vectorization from different HRSI. The proposed models are capable of extracting road surface and vectorizing road networks simultaneously and efficiently as well as alleviating the shortcomings of the traditional machine learning (ML) and pre-existing deep learning methods for the given task.

Firstly, in objective 1, I solve the issues of conventional ML methods by implementing robust DCNN approaches for road surface segmentation from different HRSI. The presented networks are implemented to the various remote sensing datasets for road surface segmentation and compared with other state-of-the-art deep learning-based networks, which the results prove the superiority of the proposed networks in the road segmentation task.

Secondly, in objective 2, I propose a shape and connectivity-preserving road identification deep learning-based architecture called SC-RoadDeepNet to overcome the discontinuous results and road shape and connectivity quality of most of the existing road extraction techniques. The proposed model comprises a new measure based on the intersection of segmentation masks and their (morphological) skeleton called connectivity-preserving centerline Dice (CP_clDice) that aids the model in maintaining road connectivity. The qualitative and quantitative assessments demonstrate that the proposed SC-RoadDeepNet can improve road extraction by tackling shadow and occlusion-related interruptions and produce high-resolution results, particularly in the area of road network completeness.

Thirdly, in objective 3, I present a new automatic deep learning-based network named road vectorization network (RoadVecNet), which comprises interlinked UNet networks to simultaneously perform road segmentation and road vectorization with achieving important information such as width/length and location of the road network. Particularly, RoadVecNet contains two UNet networks. The first network can obtain more coherent road segmentation maps and the second network is linked to the first network to vectorize road networks. Classification results indicate that the RoadVecNet outperforms the state-of-the-art deep learning-based networks for road surface segmentation and road vectorization. In short, the proposed methods and the outcomes (high quality and accurate road network data) of the study has high potential in environmental applications such as land use change detection in urban areas, and emergency tasks and also commercial value in navigation and road maps updating.

Keywords: Deep convolutional neural networks; machine learning; remote sensing; road segmentation; road vectorization; road maps verification; road database updating

CHAPTER 1

INTRODUCTION

This chapter provides a broad overview and research background for employing high-resolution remote sensing images (HRSI) to automatically update and verify road maps using advanced machine learning approaches. The major backdrop of the study, problem statement, indicated research objectives and aims, research plan, particular research questions, novelty and main contribution of the research, and thesis arrangement are also revealed in this chapter. It emphasizes the need to employ HRSI for automatic updating and verification of road maps.

1.1. General introduction

Spaceborne, airborne, and drone-based sensors using advanced Earth observation and remote sensing technologies have obtained large amounts and different types of high-resolution remote sensing images (HRSI). Such images are extensively used in several applications, such as urban planning [1], disaster management [2], and emergency tasks [3]. Among topographic object classes, road objects are essential urban features. Therefore, the constant updating and verification of road maps is necessary to achieve several geospatial information systems (GIS) goals, such as emergency functions, automated means of navigation, urban planning, and traffic control. A road database can be created and updated using feature extraction from spatial high-resolution satellite imagery [4]. Consequently, generating automatic novel techniques for extracting road classes from high-resolution satellite images and keeping road networks up-to-date in GIS databases are

useful for a variety of applications [5]. High-resolution remote sensing imagery can produce a massive amount of data and has become the main data source for extracting road regions and updating geospatial databases in real time. Although road extraction from remote sensing imagery recently gained considerable attention, this task remains challenging owing to irregular and complex road sections and structures [6]. Other features, such as building roofs, pedestrian areas, and car parking appear similar in satellite images, thereby resulting in insufficient road contexts in images. Meanwhile, roadside buildings, tree shadows, and vehicles on roads can be identified from high-resolution remotely sensed imagery [7]. Given the aforementioned issues, road class extraction from high-resolution remotely sensed imagery is difficult. Manual and traditional approaches for road extraction from high-resolution remote sensing imagery are costly, time consuming, and fraught with errors owing to human operators [8]. Therefore, various road extraction approaches, such as supervised [9] and unsupervised [10] techniques, were suggested for extracting road regions from remotely sensed imagery. Such approaches use textural [11], geometric, and photometric [12] information to extract roads through classification [13]. Techniques for road extraction can be categorized into two categories: (1) automatic and semiautomatic approaches and (2) road area and centerline extraction methods. Automatic techniques are useful in real-time applications and do not require human collaboration, unlike semiautomatic approaches. Road area extraction techniques concentrate on road segmentation and classification, whereas road centerline extraction methods focus on road skeleton recognition [14]. Recently, artificial intelligence algorithms have shown considerable development in feature extraction and segmentation from remote sensing images, thereby persuading researchers to distinguish road sections from high-resolution remote sensing imagery owing to the considerable efficiency of deep learning approaches

in different applications [15-17]. Deep learning is a rapidly growing area in machine learning and has become an effective tool for expediting image processing and object detection. Moreover, deep learning has been widely implemented in remote sensing images, especially in mapping urban land cover with highly accurate results [18]. Therefore, as indicated in the objectives and novelty section, multiple types of deep learning approaches were developed for autonomous road extraction and vectorization from various HRSI in this study. It can be observed that all of the generated models outperformed several traditional machine learning (ML) and state-of-the-art deep learning (DL) models reported in the literature review in terms of both quantitative and qualitative findings.

1.2. Research background

This section provides a summary of traditional road extraction methods. In addition, it discusses the development of deep learning methods in processing remotely sensed images and computer vision, specifically, road semantic segmentation from HRSI.

At present, road extraction and monitoring operations are performed manually, which is ineffective and costly. Therefore, the automatic extraction and detection of roads from high-resolution images would be efficient and cost effective. Previously, remotely sensed imagery, such as multispectral and hyperspectral images, with high-spectral bandwidths was used for traditional remote sensing-based road extraction [19]. The present application of extracting road sections from remote sensing imagery at the macrolevel can be used in urban planning given the huge volume of available high-spectral resolution satellite and low-spatial resolution remotely sensed images [20]. Road extraction methods principally use the depth of spectral information to extract road sections from hyperspectral and

multispectral satellite images [21]. Within the last decade, extremely high-resolution remote sensing imagery, such as orthophoto and unmanned aerial vehicle (UAV) images obtained by advanced remote sensing technologies, was increasingly utilized for shadow classification, road extraction, and vehicle detection. Such fields confirmed the potential of images with high spatial resolutions [22].

Various studies have extracted road parts from high-resolution remotely sensed images using two main techniques, namely, data-driven and heuristic methods. Data-driven methods generally use the information of large data to conduct road extraction from satellite images. Recently, several data-driven approaches were considered for extracting road classes from remote sensing imagery containing conditional random fields (CRFs) [23], clustering [24], and Markov random fields (MRFs) [25]. By contrast, heuristic methods involve texture progressive analysis [26] and mathematical morphology [27], and often use certain information about road sections. Thus, these approaches are useless in handling different types of roads compared with data-driven techniques. However, traditional segmentation approaches fail to achieve high accuracy in road extraction and cannot handle multiscale roads, particularly narrow road sections with high width variance. The reason for this inability is that compared with normal images, high-resolution remote sensing images gain more detail. Thus, narrow road regions become apparent in such images, thereby introducing novel difficulties for road segmentation from high-resolution satellite images. Also, most of the preliminary studies for road extraction are on the basis of unsupervised learning like global optimization and graph cut methods [28] that rely on the color features and they have one general constraint, which is color sensitivity. Therefore, if the colors of roads in remote sensing imagery consist of more than one color, these segmentation algorithms will not attain good results and not perform well in road

extraction and classification. Therefore, new robust techniques, such as deep learning methods, are needed to extract road networks with various scales accurately from remote sensing imagery [28]. This is due to the fact that these approaches may easily encode spectral and spatial information from raw images without the need for any preprocessing. They are also a hierarchical structure of deep neural networks and include a number of interconnected layers that may learn a hierarchical feature representation from the data and extract the deep features of the input data.

In different fields, such as image classification, scene recognition, object detection, and semantic segmentation, advanced cutting-edge convolutional neural networks (CNNs) presently exceed other methods [29]. Compared to the unsupervised approaches that rely on the color for segmentation, more than one feature other than color, such as texture, shape, and line can be extracted by deep learning methods, among others. One of the most well-known methods initially identified to generalize CNNs in computer vision is the AlexNet18 model, which won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) challenge in 2012 [30]. Recently, a CNN model called the fully convolutional network (FCN), which was suggested by [31], revealed promising results in dense semantic segmentation. In addition, remotely sensed image processing, such as object identification in high-resolution remote sensing images [32], semantic labeling of satellite images [33], and image classification [34], was conducted using modern CNN models. The FCN demonstrated satisfactory results in the semantic segmentation of high-resolution remote sensing imagery [35]. Specifically, CNNs and the FCN were also synthesized for road semantic segmentation from remotely sensed imagery to learn road features and extract road regions automatically [36, 37]. One of the initial efforts of implementing deep learning methods for road extraction from remote sensing images was made by [38]. For detecting

road parts from remote sensing data, they applied restricted Boltzmann machine (RBMs). Also, they used preprocessing and postprocessing steps for achieving better results. Saito, et al. [36] proposed a method for roads and buildings extraction from raw remotely sensed images that was different from [38]. This approach was applied on a Massachusetts road dataset that obtained better outcomes. In recent years, many studies proposed that a deeper neural framework showed better results [39]; however, training of such a model is challenging because of the gradient vanishing issue. To address this issue, a deep residual learning architecture is suggested by [40] to simplify training by using an identity mapping [41]. Because conventional road segmentation methods fail to obtain high accuracy results in road extraction and vectorization and are unable to handle multiscale roads, particularly tiny road sections with large width variations, we can argue that DL methods are more robust in automatic updating of road networks with various scales accurately. As a result, there is a need for this research, which aims to develop powerful diverse DL techniques to automatically update road networks from various types of HRSI.

1.3. Problem statement

A huge number of spatial data is now easily accessible on the Web by the fast progression of Internet development and spatial data acquisition methods. But a great number of available maps have been developed with poorly crafted approaches during the last years, whose geometry is not very accurate. In addition, because the focus is especially on updating road networks in urban areas, it can be seen that cities across the world are growing and developing every year on the basis of effective development planning. Also, natural calamities such as earthquakes, floods, and landslides have caused damage to most cities. Thus, achieving real-time information regarding urban features like road networks and updating the maps are required for better urban planning and disaster management

[42]. Poor performances of the conventional approaches as well as inadequate deployment of certain pre-existing methodology for automatic updating and verification of road maps are questionable. This is due to the presence of other features in satellite imagery, such as building roofs, pedestrian spaces, and car parking with similar spectral characteristics, which result in insufficient road contexts in the images. Furthermore, the structures of road networks are complicated and irregular [43]. Moreover, the HRSI has a lot of blended pixels, which makes it difficult to distinguish between other objects and road networks. As a result, urban road networks with rich spectral information in image data can provide false border information [6].

Failure to develop a strong deep learning model that incorporates all relevant characteristics for updating and verifying urban road networks from various HRSI. However, as discussed in the Literature review part in Chapter 2, obtaining real-time information from urban road features and updating maps, one of the primary components of a city that plays a crucial role in its development and extension, is required [44]. To date, there has been no success in implementing an appropriate methodology for upgrading and vectorizing urban road networks [45]. Therefore, to assess the capability of HRSI in obtaining road information and up-to-date traffic maps, automatic road networks vectorization models are required to be developed. In addition, the applicability of these models in getting high-accurate findings must be examined in order to detect limitations, address gaps in the literature, and comprehend the model's strength.

Unfortunately, no major research has been done in the development of a powerful model for obtaining accurate and complete results of road network updating and road map verification using DL models and HRSI. Manual approaches to update a variety of spatial

data sets as a target area anywhere in the world is very time-consuming and error-prone. Conventional approaches have difficulties when detecting roads obscured by trees or buildings. The context information modeling mechanisms of traditional methods cannot build topological links between road segments split by obstacles, resulting in fragmented and discontinuous results for road extraction. Modern methods have been produced to get high-precision geometric data, and it is feasible to enhance the accuracy of geometric mapping data by using modern techniques. Moreover, recent advances in remote sensing technologies make it possible to capture images with high precision and clarity. In this regard, combining high-resolution satellite imagery and robust advanced machine learning methods can be very effective for precisely extracting roads and updating GIS maps. Therefore, from the perspective outlined in the motivation, it is clear that more research is needed to develop advanced models for simultaneous road surface semantic segmentation and vectorization, as well as obtain real-time road information.

1.4. Research gap

Road networks form the majority of modern transportation infrastructure because they are significant man-made ground objects [46]. Previously, the most common method of extracting roads was through manual visual interpretation, which takes a long time and costs a lot of money, and the obtained outcomes may differ because of the interpreter's discrepancies. The technology of automatic road extraction has been a popular topic in this field because it can increase the effectiveness of road extraction [45]. However, high-resolution imagery can reveal the vehicles on the road and the shadows of buildings or trees on the roadside. Furthermore, the road segments are irregular, and the roads structures are complex [47]. The abovementioned challenges make extracting road networks and updating the road maps from high-resolution data more difficult [48]. Some scholars have

employed traditional methods or machine learning algorithms to overcome these difficulties, as evidenced by substantial studies in the literature [7, 49, 50]. Also, the deep learning techniques, characterized by convolutional neural networks (CNNs), have attained a milestone in the computer vision field, owing to the exponential development of accessible data and computational capacity [51],[39]. In recent years, researchers have preferred to use CNN-based algorithms to extract roads from remote sensing data because road extraction can be regarded as a binary segmentation issue. There are, however, some limits to the execution of these works. As a result, in this part, I highlight the major research gaps identified after a thorough literature review:

1. Failure to establish a robust deep learning approach for updating and verifying urban road networks from diverse HRSI that combines all essential properties.

2. Given that threshold settings fluctuate between imagery, conventional approaches can only perform with a limited set of data and cannot be tested in complex environments. Furthermore, most traditional machine learning approaches rely on color features and have one common constraint: color sensitivity. As a result, if the colors of roads in remote sensing data contain more than one color, these segmentation algorithms will not produce satisfactory results and will fail to extract and classify roads.

3. The existing DL approaches in heterogeneous areas cannot efficiently detect the road parts, specifically, where the roads in complex regions are covered by obstructions, such as cars, shadows, and trees.

4. Conventional fully convolutional networks (FCN-based) approaches convey context information through convolutional and down-sampling operations in the local receptive fields. Thus, they have difficulties when detecting roads obscured by trees or buildings.

The context information modeling mechanisms of traditional FCNs cannot build topological links between road segments split by obstacles, leading to fragmented and discontinuous results for road extraction.

5. Most researchers have performed the DL method for road surface segmentation from HRSI without considering the issue of shape and road connectivity challenges.

6. Most of the works were done on road surface segmentation and road centerline extraction from remote sensing data, not road vectorization.

7. Developing new DL models that could address the issue of road discontinuity because of the obstacles is still a huge gap. Moreover, proposing DL models that could extract road surface and vectorize road networks along with achieving accurate and simultaneous road information such as road width and location is still the main research gap.

1.5. Scope of the study

It has been an open and active research issue in remote sensing to extract road from HRSI and automatically update road maps [52]. Studies on automatic and simultaneous road surface segmentation and vectorization using robust DL methods from HRSI are in demand. Therefore, automatic updating and verification of road maps that have evolved for urban planning and development, disaster management, navigation etc. through developing advanced machine learning models and high-resolution remote sensing images are the main scopes.

Therefore, this research's scope deal with:

1. Road network maps updating and verification.

2. Preparing various types of high-resolution remote sensing datasets.

3. Labelling the images and providing high-quality ground truth samples.

4. Developing traditional classification and segmentation ML methods for road extraction.

5. Investigating the significance of additional features such as textural, geometry, and spectral in improving the segmentation results.

6. Analyzing the pre-processing steps like data augmentation, image enhancement, and applying filters in increasing the quality and the size of the dataset.

7. The robust DCNN approaches for generating high-resolution road segmentation maps.

8. Taking the advantages of boundary learning (BL) and connectivity-preserving techniques to address discontinuous results and connect broken road networks.

9. The robust DCNN method for automatic road vectorization with obtaining accurate road's width and location.

Preparing high-quality datasets of HRSI, which include original images and ground truth images, is a key factor in updating road maps. Therefore, in this research, I used different types of remote sensing datasets, some of which are open-source benchmarks, and some were created manually. For creating the original images and corresponding ground truth images, I used ArcGIS 10.8 to label the images manually. The traditional ML classification and segmentation methods developed in this research are based on decision trees (DT), k-nearest neighbors (KNN) and support vector machines (SVM), connected components labeling, multiresolution segmentation technique, Trainable Weka Segmentation method, and Level Set. These methods were applied on high-resolution unmanned aerial vehicles (UAV) and Orthophoto images for road extraction. More additional information like spectral, geometry, and texture information was also added to the methods for improving

the performance. However, since the achievements of the whole Ph.D. thesis contains the development of a robust model for generating high-resolution road segmentation maps and the scope is more toward the automatic road maps vectorization and updating in the complex urban areas, I also performed DL models to achieve high-accurate results and alleviate the limitation of conventional ML methods. Therefore, different types of deep convolutional neural networks (DCNNs) such as generative adversarial network (GAN) with a modified UNet generative model (GAN+MUNet), VNet, Multi-Level Context Gating UNet (MCG-UNet) Network, BConvLSTM with Dense Convolutions UNet (BCD-UNet) Network and Convolutional Neural Network (CNN) with principal component analysis (PCA) and object-based image analysis (OBIA) were implemented to the various datasets for road surface segmentation, which achieved higher accuracy than traditional methods. In the next stage, I developed a shape and connectivity-preserving road detection deep learning-based architecture (SC-RoadDeepNet) to address shape-accuracy and connectivity challenges that occur with most of the pre-existing methods. In a later step, I developed a new automatic deep learning-based network named Road Vectorization Network (RoadVecNet), which comprises interlinked UNet networks to simultaneously perform road segmentation and road vectorization and achieve accurate information of road's width and location.

1.6. Research aim and objectives

The aim of the research is to develop deep convolutional neural networks (DCNNs) to automatically and simultaneously extract road surfaces from various HRSI and then update road maps based on achieving accurate road's width and location information.

To address the research gaps in the literature, the current study developed different types of DCNNs, which are comprehensive and sophisticated. The suggested models are implemented to the different remote sensing images for road surface segmentation and vectorization. The principal objectives of the current study are listed as below:

1. To **develop new robust deep convolutional neural networks (DCNNs)** for road surface segmentation from various HRSI data.

2. To **integrate road shape and connectivity-preserving techniques into DCNNs** for dealing with road shape-accuracy and connectivity challenges.

3. To **develop a new DCNN model for simultaneous and automatic road extraction and vectorization** with achieving road's location and width information from HRSI that is essential for road database updating.

1.6.1. Objective 1

The designed approach's primary objective is to develop new powerful DCNN approaches like GAN+MUNet, VNet, MCG-UNet, and BCD-UNet for road surface segmentation from multiple HRSI such as Aerial, Orthophoto, Google Earth, and UAV images. In the designed approaches, I also took advantage of some additional modules or loss functions to improve the performance of the models in road extraction. For example, I implemented a basic efficient loss function called boundary-aware loss (BAL) that allowed the networks to concentrate on hard semantic segmentation regions such as overlapping areas, small objects, sophisticated objects, and boundaries of objects and produce high-quality segmentation maps. Moreover, I used new loss functions called cross-entropy-dice-loss (CEDL) or Focal loss weighted by the median frequency balancing (MFB_FL) to decrease the class imbalance influence and improve the road extraction results. More details

regarding these modules and functions are explained in the Methodology part Chapter 3. In summary, the proposed DCNN models could achieve higher accuracy and produce high-quality road segmentation maps compared to the traditional ML techniques and other comparative DL models.

1.6.2. Objective 2

This stage of the study involves addressing the issue of most of the conventional ML methods, and state-of-the-art DL approaches for road surface segmentation from remote sensing data. Most of the approaches have trouble identifying road networks hidden by trees or buildings, resulting in fragmented road extraction. Thus, in this stage of the research, I developed a new shape and connectivity-preserving road detection deep learning-based architecture (SC-RoadDeepNet) to build topological links between road segments split by obstacles, resulting in better and continuous results for road extraction. In the developed model, I offered a connectivity-preserving centerline Dice (CP_clDice), a new measure based on the intersection of segmentation masks and their (morphological) skeleton to preserve road connectivity and obtain accurate segmentations. I also utilized road boundaries to make road semantic features more proper for actual road form, solve irregular semantic features, and enhance the boundary of road semantic polygons. I leverage each road's binary edge-map to penalize boundary misclassification and fine-tune the road shape.

1.6.3. Objective 3

This stage develops a novel DCNN approach named RoadVecNet to extract road surface and then vectorize the road network simultaneously. As we have seen in the Literature Review Chapter 2, most of the methods have been applied for road surface segmentation

and road centerline extraction from HRSI, which could not get accurate information about road width and location. Therefore, in this stage of the research, I addressed the road segmentation and vectorization issues with detecting consistent road parts and vectorizing the road network by determining and extracting road vector rather than road centerline to get accurate information about the road network's width and location. In fact, the proposed approach is comprised of two convolutional UNet networks that are interlinked into one architecture for automatic and simultaneous road surface segmentation and vectorization. The initial framework was used to identify road surfaces, while the second framework was used to vectorize roads with achieving the road location and width information.

Furthermore, the current research's models include a variety of hyperparameters and modules, making them robust and effective. Moreover, the overall proposed models are innovative and were created by studying a variety of earlier and newer models for road surface segmentation and vectorization from HRSI.

1.7. Research questions

1.7.1. Questions related to the objective 1

In the first objective of this study, some specific research questions were addressed such as:

(i) Is it possible to classify and extract road features from remotely sensed images with high accuracy?

(ii) What are the benefits of deep learning methods compared to conventional machine learning algorithms for road surface segmentation from HRSI?

(iii) To what extend deep learning algorithms can extract roads from remote sensing images accurately?

According to the aforementioned research questions, the major goals of this study were set to extract road networks by (1) Presenting different types of high-resolution remote sensing data, (2) applying some conventional ML approaches for road surface segmentation, (3) Developing new robust DCNN techniques for road extraction, (4) Comparing the developed DCNN models with the traditional ML and pre-existing state-of-the-art DL models in road segmentation, (5) Adding additional hyperparameters, modules, and other functions to improve the road segmentation results, (6) and checking the efficiency of the proposed DCNN techniques in producing high-resolution road segmentation maps. Accurate road networks extraction from HRSI data is challenging and hard. However, in this research, I tried to develop effective DCNN methods for road network segmentation from different high-resolution remote sensing data more accurately than the traditional and other comparative DL methods.

1.7.2. Questions related to the objective 2

In the second objective of this study, some other research questions were also addressed, such as:

(i) Is it possible to solve the issue of detecting roads obscured by other barriers such as vehicles, trees, shadows, or buildings in the images?

(ii) Is it possible to address the road shape and connectivity-preserving challenges?

(iii) To what extend DCNN models can build topological links between road segments split by obstructions?

This research has three objectives to help reach this goal such as: (1) Develop a new robust DCNN model that accumulates important features and thus enables better feature representation for segmentation task, (2) Use boundary learning (BL) technique to leverage

each road's binary edge-map, penalize boundary misclassification and fine-tune the road form, and (3) Use a new connectivity-preserving centerline Dice (CP clDice) technique to retain road connection and produce accurate segmentations.

1.7.3. Questions related to the objective 3

Other research issues were also addressed in the third objective of this study, such as:

(i) Can we update road maps automatically using deep learning methods?

(ii) What are the requirements for updating road datasets?

(iii) Is it possible to simultaneously extract road network from HRSI and then vectorize?

To achieve this goal, this study includes: (1) Developing new a DCNN model called RoadVecNet that simultaneously extracts road surface from HRSI and then vectorizes the road network, and (2) Obtaining precise information on the width/length and location of the road network, which are the main requirements for updating road datasets.

1.8. The research's novelty and main contribution

In this research, different types of robust DCNN models were developed for urban road network updating and verification from different HRSI. The purpose of this research is to deal with the lack of comprehensive, advanced machine learning techniques for road surface segmentation and vectorization. As a principal contribution, the new DCNN models were conducted to address the issue of achieving high-resolution road segmentation and vectorization maps from HRSI even under large and continuous areas of obstacles with traditional ML techniques and pre-existing DL methods. Thus, the detailed mapping of road segmentation and vectorization from HRSI was conducted to update urban road maps.

The present study is designed to develop new DCNN models with defining new parameters, modules, and functions for obtaining high-quality segmentation maps, proposing a new DL method with BL and connectivity-preserving ability to solve the issue of road discontinuous, and developing a new model for simultaneous road surface segmentation and vectorization. These models were implemented to the various types of remote sensing images with a high spatial resolution to test for the first time. For all objectives, the section "3. The methodology's implementation" provides a detailed description of how the developed models are implemented. All the methods are new and have not been implemented in the literature that provides more coherent and satisfactory road segmentation and vectorization maps. The developed models, maps, and quantitative results achieved in this study could be used by decision-makers and urban planners to efficiently model traffic information, help traffic management, and update city planning and development strategies. The findings of this study may reduce the need for planners and local surveying departments to undertake on-site investigations.

1.9. Thesis organization

The thesis is divided into five chapters. Below is a list of the contents carried out by the chapters in detail.

The introduction to the topic and research background, the research problem, the research gap, the objectives and aim of the research, the research questions, the scope of the study, the novelty and key contribution of the research, and the thesis arrangement are all covered in detail in **Chapter 1**.

The literature of urban road network updating and verification from HRSI is demonstrated in **Chapter 2**. The first section of the chapter goes through the previous studies on deep

learning approaches that have been used to extract road sections from remote sensing images. Based on the type of DL models employed, I divided the results into multiple subsections. In the second part, a comparison of the models' advantages and disadvantages, along with application variability, is given. The paper also offers descriptions of the main conclusions.

The methodology and proposed DCNN models are discussed in **Chapter 3** of the thesis. The different types of remote sensing data, overall methodology, and execution of the developed techniques for road segmentation and vectorization are all demonstrated and discussed in this chapter.

Chapter 4 discusses the results of road surface segmentation and vectorization produced by the various proposed models in terms of both quantitative and qualitative findings.

Chapter 5 summarizes the research with a detailed explanation of the study's shortcomings, significant findings, and future directions.

In this thesis, all of the papers listed on the “LIST OF PUBLICATIONS” page were incorporated with appropriate citations as required by all the chapters.

CHAPTER 2

LITERATURE REVIEW

2.1. Introduction

This chapter elaborates on prior studies on deep learning methods that were applied to different remote sensing images road sections extraction and vectorization [53]. I split the results into several subsections based on the type of deep learning methods used (Figure 2.1). The models, the data, the accuracy, and findings will be discussed in this section. In addition, I will provide the advantages and disadvantages of the models in analyzing remote sensing images for road network extraction, a brief summary, and the ideas for future research in this section. In summary, this chapter presents a broad overview of the many models used for road network extraction and vectorization from various HRSI. Based on previous studies, I categorized all the CNNs into four main models: the patched-based CNN model [38]; the FCN-based model [31]; the deconvolutional net-based models, and the GAN-based model [54]. Each type of model will be discussed in the following subsections in detail.

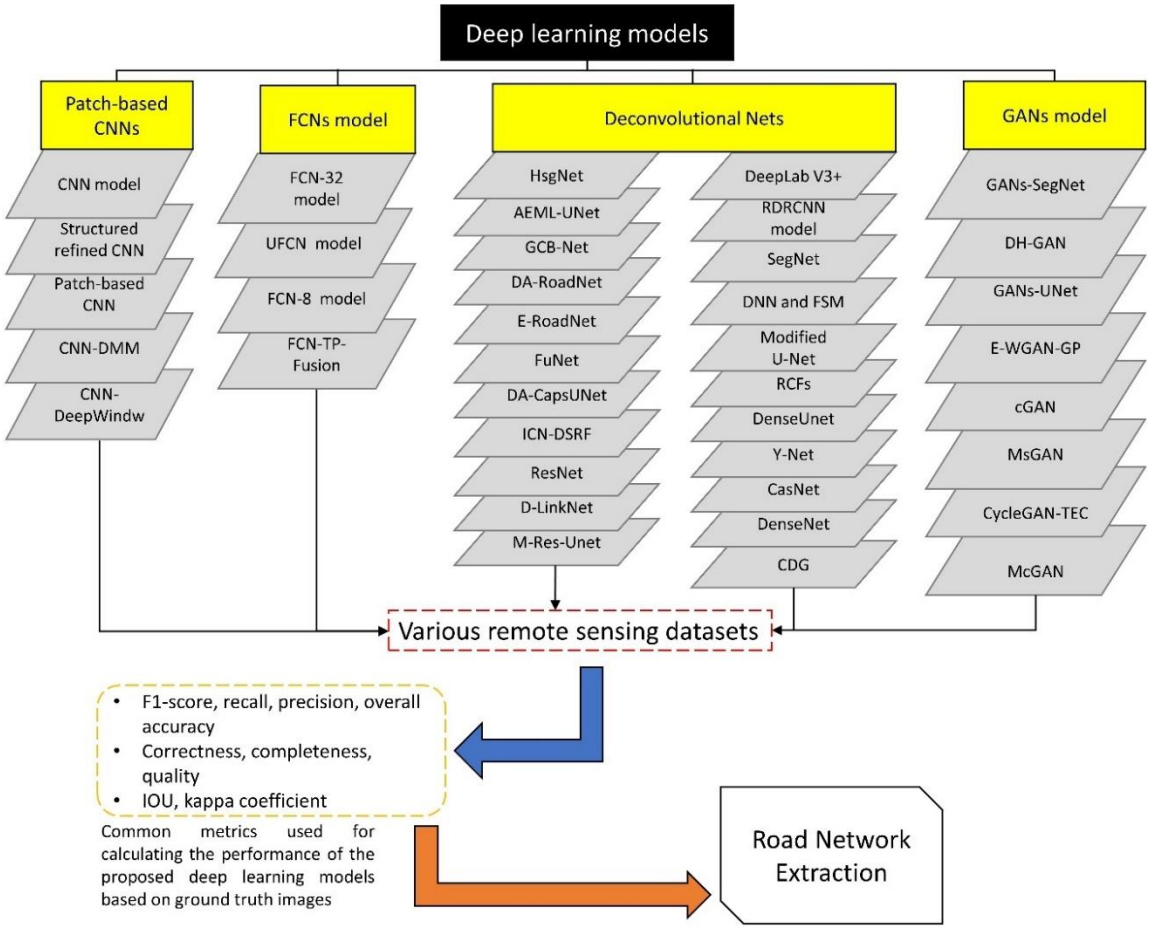

Figure 2.1. Road semantic segmentation using different deep learning models from remote sensing datasets.

2.2. Road extraction based on the patch-based CNN model

In the patch-based CNN model, the possibility of road dispensation is firstly predicted piece-by-piece with a particular stride and then the label map of the whole image is produced by assembling all of the label patches. Figure 2.2 illustrates a general architecture of the patch-level CNNs model. The initial section is convolutional and max pooling layers chased by fully connected layers acting as a linear discriminator. In this section, I describe the prior studies that used the CNN model for road extraction.

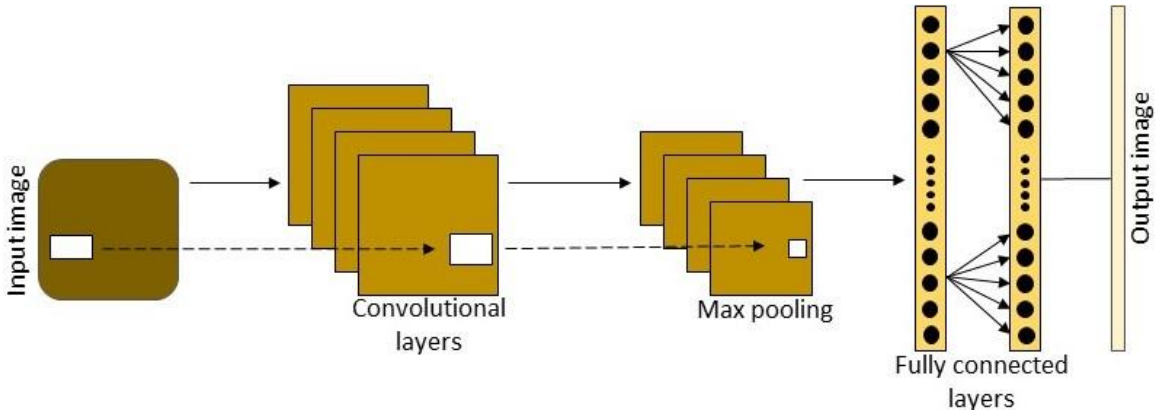

Figure 2.2. General architecture of the patch-level CNNs model.

Zhong, et al. [37] provisionally implemented the newest CNN model to extract road and building objects from satellite imagery. The model fused low-level fine-grained features and high-level semantic meaning. In addition, further hyperparameters, such as the input image size, training epoch, and learning rate, were analyzed to specify the capability of the method in the context of high-resolution remote sensing images. The Massachusetts dataset, with a 1-meter spatial resolution and 1500×1500 pixel size, containing 1711 images for the road and 151 images for the building datasets, was used for the evaluation. The Massachusetts dataset is related to the state of Massachusetts. The dataset covers over 2600 square kilometers with diverse rural, suburban, and urban areas [43]. With the integration of the pretrained FCN method with a novel four-stride pooling layer output to the last score layer, as well as fine-tuned with high-resolution spatial data, the extraction accuracy of the adjusted model was upgraded significantly to over 78%. Wei, et al. [55] used a technique on aerial images for extracting road classes based on a road structure-refined CNN model, which provided road geometric information and spatial correlation. The proposed model was merged with fusion and deconvolutional layers to obtain structured output. Furthermore, a novel road structure-based loss function was applied to

cross-entropy loss to yield a weight map by using the minimum Euclidean distance of every pixel to the road section and to model the road geometric structure. The Massachusetts road dataset, including 1172 images randomly divided into 49, 14, and 1108 images for testing, validation, and training, respectively, was used to calculate the proposed technique. Efficiency measures, namely, F1 score, recall, precision, and accuracy, were calculated for comparison, which were 66.2%, 72.9%, 60.6%, and 92.4%, respectively. The outcomes proved that the suggested model could extract roads effectively and achieve better accuracy compared with other existing road segmentation methods. However, postprocessing was needed to improve results. The link to download the public Massachusetts dataset and CNN code can be found in the online version, at https://www.cs.toronto.edu/~vmnih/data/, https://github.com/AhmedAhres/Satellite-Image-Classification.

Alshehhi, et al. [56] implemented a patch-based CNN model for extracting road and building parts simultaneously from remote sensing imagery. Global average pooling was replaced with fully connected layers to consider a medium of feature maps from the final convolutional layer. Furthermore, the authors implemented a simple linear iterative clustering method during postprocessing to integrate CNN features with low-level features, such as the compactness and asymmetry of buildings and roads. This process integrated ungrouped areas of buildings and connected–disconnected road parts, as well as improved the performance of the proposed method. The Massachusetts dataset, including 10 images for testing, 137 images for training, and 4 images for validation, and the Abu Dhabi dataset with a 0.5 meter spatial resolution per pixel, including 30 images for testing, 150 images for training, and 30 images for validation, were used for the evaluation. The authors used prevalent measure correctness to calculate the performance of the suggested approach, which was 91.7% for the Massachusetts dataset and 80.9% for the Abu Dhabi dataset. The

results showed that the approach was effective in road and building extraction. However, further processing was needed to determine boundaries precisely. Liu, et al. [57] presented an approach for road centerline extraction from high-resolution remote sensing imagery that comprised four major stages. First, a CNN model was used to classify aerial images and learn features from raw images. Second, edge-preserving filtering was applied to the classified images with the original images to exploit road edges. Third, multidirectional morphological and shape feature filtering was used during postprocessing to obtain trustworthy roads. Finally, an integrated Gabor filter model and multiple directional nonmaximum suppression were applied to extract road centerlines. The suggested method was applied to two datasets, namely, the EPFL dataset and the Massachusetts road dataset. Three accuracy measures, namely, completeness, which was 95.40%; correctness, which was 89.97%; and quality, which was 86.21%, were used to quantify the performance, which indicated the advantage of the proposed method for road centerline extraction. However, certain centerlines were not single-pixel wide in the proposed method. Li, et al. [58] employed a model based on a CNN to extract roads from high-resolution satellite imagery. First, a CNN model was applied to allocate labels to every pixel and anticipate the possibility of each pixel relating to road sections. Second, a line integral convolutional-based method was executed to maintain edge information, conjoin tiny gaps, and soften a rough map. Finally, several image-processing operations were implemented to acquire road centerlines. The authors used images from the Pleiades-1A satellite, with a spatial resolution of 0.5 meters, and the GeoEye satellite to test their model. The completeness indicator was 80.57%, the correctness indicator was 96.57%, and the quality indicator was 78.27%, which showed that the proposed model achieved high precision for road extraction in terms of correctness. However, completeness and quality percentages were low, which

was related to the complexity of the texture of various features in the images. Chen, et al. [59] combined the CNN model with Dirichlet mixture models (DMM) to extract road sections from the Shaoshan dataset. First, they filtered out most of the backgrounds with the DMM. Then, for more precise road area detection, a trained CNN model was used. The Shaoshan dataset is a 0.5m resolution Pleiades optical imagery of ShaoShan, China with the original size of 11125×7918. They cropped the original image into 49 images with the size of 1589×1131, which 29 and 20 images were selected as training and testing, respectively. They obtained the completeness, correctness and quality metrics with 85.88%, 88.43%, and 77.21%, respectively. They showed that the suggested method produced good road extraction outcomes in the experiment. However, due to pixel-by-pixel computations, DMM has a significant computational complexity. Lian and Huang [60] presented a unique approach called DeepWindow for extracting the road part from remote sensing data. Without the prior road segmentation, DeepWindow tracks the road directly from the imagery using a sliding window that is guided by a CNN-based decision function. They conducted extensive tests using two datasets: Massachusetts and a Google Earth dataset with the size of 600×600 and spatial resolution of 1.2m. The experiments demonstrated that their technique can detect the road accurately with F1-score=82.5% for Massachusetts and F1-score=90.7% for Google Earth dataset, however, when the road parts are completely obscured by noise, the approach failed to extract them.

2.3. Road extraction based on the FCNs model

Compared to the CNN model that utilizes a dense layer to achieve a fixed-length feature vector and only accepts images with a fixed size, the FCNs model uses the interpolation layer after the final convolutional layer to upsample the feature map and restore the similar input size, as well as accepts input images of any size. In the FCNs, the final dense layers

are replaced with convolutional layers, and then the output is a label map. A general architecture of the FCNs model is presented in Figure 2.3. In the following, the previous research related to the FCNs model and road extraction are explained.

Varia, et al. [61] applied a deep learning technique, namely, the FCN-32 for extracting road parts from extremely high-resolution UAV imagery. UAV-based imaging systems, which commonly use drones, can be used for the real-time assessment of several applications, monitoring tasks, and large-scale mapping, and are managed autonomously by onboard computers or remotely by human operators. UAV-based remote sensing systems are used in various remote sensing applications, such as object recognition [62] and digital elevation model (DEM) generation [63]. Compared with traditional remotely sensed systems, UAVs have multiple advantages, including improved security, high speed, low cost, and high flexibility. In addition, improved details can be provided by high-resolution images taken by drone systems for object extraction and detection. The suggested techniques were evaluated on a UAV image dataset with 189 training and 23 test images. The training time for the FCN-32 was approximately 370 seconds per image.

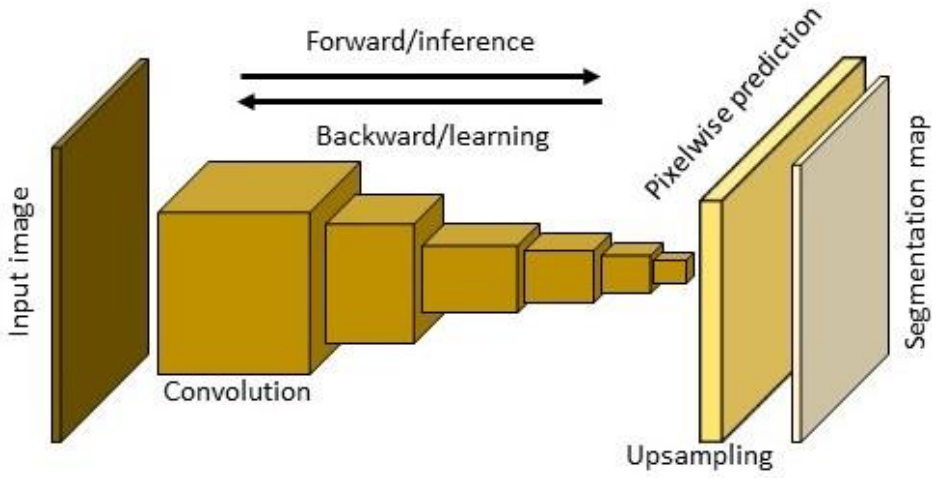

Figure 2.3. General architecture of FCNs model.

The authors evaluated quality, correctness, and completeness assessment measures to show the models' efficiency for road extraction and found that the proposed models achieved satisfactory results. Moreover, they are effective for road extraction from UAV images. However, the models misclassified nonroad areas as road areas in certain areas with high complexity, thereby resulting in a large number of false negatives (an outcome where the model incorrectly predicts the negative class) and reducing the percentage of completeness and quality in the final output. The suggested models were highly dependent on the number of images fed into them for training. Thus, they should be applied to many images with a large variety for better training and improved accuracy.

Kestur, et al. [64] presented a novel architecture based on the FCN called the U-shaped FCN (UFCN) to extract roads from UAV images. The model was used on a UAV dataset with 109 images, approximately 70% of which were used for training and 30% for testing. The authors applied data augmentation during the training step to increase dataset size efficiently to improve training. The prediction took 1.95, 7.68, 43.87, and 1.09 seconds per image for UFCN, SVM, 1D-CNN, and 2D-CNN, respectively. The 1D-CNN model was slower than the UFCN model because of the computationally intensive architecture of the 1D-CNN network. Metric indicators, namely, F1 score, recall, precision, and overall accuracy, were calculated to assess classification performance, which were 89.6%, 86.8%, 92.5%, and 95.2%, respectively. The authors also compared their model with a two-dimensional CNN model, a one-dimensional CNN model, and an SVM model. They found that the model outperformed all the aforementioned methods in terms of accuracy and prediction time. Although the result achieved by the proposed model was promising, the dataset could be extended over a large area to use the suggested method for road extraction from extremely high-resolution remote sensing imagery. An FCN-8 network was proposed

by [65] for road extraction from SAR images. The method was implemented on the TerraSAR-X dataset with 20% for testing and 80% for training. The experimental outcomes proved that the proposed model was able to extract the road part accurately. Wei, et al. [66] performed a multistage deep learning model based on FCN for accurate and simultaneous road surface and centerline extraction. The proposed method includes three main parts: segmentation (based on FCN), points tracking, and fusion (FCN+PT+Fusion). The frameworks were verified on the Massachusetts, Shaoxing, and Google Earth images. The Shaoxing dataset contains 532 images of size 1024×1024 and resolution of 0.6m, while the Google Earth dataset includes 2368 images with the size of 1024×1024 and resolution of 0.6m per pixel. For the road segmentation outcomes, IOU was evaluated that obtained with 78.65%, 61.78%, and 52.47% for Massachusetts, Shaoxing, and Google Earth datasets, respectively. However, the technique failed to detect road segments well in heterogeneous environments. Furthermore, the technique could not obtain correct information regarding road width and location for road centerline extraction. The access link to the open source code of FCN models for satellite image segmentation can be found at https://github.com/Mattymar/satellite-image-segmentation.

2.4. Road extraction based on the deconvolutional neural networks (Dense Nets)

Deconvolutional networks struggle to extract hierarchical features from images that closely pertain to a number of deep learning methods from the machine learning community. These models comprise an encoder and decoder part, which a bottom-up mapping from the input image to the latent feature space is provided by the encoder part while the latent features are mapped back to the input image using the decoder part. A general architecture of deconvolutional networks is shown in Figure 2.4. Following this, the previous works

related to using deconvolutional models for road extraction from remote sensing datasets are highlighted.

Panboonyuen, et al. [28] presented a technique based on a modified deep encoder–decoder neural network to extract road objects from remote sensing imagery. To improve the suggested model, the authors enhanced certain phases of the suggested approach containing the incorporation of the exponential linear unit (ELU) function against the rectified linear unit function. In addition, the authors increased the number of training datasets by rotating images to eight different angles incrementally and used a landscape metrics (LM) method to eliminate false road parts and improve the general accuracy of the output. The designed model was tested on the Massachusetts dataset containing 49, 14, and 1108 images for testing, validation, and training, respectively. The most common metrics, namely, F1 score, recall, and precision, were also used for the performance evaluation, which gained 85.7%, 86.1%, and 85.4%, respectively. The results proved that the suggested approach yields satisfactory results and outperforms state-of-the-art approaches in road extraction from remote sensing imagery in terms of performance metrics.

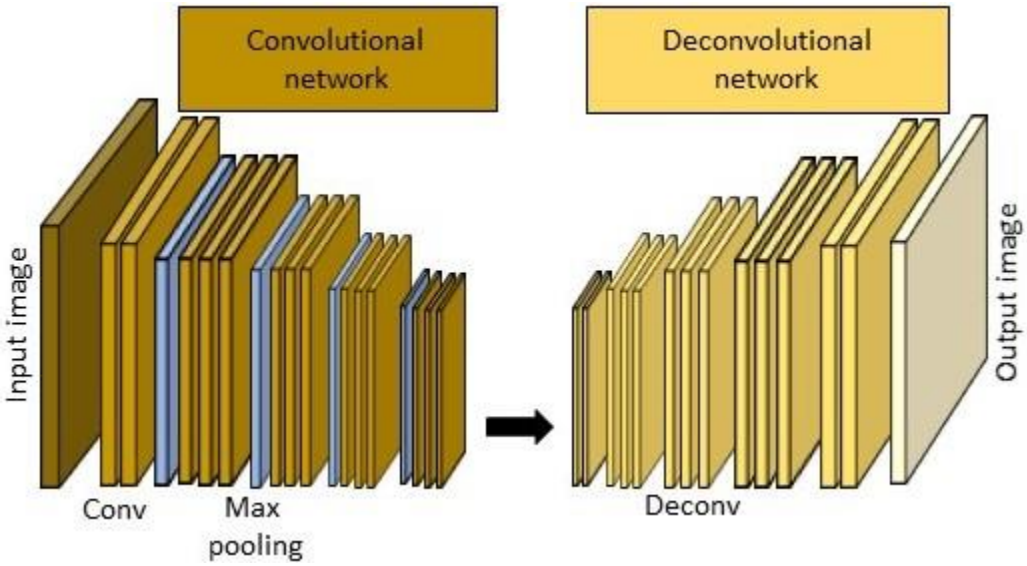

Figure 2.4. General architecture of deconvolutional networks.

Wang, et al. [47] introduced a semiautomatic technique based on the finite state machine (FSM) and DNN, including two main steps, namely, training and tracking, for road extraction from high-resolution remote sensing imagery. In the training step, the model was trained to recognize the pattern of an input image. To generate training samples, a vector-guided labeling approach that elicited huge image-direction mates from available vector road maps and images was defined. In the tracking step, a fusion strategy was used to detect the size of a detection window, and the trained DNN was used to recognize extracted image patches. In general, the DNN was applied to the proposed method to determine a pattern from complicated scenes, and the FSM was used to control the behavior of trackers and translate identified patterns into states. The model was applied to two datasets, namely, aerial and Google Earth images, which were divided into 60%, 20%, and 20% for training, testing, and validation, respectively. Completeness, correctness, and quality percentage indices were used for the performance assessment, which were 75%, 70%, and 74%, respectively, thereby proving that the suggested method could effectively exploit road classes from high-resolution remote sensing imagery in areas that were not highly complex. However, the proposed method could not operate properly in extremely complicated positions where road and other occlusions roughly contribute equal reflectance characteristics.

Panboonyuen, et al. [67] developed a new enhanced deep convolutional encoder–decoder model based on SegNet to segment road classes from high-resolution remote sensing imagery. A new activation function, namely, the ELU, was incorporated into the model to improve accuracy. The LM method was applied to remove falsely categorized road classes and identify road patterns. In the final step, the authors used CRFs to sharpen extracted roads. The proposed model was applied to two aerial and satellite datasets: 1) the

Massachusetts dataset, including 1171 images divided into 1108, 14, and 49 images for training, validation, and testing, respectively, and 2) the Thailand Earth Observation System (THEOS) dataset containing 855 satellite images. The authors used F1 score, recall, and precision performance measures, which achieved 87.6%, 89.4%, and 85.8%, respectively, for the Massachusetts dataset and 64.9%, 58.4%, and 75.1%, respectively, for the THEOS dataset. The results indicated that the suggested approach outperforms other existing road segmentation techniques. However, this framework only works on extremely high-resolution remote sensing images, and distinguishing road sections from low- and medium-resolution remote sensing imagery is challenging. Constantin, et al. [68] introduced a modified UNet CNN for extracting road classes from high-resolution remote sensing imagery. The authors applied a novel binary cross entropy loss function and Jaccard distance fusion to train the model to decrease the number of false positives (an outcome where the model incorrectly predicts the positive class) and enhance the accuracy of binary classification. The proposed method was tested on the Massachusetts dataset, including 49 aerial test images, 14 validation data, and 1108 training data, with extra data augmentation to extend the dataset. For the accuracy assessment, overall accuracy, F1 score, recall, and precision were calculated, which were 97.14%, 74.54%, 75.48%, and 74.15%, respectively. Although the proposed model achieved a high accuracy of over 97%, its accuracy for other parameters was low. Therefore, additional pre- and postprocessing operations are necessary to improve the classification efficiency of the proposed approach for road extraction.

Zhang [52] developed a deep residual UNet model similar to a UNet architecture for road semantic segmentation from high-resolution remote sensing imagery. The proposed network was designed based on residual units, which simplify network training. Rich skip

connections were also used inside the model, which allowed few parameters and facilitated information propagation while achieving improved performance. The authors used their model on the Massachusetts road dataset, including 1171 images divided into 49, 14, and 1108 images as the test, validation, and training data, respectively. The authors compared the suggested model with the UNet model and two other deep networks (e.g., CNN and CNN+postprocessing) for road extraction and found that the suggested technique was more efficient in extracting roads from high-resolution remote sensing imagery in terms of precision and recall. However, the introduced approach could not identify road sections in parking lots and under trees. Hong, et al. [44] employed a method based on richer convolutional features (RCFs) for road segmentation from high-resolution remote sensing imagery. The proposed model contains four principal phases. (1) Training and testing samples were generated based on dataset preprocessing on the main image. (2) The RCF network was trained on the training samples and implemented on the testing images to generate strict road feature maps. (3) Autothreshold segmentation was applied to remove nonroad information and produce a road binary map. 4) Finally, road sections were extracted and vectorized. The authors applied their method on the Massachusetts road dataset, including 865 images. Four metrics, namely, precision, recall, F1 score, and overall accuracy, were used to determine the capability of the proposed method for road extraction, which were 85.8%, 98.5%, 91.5%, and 96.3%, respectively. Although the suggested approach achieved high accuracy for road class extraction from high-resolution remote sensing imagery, it could not gain precise road width information owing to combined pixel and model structure issues.

Xin, et al. [69] applied the DenseUNet model that takes advantage of UNet as primary structure for road extraction from remote sensing images. The DenseUNet model included

skip connection and dense connection units that facilitated the merging of various scales by joints at different network layers. Also, in DenseUNet, the convolution operations with were replaced with up-sampling operations. Two main datasets, namely, the Massachusetts and Conghua datasets, were used to calculate model efficiency. The image resolution of the Conghua dataset was 0.2 m and consisted of three red, blue, and green bands (RGB). A total of 47 aerial images were used in this dataset, with each image consisting of 3×6000×6000 pixels. In this dataset, 80% of the data were used for training and the remaining 20% were used for model validation. The Massachusetts dataset was separated into 49 images, 14 data items, and 1108 data items for testing, validation, and training, respectively. The authors used precision, recall, F1 score, Intersection Over Union (IOU), and the Kappa coefficient to calculate the efficiency of the proposed method for road extraction. The respective values were 78.25%, 70.41%, 74.07%, 74.47%, and 70.32% for the Massachusetts dataset and 85.55%, 78.51%, 76.25%, 80.89%, and 80.11% for the Conghua dataset. The outcomes showed that the suggested technique has the advantage of low noise and high precision.

Li, et al. [70] suggested a new convolutional neural network called the Y-Net, which includes two main fusion and feature extraction modules for extracting road parts from high-resolution remote sensing imagery. A feature extraction module consisting of a deep downsampling-to-upsampling subnetwork was introduced for semantic feature extraction, and a convolutional subnetwork without downsampling was introduced for detail feature extraction. The authors applied a fusion module to combine features for segmenting road classes. Moreover, the proposed technique was tested on the public Massachusetts dataset and a private dataset from the Jlin 1 business satellite. Both datasets were split into a training dataset with 12,376 images, a validation dataset with 474 images, and a testing

dataset with 531 images. The authors calculated mean region IOU (mean IOU), the Dice coefficient, mean accuracy, the Matthew correlation coefficient, and pixel accuracy for the accuracy assessment of the proposed model, which were 77.09%, 85.58%, 82.53%, 71.56%, and 97.36%, respectively. The experiment results showed the superiority and potential of the model for road semantic segmentation from remote sensing imagery. However, the proposed approach possesses several road extraction limitations. A small portion of the remote sensing imagery is occupied by a number of road pixels; thus, class imbalance is a considerable dilemma in road segmentation, particularly in narrow road sections. Thus, the method does not perform well in such areas. In addition, the proposed method requires additional time for training, which could be reduced by introducing transfer learning and generative adversarial network (GAN) fusion in the model, thereby improving accuracy. In general, deep learning models can achieve high accuracy in road extraction from remote sensing imagery compared with other machine learning approaches.

Cheng, et al. [71] presented a new deep learning technique called the cascaded end-to-end (CasNet) deep learning model for detecting road classes and extracting road centerlines from extremely high-resolution remote sensing imagery. The suggested model includes two networks. The first is for detecting road regions, and the second is for extracting road centerlines, which are cascaded to the previous one and take full advantage of feature maps provided previously. The authors used a thinning method to achieve a single-pixel width and smooth road centerline. The model was evaluated on Google Earth images with 224 images. The Earth images obtained using Google Earth were in the form of aerial or satellite images with RGB color and different spatial resolutions based on the data source [48]. The dataset was randomly divided into 180, 14, and 30 images for training, validation,

and testing, respectively. Several regularization methods and data augmentation approaches were applied to reduce overfitting and increase the size of the dataset. Classification metrics, namely, quality, correctness, and completeness, were introduced to evaluate the road extraction performance of the proposed model, which were 88%, 92%, and 94%, respectively. The results showed that the method is effective for road centerline extraction and road detection. However, the proposed method does not perform well in areas where roads are covered by tree occlusions. Therefore, additional high-level semantic information is needed to improve the performance of the method and to extract obstructions effectively. Xu, et al. [72] used a new technique based on a densely connected convolutional network (DenseNet) by introducing local and global road information to segment roads from high-resolution remote sensing images. The method was applied to Google Earth data with a 1.2-meter spatial resolution containing 224 images. The authors calculated F1 score, accuracy, precision, and recall measurement indicators for the accuracy evaluation, which were 95.72%, 96.3%, 96.30%, and 95.15%, respectively. The results proved that the introduced technique is efficient for road extraction. The experiment results were compared with other semantic segmentation methods, such as the DeepLab V3+, FCN, and UNet models, and showed that the proposed method outperformed the others.

Buslaev, et al. [73] developed a deep learning technique based on the UNet family to extract roads from remote sensing imagery. The authors used an encoder similar to the RezNet-34 network, and a decoder was used based on the vanilla UNet decoder. The authors also produced a loss function that considers binary cross-entropy and IOU simultaneously. In addition, data augmentation was used to improve the performance of the method. The model was evaluated on a dataset collected by the DigitalGlobe satellite,

with a 50 cm pixel resolution and 6226 images. Furthermore, 1243 validation images were provided to calculate the performance of the model. IOU was used as a metric for the accuracy assessment of the suggested method, which was 64%, thereby indicating satisfactory results for road extraction. However, the model can be further improved by preparing high-quality labeled masks and amending data augmentation. Zhou, et al. [74] introduced the D-LinkNet model for road semantic segmentation from remote sensing imagery. The proposed model contains an encoder–decoder structure, dilated convolution, and a pretrained encoder for extracting road sections. A dilated convolution is a beneficial alternative to pooling layers, which is a valuable kernel for expanding and modifying receptive feature point fields and keeping detailed information, such as narrowness, connectivity, and complexity, without reducing the resolution of feature maps. The proposed technique was tested on the DigitalGlobe road dataset with 6226, 1243, and 1101 data items for training, validation, and testing, respectively. The IOU metric was evaluated and showed that the method has road extraction capabilities but retains several issues concerning road connectivity and recognition.

Doshi [75] applied an integrated model based on the ResNet and an inception-style encoder called the residual inception skip net to extract roads from satellite images. The introduced model was implemented on a dataset with a 0.5-meter pixel resolution and 6226 images. The dataset was gathered by DigitalGlobe satellites. The dataset was randomly divided into 85% and 15% for training and testing, respectively. The IOU metric was calculated to assess the accuracy of the model, which was 61.3%, thereby showing that the suggested united method can generally exceed the two other baseline approaches (i.e., UNet and DeepLab). However, various postprocessing strategies, such as the use of CRFs, can definitely promote and optimize the performance of the suggested method. Xu, et al. [76]

applied a deep CNN based on deep residual networks to extract roads from WorldView-2 satellite images. A Gaussian filter was first applied as a preprocessing operation to eliminate noise. Next, the M-Res-UNet model was introduced for road semantic segmentation. The authors calculated precision, recall, and F1 score to assess the classification performance, which were 90.04%, 95.17%, and 92.77%, respectively. The proposed method could extract road classes efficiently and achieve improvements for the assessment factors. However, the approach did not perform well in certain areas wherein objects such as cars and building roofs had similar colors and spatial distributions. The authors generated ground truths using vector maps by setting a buffer in which all road areas with similar widths affected the accuracy of the model. Therefore, generating trustworthy labels and considering topological relationships could improve accuracy. Henry, et al. [65] used DeepLabV3+ and Deep Residual UNet to extract road sections from SAR images. The authors also used a control variable and mean squared error in the training process over the spatial tolerance of the network to promote the capability of the method. Each road was manually labeled, from major apparent highways to minor detectable roads. The authors applied the proposed approaches on a TerraSAR-X dataset with 80% for training and 20% for testing. For the accuracy evaluation, IOU, precision, and recall indices were calculated, which were 45.46%, 71.69%, and 75.17%, respectively. The results showed that though the FCNN models obtained satisfactory quantitative outcomes, the models missed multiple road sections and predicted unanticipated features, such as forests and hills.

He, et al. [77] implemented a transfer learning technique for road segmentation from high-resolution remote sensing imagery. First, the authors applied a deep network based on an improved UNet model for training. Second, cross-modal data were used to fine tune the

first two layers of a pretrained network to adjust the local features of the cross-modal data. An autoencoder was used to convert the data into three bands and extract local features for the cross-modal data of various bands. For the evaluation, the proposed method was tested on 6626 WorldView-3 images with a 0.5-meter spatial resolution per pixel. The images were split into 6035 and 591 images for training and testing, respectively. F1 score, precision, recall, and IOU indicators were used to evaluate performance, which were 58.03%, 59.23%, 59%, and 42.03%, respectively. According to the results, the suggested model could extract road sections efficiently but could not achieve high accuracy in complex environments where other objects exhibited reflectances similar to road classes. Xia, et al. [78] applied a DeepLab architecture for road extraction from high-resolution satellite images. The authors first implemented a semiautomatic approach to produce labeled data. A road benchmark was generated automatically then revised manually based on the construction characteristics and road patterns built by the transportation industry. The authors studied data influenced by color distortion as a type of road. Subsequently, they trained a DCNN model with deep layers to learn different road attributes. The designed method was tested on a GF-2 dataset, with spatial resolutions of 1 and 4 meters for the panchromatic and multispectral scanners, respectively. The experiment results illustrated that the suggested approach can recognize road classes from complicated positions with an accuracy of more than 80% in indistinguishable regions. However, smoothness estimation for curved lines is not successfully achieved by the proposed approach. Gao, et al. [79] introduced a new framework called the refined deep residual CNN to extract roads from high-resolution satellite imagery. The proposed method comprises two main units, namely, residual connected and dilated perception units. The authors applied a postprocessing step based on a tensor-voting technique and math

morphology to incorporate split roads and promote the performance of the proposed model. The suggested method was implemented on two datasets: (1) Massachusetts road images with a 1-meter spatial resolution per pixel, including 60, 6, and 10 images for training, validation, and testing, respectively, and (2) GF-2 road images with a 0.8-meter spatial resolution consisting of 60, 16, and 10 images for training, validation, and testing, respectively. The authors calculated IOU, accuracy, recall, precision, and F1 score indicators to assess the quantitative performance of the suggested approach, which were 65.91%, 98.10%, 77.94%, 83.88%, and 80.58%, respectively. The experimental results confirmed the efficiency advantage of the proposed technique for road extraction from remote sensing imagery. However, further processing is needed to achieve high accuracy in outline boundaries and complex urban areas. Xie, et al. [80] applied a new road extraction method using a high-order spatial information global perception framework (HsgNet), which uses LinkNet as its basic network and embeds a middle block between encoder and decoder. The middle block learns to maintain various feature dependencies and channels' information, long-distance spatial relationship and information, and global-context semantic information. They implemented the proposed model on the DeepGlobe dataset that consists of 622 test images, 622 validation images and 4971 training images with a spatial resolution of 0.5 m and image resolution of 1024×1024, as well as the SpaceNet dataset that includes 567 test images and 2213 training images with an image size of 512×512. For evaluating the performance of the proposed method for road extraction, they calculated measurement metrics such as precision, recall, F1 score and IOU that obtained 83%, 82%, 71.1%, and 71.1%, respectively, for the DeepGlobe dataset and 81.6%, 84.5%, 83%, and 71%, respectively, for the SpaceNet dataset. The

experimental results showed that the suggested model performed well for road extraction from high-resolution remote sensing imagery.

Chen, et al. [81] extracted road parts from three datasets called the large road segmentation dataset of New York (LRSNY), Shaoshan, and Massachusetts based on adaboost-like end-to-end multiple lightweight UNets model (AEML UNets). The proposed approach was made up of several lightweight UNet components, which the output of the previous UNet was used as an input for the next UNet. They separated the original Massachusetts images (1500×1500) into 256×256 for their experiment, resulting in 27700, 350, and 1225 images for training, validation and testing, respectively. For Shaoshan dataset, they generated 14580 training images with the size of 256×256 to fit their model input size. The LRSNY dataset is optical images with 0.5 m spatial resolution that includes 716 training, 220 validation, and 432 test images with the size of 256 × 256. In their experiment, they achieved the IOU with 88.21% for the LRSNY dataset, 75.08% for the Shaoshan dataset, and 64.77% for the Massachusetts dataset. The result proved the effectiveness of the model in road extraction from different datasets; however, the model showed serious issues with incorrect extractions, especially in regions obstructed by car parking lots and trees. Chen, et al. [81] proposed a global context-aware and batch-independent network (GCB-Net) for continuous and complete road networks extraction. To successfully incorporate global context characteristics, the global context-aware (GCA) block was added to the encoder-decoder part in GCB-Net. To improve the original basic model, they used the filter response normalization (FRN) layer that remove batch reliance and enhance the model's robustness and accelerate learning. They applied their model to the three CHN6-CUG, SpaceNet, and DeepGlobe road datasets. The CHN6-CUG includes Google Earth images with the size of 512×512 and a resolution of 0.5m per pixel. They divided the dataset into

3608 and 903 images for training and testing, respectively. They divided the SpaceNet dataset with 0.3m resolution and size of 650×650 into 567 images for testing and 2213 images for training. For the DeepGlobe dataset, they cropped the original images into the size of 512×512 and finally created 42255 and 6116 imagery for training and testing, respectively. They obtained an F1 score of 81.54% for the DeepGlobe dataset, 76.33% for the SpaceNet dataset, and 72.70% for the CHN6-CUG dataset. The outcomes of the experiments showed that the suggested framework outperformed other state-of-the-art techniques. However, the baseline network was difficult to segment due to the significant heterogeneity of road networks in Wuhan. Wan, et al. [82] performed a shallow encoder-decoder model with densely connected blocks called dual-attention network (DA-RoadNet) for road extraction, which can reduce the amount of road structural data lost as a result of successive downsampling operations. Also, they included a hybrid loss function to deal with class imbalance. They performed the method on the Massachusetts and DeepGlobe datasets. They cropped the Massachusetts images into image tiles at 256×256 and DeepGlobe images into 512×512. They selected 3736 training, 1245 validation and 1245 testing images for DeepGlobe, while 725 training, 14 validation and 49 testing images for Massachusetts dataset. They attained the quantitative results for the F1-score=78.19% for Massachusetts and F1-score=71.54% for DeepGlobe. However, in order to produce more complete and accurate results, the topology of the roads must be incorporated into the model. To extract road from satellite imagery, Shan and Fang [87] offered a DCNN model with encoder-decoder structure called E-Road network comprises of ResNet-18 with atrous spatial pyramid pooling (ASPP) method. To recover a clear and sharp boundary of road, the PointRend algorithm was used. They trained the model on the DeepGlobe dataset with 6226 training, 1243 validation, and 1101 testing images with a size of 512×512.

Although the proposed model could achieve accurate results with an IOU of 85.20%, more challenging datasets with complex environments are required to be tested to prove the model's efficiency in road extraction.

Zhou, et al. [83] presented a fusion network (FuNet) that combines location data and remote sensing images for road extraction. FuNet has an IteR (universal iteration reinforcement) module that improves network learning capabilities. BeiJing Dataset with a size of 1024×1024 and 0.5m spatial resolution was used for the training (278 images) and testing (70 images). The proposed model achieved an IOU of 63.31%; however, multi-source data integration like road spatial relationship and direction should be used to improve the extraction results. In [84], a dual-attention capsule UNet (DA-CapsUNet) that integrates the powerful features of attention mechanisms and the beneficial aspects of capsule representations was suggested for extracting road networks from remote sensing imagery. The presented technique was evaluated on the 20000 Google Earth images with a spatial resolution of 0.3-0.5m and the size of 800×800. On the test set, the suggested DA-CapsUNet yielded promising road segmentation outcomes with an F1 score of 91.30%. However, the DA-CapsUNet model failed to keep the road's completeness for regions where road portions were highly obscured by dark shadows or covered by the buildings or trees. In another work [85], a multitask road-related extraction network (MRENet) was developed for simultaneous road surface and centerline extraction from GF-2 satellite images. The proposed approach obtained an F1 score of 71.41% for road surface extraction and 70.09% for road centerline extraction; however, it demonstrated errors in maintaining the road connectivity at most intersection areas. Wang, et al. [86] applied an encoder-decoder deep learning method called inner convolution integrated network and directional CRFs (ICN-DCRF) for road extraction. The suggested technique provided good extraction

results, with an F1 score of 84.6%, according to the experimental tests on the Massachusetts dataset. The proposed technique, on the other hand, did not recognise long occlusions and some blurred roads accurately. Wang, et al. [87] proposed a deep learning network named coord-dense-global (CDG) to detect road networks from GF-2 and Massachusetts datasets. The model was built on three main steps: a global attention module, a dense convolutional network (DenseNet), and a coordconv module that translates coordinates into feature maps. They evaluated the F1 score to assess the performance of the model, which achieved 76.10% for Massachusetts and 72.62% for GF-2 images. Although the method improved outcomes, the model's extraction effect was pretty weak when roads are substantially blocked by many trees. The links to download the public datasets and official code repositories of the aforementioned deep learning models can be found in the online version at https://github.com/robmarkcole/satellite-image-deep-learning; https://github.com/jeradhoy/DeepSatelliteData, https://github.com/divamgupta/image-segmentation-keras.

2.5. Road extraction based on the GANs model

GANs comprises two main generative and discriminator models, in which the generative term tries to obtain the data dispensation and the discriminator part tries to determine the likelihood that a representation refers to training data instead of being created by a generative model [88]. The generic architecture of GANs model is presented in Figure 2.5. In this section, previous work related to applying the GANs model for road segmentation is highlighted.

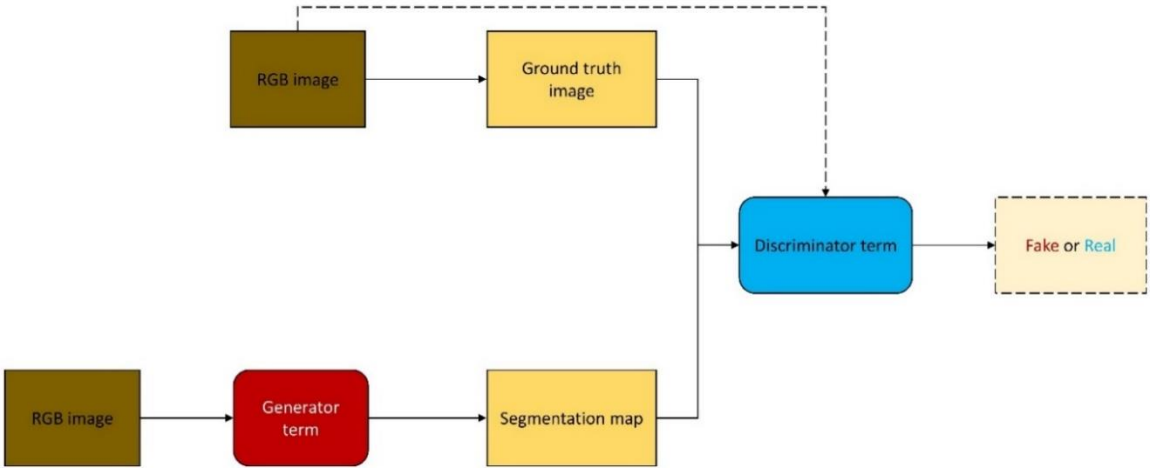

Figure 2.5. Generic architecture of GANs model.

Costea, et al. [89] presented a new method named dual-hot generative adversarial networks (DH-GAN) to detect intersections and roads from UAV images at the higher semantic level of road graphs during the first step. Then, they applied a smoothing-based graph optimization method for pixel-wise road segmenting and finding the road graph. They used the F1 score, precision, and recall for evaluating the performance of the model, which were 86%, 89.84%, and 82.48% that proved the efficiency of the proposed model for road extraction, and also was able to minimize the memory costs. Varia, et al. [61] applied the GANs model for road extraction from UAV images. They used the UNet model for the generator part, and the model was trained on 189 UAV images and evaluated on 23 test images. The training took 300 seconds per image for GANs-UNet. They achieved an accuracy of 96.08 for the F1 score, which shows that the proposed model was more efficient for road extraction from UAV images. Shi, et al. [88] implemented the GANs model for attaining a smooth road segmentation map from Google Earth images with 550 images: 320 images were used for training, 100 images for validation, and 130 images for testing. They also used data augmentation procedures to increase the size of the dataset. An encoder–decoder SegNet model was used for the generative part to generate a high-

resolution segmentation map. The accuracy that they achieved for recall, precision, and F1 score was 91.01%, 88.31%, and 89.63%, respectively, which shows the superiority of the proposed model for road extraction. Yang and Wang [90] applied the E-WGAN-GP approach, which is an ensemble Wasserstein generative adversarial network with gradient penalty (WGAN-GP) technique to extract road from remote sensing data in rural environments. To overcome the class imbalance difficulties in road extraction, they added a spatial penalty component to the WGAN-GP model's loss function. They tested the method on the GF-2 images, including 36000 training images and 4000 validation images with the size of 512×512, and DeepGlobe dataset containing 5500 training, 500 validation, and 300 test images with the size of 1024×1024. They achieved F1-score=85% and IOU=73% for GF-2 dataset and F1=82% and IOU=73% for DeepGlobe dataset.

Cira, et al. [91] implemented a conditional generative adversarial network (cGAN) for road surface areas extraction from a high-resolution aerial dataset including 6784 training tiles and 1696 testing tiles with the size of 256×256. The IOU metric (75.30%) was used to assess the performance of the model in road extraction. However, the method showed shortcomings, particularly in urban domains. Zhang, et al. [92] presented a multi-supervised generative adversarial network (MsGAN) as a learning-based method for road extraction that is jointly trained by the road network's topology and spectral characteristics. The suggested model was tested on two road datasets called Massachusetts and Pleiades-1A remote sensing images with a spatial resolution of 0.5m. The model showed satisfactory performance in road extraction with achieving quantitative statistic (F1 score) of 86% for Pleiades-1A and 86.2% for Massachusetts images. However, in regions where roads are blocked for a long distance, the method produced errors. In [93], a new approach on the basis of cycle generative adversarial network (CycleGAN) and transfer learning with

ensemble classifier (TEC) was performed for road network extraction from UAV imagery. The performance of the proposed techniques was evaluated on 13 test images based on performance measures such as completeness=87%, correctness=82% and quality=71%. The model needs to be tested on more UAV images for extracting roads in complicated settings such as city avenues and roads. In another work [94], a multi-conditional generative adversarial network (McGAN) was implemented to extract roads from remote sensing data. The Massachusetts dataset and Pleiades-1A remote sensing images were used in the experiment to assess the suggested method. Experiments showed that the suggested method produced acceptable quantitative results with F1 score=84.9% for Massachusetts and F1 score=84.1% for Pleiades-1A datasets. The proposed method produced good quantitative results, according to the experiments. The method, on the other hand, was unable to refine the discontinuous structures in some complex regions. The access link to the GANs model code for image segmentation can be found at https://github.com/eriklindernoren/Keras-GAN/tree/master/pix2pix.

2.6. Discussion

Several deep learning techniques have been suggested for extracting road classes from high-resolution remote sensing imagery. However, demands to obtain improved precision for segmented road outcome sets remain. Compared with other machine learning methods, deep learning techniques have shown notable development in object segmentation from images. However, their efficiency in road extraction can be scaled based on the processing power, model complexity, and the size of the training data. This review of existing research proves that compared with other machine learning and traditional techniques; deep learning methods have obtained higher precision in extracting road parts from high-resolution remote sensing imagery.

All the CNNs were classified into four major models: the patched-based CNN model [38]; the FCN-based model [31]; deconvolutional net-based models, such as UNet [95], SegNet [96], and DeepLab [97]; and the GAN-based model [54]. GANs contains two sections called the generator and discriminator parts, which have recently gained considerable attention [98]. The generator part struggles to make fake images from realistic ones, whereas the discriminator part strives to identify feigned images from actual images. Finally, dynamic balance can be achieved by the two parts, and an image can be segmented by the generator portion.

Table 2.1. Strengths and limitations of various deep learning methods for road extraction.

Approaches	Complexity	Output	Smoothness
Models based on GANs	• Model breakdown and lack of convergence for complex and large data • Complex training	• Efficient and robust • Provide constant output	• Capable of achieving boundary information and smooth segmentation map
Models based on CNNs	• Require few parameters • Require extensive samples • Low computing process	• Not highly efficient in providing constant output • Do not perform well in highly complex positions • Ignore the correlation among neighboring pixels • Attain pixel-to-pixel reasoning	• Require high processing to identify boundaries and create a smooth segmentation map

Models based on FCNs	• Low adaptability with complex data and depend on images and masks for training	• Issues with road connectivity • Low position accuracy, lack of spatial consistency	• Cannot successfully achieve smoothness estimation for curved lines
Models based on deconvolutional nets	• Require large amounts of memory and storage • Require additional time for training and high computing process	• High spatial accuracy • Efficient and robust for achieving consistent output	• Able to obtain a smooth segmentation map

In FCN models, each pixel can be inferred end-to-end by examining the patch-to-pixel anticipation. In these models, convolutional layers are replaced by final dense layers in which the output of the label map is the last convolutional layer. Deconvolutional net-based models are identified by deconvolutional layers, which are called decoder sections. Finally, the image block around a pixel can be used to train and anticipate input in the patch-based CNN model. The throughput outcomes of the aforementioned studies have shown that the deconvolutional networks are the most popular models that most of the researchers apply for the purpose of road semantic segmentation from high-resolution remote sensing imagery. I elaborate on the advantages and disadvantages of the discussed approaches to develop a general comparison (Table 2.1).

Table 2.1 shows that each model has its own limitations and strengths. For example, simple interpolation is utilized in the upsampling of the FCN models, thereby causing the models to achieve low precision. However, pixel-to-pixel reasoning can be obtained as well as end-to-end can be learned by FCNs inspired by CNN-based models that need expansive

samples, ignore the correlation among neighboring pixels, and require a high processing step to recognize precise road boundaries. While FCNs models encounter problems with road connectivity and cannot make smoothness predictions for curved lines as well as the segmentation map encounters with low spatial constancy, the DeconvNet model can obtain higher spatial precision and contains high adaptability compared with FCNs, as the former uses low-level information in deconvolutional layers. However, a large amount of storage and memory as well as a high computing process is required for applying this model. By contrast, the GANs model is more efficient because this model can achieve a constant segmentation map with road boundary information. However, the model encounters problems with a lack of convergence, gradient destruction, and complex training.

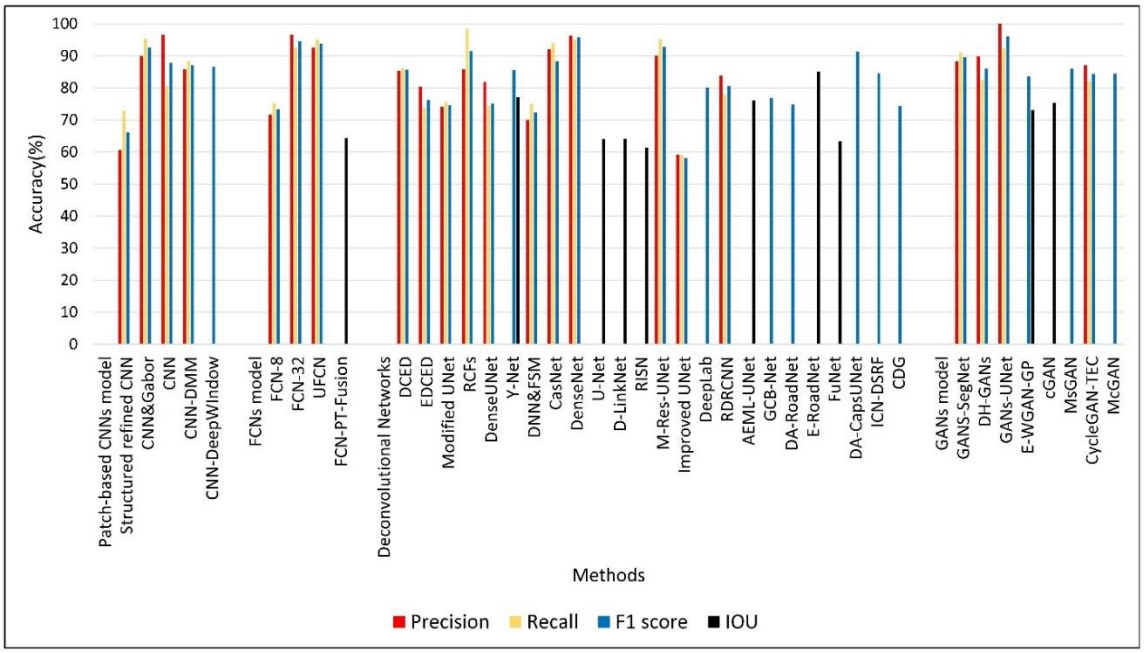

Figure 2.6. General comparison of deep learning models applied to different road datasets.

In addition, I attempt to compare the accuracy of different deep learning models applied to remote sensing datasets based on the common metrics [84] used to evaluate the efficiency of the proposed approaches for road extraction. Popular evaluation measures are calculated

based on a confusion matrix comprising four main factors, namely, false negative (FN), true negative, true positive, and FP [99, 100]. A general comparison of all the methods used in all datasets is provided to elaborate on the most efficient technique for road extraction (Figure 2.6). All the aforementioned works and corresponding values are plotted using an x-axis and y-axis, respectively. Only the methods that include a dataset and research performance reports are compared.

I consider the F1 score metric, which is a trade-off measure between recall and precision, to compare the results achieved by different deep learning models for road extraction, except for some models, as the authors utilized only the IOU indicator for the performance evaluation. Figure 2.6 shows that the F1 score percentage is high for the GANs-UNet model, DenseNet method, and FCN-32 applied to UAV and Google Earth images, with accuracies of 96.08%, 95.72%, and 94.59%, respectively. One of the elegant fully convolutional neural networks named UNet model was used for a generative model in the GANs framework to create a high-resolution segmentation map with more accuracy. Also, the model was applied on UAV images that consist of very high spatial resolution with a variety for the angle of capture, color, shapes, and orientation, which led to achieving a highly precise road segmentation map compared to the other deep learning models. Figure 2.7 illustrates the results achieved for road segmentation from UAV images (Figure 2.7a, b) with image dimension of 128×128, Google Earth images (Figure 2.7c) with a spatial resolution of 1.2 m and image dimension of 256×256, and the Massachusetts dataset (Figure 2.7d) with a spatial resolution of 1 m and image dimension of 375×375, by using the FCN-32, GANs-UNet, DenseNet, DeepLab V3+, CNN, and RSRCNN methods. The first and second columns are original and ground truth images, while the third and fourth columns depict the results achieved by the state-of-the-art methods. As it can be seen from

Figure 2.7, the GANs model applied on UAV images performed better and predicted less FP and FN pixels when compared to other methods. Also, a smooth segmentation map with more details of boundary information is attained by the proposed model. In contrast, the CNN model applied on the Massachusetts dataset was unable to achieve high accuracy in road extraction compared to the RSRCNN method that was applied on the same dataset. The extracted road parts by CNN have a significant issue of fuzzy boundaries and "salt and pepper" phenomena because the CNN model only counts on texture and spectral features; the mixed pixels in road borders lead to misclassification while the other methods improve the classification performance by restraining the effect of mixed pixels by the segmentation process. In the models such as DenseNet and GANs, road features are extracted from every convolutional layer and then integrated on multiscales. Multiscale merging of road features not only uses high-level semantic information to avoid influence of width changes, curvatures, and shadows to achieve precise road boundaries, but also utilizes low-level information to preserve detailed information of road features. As a result, the CNN model predicted more nonroad pixels that lead to extract larger road parts compared to the reference map with low accuracy.

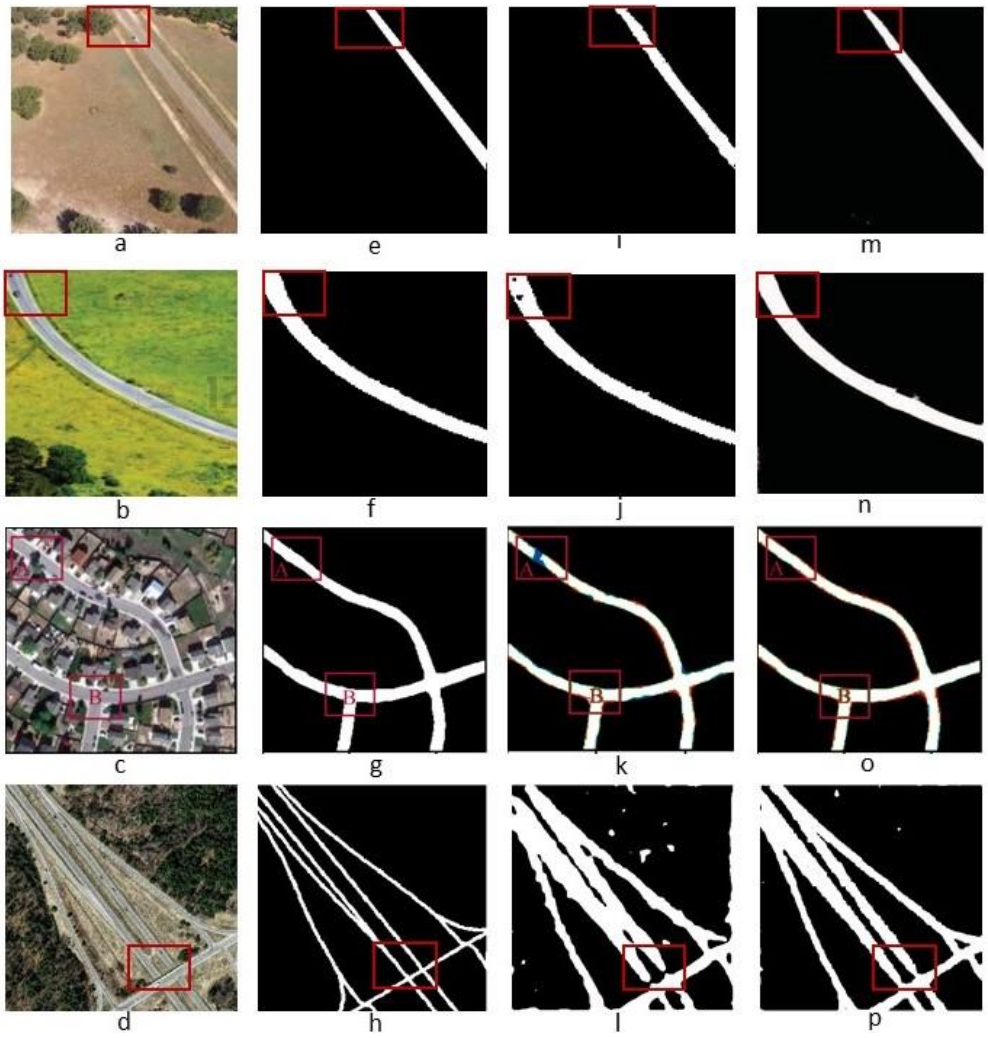

Figure 2.7. Extracted road parts using deep learning methods from high-resolution remote sensing images: (a,b,c,d) original images; (e,f,g,h) corresponding reference maps; (i,j) results of FCN-32 and (k) result of DeepLab V3+; (m,n) results of GANs-UNet and (o) result of DenseNet model; and (l,p) results of CNN and RSRCNN methods, respectively.

2.7. Summary

Despite the fact that a variety of methodologies have been used to identify road networks from remote sensing data, they all have flaws. In other words, pre-existing techniques could not detect road parts well in heterogeneous areas. Thus, by integrating new hyperparameters, modules, and other functions, I tried to develop robust DCNN methods to accurately extract road network high-resolution remote sensing images and tackle the

shortcomings of existing methods in road extraction (**Objective 1**). Also, due to the complex characteristics of covered roads, typical FCNs-based approaches will not be capable of detecting them accurately. Furthermore, because these techniques are mainly encoder-decoder architectures, the boundary and connectivity precision of the road extraction findings would diminish during the downsampling phase. The number of feature maps in the encoder rises as the model goes deeper, while the spatial resolution diminishes. Feature maps' spatial resolution is gradually regained in the decoder arm via the up-sampling layer; however, edge information is degraded. Because roads are man-made objects with distinct borders, concentrating on boundary and connectivity precision increases the road network quality. Convolutional and down-sampling processes in the local receptive fields are used in traditional FCN-based techniques to convey context information. As a result, they have trouble detecting road networks that are hidden by trees or buildings. Traditional FCNs' context information modeling processes are unable to create topological linkages between road segments broken by barriers, leading to fragmented and discontinuous road extraction outputs. Therefore, in this research, I developed a shape, and connectivity-preserving road detection deep learning-based architecture (SC-RoadDeepNet) is suggested to address the shape-accuracy and connectivity challenges (**Objective 2**). Moreover, some of the previous studies applied deep learning models to extract road surface and centerline simultaneously. However, for road centerline extraction, the existing approaches could not extract road centerline well around road intersections and could not get accurate information about road width and location (Not Road vectorization is done). Therefore, in this study, I developed a new deep learning model called RoadVecNet to extract the road surface and vectorize the road network simultaneously by identifying and extracting road vector instead of road centerline

to obtain correct location and width information about the road network (**Objective 3**). According to the literature review, this study can provide the following important outcomes.

1. The capabilities of deep learning methods for road extraction are more effective than those of conventional approaches.

2. When the complexity of images is high, and various road types are present, the accuracy of the models is low. Therefore, mixing additional robust functions and modules to the DL techniques is recommended and useful to achieve satisfactory results.

3. Occlusions, such as shadows, cars, and buildings, are similar to road features, such as colors, reflectance, and patterns. Road extraction remains challenging owing to such issues. Also, most of the methods resulted in fragmented and discontinuous road extraction results, where the aforementioned issues cover the roads. Thus, developing a robust DCNN model to preserve the shape and connectivity of road networks is recommended and beneficial.

4. Further research is required to build detailed techniques with high precision. CNNs trained by one dataset may be inconsistent with other scenes. Nonetheless, if training datasets are adequate and a deep learning model can be created effectively, then the model can be implemented properly on the most prevalent datasets.

5. Most of the methods focused on road surface segmentation and centerline extraction without achieving accurate information regarding the road's width and location. Therefore, further research is required to build detailed techniques with high precision for road vectorization to not only extract road surfaces accurately but also vectorize the road network and obtain the above-mentioned essential information.

In the literature review, state-of-the-art DCNN models that represent common and newly advanced methodologies were described. In conclusion, introducing several new robust methods related to road semantic segmentation is important, and research on different proposed techniques with cutting-edge technology for road vectorization is increasing.

CHAPTER 3

MATERIALS AND METHODOLOGY

3.1. Introduction

Several approaches, including conventional ML methods and DCNN models applied for road extraction, and vectorization from different high-resolution remote sensing data are illustrated in this chapter. The overall methodology, detailed methodology execution, and performance assessment are all presented. The utilized materials and data and experiment settings were thoroughly discussed. First, some traditional ML approaches such as Trainable Weka segmentation and Level Set methods applied on UAV images and integrated technique of segmentation (multiresolution segmentation method) and classification methods (DT, KNN, and SVM) with connected components analysis implemented on orthophoto images for road extraction are presented. Second, various kinds of DCNN methods with additional modules and loss functions such as GAN+MUNet, VNet, MCG-UNet network, and BCD-UNet network implemented to the different remote sensing images for road surface segmentation are described. The SC-RoadDeepNet model was developed to extract accurate road surfaces from different images and solve the issue of road shape and connectivity challenges. Finally, the RoadVecNet approach was used to extract road surface and vectorize road network accurately and simultaneously. Figure 3.1 depicts the overall framework for road databases updating system that briefly present three objectives: road surface segmentation using different robust DCNN models with additional parameters from various HRSI, dealing with broken road parts and connectivity-preserving challenges to improve the road segmentation maps

and achieve the road information like location and width/length based on road vectorization technique that is essential for road database updating. The methodology includes five main steps: Step 1 includes data preparation. In this step, different types of high-resolution remote sensing images were prepared to evaluate the proposed techniques for road extraction and vectorization. Step 2 contains applying data augmentation methods or image enhancement techniques to increase the size and quality of some training datasets. In Step 3, traditional ML methods and state-of-the-art DL models were implemented for road surface segmentation, and the results were compared. In Step 4, I applied a new DCNN technique called SC-RoadDeepNet to delete non-road noises, connect broken roads, and preserve the shape of road networks. Finally, In Step 5, a new DCNN method (RoadVecNet) was developed for road surface segmentation and road vectorization automatically and simultaneously that can achieve important information like width and location of road networks, which is essential for updating road database. In the following, the methods are described in detail based on each objective and published works mentioned on the “LIST OF PUBLICATIONS” page.

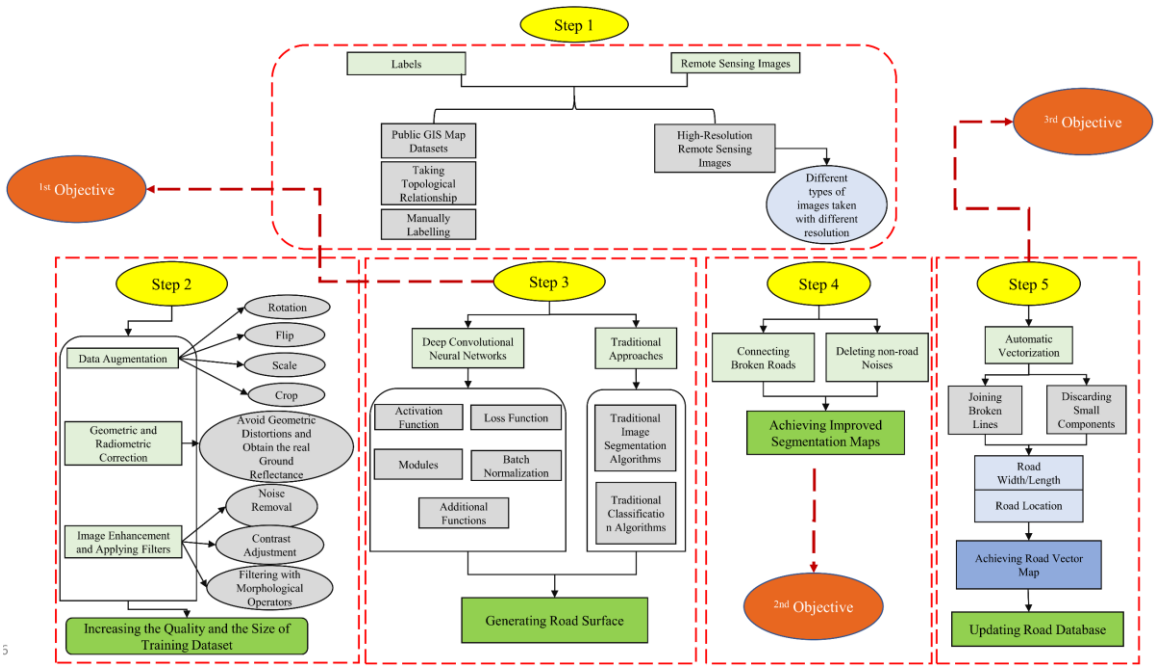

Figure 3.1. Overall flowchart of research methodology for road database updating

3.2. Conventional ML methods for road surface extraction

In this study, some traditional ML techniques such as multiresolution segmentation method, DT, KNN, SVM, connected components method, Trainable Weka segmentation method, and Level Set were applied for road extraction from HRSI, which the implementation of the methods are described in this section. The Trainable Weka segmentation method and Level Set were implemented to the UAV images for road extraction, while the multiresolution segmentation, DT, KNN, SVM, and connected components methods were applied to the orthophoto images to extract road networks.

3.2.1. Level Set segmentation approach

This study proposed a new approach based on Trainable Weka Segmentation (TWS) and LS techniques for road extraction from UAV. Also, a series of filtering processes such as

detectors for edge detection, filters for texture, filters for noise depletion, membrane finder and new morphological filtering approach were applied for improving extraction precision.

The suggested road extraction method consists of the following steps. First, some training data (200 samples) are selected as input for the TWS algorithm. Then, the algorithm is implemented for image segmentation. One of the essential stages in image processing and recognition is segmentation [101]. In the image segmentation process, images are divided to disjoint and uniform areas on the basis of color, texture and depth [102]. These similar sections are supposed to match with the real classes in an image during processing. Thus, image segmentation plays an important role in image processing. After segmentation, the subsequent processes, such as identification and interpretation, are implemented. Therefore, the outcome achieved from image segmentation is essential in high-level image processing.

In the next part, the LS method is performed for extracting roads from UAV images. Some roads can be identified more easily because they are more recognizable and include less noise. As roads in the corresponding image assign some general visual features, the information from the previously elicited roads and other objects, such as spectrum, can be utilized to interpret the process of classifying roads that are less obvious or massively influenced by surrounding objects. Otherwise, these roads are not simply separable from patterns created by other objects. For instance, a collection of small blocks may resemble with the road class from the corresponding block. Subsequently, for improving road class extraction accuracy, morphological operators are applied. Considered as the most common operators in feature extraction, opening and erosion operators are used in this study [103]. Numerous deficiencies can be found in binary images. Particularly, binary sections, which are provided by uncomplicated thresholding methods, are deteriorated by texture and noise.

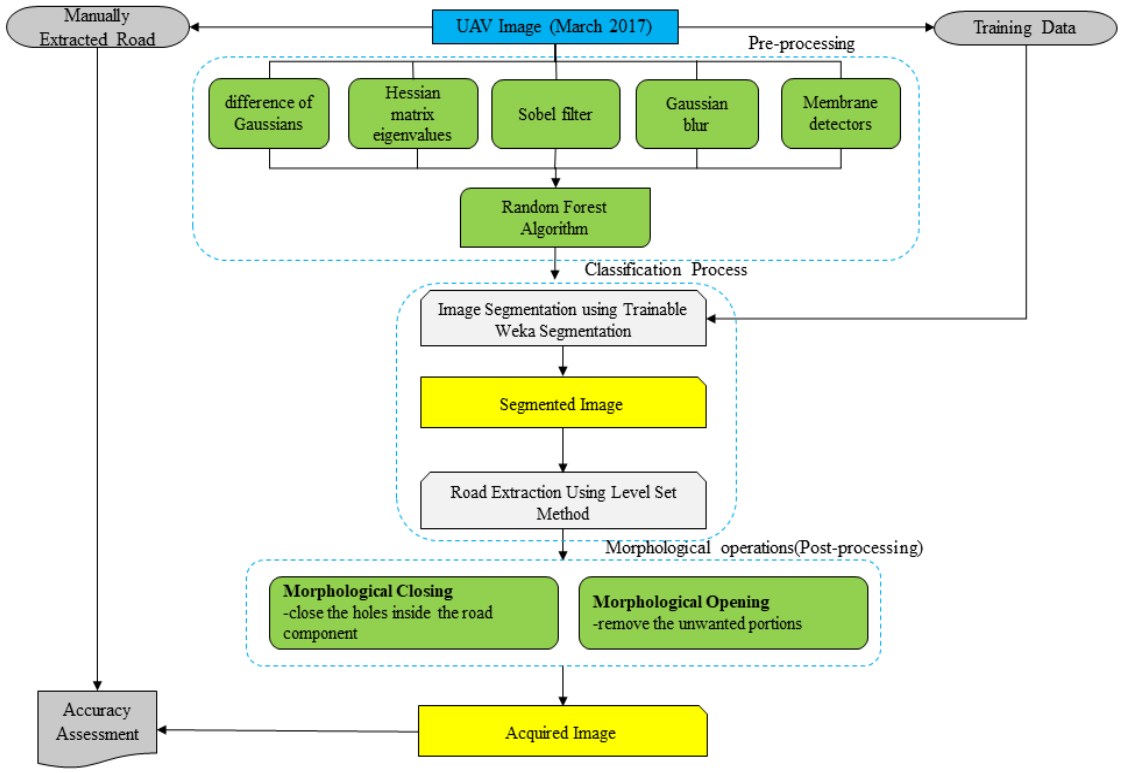

Figure 3.2. The methodological framework of Level Set segmentation approach for road extraction.

Morphological operators in image processing eliminate these defects by considering the structure and formation of the image [7]. Finally, the accuracy of road extraction from the images is calculated. All the steps listed above are shown in Figure 3.2.

3.2.1.1. Data

Data from UAV images from the Shiraz region were used to evaluate the Level Set method for road extraction (Figure 3.3). Shiraz City is in the southwestern part of Iran in Fars Province (29.61° N, 52.53° E) with an elevation of 1500 m above sea level. A UAV, also known as a drone, is an aircraft remotely or autonomously managed by a human operator or an onboard computer, respectively. UAV-based remote sensing can be used for large-scale mapping and monitoring activities and real-time evaluation of multiple applications [104].

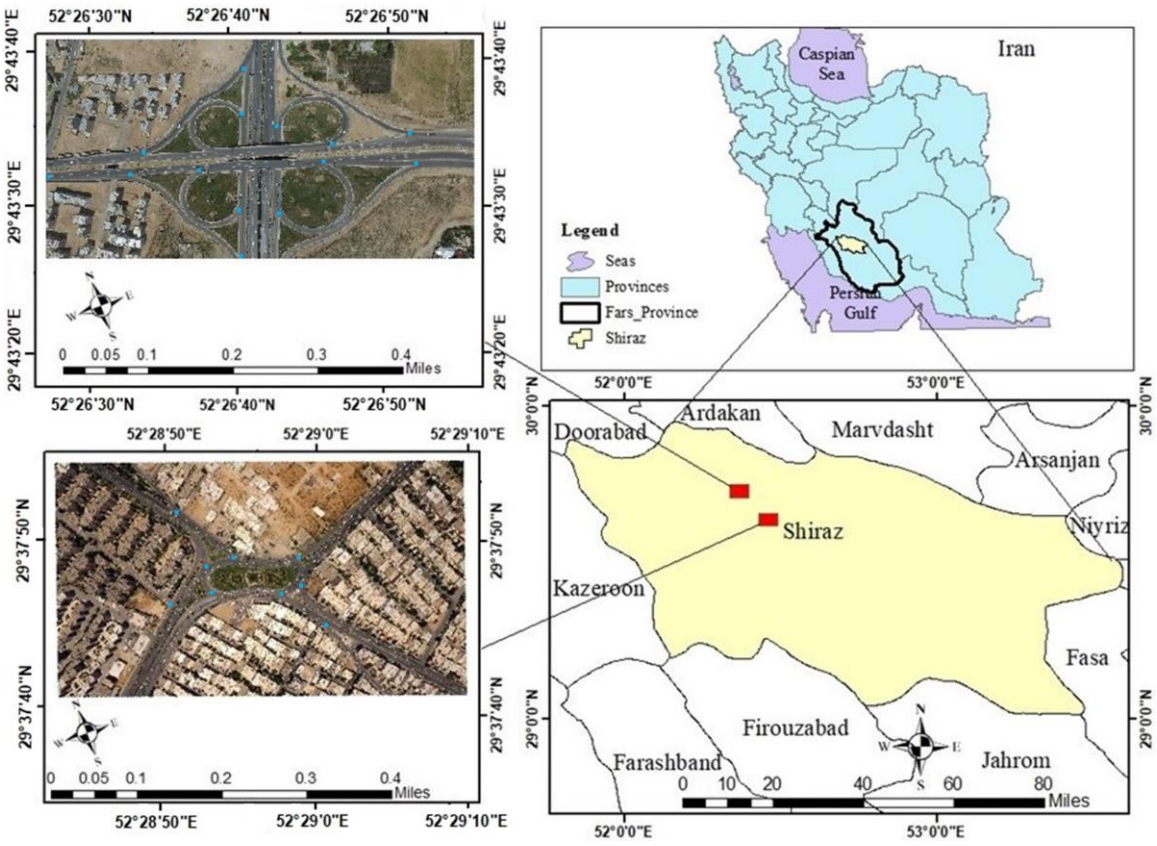

Figure 3.3. Study area location map (Shiraz, Iran) and UAV image used for road extraction based on Level Set method. Some examples of GCPs are shown on the images in blue color.

The UAV images were collected on March 04, 2017, using Phantom 3 drone, with a flying height of 1000 meters. These images have no spectral band and are taken as RGB with a resolution of 5 cm. 108 images acquired in 60 % forward lap and 30 % side lap based on oblique left-tilted method, which all the images were finally mosaiced.

3.2.1.2. Geometric and atmospheric correction

Since the UAV images was captured by Phantom 3 drone, it was essential to calibrate it geometrically before any processing to correct the geometric errors. For geometric calibration, first some ground control points (GCPs) were collected. These points were collected from clearly identifiable points (corners, intersection roads and solitary trees).

Geometric calibration was applied in ArcGIS 10.6 software and included three main steps: (A) recognition of transformation points in the UAV images, (B) using the least square transformation, and calculation of the accuracy of the process. The chosen points were well distributed throughout the images. Then, least square approach used to evaluate the coefficients, which are necessary for geometric transformation process. In this step, polynomial equations were applied to identify the root mean square (RMS) error between the aligned data points and source data points. For atmospheric correction, since UAV images were taken in a good weather condition and low altitude, so atmospheric correction process was not applied on the images.

3.2.1.3. Trainable Weka segmentation

One of the Fiji plugins is the TWS, which merges several algorithms in machine learning with a collection of selected image characteristics to create pixel-based segmentation [105]. For image data segmentation, TWS converts the problem of segmentation into a problem of pixel classification, in which each pixel should be categorized to a particular class or section. The collection of input pixels that are specified is displayed in the property space and then applied as a training collection for a selected classifier. During the training of the classifier, it would be handled to categorize either entirely as new image data or the rest of the input pixels. TWS includes a combination of visualization tools and algorithms for predictive modelling and data analysis, together with graphical user interfaces for easy access to this functionality. Also, it contains a wide variety of image characteristics, and most of them are elicited by usual plugins or filters accessible as a section of Fiji. The features existing in TWS can be classified as (1) detectors for edge detection, which target demonstrating object borders in an image (e.g. Gabor filters, difference of Gaussians,

Hessian matrix eigenvalues, Sobel filters, Laplacian); (2) filters for texture analysis for evoking texture data (containing filters, such as entropy, variance, minimum, maximum and median); (3) filters for noise depletion, such as Lipschitz and Kuwahara, anisotropic diffusion, bilateral filter and Gaussian blur [105]. In this study, the difference of Gaussians, Hessian, Sobel filter, Gaussian blur and membrane detectors were used as training features to indicate the boundaries of road objects in an image, reduce noise in the image and localize the membrane-like structures of certain size and thickness. Furthermore, TWS provides users with customizing features. Accordingly, a rather easy script is required to add user-defined features in the segmentation process, in combination with the existing filters or alone. This can help users to create all kinds of linear and nonlinear features. Furthermore, a fast random forest (RF) algorithm is applied as a classifier because of its efficiency and robustness. The RF algorithm is an ensemble classifier, which uses a randomly elected subset of training variables and samples to generate several decision trees. The RF method is a technique of machine learning, which is frequently applied to image classification and creation of connected objects, such as roads and vegetation [106]. In addition, this technique needs fewer parameters when running, compared with other machine learning methods, such as artificial neural network (ANN) and SVM, whilst achieving high accuracy and good results [107].

3.2.1.4. Level set approach

The active contour method fits closed or open splines to edges or lines in an image. This model acts by minimising an energy, which is defined partially by the image and by the spline's shape, length and smoothness. Minimisation is performed explicitly in the image energy and implicitly in the shape energy [108]. Kass, et al. [109] introduced

an active contour method. Osher and Sethian [110] launched the LS approach to calculate and investigate the progression of a contour with incomplete differential equation because of some constraints, such as confronting topological differences and development of boundary indentations. The basic idea of LS is to show the surfaces as the zero level set of a higher dimensional hyper-surface. Using this technique, not only more accurate numerical implementations can be provided but also topological changes can be handled very easily. Primarily, it means that the closed curvatures in a two-dimensional surface are considered as a constant surface of a three-dimensional space. The definition of a smoothing function ϕ (x, y, t) stands for the surface while the set of definitions ϕ (x, y, t) =0 for the curves. Therefore, the progression of a curve can be converted into the progression of a three-dimensional LS function. Given a Level Set function ϕ (x, y, t = 0), which zero LS matches to curve. With the curve as the boundary, the entire surface can be separated into an inner region and an outer section of the curve. Set a Signed Distance Function (SDF) on the surface:

$$\phi (x, y, t = 0) = d \quad (1)$$

Where, the value of d is the shortest distance between the point of x on the surface and the curve. In the whole evolutional process of the curve, its points will fit into the following formula:

$$\phi (x, y, t) = 0 \quad (2)$$

The common movement formula of LS is:

$$\Phi_1 + \left| F \nabla_{\phi} \right| = 0 \quad (3)$$

F is the speed function, which is a function related to evolving surface features (e.g. curvature, normal direction, etc.) and image features (e.g. gray, gradient). When applied into image segmentation, the design of F depends on the information of image and the ideal value is zero on the edge of the target. LS method usually shows a large influence in solving the obstacles of corner point constructing, curve breaking and combining because of its stability and irrelevancy with topology. Consequently, it is applied in a broad area. Nevertheless, there are some drawbacks to this method. Following the edge-stopping function depends on the image gradient, only objects with edges defined by gradient can be segmented. Other drawback is that in practice, which the curve may eventually pass through object boundaries due to of edge-stopping function is never exactly zero at the edges.

The forward progression of a border is persuaded by the LS approach by using a speed function, which is common to the boundary curve [108]. The problem of extracting roads is regarded as a border transition issue inside the framework of the LS method. As long as the speed function is more than zero, the LS method will spread. At the borders of the right road edge, the speed function must be higher than zero for road extraction. Therefore, to move the zero-level curvature towards the object borders, an outer energy is specified. Given that the LS framework provides borders and automatic topological differences, the LS approach has been applied widely for several purposes, including border improvement. These kinds of features prepare a rational option for extracting roads for the LS framework, as the border of extracted roads might (1) disintegrate because of car appearance, (2) occupy pointed corners because of junctions and (3) adjust its topology arbitrarily (e.g., roads can be mixed anywhere along the road). Consequently, the road extraction query is set as a border evolving difficulty into the framework of LS.

3.2.2. Integrated technique of segmentation and classification methods with connected components analysis

This work proposed an integrated method combining segmentation and classification methods with connected components analysis to extract road class from orthophoto images [111]. The proposed technique is threefold. First, multiresolution segmentation was performed to divide the images into segments based on their spectral values. A total of 567 segments were selected as labeling data for training classification methods based on the segmented images randomly. Then, three main classification approaches, namely, SVM, DT, and KNN, were applied to the segmented image and trained based on sampling data to classify the image into two principal classes: road and non-road class. Finally, connected components analysis and morphological operations were performed to group the pixels together in terms of similar connected components and delete holes and noises to improve the accuracy of the proposed road extraction method. Figure 3.4 illustrates the flowchart of the suggested method along with the entire process for road part extraction from orthophoto images.

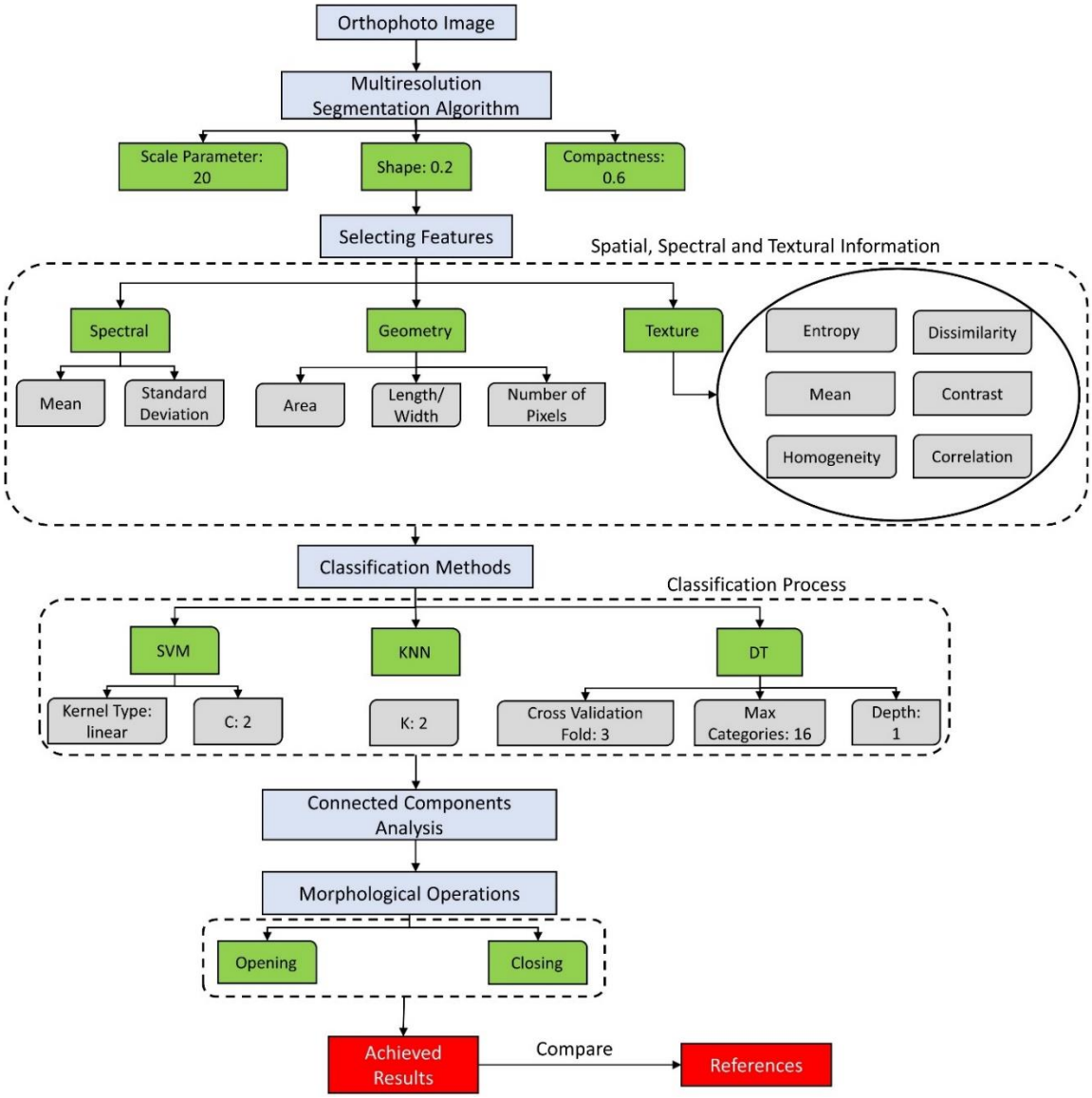

Figure 3.4. Flowchart of the proposed road extraction method from orthophoto images.

3.2.2.1. Orthophoto data and geometric correction

Orthophoto images obtained from the state of Selangor in Peninsular Malaysia with spatial resolution of 7 cm are utilized in this paper (Figure 3.5). An Optech Airborne Laser Terrain Mapper 3100 instrument in an airborne laser scanning of light detection and ranging (LiDAR) system was used to collect orthophotos from the specific area on November 2, 2015. A LiDAR system basically includes a specific GPS (global positioning system)

receptor, a scanner, and a laser. The most regularly utilized platforms for collecting LiDAR data over large regions are helicopters and airplanes. Laser scanning systems are classified as topographic and bathymetric. Topographic LiDAR maps the land based on a near-infrared laser, whereas bathymetric LiDAR measures seafloor and riverbed elevation and maps land based on water-penetrating green light [112]. The flight height for data collection was 1510 m in a bright sky. The geometric calibration of the orthophoto images was performed to eliminate geometric error and designate single pixels in their appropriate planimetric (x, y) map positions [113]. Subsequently, several well-distributed ground control points in the entire image were selected, and then the least square technique was performed to determine the coefficient. Finally, polynomial equations were formulated to determine the root mean shift error between the X, Y of reference, and the adjusted coordinates.

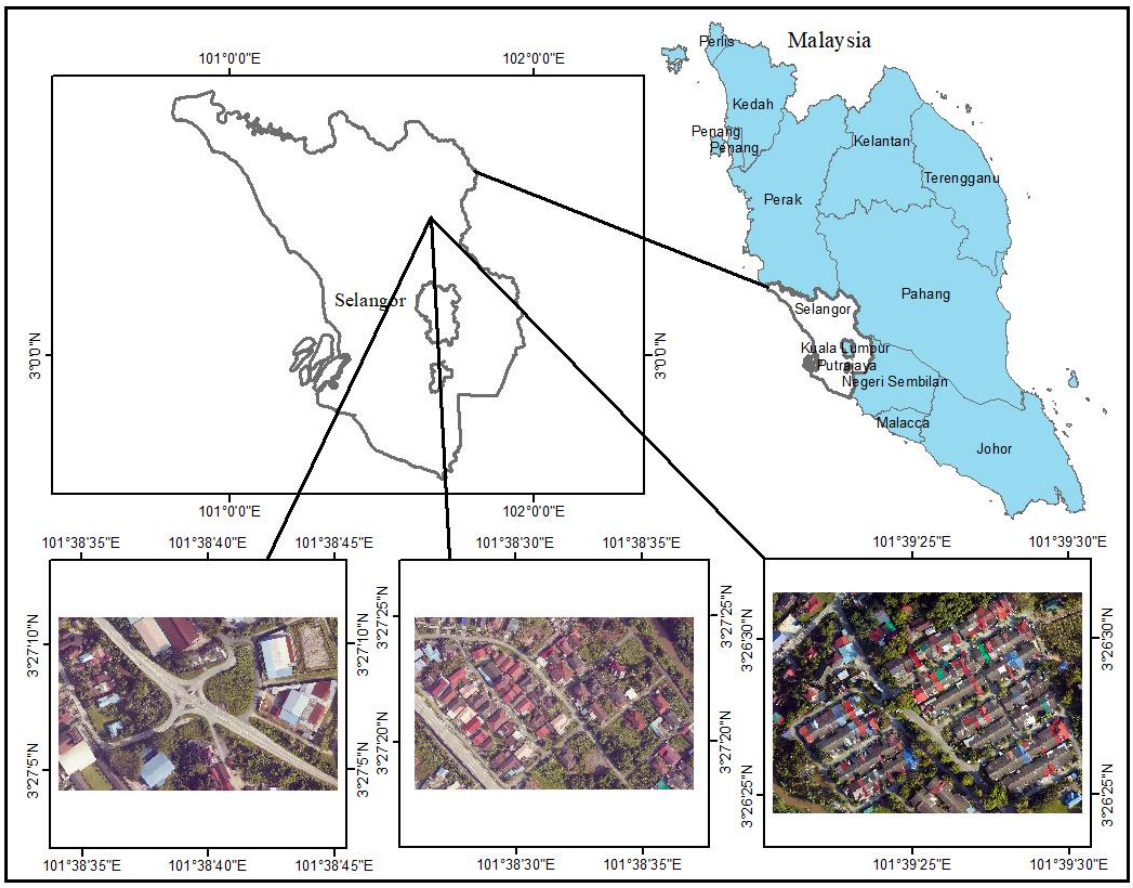

Figure 3.5. Orthophoto images showing the location of the study area.

3.2.2.2. Segmentation process

Image segmentation is a crucial step because it will produce the primary entities for the subsequent processes. The quality of image segmentation has a notable impact on the succeeding operations, making it a crucial yet challenging aspect of OBIA [114]. The algorithms for image segmentation can be divided into four main categories: edge-based, pixel-based, region-based, and mixture methods. The multiresolution segmentation technique is applied in this study for image segmentation [115]. The scale, shape, and compactness parameters for the proposed segmentation method were set to 20, 0.2, and 0.6, respectively, to obtain high accuracy in the classification process. The proposed segmentation method is a region-based method, which reduces the non-homogeneous segments using spectral and shape characteristics [116]. In this method, each pixel of the image is considered as an object. Then, using a fusion factor, objects were joined together to make a large one during a repetitive process. Equation 4 shows the fusion factor, which demonstrates the cost of fitting [115].

$$f = W_{color}\, h_{color} + W_{shape}\, h_{shape} \qquad (4)$$

where h_{shape} is the difference in the shape dissimilarity, h_{color} is the difference in the spectral dissimilarity, W_{shape} is the weight of shape dissimilarity, and W_{color} is the weight of spectral heterogeneity. Furthermore, $W_{color} + W_{shape} = 1$. Equation 5 defines the difference between two objects on the basis of spectral heterogeneity in a multispectral image with B band.

$$h_{color} = \sum_{b=1}^{B} W_b \{ n_m \sigma_{b,m} - \left(n_1 \sigma_{b,1} + n_2 \sigma_{b,2} \right) \} \qquad (5)$$

where n is the number of pixels in every object; σ is the standard deviation of spectral values; indexes 1, 2, and m represent the first, second, and the combined object, respectively; and W_b is the band weight. Smoothness and compactness dissimilarity represent the

difference between the shape heterogeneity of two objects [117]. The difference in shape dissimilarity is expressed by Equation 6. W_{comp} and W_{smooth} are the compactness and smoothness dissimilarities, respectively.

$$h_{shape} = W_{smooth}\left\{n_m \frac{\ell_m}{p_m} - \left(n_1 \frac{\ell_1}{p_1} + n_2 \frac{\ell_2}{p_2}\right)\right\} + W_{comp}\left\{\ell_m \sqrt{n_m} - \left(\ell_1 \sqrt{n_1} + \ell_2 \sqrt{n_2}\right)\right\} \quad (6)$$

where p shows the minimum bounding box perimeter of the object, and ℓ represents the genuine length of the object. $W_{smooth} + W_{comp} = 1$.

3.2.2.3. Selecting features

In this work, OBIA, which considers not only spectral information but also spatial and textural features, was applied to deal with color sensitivity and enhance the efficiency of the suggested road extraction approach. Pixels in the image are first grouped into objects on the basis of either spectral correlation or an outer parameter, such as ownership, soil, or geological unit in the OBIA [118]. The parameter values, such as standard deviation and mean, were considered for each band in the image for the spectral values. The different shapes and elongation of road objects facilitated the easy identification of the proposed method. Geometric features (e.g., length/width, area, and number of pixels) were also considered to ease the classification process. Finally, for the textural values, contrast, entropy, dissimilarity, homogeneity, and correlation values were considered. These features are generally applied to alleviate the classification process and improve the efficiency of road extraction approaches. These features were fed into the classifiers as a training part to accurately classify the image into the road and non-road sections.

3.2.2.4. Classification process

After image segmentation, classifiers, such as SVM, KNN, and DT, were selected to categorize the orthophoto images into two principal classes: road and non-road. This section presents individual discussions of the above classifiers.

3.2.2.4.1. SVM classifier

SVM, which is one of the supervised machine learning approaches, exhibited ample ability in image classification compared with that of the traditional techniques, such as neural networks [119]. The SVM classifier is a linear classification approach that creates a hyperplane to separate data. The process of separating data into classes is followed by identifying the best hyperplane and maximum margin. SVM transforms data according to the predesignated sections in a novel space, wherein data can be detached and classified linearly. Then, a linear equation that provides a maximum margin between two classes is formulated by finding a support line in multi-dimensional space using SVM [11]. The practical application of the SVM method depends on the hypothetical maximum margin classifier. Given that hyperplane is a line separating the input variable space, a hyperplane in the SVM classifier detaches points from the input variable space based on their class (0 or 1). All the input points can be completely split by this line into a two-dimension space (Equation 7).

$$B_0 + \left(B_1 \times X_1\right) + \left(B_2 \times X_2\right) = 0 \quad (7)$$

where X_1 and X_2 are the input variable, B_0 is set up by the learning algorithm, and B_1 and B_2 specify the slope of the line. In this study, the kernel type for SVM is considered to be a linear kernel explaining the distance measure or similarity between new data and support vectors. The performance of the SVM method is shown in Figure 3.6.

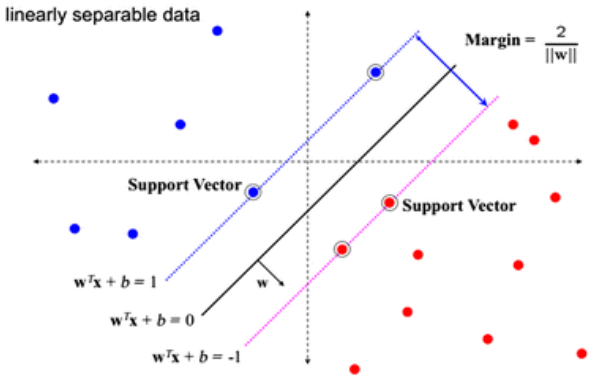

Figure 3.6. SVM performance in categorizing data **[120]**.

The dotted lines in the figure represent corresponding class support vectors, and the data are presented into two categories (red and blue). The long black line is the SVM. Each kind of support vector has a characteristic formula that describes the boundary of each group.

3.2.2.4.2. KNN classifier

One of the non-parametric techniques in machine learning methods is KNN, which has been utilized in statistical applications since the early 1970s [121]. The fundamental concept of KNN is the discovery of a collection of k samples in the calibration dataset nearest to uncertain samples based on distance functions. By evaluating the average of the response variables (e.g., attributes of KNN class), the class of uncertain samples is specified from these k samples [122]. Therefore, k is the key tuning parameter of KNN and plays a crucial role in ensuring the efficiency of KNN in image classification. The bootstrap process is used to identify the k parameter [123]. Different k values from 1 to 10 were inspected in this study to find the ideal k value from all the training datasets, which finally yielded 2.

3.2.2.4.3. DT algorithm

Regarding the dispensation of data, the DT method can be executed without any previous statistical presumptions because it is a non-parametric classifier. The basic structure of the

DT algorithm has three main parts, which include one root node, numerous interior nodes, and a collection of final nodes [124]. The data are recessively broken down into a DT based on the assigned classification structure. Using a breaking test of the form $x_i > c$ for univariate or $\sum_{i}^{n} a_i x_i \leq c$ for multivariate decision trees, a decision rule necessary at every node can be performed. Where c is the decision threshold, a is the linear coefficient vector, n is the chosen feature, and x_i presents the evaluation vectors. Compared with traditional methods, such as the minimum-distance-to-means approach, the DT method has high precision. However, several variables, such as decision threshold, boosting, and pruning approaches, can affect the efficiency of DT in classification [125]. Some parameters, such as max categories, cross-validation fold, and depth, are set to 16, 3, and 1, respectively, for the DT method to achieve optimal results.

3.2.2.5. Connected component analysis and morphological operations

After applying the classification methods and obtaining the results, connected components labeling was performed to extract road sections. Image pixels were grouped into components using connected components analysis on the basis of pixel connectivity, wherein all pixels in the connected component have the same pixel intensity values and are labeled with color or gray level based on each component [126]. The image can be partitioned into segments using these connected components. Morphological operators can be used to extract connected components. Analyzing connected components can be very useful for several applications, such as line detection and road extraction [103].

The trivial operation was applied to extract connected component based on some criteria. Assume that P(i) is the connected component, P is the image, and T is the length of the main axis. The trivial opening can then be expressed as follows:

$$R_0 = \{P | Long\ axis\ of\ \ minimum\ ellipse\ enclosing\ P(i) \geq T\} \quad (8)$$

where R_0 is the connected component. According to the T, trivial operation is utilized for suitable connected components extraction. The entire region of connected components is preserved if that component satisfied condition T and is removed otherwise. After extracting the required connected components in terms of road section, common morphological operations, such as opening and erosion operations, were used to fill gaps, remove noises, delete non-road parts from the image, and improve the accuracy of the extracted road class using the proposed methods [127].

3.3. State-of-the-art DCNN models for road surface extraction (Objective 1)

Based on the first objective, several new robust DCNN models such as VNet, GAN+MUNet, MCG-UNet, and BCD-UNet were implemented for road surface segmentation from various HRSI data such as Google Earth and Aerial images, which the implementation of the approaches is detailed in this part. In the designed approaches, some additional modules or loss functions were also used to improve the performance of the models, solve the issues of pre-existing ML and DL methods in road extraction, and produce high-resolution road segmentation maps even under complicated backgrounds.

3.3.1. Generative Adversarial Network (GAN) and modified UNet model (MUNet)

This work lied in proposing a GAN with a modified UNet generative model (GAN+MUNet) to extract roads from high-resolution aerial imagery [128]. Compared to prior GAN-based road extraction approaches such as GAN+FCN proposed by [42], GAN+SegNet presented by [88], Ensemble Wasserstein Generative Adversarial Network (E-WGAN) proposed by [90], Multi-supervised Generative Adversarial Network (MsGAN) performed by [92], and Multi-conditional Generative Adversarial Network

(McGAN) implemented by [94], I introduced the modified UNet model (MUNet) for the generative term to create a high-resolution smooth segmentation map, with high spatial consistency and clear segmentation boundaries. The proposed model did not require high computational time and a large training dataset and still improved performance and addressed the challenges of aforementioned methods for road extraction from remote sensing imagery. Also, the proposed method preserved the edges and structure of roads and generated high-quality road segmentation maps in agreement with ground truth labels.

Figure 3.7 shows the overall methodology for training and evaluating the proposed GAN-based approach for road network extraction organized as four major steps: (i) generation of training and testing samples; (ii) local Laplacian filtering (LLF)-based pre-processing to enhance image quality; (iii) GAN optimization using the training samples, and extraction of the road network from images in the test set using the generator from the optimized GAN; and (iv) performance quantification for the proposed method using common metrics.

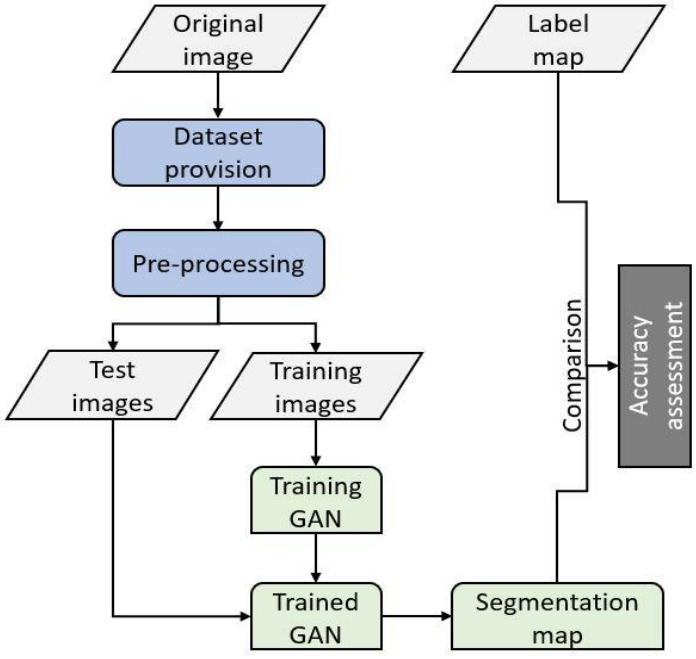

Figure 3.7. Workflow for training and evaluating the proposed GAN-MUNet approach.

3.3.1.1. Pre-processing

As a pre-processing step, I used LLF to enhance the quality of images prior to using them in the proposed model for training/testing. LLF is a nonlinear image filtering framework based on Laplacian pyramids (LP) that enables edge-aware processing using simple local processing operations. The filtered LLF image is obtained by rendering its LP coefficient by coefficient based on locally adaptive processing of the input image [129]. LLF was introduced in [130], where it was verified that this filtering technique can enrich image details without introducing halos or other artifacts and can be effectively used for range compression and tone mapping. With appropriate approximation and parallel implementation, LLF can be significantly speeded up to enable interactive use [129].

3.3.1.2. GAN Framework for semantic segmentation

As illustrated in Figure 3.8, the GAN framework [98] uses two subnetworks: a generator *G* and a discriminator *D*. The generator attempts to generate data representative of the ground truth provided for training, whereas the discriminator attempts to distinguish true ground truth data from data produced by the generator. The two subnetworks are jointly trained in an adversarial game to obtain the min-max operating point where the road maps created by *G* minimize the maximum discrepancy for *D* between the true and generated pairs [131]. Figure 3.9 illustrates the detailed network architecture illustrating the structure of the generator and discriminator. For the generator, I utilized the MUNet model that includes two corresponding arms, a contracting (downsampling) encoder and an expanding (upsampling) decoder, with skip-connections that append every upsampled feature map at the decoder with the corresponding one in the encoder that has the same spatial resolution [132].

The generator subnetwork seeks to learn a map $G : x \rightarrow y$ that produces a binary segmentation map y from the input image x based on the distribution p seen in the training data. The discriminator maps a pair $\{x, y\}$ comprised of an input image and a segmentation map to a value between 1 and 0 indicating the discriminators' estimate of whether y represents a ground truth mask or an estimate from a generator subnetwork.

For road map segmentation, the GAN objective function is then formulated as

$$L_{GAN}(G, D) = E_{x, y \sim p_{data}(x, y)}[\log D(x, y)] + E_{x \sim p_{data}(x)}[\log(1 - D(x, G(x)))] \quad (9)$$

Note that maximization of the objective function aligns with maximization of $D(x, y)$ and minimization of $D(x, G(x))$, which seeks to train the discriminator subnetwork D to make right decision. On the other hand, the generator subnetwork G should generate outputs that are indistinguishable from the true data to hamper the discriminator D from making right decision and should therefore be chosen to minimize the objective function. I defined the objective function as minimax of the objective function in (1) with maximization over choices of D and minimization over choices of G, as the final purpose is to achieve realistic probability outputs from G.

In addition to the GAN objective function, I also used a second binary cross-entropy loss function that is common in segmentation and has also recently been incorporated in a GAN framework for segmentation [132] of retinal images,

$$L_{SEG}(G) = E_{x, y \sim p_{data}(x, y)}[-y . \log G(x) - (1 - y) . \log(1 - G(x))] \quad (10)$$

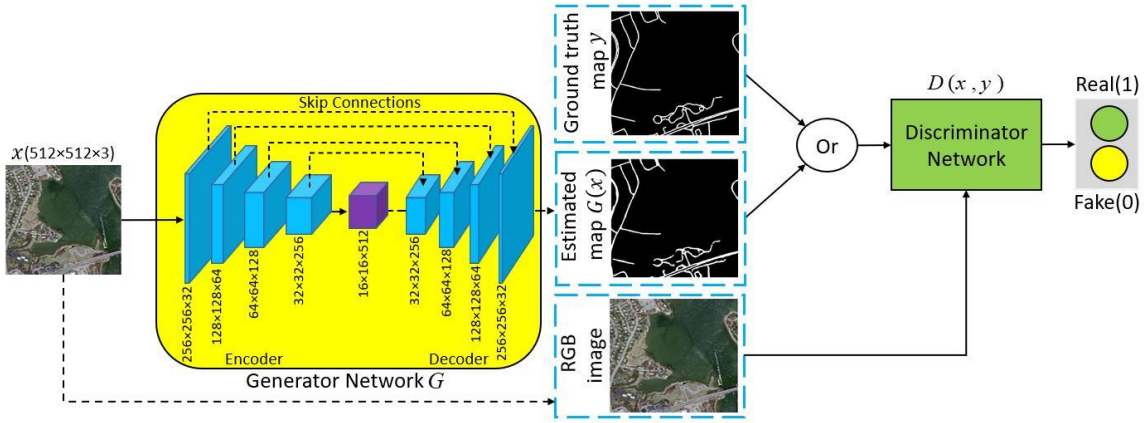

Figure 3.8. GAN training to generate a road segmentation map from an RGB image; the generator network seeks to create a representation that cannot be distinguished from the ground truth image by the discriminator network, which in turn is trained to best distinguish generated samples from real ground truth data.

Combining both the segmentation loss and the GAN objective function, the optimal generator network for road map segmentation is obtained as

$$G^* = \arg\min_G [\max_D L_{GAN}(G, D)] + \lambda L_{SEG}(G) \quad (11)$$

where the impact of the two objective functions can be balanced by the weighting parameter λ. In practice, I used the Prop-GAN architecture to train from a low to a high resolution on the ground truth segmentation maps. During training, I incrementally added layers to the generator and discriminator to increase the spatial resolution of the generated segmentation maps. Per pixel semantic class labels is the output of the generator. I first created per-pixel likelihood scores of belonging to every semantic label, and then sampled every semantic class per pixel to synthesize segmentation layouts. Then, I used tanh function on the generator's last layer to calculate the per-pixel probability scores, which resulted in probability maps. The synthesized samples fed to the Prop-GAN discriminator should still have distinct labels, similar to the real samples. As a result, I computed minimax

for both forwards and backwards passes, with the goal of achieving practical probability outputs.

3.3.1.3. Generator and discriminator architecture

The detailed architectures of the generator and discriminator subnetworks used in our work are shown in Figure 3.9. The generator uses the MUNet architecture [133] and it is built from scratch and trained according to our dataset. The upper half corresponds to the contracting encoder arm where resolution decreases and feature depth increases as one proceeds from left to right and the lower half corresponds to the expanding decoder arm where resolution increases and feature depth decreases as one proceeds from right to left. The feature map size for the downscaling and upscaling layers of the generator is listed in Table 3.1.

Table 3.1. The detailed architecture of the generator subnetwork including downscaling and upscaling parts.

Instruction	Layers	Kernel Size	Feature Map Size
Input	Input	-	(Batch size, 512,512,3)
Downscale	Conv2D	4×4	(Batch size, 256,256,32)
	Conv2D		(Batch size, 128,128,64)
	Conv2D		(Batch size, 64,64,128)
	Conv2D		(Batch size, 32,32,256)
	Conv2D		(Batch size, 16,16,512)
Upscale	Deconv2D	4×4	(Batch size, 32,32,256)
	Deconv2D		(Batch size, 64,64,128)
	Deconv2D		(Batch size, 128,128,64)
	Deconv2D		(Batch size, 256,256,32)
Output	Output	-	(Batch size, 512,512,3)

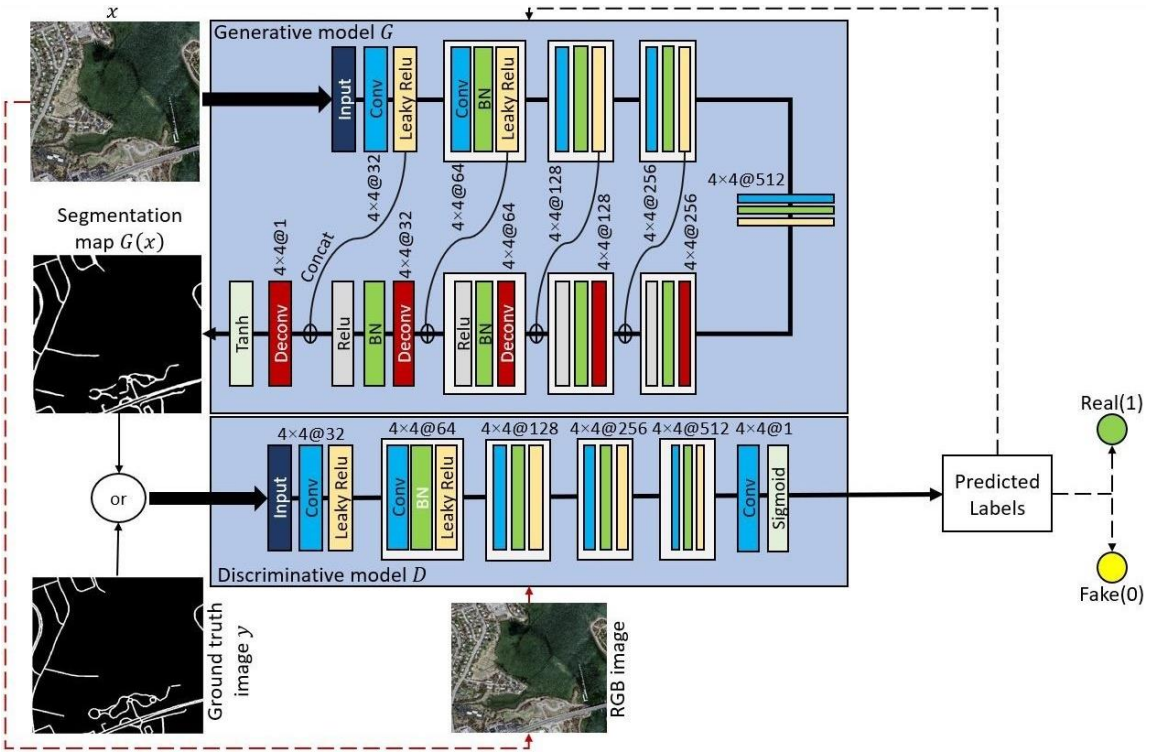

Figure 3.9. Detailed structures of generative and discriminative networks comprising the proposed GAN for road network segmentation.

The skip connections characteristic of the UNet architecture [95] connect corresponding resolution layers between the encoder and decoder arms allowing for the insertion of details in the upsampling for each resolution expansion. Compared to UNet, the changes in the MUNet architecture include: the introduction of batch normalization, the use of the ReLU activation function in the decoder and Leaky ReLU for the encoder, and elimination of the pooling layer. Specifically, as shown in Figure 3.9, in the contracting arm of the MUNet, I used convolutional layers with a kernel size of 4×4 followed by batch normalization and Leaky ReLU activation function, and in the expanding arm, I used deconvolution layers with a 4x4 stride followed by batch normalization and ReLU activation function. Finally, for mapping every 32-component feature vector to the desired number of classes (road and non-road), I used the final deconvolution layer with the 4x4 stride and a tanh activation

function [31] for mapping predicted values to classification probabilities. The ReLU and Leaky ReLU activation functions are, respectively, defined as

$$ReLU(k) = \begin{cases} 0 & if\ k <= 0 \\ k & if\ k > 0 \end{cases} \quad (12)$$

$$LReLU(k) = \begin{cases} k & if\ k > 0 \\ \alpha k & if\ k <= 0 \end{cases} \quad (13)$$

where α is a small constant between 0.1 and 0.3 [134].

The discriminator architecture used in our work is also shown in Figure 3.9. The ground truth data and segmentation results are fed into the discriminative term to find whether the generator output is fake (0) or real (1). The discriminator uses a fully convolutional architecture with 17 layers, with a structure that mimics the encoder arm of the generator comprising of convolutional layers with a kernel size of 4×4 and stride of 2×2 followed by batch normalization and Leaky ReLU activation function. The final layer used a sigmoid function to produce a value between 0 and 1 indicative of the discriminator's assessment of the probability that the presented road segmentation map corresponds to labeled ground truth.

3.3.1.4. Dataset

For our benchmarks, I used the Massachusetts dataset [135], which is the largest existing road dataset. This dataset includes 1,171 aerial images with original spatial dimensions of 1500×1500. For validating the proposed model on the dataset for road extraction, 100 images with complete information and good quality were selected. The original images were divided into eight parts with a size of 512×512 to accommodate computational constraints. Consequently, 761 images were used as the final dataset in the experiments. The dataset was

divided into 733 images for training and validation: and 28 images for testing. Data augmentation techniques, such as horizontal flip, vertical flip, zooming, and rotation, were used to increase dataset size for training of the proposed method.

3.3.1.5. Parameters and implementation

For LLF, the sigma and alpha parameters were set as 0.2 and 0.3, respectively. Training of the GAN network to optimize the loss function was performed using the extensively utilized Adam optimizer [134] with learning rate of 0.001, beta_1 of 0.9 and beta_2 of 0.999. A dropout probability of 0.5 was used during model training to avoid overfitting. The proposed model was trained with batch size 1 for 100 epochs and the trained model was then applied to the test data to extract roads. The extracted labels were compared against the ground truth labels for evaluating the performance. The whole process of the proposed method for road extraction from remotely sensed imagery was implemented on a GPU Nvidia Quadro P5000 with a computing capacity of 6.1 with 2560 shading units, 160 texture mapping units, and 64 render output units (ROPs), and a memory of 16 GB under the framework of Keras with Tensorflow backend.

3.3.2. VNet network and cross-entropy-dice-loss (CEDL)

In this research, I used a novel deep learning-based convolutional network called VNet model with 2D convolutional kernel to extract road networks from two different high-resolution remote sensing imagery such as Massachusetts road dataset (Aerial images) and Ottawa road dataset (Google Earth images) and produced a high-resolution segmentation output [136]. The proposed method trained end-to-end and leverage the power of fully convolutional neural networks to process high-resolution remote sensing imagery. In the suggested VNet network, pooling layers were replaced with

convolutional layers that resulted in a shorter memory footprint throughout the training process. Also, a new objective loss function on the basis of cross entropy and dice loss (CEDL) was used to (i) combine local information (CE) and global information (DL), (ii) diminish the influence of class imbalance, and (iii) improve the road segmentation results. In addition, a new non-linearities activation function named parametric rectified linear unit (PRelu) was applied rather than rectified linear unit (ReLU) function to enhance accuracy at a negligible additional computational cost and its performance is better than ReLU for large-scale data processing. The overall methodology of the suggested VNet-based method for road extraction from high-resolution remote sensing imagery is shown in Figure 3.10. At the first step, two different road datasets called Massachusetts and Ottawa were used to prepare the training, validation and test images for training and evaluating the proposed method. Then, the architecture of the proposed VNet approach along with the new CEDL function was defined. Following this, the training samples were used to train the VNet model and then test images were used to extract road networks and evaluate the performance of proposed methods.

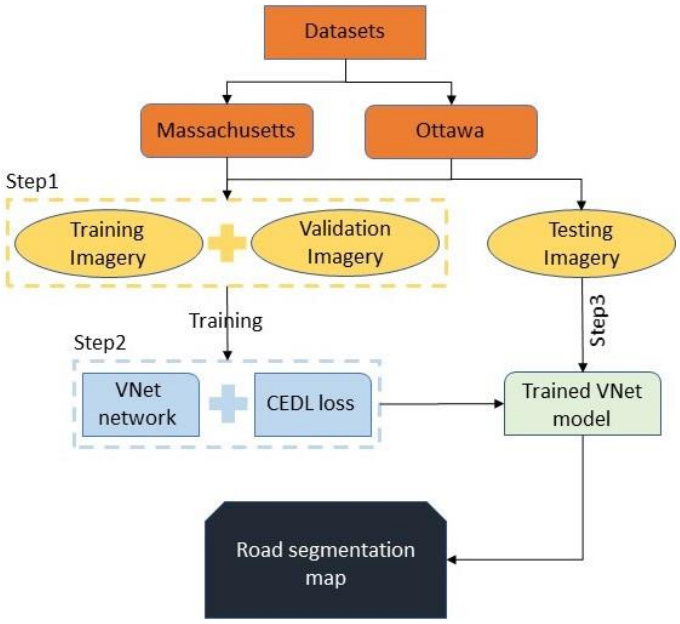

Figure 3.10. The overall framework of the proposed VNet network for road extraction.

3.3.2.1. VNet architecture

A schematic presentation of the proposed VNet model is shown in Figure 3.11. The proposed VNet approach is comprised of two main parts: the left section that includes a compression path and the right part that decompresses the input till its initial size is attained. Convolutions with appropriate padding are all performed, aiming to both exploit features from the input and decrease its resolution using proper stride at the end of each stage. The architecture of the proposed VNet network is similar to the widely used UNet [95] model, but with some differences.

The left part of the VNet architecture is split into various phases operating at different resolutions. One to three convolution layers exist in each stage. A residual function can be learned in each phase as I formulate each stage similar to the method illustrated in [40]. In other words, in order to enable learning a residual function, the input of every phase is processed through the non-linearities and utilized in the convolution layers and then appended to the output of the final convolution layer of that phase. This network guarantees

convergence in comparison with non-residual learning architecture such as UNet. Also, in each stage, the convolutional layers with the size of 5×5 is performed. The convolution process is expressed using Equation (14).

$$x_k(ii,jj) = \sum_{n=1}^{N}\left\{\sum_{p=0}^{W_f-1}\sum_{q=0}^{h_f-1} x_n(i\cdot s_f+p, j\cdot s_f+q)\cdot h_k(p,q)\right\} + b_k \quad (14)$$

where b_k is the bias parameter of the k-th filter that is shared among all locations (p,q), s_f is the sampling stride, h_k (p,q) is the weight value at (p,q) of the k-th filter, x_k (ii,jj) is the pixel value at (ii,jj) in the k-th filter size of the input map, and x_n (ii,jj) is the pixel value at (ii,jj) in the n-th channel of an input feature map.

The resolution of data is reduced as it proceeds through various phases along the compression path and this is implemented using convolutional layer with size of 2×2 and stride 2. The size of the resulting feature maps is halved as the second operation considers only non-overlapping 2×2 patches and extract features [137]. I replaced max-pooling layers with convolutional layers in our method that serves as the same objective as pooling layers incited by [137]. I used these convolutional operations for doubling the number of feature maps. This is due to the formulation of the method as a residual framework, and since the number of feature channels double at every phase of the VNet compression path. Using convolutional layers instead of pooling layers results to the network to have a smaller memory footprint throughout the training. Pooling operation did down-sampling the features, but I wanted to keep all the features as possible. Therefore, the advantages of using convolutional layers rather than pooling layers in our proposed method is that to process inputs in higher resolution and detect fine-details as well as capture more

contextual information by broadening the view of input data [138]. Decreasing the size of input and increasing the receptive field of the features being assessed in the following layers ofnetwork is operated by down-sampling step. In the left section of the network, the number of features that are assessed by each phase is two times higher than one of the prior layer. For activation function (Equation 15), there are several functions such as $\tanh$, rectified function and so on that can be used.

$$Z(x_k(ii,jj)) = f\left(\sum_{k=1}^{k} x_k(ii,jj)\cdot w_k + b_k\right) \Leftrightarrow Z = f(X \cdot W + b) \quad (15)$$

where w is a weight vector, b is a bias vector, and x_k (ii,jj) is used as input to the activation function of the neural network that is the output of convolution operation.

In this work, a non-linearity function called PRelu (Equation 16) proposed by [139] was implemented throughout the model. PRelu function can be optimized concurrently with other layers and can be trained using back-propagation. This function enhances accuracy at a negligible additional computational cost, and adaptively learns the parameters of the rectifiers. For the large-scale image classification, the authors reported that its performance is better than ReLU function.

$$y_i = \begin{cases} x_i & if\ x_i \geq 0 \\ \dfrac{x_i}{a_i} & if\ x_i \leq 0, \end{cases} \quad (16)$$

where a_i defines as a fixed parameter in the range of $(1,+\infty)$ and it is learned via back-propagation in the training.

In order to assemble and gather the essential information to output of two channels segmentation map, the right part of the network expands the spatial support of the lower resolution feature maps and extract features. The last convolutional layer with the kernel

size of 1×1 produces the output with a similar size of the input data and computes the two features maps. Also, I used sigmoid function in this layer that converts these two feature maps into probabilistic segmentation maps of the background and foreground areas. Compared to the UNet architecture, after every phase of the right part of the network, a deconvolutional operation was followed by one to three convolution layers. This includes half the number of 5×5 kernels that were applied in the past layer and was used to increase the size of input. I also resorted to learn residual functions in the convolutional phases of the right part of the network similar to the left portion. Next, the extracted features were forwarded from early phases of the left portion of the network to the right section similar to [95] that is shown in Figure 3.11 by horizontal links. Subsequently, I improved the quality of last contour prediction in this way by gathering fine-grained details that would have been otherwise missed during the compression stage. It is also observed that the convergence time of the model has been improved by these connections.

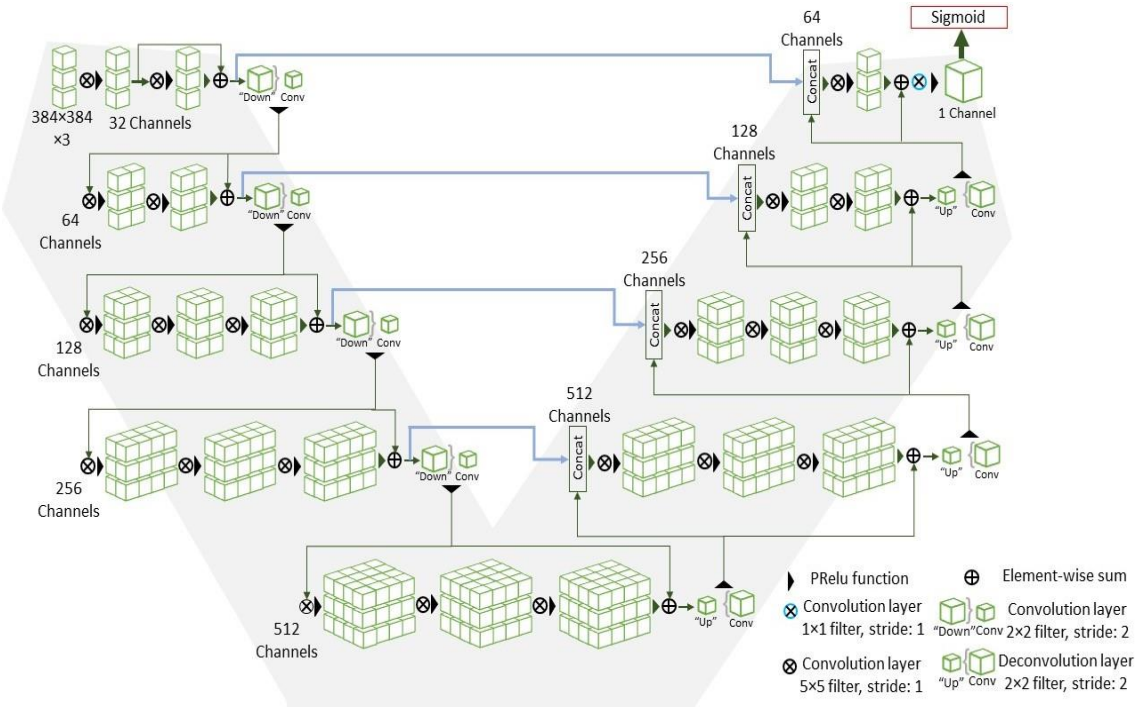

Figure 3.11. The architecture of VNet network including two mains expansive (right side) and contracting parts (left side).

3.3.2.2. Loss function

In feature semantic segmentation from high-resolution remote sensing images such as road networks segmentation, it is common that road pixels occupy just a pretty tiny area of the image. This usually can be the cause of confining learning process in the regional minima of the loss function, generating a model whose anticipations are heavily prejudiced to the background. Therefore, the foreground area is usually only partly identified or even missed. To tackle this problem, multiple prior methods on the basis of re-weighting the samples where background areas are assigned less significance than foreground areas ones through learning such as weighted cross-entropy [140] and dice loss [141] have been presented. Equation 17 defines the dice loss (DL) between two binary classes whose values are ranging between 0 and 1.

$$DL = \frac{2\sum_{i}^{N} p_i g_i}{\sum_{i}^{N} p_i^2 + \sum_{i}^{N} g_i^2} \quad (17)$$

where $g_i \in G$ is the ground truth pixels, $p_i \in P$ is the predicted binary pixels and N defines as total pixels. The dice formulation can be modified with producing the gradient measured regarding the j-th pixels of the anticipation (Equation 18). As a result, for establishing the right balance between background pixels and foreground ones, I do not require to allocate weights to the various classes samples using this formulation.

$$\frac{\partial D}{\partial p_j} = 2\left[\frac{g_j(\sum_{i}^{N} p_i^2 + \sum_{i}^{N} g_i^2) - 2p_j(\sum_{i}^{N} p_i g_i)}{(\sum_{i}^{N} p_i^2 + \sum_{i}^{N} g_i^2)^2}\right] \quad (18)$$

In this study, since we have the same issue of imbalance classes such as road pixels (foreground) and non-road pixels (background) I introduced a new dual objective loss

function (CEDL) that incorporates both cross-entropy loss function (CE) and dice coefficient (DL) to reduce the influence of class imbalance issues. Equation 19 defines the new loss function (L) that is a mixture of CE and DL. Note that DL returns a scalar while CE returns a tensor of every image in the batch. In other words, I mixed global information (DL) and local information (CE) to extract road network more accurately.

$$CEDL = CE(p_i, g_i) + DL(p_i, g_i) \quad (19)$$

3.3.2.3. Datasets

Massachusetts road dataset: This dataset [135] contains 1171 aerial imagery with the primary spatial resolution of 0.5 m and dimension of 1500×1500. The target maps were usually generated by rasterizing road centerlines obtained from the OpenStreetMap project Due to computational restrictions, I split the main images into smaller parts with the size of 384×384 that include good quality and complete information. The dataset that I used for validating the proposed method for road extraction includes 4135 images that I divided it into 215 images for test, 120 images for validation and 3800 images for training. For expanding the dataset, some data augmentation methods such as vertical flip, rotation and horizontal flip are also used. Moreover, for overcoming the over-fitting issue, I added a dropout of 0.5 to deeper convolutional layers. Some examples in the Massachusetts road dataset are illustrated in Figure 3.12.

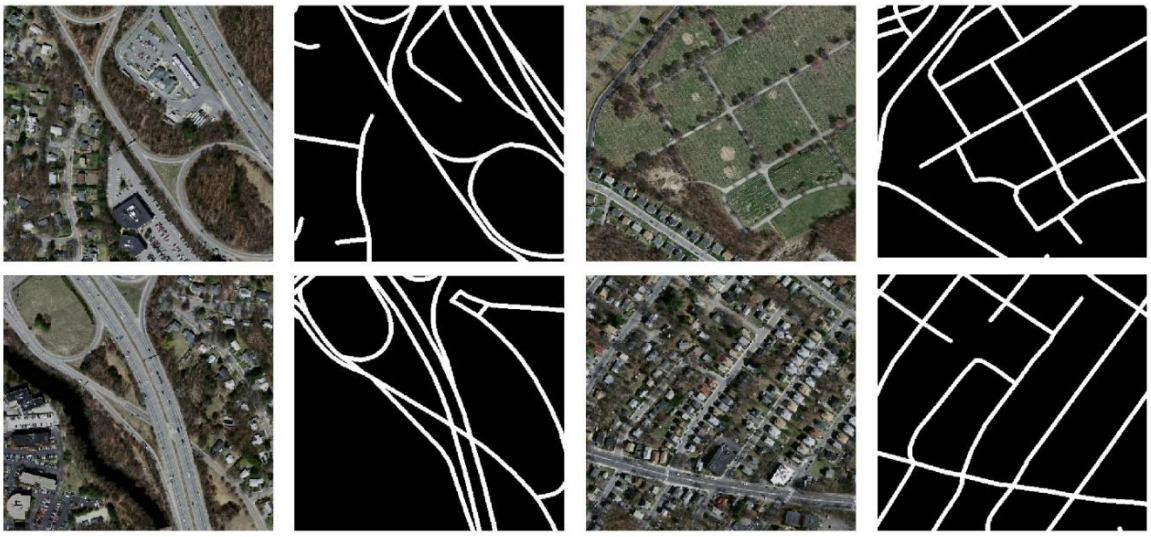

Figure 3.12. Some sample imagery in Massachusetts road dataset. The main imagery and corresponding ground truth maps are illustrated in the first and second columns, respectively.

Ottawa road dataset: This dataset includes Google Earth images with spatial resolution of 0.21 m that encompasses 21 typical urban regions covering about 8 km^2 of Ottawa, Canada [46]. The road labels were annotated manually and compared to other datasets such as [45], [47] and [10]. This dataset is more challenging and comprehensive as it covers different urban areas with different complexity. In this study, I divided the dataset into images with a size of 384×384 to validate the proposed method. The final dataset contained 1005 images that were split into 899 training images, 62 validation images and 44 test images. Also, I used data augmentation techniques like flipping horizontally and vertically and rotating the images to expand the dataset. Some examples in the Ottawa road dataset are depicted in Figure 3.13. The whole process of applying the proposed model for road network extraction from high-resolution remote sensing imagery is functioned on a GPU Nvidia Quadro RTX 6000 with a computation capacity of 7.5 and a memory of 24 GB under the framework of Keras with Tensorflow backend.

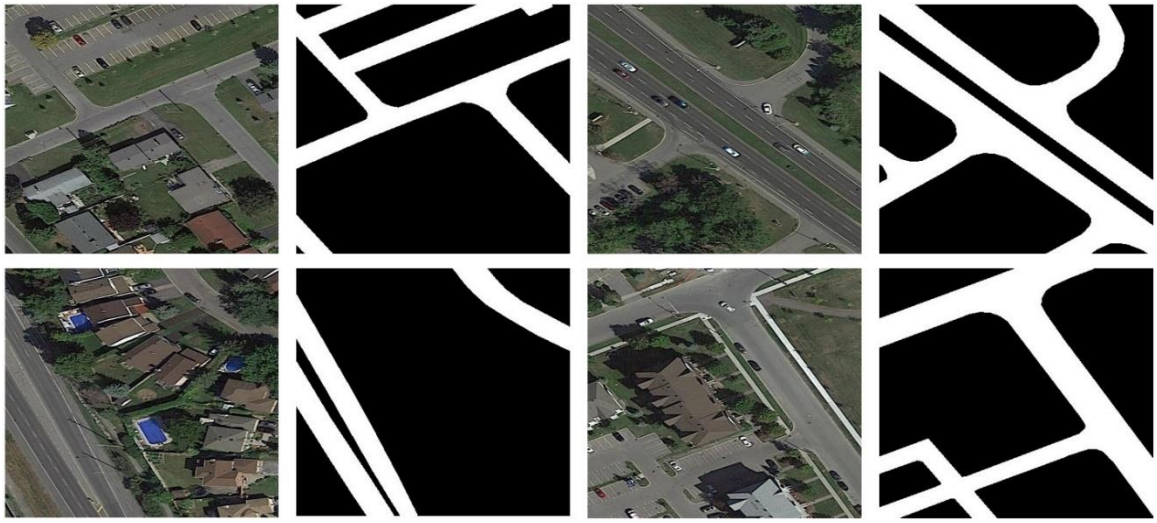

Figure 3.13. Some sample imageries in Ottawa road dataset. The main imagery and corresponding ground truth maps are illustrated in the first and second columns, respectively.

3.3.3. Multi-level context gating UNet (MCG-UNet) and bi-directional ConvLSTM UNet (BCL-UNet) models

In this work, I implemented two end-to-end frameworks, the MCG-UNet and BCL-UNet models [142], which are an extension of the UNet model, and which have all the advantages of UNet, dense convolution (DC) mechanism, bi-directional ConvLSTM (BConvLSTM), and squeeze and excitation (SE) to identify road object from aerial imagery. The BCL-UNet model only takes the advantages of BConvLSTM, whereas the MCG-UNet model also takes the benefit of SE function and DC. The densely connected convolutions (DC) were used to increase feature reuse, enhance feature propagation, and assist the model to learn more various features. The BConvLSTM module was applied in the skip connections to learn more discriminative information by combining features from encoding and decoding paths. The SE function was employed in the expanding path to consider the interdependencies between feature channels and extract more valuable information. A BAL loss function was also used to focus on hard semantic segmentation regions, such as overlapped areas of objects and complex regions, to magnify the loss at the edges and

improve the model's performance. I used this strategy to improve the border of semantic features and make them more appropriate for actual road forms. By adding these modules to the models and using BAL loss, the model's performance for road segmentation was improved. The overall methodology of the presented techniques is depicted in Figure 3.14. The proposed framework includes three main steps. (i) Dataset preparation step was firstly applied to produce test imagery and training and validation imagery for road objects. (ii) The presented networks were then trained on the basis of training imagery and validated based on validation imagery. After that, the trained frameworks were applied on the test images to generate the road segmentation maps. (iii) Common measurements factors were finally used to assess the model's performance.

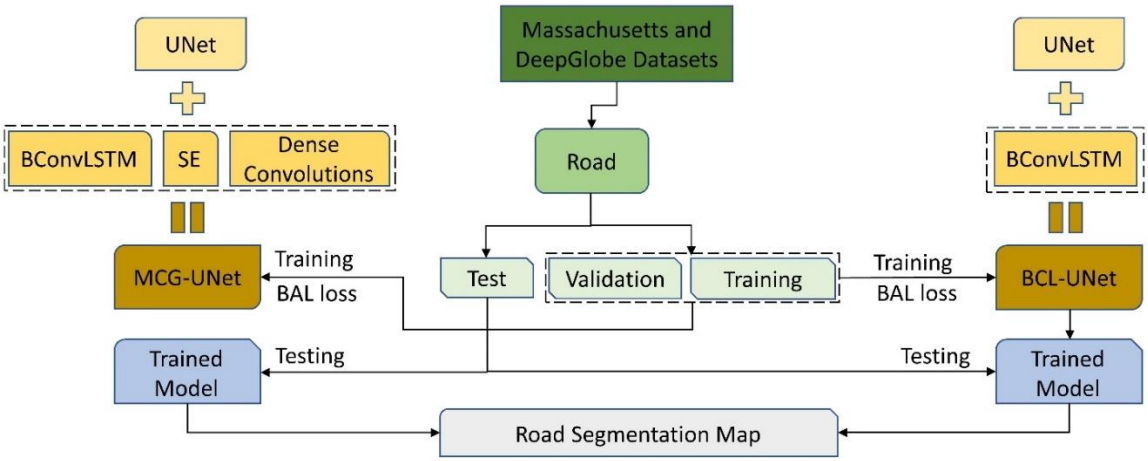

Figure 3.14. Overall flow of the offered BCL-UNet and MCG-UNet frameworks for road surface segmentation.

3.3.3.1. BCL-UNet and MCG-UNet architectures

The proposed BCL-UNet and MCG-UNet models are inspired by dense convolutions [143], SE [144], BConvLSTM [145], and UNet [95]. The architectures of the UNet and the proposed BCL-UNet and MCG-UNet are shown in Figures 3.15–3.17, respectively. The widely used UNet model comprises the encoding and decoding paths. In the contracting

path, hierarchically semantic features are extracted from the input data to take context information. A huge dataset is required for training a complicated network with a massive number of parameters [95]. However, deep learning-based techniques are mainly localized on a particular task, and collecting a massive volume of labeled data is very challenging [146]. Therefore, I used the concept of transfer learning [146] by employing a pretrained convolutional network of VGG family as the encoder to deal with the isolated learning paradigm, leverage knowledge from pre-trained networks, and improve the performance of the UNet. To make utilizing pre-trained networks feasible, the encoding path of the proposed model was designed similar to the first four VGG-16 layers. In the first two layers, I used two 3×3 convolutional layers chased by a 2×2 max pooling layer and ReLU function. In the third layer, I used three convolutional layers with a similar kernel size chased by a similar ReLU function and max pooling layer. At every stage, the quantity of feature maps was doubled. In the final step of the contracting path, the main UNet model included a series of convolutional layers [147]. This allowed the networks to learn various sorts of features. However, in the successive convolutions, the model might learn excess features. To moderate this issue, I used the idea of "collective knowledge" by exploiting densely connected convolutions [143] to reutilize the feature maps through the model and improve the model performance. Inspired by this idea, I concatenated feature maps learned from the current layer with feature maps learned from all prior convolutional layers and then forwarded to utilize as the next convolutional layer input.

Using densely connected convolution (DCC) instead of the usual one [143] has some benefits. First, it prompts the model to avoid the risk of vanishing or exploding gradients by getting advantages from all the generated features before it. Furthermore, this idea allows information to flow through the model, in which the representational power of the

networks can then be improved. Moreover, DCC assists the models to learn various collections of feature maps rather than excessive ones. Therefore, I employed DCC in the suggested approaches. One block was introduced as two successive convolutions. There is a sequence of N blocks in the final convolutional layer of the contracting path that are densely connected. The feature map concatenation of all previous convolutional blocks, e.g., $[x_e^1, x_e^2, ..., x_e^{i-1}] \in R^{(i-1)F_l \times W_l \times H_l}$ was considered as an input of the i^{th} $(i \in \{1, ..., N\})$ convolutional block and $x_e^i \in R^{F_l \times W_l \times H_l}$ was considered as its output, where the number and size of feature maps at layer l are defined as $W_l \times H_l$ and F_l, respectively. A sequence of N blocks that are densely connected in the final convolutional layer is presented in Figure 3.18.

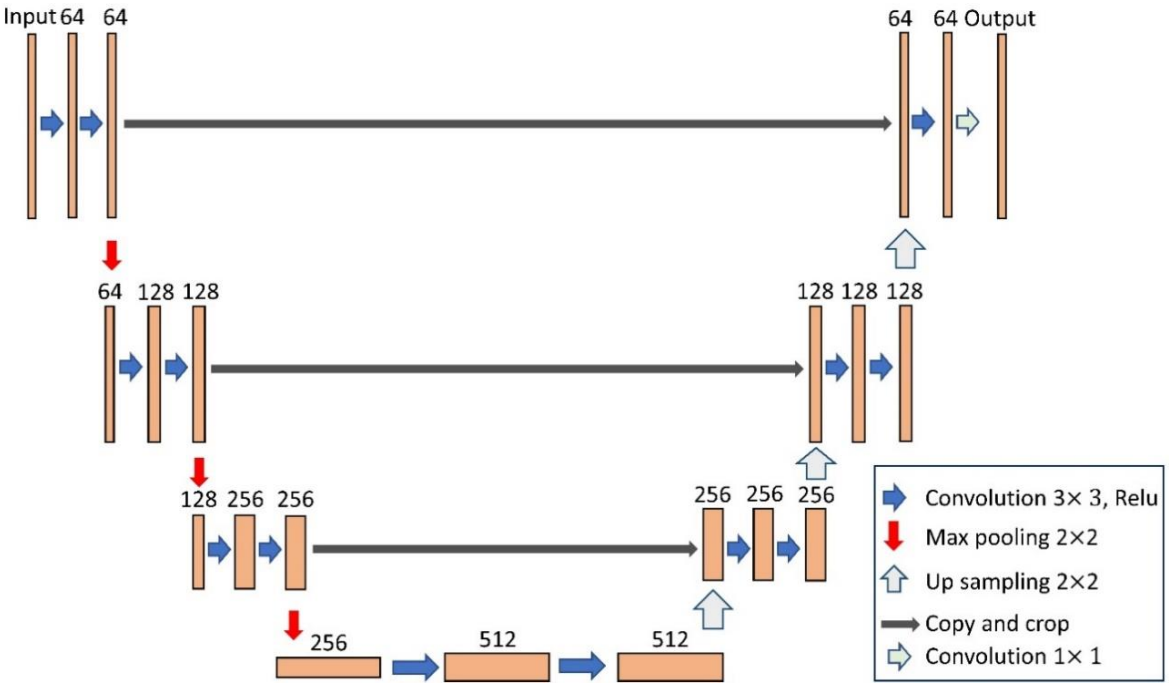

Figure 3.15. UNet model without any dense connections and with BConvLSTM in the skip connections.

In the expansive path, every phase starts with an upsampling layer over the prior layer output. I used two significant modules, namely, BConvLSTM and SE, for the MCG-UNet and BConvLSTM module for BCL-UNet to augment the decoding part of the original UNet and improve the representation power of the models. In the expanding part of the main UNet model, the corresponding feature maps were concatenated with the upsampling function output. For combining these two types of feature maps, I employed BConvLSTM in the proposed frameworks. The BConLSTM output was then fed to a set of functions containing two convolutional modules, one SE function, and another convolutional layer. SE module takes the output of the upsampling layer, which is a collection of feature maps. On the basis of interdependencies between all channels, this block uses a weight for every channel to promote the feature maps to be more instructive. SE also allows the framework to utilize global information to suppress useless features and selectively emphasize informative ones. The SE output was then fed to an upsampling function. Figures 3.19a,b illustrate the structure BConvLSTM in BCL-UNet framework and BConvLSTM with SE modules in MCG-UNet framework, respectively. Presume that $X_d \in R^{F_{l+1} \times W_{l+1} \times H_{l+1}}$ defines a set of exploited feature maps from the prior layer in the expansive part. We have $H_{l+1} = \frac{1}{2} \times H_l$, $W_{l+1} = \frac{1}{2} \times W_l$ and $F_{l+1} = 2 \times F_l$, which we assume as $X_d \in R^{2F \times \frac{W}{2} \times \frac{H}{2}}$ for simplicity. As illustrated in Figures 3.17 and 3.18, the set of feature maps first goes through an upsampling function chased by convolutional layer with size 2×2, in which these functions halve the channel number and double the size of every feature map to produce $X_d^{up} \in R^{F \times W \times H}$. In the decoding part, the size of the feature maps is increased layer-by-layer to achieve the primary size of input data. These feature maps are then converted into prediction maps of the foreground and background parts in the last layer based on the

sigmoid function. The detailed configurations of all approaches, the number of parameters and layers, batch size, and input shape are shown in Table 3.2. In the following, the batch normalization (BN), BConvLSTM, and SE modules are described.

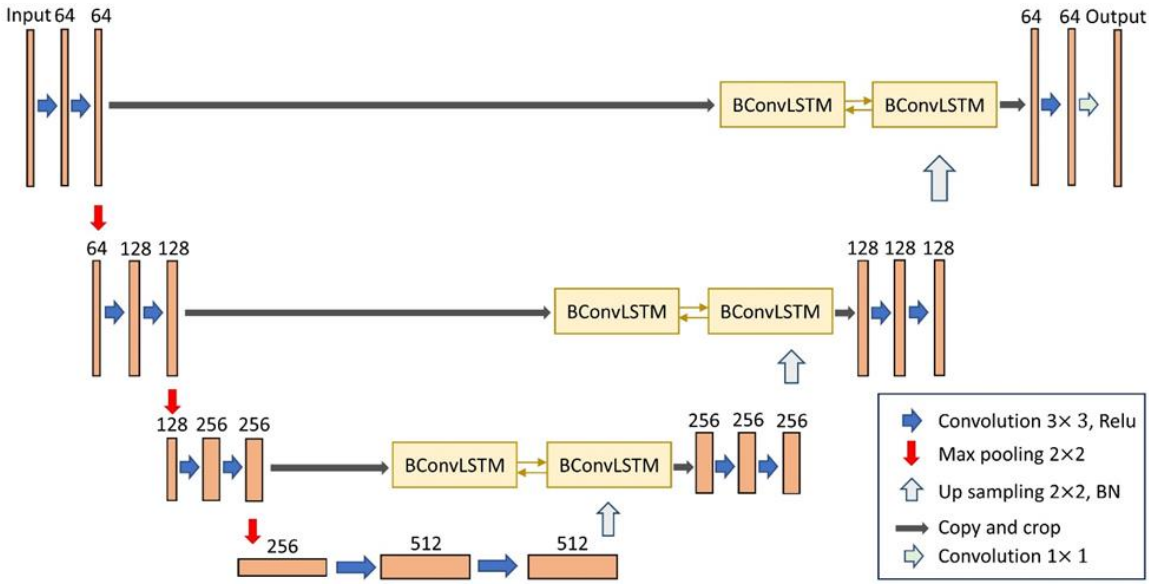

Figure 3.16. BCL-UNet model without any dense connections and with BConvLSTM in the skip connections.

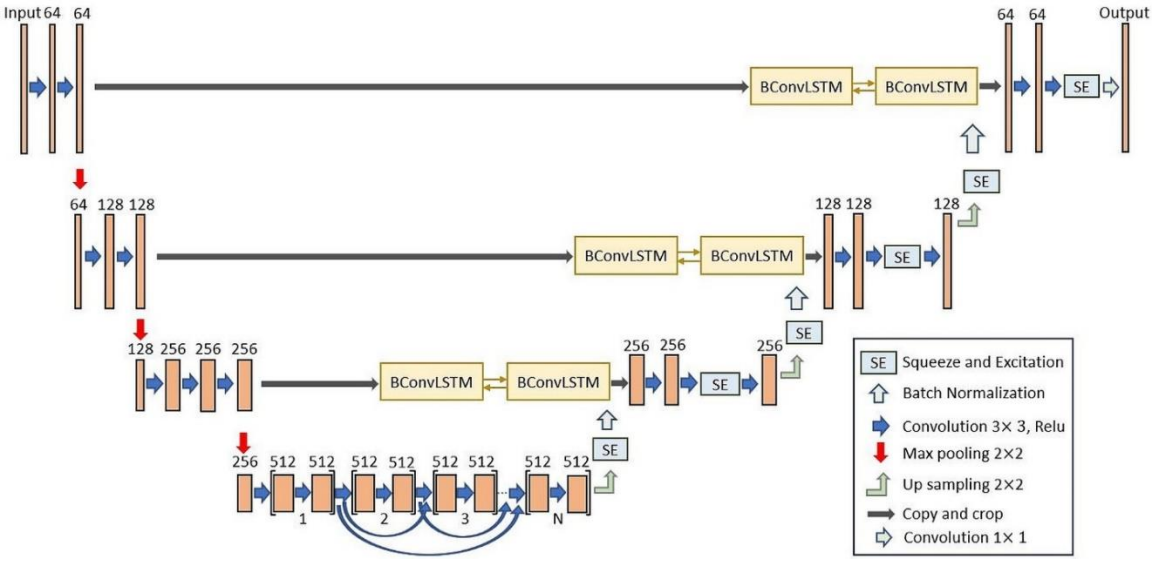

Figure 3.17. MCG-UNet model with dense connections, with the SE function in the expansive part and BConvLSTM in the skip connections.

3.3.3.2. SE function

The SE function [144] was suggested to gain a clear relationship between the convolutional layers channels and improve the representation power of the model by a context gating mechanism. By allocating a weight for every channel in the feature map, this function encodes feature maps. The SE module comprises two main sections named squeeze and excitation. Squeeze is the first operation. I accumulated the input feature maps to SE block to generate channel descriptor by applying global average pooling (GAP) of the entire context of channels. We have $X_d^{up} = [X_1^{up}, X_2^{up}, ..., X_F^{up}]$, in which the input data to SE function is $X_f^{up} \in R^{W \times H}$, and spatial squeeze (GAP) is calculated as:

$$z_f = F_{sq}(X_f^{up}) = \frac{1}{H \times W} \sum_i^H \sum_j^W X_f^{up}(i, j) \quad (20)$$

where the size of the f^{th} channel, the channel spatial location, and the spatial squeeze function are expressed as $X_f^{up}(i, j)$, $H \times W$, and F_{sq}, respectively. In other words, z_f can be produced by compressing every two-dimensional feature map using a GAP. The initial stage (Squeeze) introduces the global information, which is then fed to the next stage (Excitation). The excitation stage comprises two dense (FC) layers as shown in Figure 3.17. To shape $1 \times 1 \times \frac{F}{r}$ and $1 \times 1 \times F$, the pooled vector is initially encoded and decoded, respectively. Next, the excitation vector is generated as $s = F_{ex}(z; W) = \sigma(W_2 \delta(W_1 z))$, where r is the reduction ratio, σ denotes the sigmoid function, δ is Relu, and $W_1 \in R^{\frac{F}{r} \times F}$ denotes the initial fc layer $R^{F \times \frac{F}{r}}$ parameters. The SE block output is produced as $\tilde{X}_f^{up} = F_{scale}(X_f^{up}, z_c) = s_c X_f^{up}$, where s_c is the scale factor, F_{scale} is the input feature map, and $\tilde{X}_d^{up} = [\tilde{X}_1^{up}, \tilde{X}_2^{up}, ..., \tilde{X}_F^{up}]$ is defined as a multiplication between the channel's attention

on a channel-by-channel basis. In [144], a dimensionality-reduction and a dimensionality-increasing layer with ratio r were utilized, respectively, in the initial FC layer and the second one to aid generalization and limit model complexity.

Table 3.2. Detailed configurations of all approaches.

Approaches	Number of Parameters	Number of Layers	Batch Size	Input Shape	Computer Configuration
UNet	9,090,499	30	2	768×768× 3	A GPU: Nvidia Quadro RTX 6000 24 GB and a computation capacity of 7.5 Python: 3.6.10 TensorFlow: 1.14.0
BCL-UNet	13,580,995	42	2	768×768× 3	
MCG-UNet	27,891,901	74	2	768×768×3	

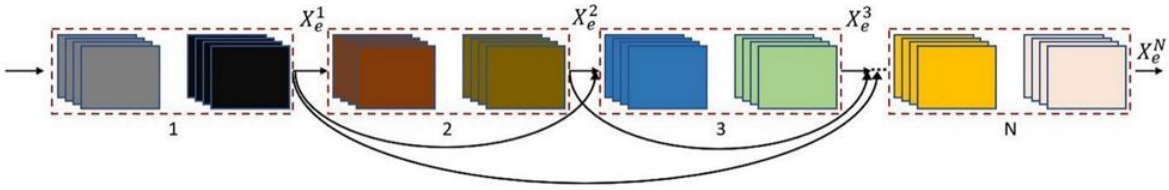

Figure 3.18. Densely connected convolutional layers of MCG-UNet.

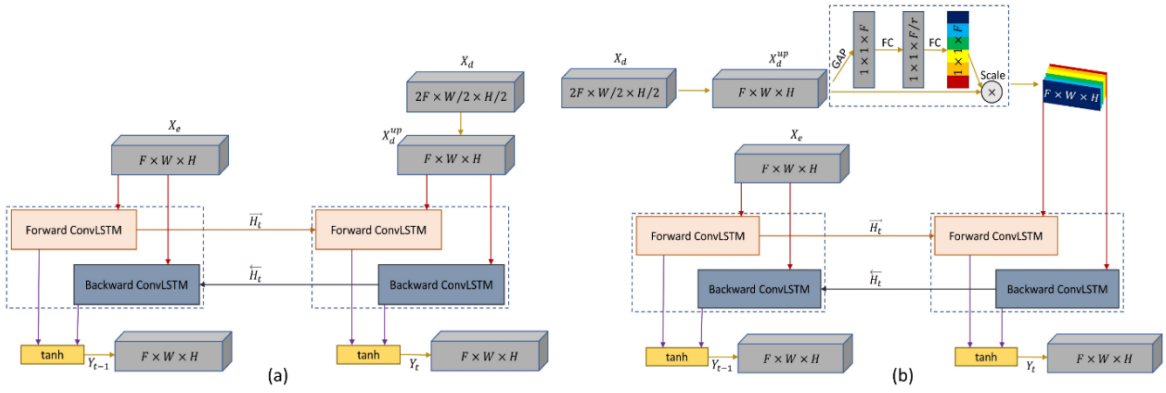

Figure 3.19. (**a**) Structure of BConvLSTM in the expansive part of the BCL-UNet model, and (**b**) BConvLSTM with the SE module in the expansive part of the MCG-UNet model (**b**).

3.3.3.3. BN function

The dispensation of the activations alters in the intermediate layers in the training stage and this issue slows down the training process. This is because every layer in each training stage must learn to adjust themselves to a novel distribution. Therefore, the BN function [148] is used to enhance the consistency of the networks. The batch mean is subtracted and then divided by the batch standard deviation using the BN function to standardize the inputs to a layer in the models. The BN function improves the performance of the networks in some cases and efficiently hastens the speed of training process. BN uses $\tilde{X}_d^{up}$ as an input after upsampling to generate $\hat{X}_d^{up}$. Additional details are available in [148].

3.3.3.4. BConvLSTM function

The standard long short-term memory (LSTM) networks utilize full relationships between transmissions of input-to-state and state-to-state and do not take the spatial correlation into account, which is the major disadvantage of these networks [149]. Therefore, ConvLSTM was suggested by [150] to exploit convolution operations into transmissions of input-to-state and state-to-state and tackle this issue. ConvLSTM includes a memory cell, a forged gate, an output gate, and an input gate, which work as controlling gates for accessing, updating, and clearing the memory cell. The ConvLSTM function can be calculated as:

$$
\begin{aligned}
i_t &= \sigma(W_{xi} \times X_t + W_{hi} \times H_{t-1} + W_{ci} \times C_{t-1} + b_i) \\
f_t &= \sigma(W_{xf} \times X_t + W_{hf} \times H_{t-1} + W_{cf} \times C_{t-1} + b_f) \\
C_t &= f_t o C_{t-1} + i_t \tanh(W_{xc} \times X_t + W_{hc} \times H_{t-1} + b_c) \\
o_t &= \sigma(W_{xo} \times X_t + W_{ho} \times H_{t-1} + W_{co} o C_t + b_c) \\
H_t &= o_t o \tanh(C_t),
\end{aligned}
\tag{21}
$$

where b_c, b_o, b_f, and b_i are bias terms, H_t is the hidden state, X_t is the input state, o is the Hadamard and $\times$ denotes the convolution functions, C_t is the memory cell, and W_{X*}

and W_{h*} are Conv2D kernels corresponding to the input and hidden state, respectively. To encode X_e and $\hat{X}_d^{up}$, I applied BConvLSTM [145] in the proposed BCD-UNet and MCG-UNet models that derive the output of BN step. The BConvLSTM function decides for the current input based on processing the data dependencies in both forward and backward directions. In contrast, a standard ConvLSTM only processes the dependencies of the forward way. In other words, the BConvLSTM processes the input data into two paths (forward and backward) utilizing two ConvLSTM. The output of BConvLSTM can be formulated as:

$$Y_t = \tanh(W_y^{\vec{H}} \times \vec{H}_t + W_y^{\overleftarrow{H}} \times \overleftarrow{H}_t + b) \quad (22)$$

where $Y_t \in R^{F_t \times W_t \times H_t}$ denotes the last output with bidirectional spatio-temporal information, $\overleftarrow{H}_t$ and $\vec{H}_t$ are the backward and forward hidden tensors, respectively, b is the bias term, and $\tanh$ is a non-linear hyperbolic tangent used to mix the output of both states. Analyzing the forward and backward data dependencies will boost the predictive performance.

3.3.3.5. Boundary-aware loss

In this work, I suggested a boundary-aware loss function (BAL), which is a simple yet efficient loss function. I first extracted boundaries E_i by filter $f_E = 2 \times 2$ from semantic segmentation labels l_i for every class i (Equation (23). Then, at the boundary image, I adopted Gaussian blurring using a Gaussian filter f_G, summed all of the channels results E_G, and added bias β (Equation (24). I calculated the BAL by multiplying the original binary cross-entropy loss L to the Gaussian edge E_G (Equation (25)) between ground truth and prediction to suppress the inner regions of every class and amplify loss around boundaries. The Gaussian edge efficiently concentrates on not only small objects, occluded

areas between objects, and complex parts of objects, but also boundaries and corners of objects [53].

$$E_{i(x,y)} = \begin{cases} 0 & |(l_i \otimes f_E)_{(x,y)}| = 0 \\ 1 & |(l_i \otimes f_E)_{(x,y)}| > 0 \end{cases} \quad (23)$$

where $f_E = \begin{bmatrix} 1 & -0.5 \\ -0.5 & 0 \end{bmatrix}$

$$E_G = \sum_i (E_i \otimes f_G) + \beta \quad (24)$$

$$BAL = \frac{1}{n} \sum_{(x,y)} E_G(x,y) \times L(x,y) \quad (25)$$

where the number of pixels in the label l is denoted as n.

3.3.3.6. Dataset and experiment setting

I used the Massachusetts [135] and DeepGlobe [151] road datasets to test the proposed networks for road extraction. Massachusetts dataset comprises 1171 aerial imagery with a dimension of 1500×1500 pixels and a spatial resolution of 0.5 m. I selected some good-quality imagery with complete information of road pixels and then split them into the size of 768×768. The last dataset that I utilized comprised 1068 images. I divided the dataset into 64 test images and 1004 validation and training images. DeepGlobe dataset includes 7469 training and validation images and 1101 testing images with a spatial resolution of 50 cm and a pixel size of 1024×1024. Furthermore, I applied vertical and horizontal flipping and rotation as data augmentation approaches to extend our dataset. Deeper convolution layers were given a 0.5 dropout to overcome over-fitting concern [17]. An optimization method is necessary to reduce the energy function and update the model parameters while training the network. Thus, I utilized the adaptive moment estimation

(Adam) optimization algorithm in our framework with a learning rate of 0.0001 to diminish the losses and update weights and biases. The entire process of the presented approaches for road extraction in this study was implemented using Keras with a TensorFlow backend and a GPU Nvidia Quadro RTX 6000 with a 7.5 computation capacity and memory of 24 GB.

3.4. Road shape and connectivity-preserving with SC-RoadDeepNet (Objective 2)

Based on the second objective, a new robust DCNN method called SC-RoadDeepNet was performed for road surface segmentation from HRSI, which the implementation of the method is described in this section. As discussed in the literature, road extraction remains challenging owing to obstacles, such as shadows, cars, and buildings that are similar to road features in terms of patterns, reflectance and color. Furthermore, the majority of the approaches yielded broken and disconnected road extraction outputs, indicating that the roads are affected by the aforementioned difficulties. As a result, it is essential to design a robust DCNN framework to conserve the geometry and connectedness of road networks. Thus, the SC-RoadDeepNet was proposed in this work to solve the above-mentioned issues with the existing methods and produce high-quality road segmentation maps.

In the proposed model, I implemented a new deep learning-based network called the recurrent residual CNN model (RRCNN) that is on the basis of the UNet network. The presented network uses recurrent residual convolutional layers (RRCLs), UNet, and residual networks. For segmentation tasks, RRCLs accumulate important features and thus enable better feature representation. They allow us to build a UNet network with similar network parameters but better segmentation performance. I also utilized road boundaries to make road semantic features more proper for actual road form, solve irregular semantic

features, and enhance the boundary of road semantic polygons. I leveraged each road's binary edge-map to penalize boundary misclassification and fine-tune the road shape. Furthermore, I offered a connectivity-preserving centerline Dice (CP_clDice), a new measure based on the intersection of segmentation masks, and their (morphological) skeleton to preserve road connectivity and obtain accurate segmentations. Our measure states the network's connectivity rather than evenly weighting each pixel given its morphological skeletons-based formulation. I showed that CP_clDice ensures connectivity conservation for binary segmentation, allowing for a proper road network extraction.

3.4.1. The architecture of RRCNN

I proposed RRCNN (Figure 3.20), a new model for segmentation tasks that is inspired by UNet [95] (Figure 3.21), RCNN [152], and the deep residual model [40]. The original UNet model consists of two main parts: convolutional encoding and decoding units. In both portions of the model, the fundamental convolutional layers are applied, followed by ReLU activation. In the encoding part, 2×2 max-pooling layers are applied for down sampling. The convolutional transpose layers are used to up-sample the feature maps during the decoding step. Also, in the UNet network, cropping and copying method is used to crop and copy feature maps from the encoder part to the decoder part. Therefore, the benefits of all three newly established deep learning approaches are combined in the proposed approach. Assuming a pixel in an input sample on the k^{th} feature map in the recurrent convolutional layers (RCL) that is located at (i, j) and input sample x_l in the layer l^{th} of the RCNN block, the network's output $o_{ijk}^{l}(t)$ at the t time step can be expressed as follows:

$$O_{ijk}^{l}(t) = (w_k^f)^T \times x_l^{f(i,j)}(t) + (w_k^r)^T \times x_l^{r(i,j)}(t-1) + b_k \text{ , } (26)$$

where b_k is the bias, w_k^r is the weight of the k^{th} RCL's feature map, w_k^f is the standard convolutional layer's weight, $x_l^{r(i,j)}(t-1)$ is the input for the l^{th} RCL, and $x_l^{f(i,j)}(t)$ is the input for the standard convolutional layers. The RCL's outputs are passed through the rectified linear unit (ReLU) activation function f that is denoted as follows:

$$F(x_l w_l) = f(O_{ijk}^l(t)) = \max(0, O_{ijk}^l(t)), \quad (27)$$

where $F(x_l w_l)$ denotes that the outputs of the l^{th} RCNN layer are utilized in the encoding and decoding arms of the network for down-sampling and up-sampling layers, respectively. For the RRCNN model, the last output that is passed through residual units can be expressed as follows:

$$x_{l+1} = x_l + F(x_l w_l), \quad (28)$$

where, in the RRCNN's encoding and decoding arms, x_{l+1} is utilized as the input for immediate subsequent down or up-sampling layers, and the RRCNN-input block's samples are represented by x_l.

The suggested RRCNN model is the building block of the stacked recurrent residual convolutional units depicted in Figure 3.22(c). This study investigated convolutional and recurrent convolutional units in various variants for three distinct architectures, shown in Figures 3.22(a)–3(c). The first is the primary UNet architecture [95] with encoder-decoder arms and a crop and copy method (skip connection). This model's fundamental convolutional unit is depicted in Figure 3.22(a). The second is ResUNet [153], which is the original UNet model with forwarding convolutional and residual connection units, as illustrated in Figure 3.22(b).

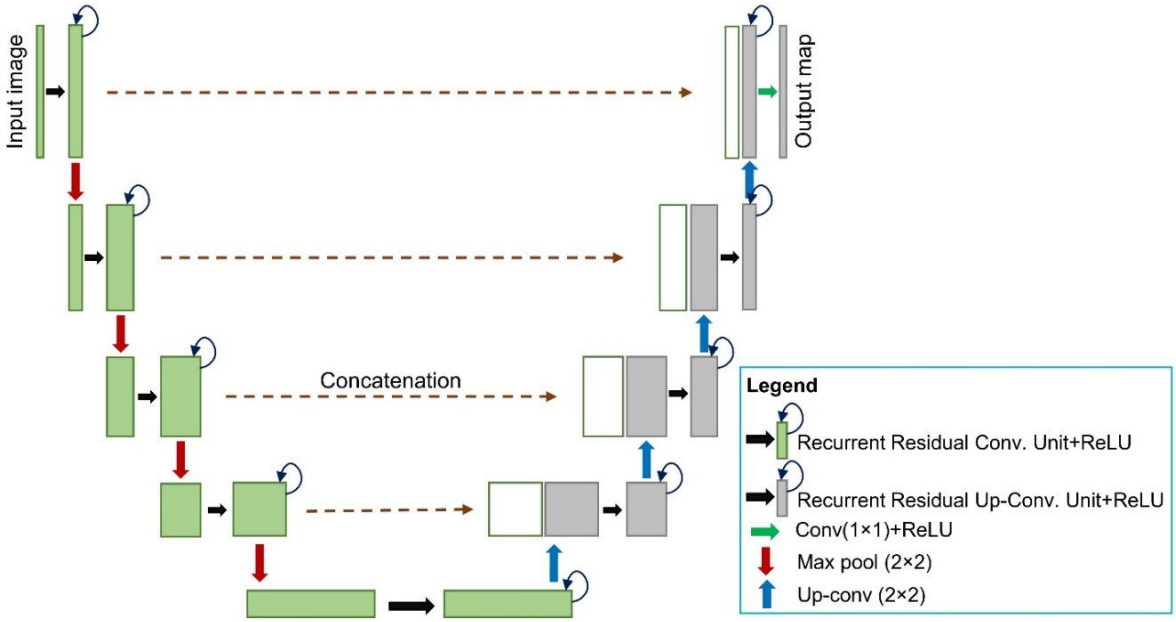

Figure 3.20. Architecture of the proposed RRCNN model, including encoder-decoder units based on recurrent RRCL and UNet networks.

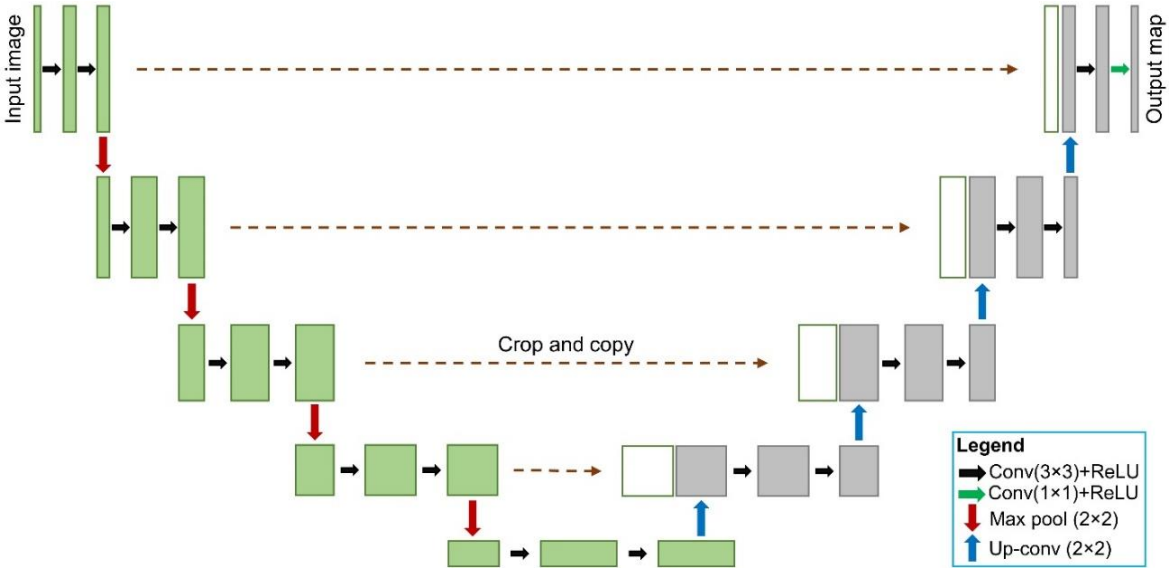

Figure 3.21. Architecture of the original UNet model, including convolutional encoder-decoder units.

The final architecture is the proposed RRCNN, including the primary UNet with RCL and residual connections, as depicted in Figure 3.22(c). When compared with UNet, the proposed architecture offers various advantages. One of those is network productivity,

which is measured in relation to the number of network parameters. Compared with UNet and ResUNet, the suggested RRCNN model is built to have similar parameters while performing efficiently on feature extraction. Recurrent or residual units do not increase the number of network parameters. However, they have a considerable effect on training/testing results. Furthermore, the proposed model's RCL units provide an efficient feature accumulation mechanism. Concerning distinct time-steps, feature accumulation guarantees more reliable and robust feature representation. As a result, it aids in the extraction of low-level features that are critical for feature extraction. I eliminated the cropping and copying method from the primary UNet network and replaced it with concatenation operation, leading to a considerably more elegant design with improved efficiency.

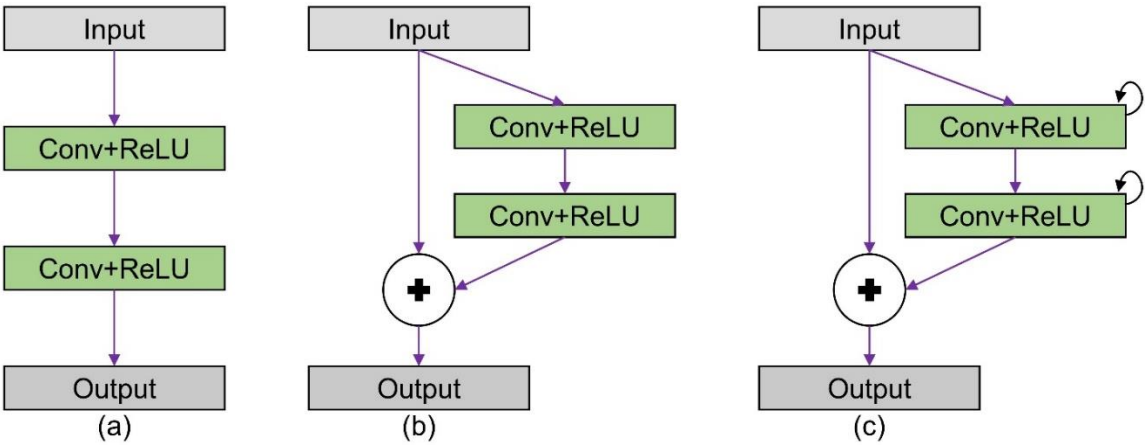

Figure 3.22. Convolution and recurrent convolution units in various variants: (a) forward convolution units, (b) residual convolution units, and (c) recurrent residual convolution units.

3.4.2. Emphasizing connectivity using CP_clDice

Figure 3.23 depicts a schematic overview of our suggested CP_clDice technique. Based on intersecting skeletons with masks, I present a new connectivity-preserving measure for evaluating road structure segmentation. The ground truth (M_G) and detected segmentation

(M_D) masks are two binary masks that I considered. From M_G and M_D, skeletons S_G and S_D are first extracted, respectively. $S_D = \{g_i\}_{i=1}^{N}$ is the detected skeleton of a detected mask M_D, while $S_G = \{h_i\}_{i=1}^{N}$ is the true skeleton of a true mask M_G, where the h_i and g_i are the skeleton points of S_G and S_D, respectively. Following that, I calculated the proportion of S_G that exists within M_D, which is called connectivity sensitivity or $C_{sens}(S_G, M_D)$, and vice-a-versa. I computed connectivity precision or $C_{prec}(S_D, M_G)$ as follows:

$$C_{sens}(S_G, M_D) = \frac{|S_G \cap M_D|}{|S_G|}; C_{prec}(S_D, M_G) = \frac{|S_D \cap M_G|}{|S_D|}. \quad (29)$$

Or $C_{sens} = \sum_{i=1}^{N} \frac{h_i M_D(h_i)}{\sum_{j=1}^{N} h_i}; C_{prec} = \sum_{i=1}^{N} \frac{g_i M_G(g_i)}{\sum_{j=1}^{N} g_i}$

The metric the measure $C_{sens}(S_G, M_D)$ is prone to false negatives in prediction, whereas $C_{prec}(S_D, M_G)$ is prone to false positives, clarifying why I referred to $C_{sens}(S_G, M_D)$ as connectivity's sensitivity and $C_{prec}(S_D, M_G)$ as its precision. I calculated CP_clDice as the harmonic mean of both measures because I want to maximize sensitivity and precision:

$$CP\_clDice(M_D, M_G) = 2 \times \frac{C_{prec}(S_D, M_G) \times C_{sens}(S_G, M_D)}{C_{prec}(S_D, M_G) + C_{sens}(S_G, M_D)} \quad (30)$$

3.4.3. Soft-skeletonization with soft CP_clDice

The following section demonstrates how I used CP_clDice formulation to train a connectivity-preserving network using our theory effectively. Our strategy relies on correct skeletons extraction. A variety of ways have been presented for this task [154]. However, most of them are not entirely distinguishable and thus unsuitable for use in a loss function. The repeated morphological thinning [155] or Euclidean distance transform [156] are two popular methods. A series of erosions and dilation operations are used in morphological

thinning. The Euclidean distance transform remains a discrete operation, prohibiting it from being used in a loss function for neural network training. As a grey-scale alternative to morphological erosion and dilation, min- and max filters are often utilized. As a result, I suggested soft-skeletonization, in which iterative min-max pooling is used as a surrogate for morphological dilation and erosion. Figures 3.24 and 3.25 illustrate the sequential steps of our skeletonization intuitively. Initial iterations (Figure 3.24) skeletonize and maintain structures with a small radius until later iterations skeletonize and maintain thicker structures, allowing for the creation of a parameter-free, morphologically focused soft skeleton. The iterative processes involved in its computation are described in Algorithm 1 (soft-skeletonization) shown in Figure 3.25. The iterations are represented by the hyper-parameter, which must be equal to or greater than the maximum witnessed radius.

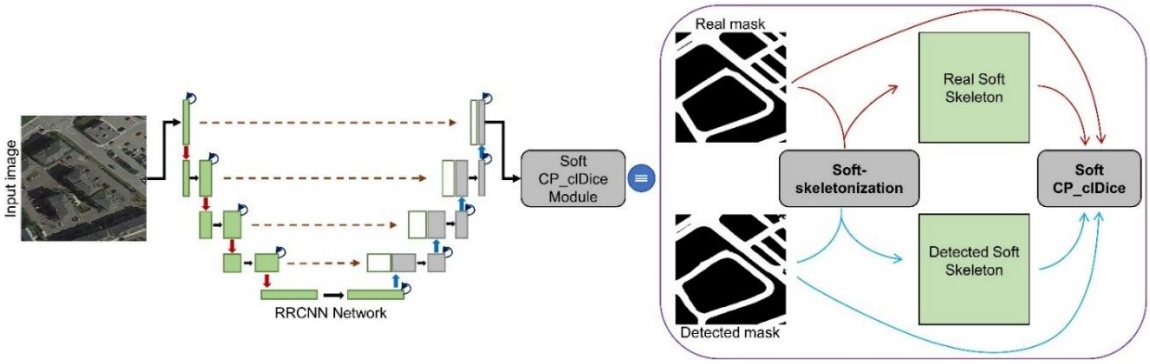

Figure 3.23. An overview of our suggested CP_clDice technique. The CP_clDice method can be implemented in any generic segmentation model. I applied the RRCNN network in this work. Pooling functions from any common deep learning toolbox can be used to build soft-skeletonization simply.

This parameter varies depending on the dataset. In our experiments, for example, $k = 5...20$, which corresponds to the pixel radius of the largest witnessed road structures. A low k results in incomplete skeletonization. Increasing the value of k does not decrease performance but lengthens computation time. Given the previously stated soft-

skeletonization, I can utilize CP_clDice as an optimizable, real-valued, and fully differentiable measure. The implementation is described in Algorithm 2 (Figure 3.25) that is known as the soft CP_clDice. The amount of linked loops determines the homotopy type for a single connected foreground component without knots. As a result, no pairwise linked loops are detected, and reference pixels are not homotopy-equal. The deformation retracted skeleton of the solid foreground must be added or removed to include or omit these extra loops. Thus, the addition of new pixels that have been appropriately detected is needed. Unlike other losses like cross-entropy and Dice, CP_clDice only analyses the solid foreground structure's deformation-retracted graphs. As a result, I assert that CP_clDice needs the minimum number of new properly detected pixels to ensure homotopy equality. Cross-entropy or Dice can only ensure homotopy equivalence in these lines provided each pixel is properly segmented. CP_clDice can ensure the equivalence of homotopy for a wider combination of pixels, which is an intuitively appealing trait because it renders CP_clDice powerful against noisy segmentation labels.

3.4.4. Cost function

I integrated our suggested soft CP_clDice with soft-Dice (a function to calculate dice loss) in the following manner to preserve connectivity while obtaining correct segmentations (our objective) rather than the learning skeleton:

$$L_c = (1-\alpha)(1-softDice) + \alpha(1-softCPclDice) \quad (31)$$

where $softDice = \dfrac{2\sum_{i}^{N} p_i o_i}{\sum_{i}^{N} p_i^2 + \sum_{i}^{N} o_i^2}$

here, N denotes as total pixels, $p_i \in M_D$ is the detected binary pixels, and $o_i \in M_G$ is the ground truth pixels.

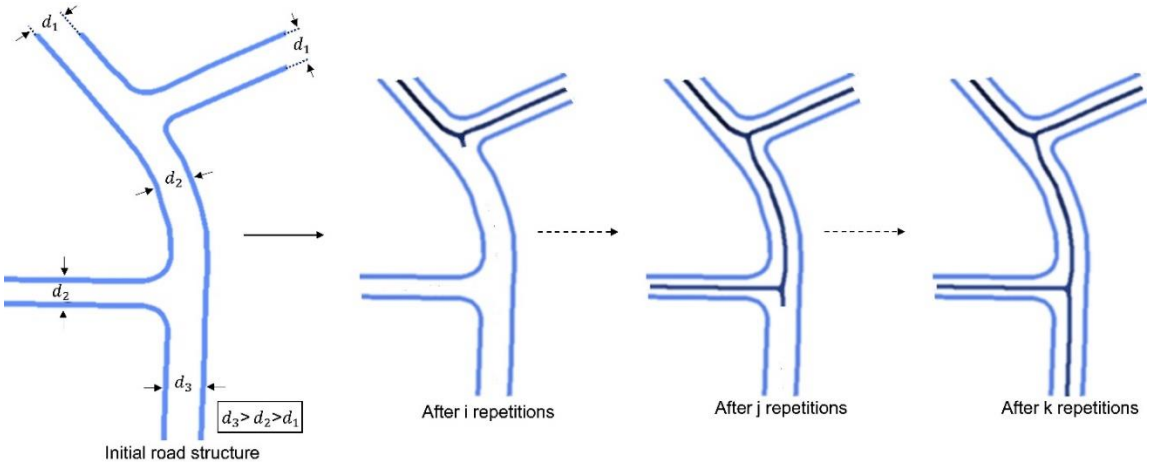

Figure 3.24. Sequential bagging of skeleton pixels (dark blue) by iterative skeletonization leads to complete skeletonization based on the initial road structure (blue), where $k > j > i$ signifies iterations and d diameter.

This study aims to learn a connectivity-preserving segmentation, not learning the centerline. As a result, I limited α options (weight for the CP_clDice element) in our experiments to $[0.1, 0.5]$ for achieving high-quality results. Furthermore, I used the binary edge-map of each road to penalize boundary misclassification, solve irregular road forms, and enhance the shape of semantic roads. In fact, reliable annotated road edges are integrated into semantic polygons to strengthen the semantic polygon's border, repair discontinuous areas, assure the road's continuity and integrity, and obtain more precise boundary positioning. I tested our CP_clDice and binary edge-map information on a new state-of-the-art deep learning model (RRCNN). I proposed a new method named SC-RoadDeepNet, a shape and connectivity-preserving method, to show the model's effectiveness in preserving connectivity while obtaining accurate segmentation.

Algorithm 1: soft-skeletonization	**Algorithm 2:** soft CP_clDice
Input: M, k $M' \leftarrow \max pooling(\min pooling(M))$ $Skel \leftarrow relu(M - M')$ **for** $m \leftarrow 0$ **to** k **do** $M \leftarrow \min pooling(M)$ $M' \leftarrow \max pooling(\min pooling(M))$ $Skel \leftarrow Skel + (1 - Skel) \circ relu(M - M')$ **end** **Output:** $Skel$	**Input:** M_D, M_G $S_D \leftarrow$ *soft-skeletonization* (M_D) $S_G \leftarrow$ *soft-skeletonization* (M_G) $C_{prec}(S_D, M_G) \leftarrow \frac{\lvert S_D \cap M_G \rvert}{\lvert S_D \rvert}$ $C_{sens}(S_G, M_D) \leftarrow \frac{\lvert S_G \cap M_D \rvert}{\lvert S_G \rvert}$ $CP\_clDice \leftarrow 2 \times \frac{C_{prec}(S_D, M_G) \times C_{sens}(S_G, M_D)}{C_{prec}(S_D, M_G) + C_{sens}(S_G, M_D)}$ **Output:** *CP_clDice*

Figure 3.25. The suggested soft-skeleton is calculated using Algorithm 1, where k is the number of iterations for skeletonization and M is the mask to be soft-skeletonized. The soft CP_clDice loss is calculated using Algorithm 2, where M_G is the ground truth mask and M_D is the segmentation mask. $\circ$ denotes the Hadamard product.

3.4.5. Datasets

This part describes the datasets used to train and assess SC-RoadDeepNet, including Google Earth imagery [42] DeepGlobe [151] and Massachusetts [135]. The Google Earth dataset has a spatial resolution of 0.21 m per pixel covering around 8 km^2. The dataset is more comprehensive and difficult to work with because of the numerous obstacles and shadows generated by avenue trees and cars along the roads. A total of 696 images are included in the dataset, which is divided into a training set and a testing set of 651 images and 45 images. Every original image has 512×512 pixels size. The DeepGlobe dataset is captured in India, Indonesia, and Thailand, containing 8570 images with 50 cm per pixel spatial resolution and covering 2220 km^2. Each image is 1024×1024 pixels in size. The training and testing datasets consisted of 1006 and 26 images in this study, respectively.

The Massachusetts dataset that I used contains 1032 training and 32 testing images with a size of 768×768 and spatial resolution of 0.5 m.

3.4.6. Experiment settings

Given that the size of our road dataset is still small, which may lead to an over-fitting issue, some data augmentation techniques are utilized to increase the dataset size. I used data augmentation tactics, such as rotating (90, 180, and 270 degrees) the images and flipping (vertical and horizontal) them to enhance the dataset's capacity. The proposed network was trained on a GPU Nvidia Quadro RTX 6000 under Keras framework and with Tensorflow backend with batch size 1 for 100 epochs across the datasets. This study used an adaptive moment estimation (Adam) optimizer with a $1e-3$ learning rate and decay of 0.9 to optimize the loss function and learn model parameters. The Sigmoid activation is also applied to sort the outcomes. The final layer gives outputs in continuous value from 0 to 1 since it is activated by the Sigmoid function. As a result, I used a 0.5 threshold to attain the final segmentation map of the input images.

3.5. Simultaneous road network segmentation and vectorization using RoadVecNet (Objective 3)

For the third aim, a novel trustworthy DCNN methodology called RoadVecNet was used for simultaneous road surfaces segmentation and vectorization from different HRSI [45], which the method's implementation is discussed in this section. Most of the techniques concentrated on road surface segmentation and centerline extraction without obtaining precise information about the road's width and position, as stated in the literature. As a result, a new high-accurate road vectorization model was developed in this study that can

not only extract road surfaces accurately but also vectorize the road network and get the crucial information indicated above.

The proposed RoadVecNet model extract the road surface and vectorize the road network simultaneously. In the extraction part, I wanted to deal with the road segmentation issues and detect consistent road parts. I also wanted to vectorize the road network by determining and extracting the road vector rather than the road centerline to obtain accurate information about the road network's width and location. The proposed approach comprised two convolutional UNet networks that are interlinked into one architecture. The initial framework was used to identify road surfaces, while the second framework was utilized to vectorize roads to achieve road location and width information. In the proposed model, I used two encoders, two decoders, and two novel modules, namely, dense dilated spatial pyramid pooling (DDSPP) [157] and squeeze-and-excite (SE) [144]. The DDSPP module was used to achieve a bigger receptive field and create feature pyramids with a more denser scale variability. The SE module was employed to consider the interdependencies between feature channels and extract more valuable information. I also used a loss function named focal loss weighted by the median frequency balancing (MFB_FL) to overcome highly unbalanced datasets where positive cases are rare. MFB_FL lessens the burden of simple samples, allowing more time to be spent on difficult samples, and improves the road extraction and road vectorization results. Accordingly, I could achieve constant road surface identification outcomes and complete and smoothen road vectorization results with accurate information of road width and location even under obstructions of shadows, trees, and complicated environments compared with other comparative deep learning-based techniques.

3.5.1. RoadVecNet architecture

An overview of the suggested RoadVecNet framework is shown in Figure 3.26. The proposed network comprises the road surface segmentation and road vectorization networks. Each UNet model includes a contracting encoder arm where the resolution decreases, and the feature depth increases and an expanding decoder arm where the resolution increases, and the feature depth decreases. I utilized filters of 32, 64, 128, and 256 to consider the number of feature maps in encoder–decoder. The skip connections characteristic of the UNet framework [95] connect each upsampled feature map at the decoder arm to the encoder's arm with an identical spatial resolution. Accordingly, the probability map that indicates the likelihood of every road and non-road pixel is obtained with the sigmoid classifier.

1) **Road surface segmentation architecture**: The detailed configuration of this network is shown in Figure 3.26(a). This network was first applied to detect the road surface, which is categorized into two: road and background categories. In this network, pre-trained VGG-19 [39] was used as an encoder because VGG-19 can be easily transferred to another task, given that it has formerly learned features from ImageNet. The key advantages of adopting the VGG-19 network are as follows: (1) its design is identical to UNet, making it easier to combine with UNet, and (2) it will allow much deeper networks to produce superior output segmentation and vectorization results. I also used the DDSPP module to extract high-resolution feature maps and capture contextual information within the architecture and the SE module to pass more relevant data and reduce redundant ones. Every block in the decoder part implements a 2×2 bilinear upsampling on the input features to double the dimension of the input feature maps. This avoids artifacts and the use of slow deconvolution layer and hence decrease the number of learning parameters, which it also

contributes to a faster total training and inference time. Then, the proper skip connections of the encoder feature maps to the output feature maps are concatenated. Thereafter, two 3×3 convolutional layers were applied, followed by batch normalization (BN) and Rectified Linear Unit (ReLU) function. The distribution of activations varies in the intermediate layers during the training step, which is a problem. This issue slows down the training phase because every layer in every training phase must learn to adjust to a new distribution. Thus, BN [148], which standardizes the inputs to a layer in the network by subtracting the batch mean and dividing by the batch standard deviation, is used to improve the stability of a neural network. The speed of a neural network's training process can be accelerated by BN [148]. Furthermore, the model's performance is improved in some cases due to the modest regularization influence. Subsequently, the SE module was used, and the mask was generated by applying a convolutional layer with the sigmoid function. In remote sensing imagery, the road samples face the class imbalance issue because of the skewed dispensation of ground objects [136]. The cross-entropy loss does not adequately account for the imbalanced classes because it is calculated by summing up all of the pixels. A typical approach for considering the imbalanced classes is to use a weighting factor [158]. The class loss is weighted using median frequency balancing by the ratio of the training set's median class frequency and the real class frequency [158]. The presentation of a weighting factor between the simple and the hard samples is the same; however, it balances the value of positive and negative samples. Therefore, the focal loss function was implemented by [159] to lessen the burden of simple samples, allowing them to focus more on the hard samples. I used the focal loss weighted by the median frequency balancing (MFB_FL) to address the imbalance issue of the training data and train the road surface segmentation network that is denoted as follows:

$$MFB\_FL_{seg}(g, f(o), \delta_1) = \alpha(1 - l_c(I_i^j))^{\gamma} \cdot BCE_{seg}, \quad (32)$$

where

$$BCE_{seg} = -\sum_{i=1}^{S}\sum_{j=1}^{P}\sum_{c=1}^{C} w_c \cdot (g_i^j = C) \log l_c(I_i^j), \quad (33)$$

$$w_c = \frac{median(m_c \mid c \in C)}{m_c},$$

where $median(m_c)$ is the median value of every m_c, m_c is the modulation of pixels in class c, w_c is the class weight, $f(I_i^j)$ is the output of the final convolutional layer at pixel I_i^j, g_i^j is the surface ground truth label, I_i^j is the jth pixel in the ith patch, C is the amount of classes, P is the amount of pixels in every patch, S is the batch size, δ_1 denotes the road segmentation model parameters, and $l_c(I_i^j)$ is defined as the road surface likelihood of pixel I_i^j

$$l_c(I_i^j) = \frac{\exp(f_c(I_i^j))}{\sum_{l=1}^{c} exp(f_l(I_i^j))}. \quad (34)$$

2) **Road vectorization architecture:** The detailed configuration of this network is shown in Figure 3.26(b). This network was then implemented to vectorize roads and extract the accurate width and location of the road network. The architecture has a similar architecture as the road surface segmentation architecture that has a contracting arm, expanding arm, skip connections, and sigmoid layer; however, it is much smaller than the road surface segmentation model. A relatively small architecture was chosen for this part for the following reasons. First, the training network has fewer positive pixels (vectorized road pixels) compared with the road segmentation framework. Thus, applying a relatively deep

network may cause overfitting. In addition, the feature maps generated by the final convolutional layer in the decode arm of the road segmentation framework have fewer complex backgrounds compared with the original image. A relatively small architecture is sufficient to deal with the vectorization task. In Figure 3.26, the inputs of the vectorization model are the feature maps generated by the final convolutional layer of the decoder arm in the road segmentation model. In every encoder block, two 3×3 convolutional layers were implemented, followed by batch normalization and ReLU. Thereafter, the SE block is used to enhance the feature map's quality. Then, a 2×2 max-pooling layer with stride 2 was applied to decrease the spatial dimension of the feature maps. All the components in the decoder arm are comparable to those of the decoder arm of the road segmentation network. To train the road vectorization model, its MFB_FL is denoted as follows:

$$MFB\_FL_{vec}(y, h(I), \delta_2) = \alpha(1 - l_c(f(I_i^j)))^{\gamma} \cdot BCE_{vec}, \quad (35)$$

where

$$BCE_{vec} = -\sum_{i=1}^{S}\sum_{j=1}^{P}\sum_{c=1}^{C} w_c \cdot (y_i^j = C) \log l_c(f(I_i^j)), \quad (36)$$

where y_i^j is the vectorized ground truth label, $h(f(I_i^j))$ is the output of the final convolutional layer in the road vectorization network, $f(I_i^j)$ is the output of the final convolutional layer at pixel I_i^j in the road segmentation model, C is the amount of classes, P is the amount of pixels in every patch, S is the batch size, δ_2 denotes the road vectorization network parameters, and $l_c(f(I_i^j))$ is denoted as the vectorized road likelihood of pixel I_i^j.

$$l_c(f(I_i^j)) = \frac{\exp(h_c(f(I_i^j)))}{\sum_{l=1}^{c} exp(h_1(f(I_i^j)))}. \quad (37)$$

I employed an end-to-end strategy to concurrently train the proposed road segmentation network and road vectorization network and utilized a distinct training dataset for every subtask. Moreover, I used the main RGB (red, green, and blue) images and the corresponding ground truth surface images for the road surface segmentation task and the main images and its corresponding ground truth vectorized images for the road vectorization task. Finally, the overall loss function in RoadVecNet, which is a combination of losses (1) and (3), can be expressed as follows:

$$\begin{aligned} &MFB\_FL(\delta_1+\delta_2) = MFB\_FL_{seg}(g, f(o), \delta_1) \\ &+MFB\_FL_{vec}(y, h(I), \delta_2) = \\ &\alpha(1-l_c(I_i^j))^{\gamma} \cdot BCE_{seg} + \alpha(1-l_c(f(I_i^j)))^{\gamma} \cdot BCE_{vec} \end{aligned}, \quad (38)$$

where the last convolutional layer's output in the road vectorization network is $h(f(\cdot))$, and the last convolutional layer's output in the road segmentation model is $f(\cdot)$. The focal loss is parameterized by γ and $_{\alpha}$, and it controls the degree of downweighting of easy examples and the class weights, respectively. The FL simplifies to BCE when γ=0. In this work, I set the values for $\gamma = 2$ and $\alpha = 0.25$ because the degree of concentrating on hard and easy samples can be increased by higher values of γ and lower values of α .

3.5.2. SE module

The SE module [144] was used to improve the model's representation power by a context gating mechanism and attain a clear relationship between the convolutional layer channels. The module encodes feature maps by allocating a weight for every channel in the feature

map. The SE module includes two major parts, called squeeze and excitation. The first operation is squeeze. The input feature maps to SE block are accumulated to generate a channel descriptor by applying global average pooling (GAP) of the entire context of channels.

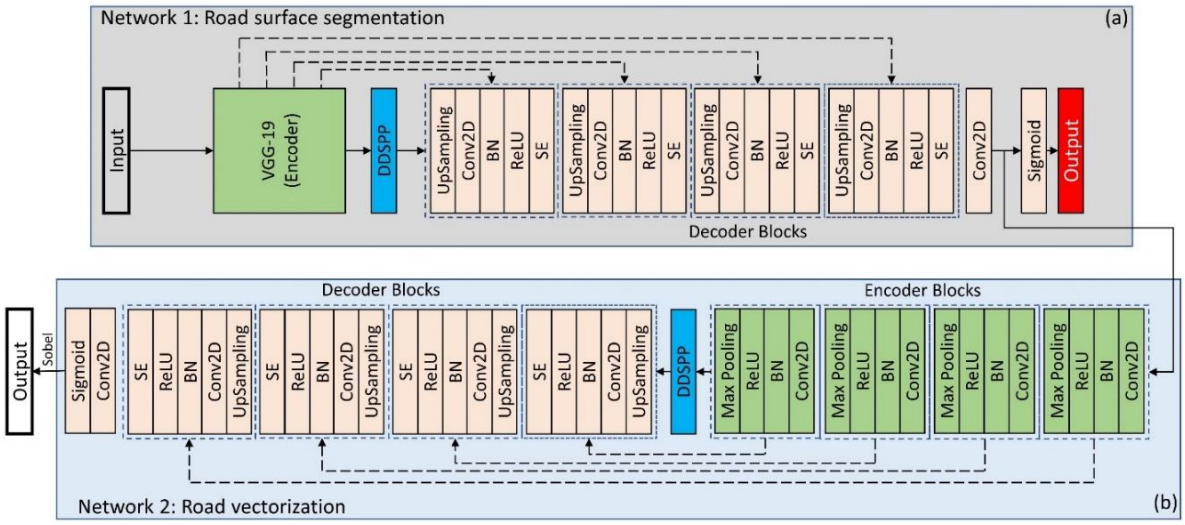

Figure 3.26. Flowchart of the RoadVecNet framework containing (a) road surface segmentation and (b) road vectorization UNet networks.

We have $X_d^{up} = [X_1^{up}, X_2^{up}, ..., X_F^{up}]$, in which the input data to the SE module are $X_f^{up} \in R^{W \times H}$, and the spatial squeeze is calculated as follows:

$$z_f = F_{sq}(X_f^{up}) = \frac{1}{H \times W} \sum_{i}^{H} \sum_{j}^{W} X_f^{up}(i, j), \quad (39)$$

where $H \times W$ is the size of this channel, $X_f^{up}(i, j)$ is a spatial location of the f^{th} channel, and F_{sq} is the spatial squeeze module. The second operation is excitation, which takes the global information produced in the squeeze stage. This operation includes two fully connected (FC) layers. The pooled vector is first encoded and then decoded to shape $1 \times 1 \times \frac{F}{r}$ and $1 \times 1 \times F$, respectively, to generate an excitation vector as $s = F_{ex}(z; W) = \sigma(W_2 \Re(W_1 z))$,

where $W_1 \in R^{\frac{F}{r} \times F}$ denotes the parameters of the initial FC layer $R^{F \times \frac{F}{r}}$, r is the reduction ratio, $\Re$ is ReLU, and σ denotes the sigmoid function. The output of the SE block is generated as $\tilde{X}_f^{up} = F_{scale}(X_f^{up}, z_c) = s_c X_f^{up}$, where $\tilde{X}_d^{up} = [\tilde{X}_1^{up}, \tilde{X}_2^{up}, ..., \tilde{X}_F^{up}]$ is a channel-wise multiplication between the channel attention, s_c is the scale factor, and F_{scale} is the input feature map.

3.5.3. DDSPP module

In this work, the DDSPP module was performed on the feature maps generated by the encoder arms to elicit further multi-scale contextual information and produce a greater number of scale features over a broader range. Atrous spatial pyramid pooling (ASPP) was first utilized in DeepLab [160] to enhance the suggested networks' performance. ASPP is a mixture of spatial pyramid pooling and atrous convolution with various atrous rates. This tool is effective in adjusting the receptive field to catch multi-scale information and in controlling the resolution of the features computed by deep learning networks. In particular, ASPP includes (a) an image-level feature that is generated by global average pooling and (b) one convolution with a 1×1 filter size and four parallel convolutions of a 3×3 filter size with different rates of 2, 4, 8, and 12, as illustrated in Figure 3.27. Then, bilinear upsampling was applied to upsample the outcoming features from the entire branches to the input size and concatenated and underwent another convolution with 1×1. However, I used a new module named DDSPP [157], which combines the benefit of cascaded modules with atrous convolution and ASPP to produce more scale features over a broader range and exploit further multi-scale contextual features. The receptive field for atrous convolution can be defined as follows:

$$F = [(K-1)(R-1)+K] \times [(K-1)(R-1)+K], \quad (40)$$

where R is the rate, and k is the convolution kernel size. For example, when R=2 and K=3, the F is then equal to 5×5. However, we can have a bigger receptive field and can create feature pyramids with a more denser scale variability by using dense connections between stacked dilated layers. Assuming that we have two convolutional operations with K_1 and K_2 kernel sizes, the receptive field can be defined as follows:

$$F = (K_1 + K_2 - 1) \times (K_1 + K_2 - 1). \quad (41)$$

The new receptive field size will result in 13×13 when the rates are 2 and 4.

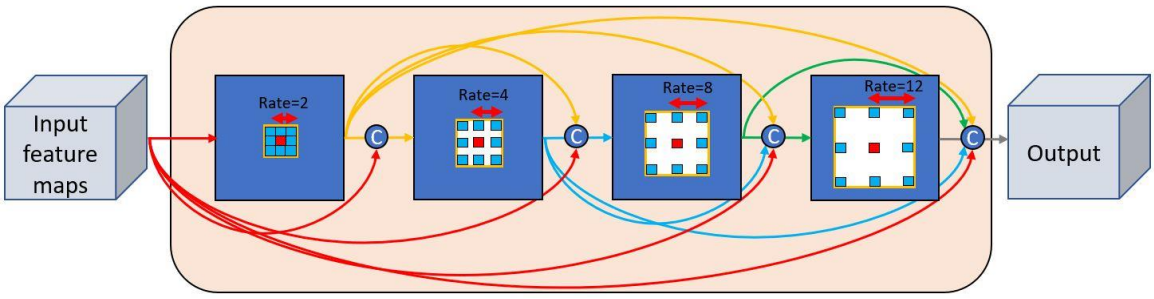

Figure 3.27. DDSPP structure. Each dilated convolutional layer's output is concatenated (C) with the input feature map and then fed to the subsequent dilated layer.

3.5.4. Inference stage

The road surface segmentation and road vectorization can be concurrently implemented through the proposed RoadVecNet in the inference stage (Figure 3.26). A probability road map was achieved by using the road segmentation network. Then, the road vectorization network transformed the features maps of the final convolutional layer generated by using a road segmentation model into vector-based possibility maps in the inference stage. Finally, the Sobel algorithm was applied to achieve a complete and smooth road vectorization network with precise road width information [161]. The Sobel algorithm is

an instance of the gradient approach. In the gradient method, the edges are detected by looking for the minimum and maximum in the image's initial derivative. The Sobel method computes an estimation of the image intensity gradient function and is a discrete differentiation method [161].

3.5.5. Experimental setting

I utilized some data augmentation strategies, such as flipping the images vertically and horizontally as well as rotating them 90°, 180°, and 270° to expand the size of our training and validation sets and train a proper model. Moreover, to dominate the overfitting difficulty, I appended a dropout of 0.5 [162] to the deeper convolutional layers of the road segmentation network and road vectorization network. A computationally affordable yet strong regularization to the model can be provided using this strategy. Adaptive moment estimation (Adam) optimizer with 0.001 learning rate was also utilized in this work to learn the model parameters, such as weights and biases via optimizing the loss function. The presented RoadVecNet was trained with batch size 2 from scratch except the backbone network that I used as the pretrained one. The trained network was then implemented on the test data for road surface segmentation and road vectorization. I implemented the optimization of the networks for 100 epochs through the datasets until no more performance improvements were seen. I applied the suggested network for road surface segmentation and road vectorization on a GPU Nvidia Quadro RTX 6000 with a memory of 24 GB and a computing capability of 7.5 under Keras framework with TensorFlow backend.

3.5.6. Dataset descriptions

Two types of remote sensing datasets called Massachusetts road imagery [135] containing aerial images with 0.5 m spatial resolution and Ottawa road imagery [163] containing

Google Earth images with 0.21 m spatial resolution were used to test the proposed network on the road segmentation and vectorization. I selected these two different datasets, which contain various road width pixels, to show the proposed architecture's superiority in road segmentation and vectorization. Each dataset includes two sub-datasets, namely, road surface segmentation and road vectorization. The detailed information of each dataset is highlighted as follows:

1) *Massachusetts datasets*: In this dataset, I used 766 images, which are split into 690 training, 48 validation, and 28 test images with a dimension of 512×512 and road width of approximately 6–9 pixels. Figure 3.28 demonstrates some samples of the original images in the first column, the corresponding reference map in the second column, and a superposition between vectorized road and road segmentation ground truth maps in the last column.

Figure 3.28. Demonstration of three representative imagery, their segmentation ground truth, and vectorized ground truth maps for the Massachusetts road imagery. (a), (b), and (c) illustrate the original RGB imagery, corresponding segmentation ground truth maps, and superposition between vectorized and segmentation ground truth maps, respectively.

2) *Ottawa datasets*: I utilized 652 images divided into 598 training, 34 validation, and 20 test images with a dimension of 512×512 and road width of almost 24–28 pixels. Figure 3.29 illustrates some examples of the main imagery, the corresponding reference map, and the superposition between vectorized road and road segmentation ground truth maps in the first, second, and last columns, respectively.

Figure 3.29. Demonstration of three representative imagery and their segmentation ground truth and vectorized ground truth maps for the Ottawa road imagery. (a), (b), and (c) demonstrate the main RGB images, corresponding segmentation ground truth maps, and superposition between vectorized and segmentation ground truth maps, respectively.

3.6. Evaluation factors

Different metrics were used to evaluate the accuracy assessment of the suggested ML and DL methods applied for road class extraction and vectorization from high-resolution remote sensing data, namely, F1 score, Recall (Completeness), Intersection over union (IOU), Precision (Correctness), Matthews correlation coefficient (MCC), and Mean intersection over union (MIOU) factors. These metrics can be calculated from the number of false positive (FP), false negative (FN), true negative (TN), and true positive (TP) pixels as:

$$\mathrm{Pr}ecision = \frac{TP}{TP + FP} \quad (42) \qquad \mathrm{Re}call = \frac{Tp}{TP + FN} \quad (43) \qquad IOU = \frac{TP}{TP + FP + FN} \quad (44)$$

$$F1 \quad score = \frac{2 \times \mathrm{Pr}ecision \times \mathrm{Re}call}{\mathrm{Pr}ecision + \mathrm{Re}call} \quad (45) \qquad MCC = \frac{TP.TN - FP.FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (46)$$

$$MIOU = \frac{1}{k} \sum_{i=0}^{k-1} \frac{TP_i}{TP_i + FP_i + FN_i} \quad (47) \qquad \mathrm{OA} = \frac{\mathrm{TP+TN}}{\mathrm{N}} \quad (48)$$

The recall represents the fraction of the labeled road pixels that are correctly classified and precision represents the fraction of the road pixel classifications that are correct [7]. The F1 score [164] combines the precision and recall metrics within a single numeric score that is considered a balanced measure of accuracy when class sizes are different. In addition, the MCC is also a correlation coefficient between predicted and recognized binary classifications, providing a value between −1 and +1 [136]. The proportion of unions and intersections between the set of classified values and the set of ground truth is computed using MIOU. In MIOU, the number of classes *k* is equal to 2, presenting the road class and background. OA is also a simple summary assessment of a case's likelihood of being correctly classified [100]. IOU factor (Quality) is calculated by dividing the total number

of mutual pixels between the real and the classified masks by the total number of present pixels in both masks.

3.7. Summary

The following are the summaries obtained from the developed ML and DCNN models for road extraction and vectorization in this chapter:

1. Different types of high-resolution remote sensing images with various resolutions, such as Orthophoto, UAV, Aerial, and Google Earth images, were used to evaluate the models.

2. The datasets were used to generate the training, validation, and test images for training and assessing the suggested methods.

3. ML approaches such as Trainable Weka segmentation and Level Set methods, as well as multiresolution segmentation, classification methods (DT, KNN, and SVM), and connected components analysis were used to extract road networks.

4. State-of-the-art DCNN models such as BCD-UNet, MCG-UNet, VNet, and GAN+MUNet were applied to alleviate the issues of ML and pre-existing DL methods in detecting the road networks from heterogeneous areas due to the presence of occlusions.

5. The SC-RoadDeepNet method was applied to address the issues of road geometry and connection while also producing high-quality road segmentation maps.

6. The RoadVecNet technique was used to precisely and simultaneously extract road surface and vectorize road networks and address the challenges of existing methods that are only used for road surface and centerline extraction.

7. Some experimental settings were done for performing the methods, and the whole process of running the proposed DCNN models for road extraction and vectorization HRSI was implemented under the framework of Keras with Tensorflow backend.

8. The efficiency of the given DCNN algorithms for road surface segmentation and vectorization was evaluated using a variety of assessment metrics.

CHAPTER 4

RESULTS AND DISCUSSION

4.1. Introduction

The results of road surface segmentation from HRSI using developed ML and DL models are presented in this chapter. The road vectorization results obtained by the developed RoadVecNet model are also shown in this part. In addition, the quantitative comparison results of the proposed models and other comparative methods are demonstrated in this chapter. Moreover, the ablation studies for testing the methods with different parameters for road extraction and vectorization are discussed in this section. In this chapter, the generated high-resolution road segmentation and vectorization maps are also presented to show the effectiveness of the proposed models for the given tasks compared to the comparative techniques.

4.2. Results of traditional ML approaches for road segmentation

In this part, the results of the proposed traditional ML methods such as the Level Set segmentation method and integrated approach of segmentation and classification approach with connected components technique are described.

4.2.1. Results of Level Set method

To implement and calculate the accuracy of the suggested road extraction method from UAV images, several software, such as ImageJ (Weka), MATLAB and ArcMap were used. The UAV images were from urban and suburban areas, involving various road features in terms of type and form. In Figures 4.1(a) and 4.2(a), the main images of suburban and

urban areas were presented respectively. The test image in Figure 4.1 was retrieved from outside the city and was not surrounded by other features, whereas the original image in Figure 4.2 was completely covered by buildings, cars and vegetation. Figures 4.1(b) and 4.2(b) illustrate the segmented image using the TWS method. Both images were segmented by the algorithm with high precision. The segmented images were used as input for the LS method for road extraction (Figures 4.1, 4.2(c)). As shown in the figure, the road boundaries were extracted accurately using the LS method despite the presence of other objects on the road, but the objects with spectral characteristics similar to roads were also extracted, appearing as small irregular shapes (noise) in the extracted image. Therefore, after road extraction using the LS method, morphological operators were applied on the images for improving accuracy, given that morphological operators fill the gaps and eliminate nonroad pixels and noises (Figures 4.1, 4.2(d)). Table 4.1 also shows the confusion matrix factors (TN, FN, FP, TP) for evaluating the accuracy of road extraction based on the TWS and LS methods from UAV images.

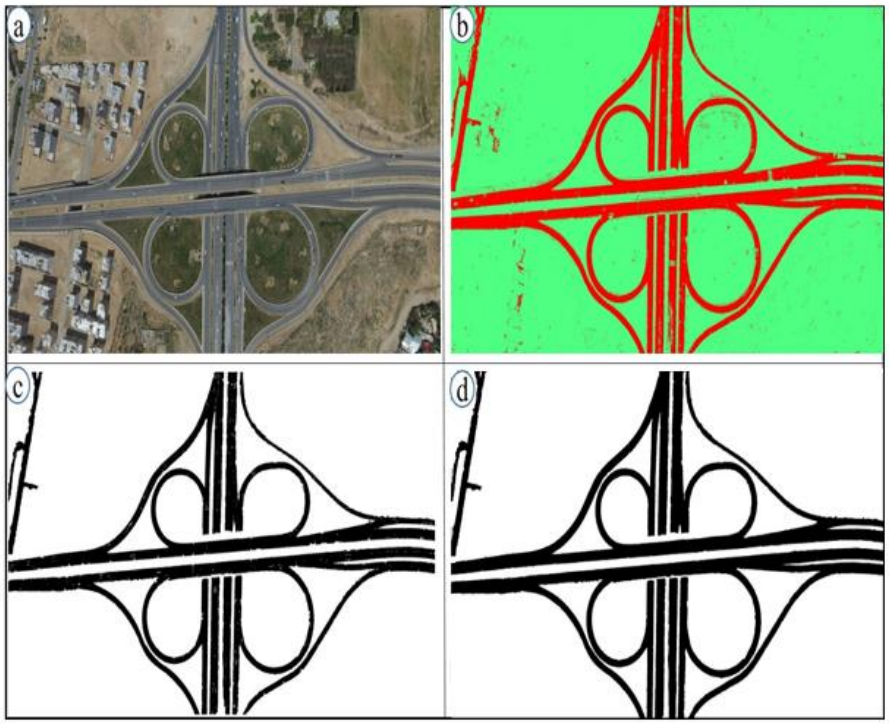

Figure 4.1. (a) Main image, (b) Segmented image, (c) result from Level Set, and (d) result from Morphological Operators.

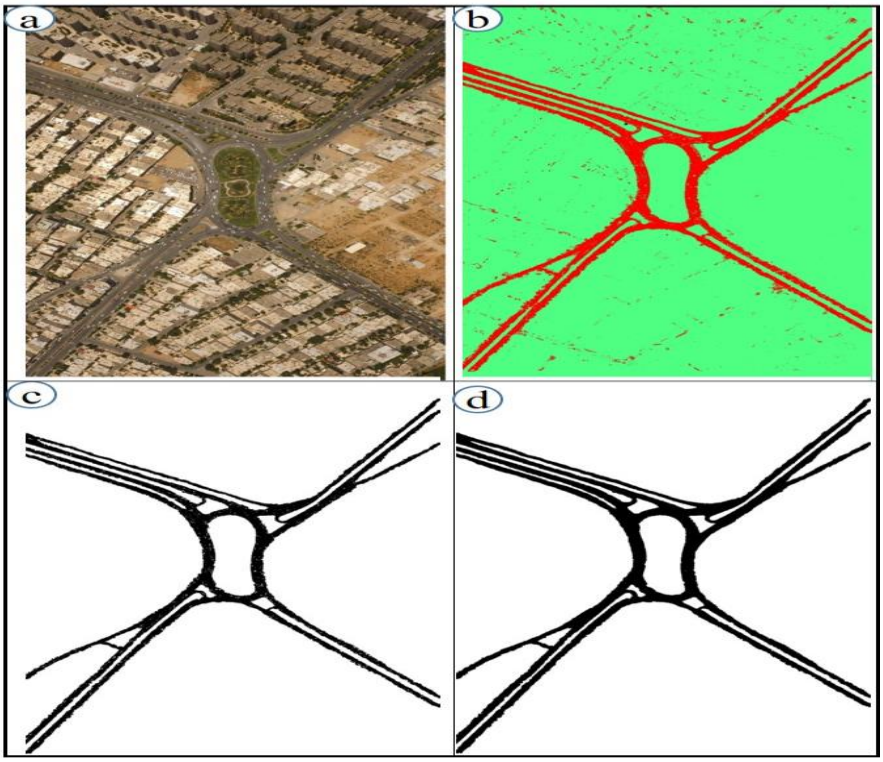

Figure 4.2. (a) Main image, (b) Segmented image, (c) result from Level Set, and (d) result from Morphological Operators.

Table 4.1. Road extraction accuracy using Confusion Matrix

	TN	FN	FP	TP
Figure 4.1	507069	8019	16103	128400
Figure 4.2	639015	7014	19241	92259

4.2.1.1. Discussion

The parameters of accuracy evaluation for the suggested road extraction approach, which are achieved on the basis of the parameters shown in Table 4.1, are exhibited in Table 4.2. As shown in the table, the results of completeness, correctness and quality of the proposed approach are 94.12%, 88.85% and 84.18%, respectively, for Figure 4.1 and 92.93%, 82.74% and 77.84%, respectively, for Figure 4.2. As specified in Table 4.1, the number of pixels not belonging to road class (FP) in Figure 4.2 is greater than the number of FP in Figure 4.1. In other words, the method could not recognise numerous pixels related to road class. Therefore, the precision of road extraction decreases slightly, especially for the correctness and quality parameters, due to the presence of vehicles. Therefore, these means extend the spectral heterogeneity, which influences the representation of linear structures by path opening. Moreover, the method produced a higher number of FPs in some sections where roads are near to built-up sections and buildings. Therefore, detaching road sections from their surroundings by solely relying on spectral characteristics is difficult, because these regions have a similar spectral reflectance as roads. Table 4.2 denotes that the suggested method is highly accurate in road extraction for both images. In terms of performance measures, the extracted road in Figure 4.2 has lower accuracy than that in Figure 4.1. The completeness percentage of the extracted roads is higher than the correctness and quality measures that are based on the performance factors. For Figure 4.2, although the suggested approach generated a good result, its extracted road still has lower accuracy than Figure 4.1. The accuracy reduction in Figure 4.2 is due to having more

similarities between road class and other features, such as vegetation and building shadows and urban complex texture and other obstacles, in which the suggested technique encounters difficulty in extracting road class and has less accuracy than Figure 4.1. However, using the TWS and LS techniques concurrently showed that the introduced approach has overall success for automatic road extraction. In this study, the TWS had a high accuracy in image segmentation, and using the segmented image as input for the LS method resulted in a highly accurate road extraction.

Table 4.2. Parameters of precision assessment

	Completeness	Correctness	Quality Percentage
Figure 4.1	94.12	88.85	84.18
Figure 4.2	92.93	82.74	77.84

To display clearly the effectiveness of the method used in this study, the performance measure factors of the suggested work were compared with other works. In this work, two test images were selected for measuring performance factors, whereas in other works the number of images taken for these factors varies. Therefore, the percentages of the average values of quality, completeness and correctness were considered for comparison with other methods. Huang, et al. [165] applied a feature fusion method based on the cross-validation line features of the statistical region merge and line segment detector, according to their spatial relation, to extract roads from remote sensing images. The performance factors are evaluated for different images, in which the average value of performance measures is taken for comparison. Miao, et al. [166] suggested a novel integrated approach according to tensor voting, kernel density estimator and geodesic technique for road centreline extraction from remote sensing images. They considered completeness, correctness and quality measures to calculate the precision of extracted road. Shi, et al. [167] applied a

method based on shape features and spectral–spatial classification for urban road extraction from a Ziyuan-3 satellite image with a spatial resolution of 6 m per pixel and an IKONOS image. They calculated completeness, correctness and quality factors, which are taken to compare with the proposed method in this study. Sujatha and Selvathi [103] suggested a technique to extract road centreline from various high-resolution satellite images automatically. They evaluated the performance measures to calculate the efficiency of their method, and the average values of those factors are presented in Table 4.3 for comparison. Results of the suggested method, along with the results from previous studies, are rounded off to the nearest integer, as shown in Table 4.3.

Table 4.3. Performance measures comparison of the proposed method with another works.

Approaches	Average of Completeness (%)	Average of Correctness (%)	Average of Quality (%)
[165]	87	89	76
[166]	87	92	83
[167]	79	77	63
[103]	90	96	87
Ours	94	86	81

The performance measures for the proposed method and other methods are plotted to illustrate the differences clearly (Figure 4.3). The x-axis is categorised into three sections: the first section presents the average value of completeness; the second section demonstrates correctness; and the last section shows the values for quality. The y-axis presents the percentages of the corresponding values. As shown in the plot, the percentage of completeness for the proposed approach lies on the top level, compared with other works. Furthermore, the values for correctness and quality of the proposed method in some

sections are higher than those of the other works, which proves that the method was remarkably efficiency for road extraction from UAV images.

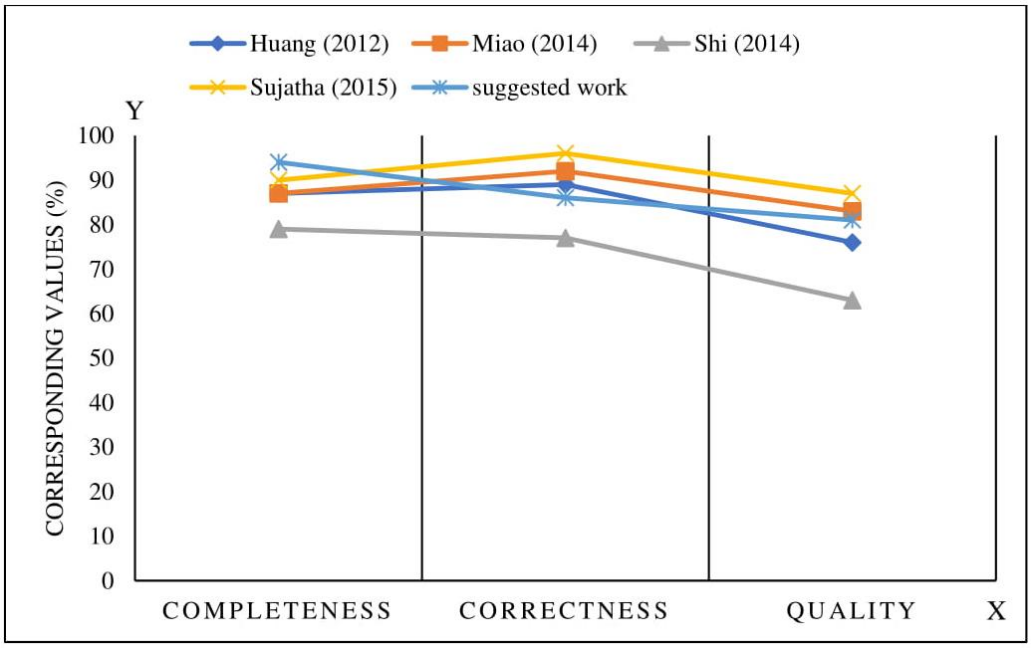

Figure 4.3. Comparison plot for performance factors.

4.2.2. Results of segmentation and classification methods with connected components analysis

Three images from different areas, in which road section is covered by some other objects, such as vegetation, vehicles, and buildings, were considered to demonstrate the efficiency of the proposed road extraction method in this work. Software, including MATLAB, eCognition Developer 64, and ArcMap, were used to apply the proposed method and calculate its efficiency in road extraction. I considered two sets of values for parameters such as scale, shape and compactness for the proposed segmentation approach to measure how the parameters of the method affect the detection accuracy. First, I set the values for the scale, shape and compactness parameters of the segmentation method to 50, 0.5 and 0.3 and then applied the classification methods, and the results are shown in Figure 4.4.

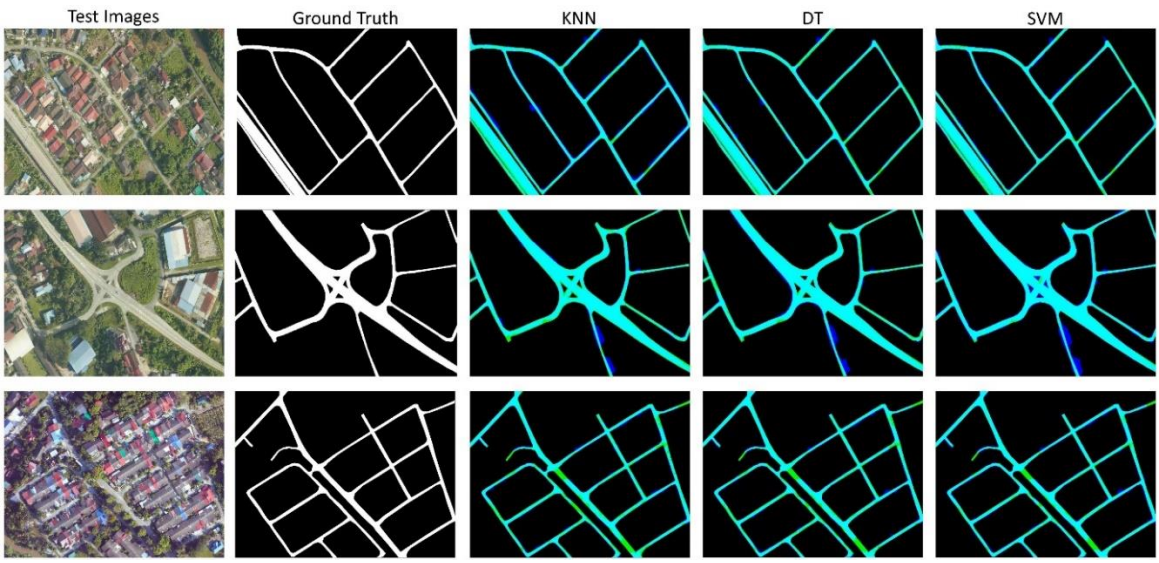

Figure 4.4. Extracted road class from orthophoto images with scale=50, shape=0.5 and compactness=0.3. First and second columns show the original image road label, respectively while third, fourth and fifth columns show the results of road detection by KNN, DT and SVM approaches, respectively.

Whereas Figure 4.5 shows the results of road detection by the methods after setting the values of scale, shape and compactness parameters to 20, 0.2 and 0.6, respectively. Both figures are illustrated in five columns and three rows. The first and second columns depict the original RGB images and original ground truth maps, respectively. The third, fourth and fifth columns depict the results of road detection by the KNN, DT and SVM approaches after integration with connected components analysis. Road parts in the main images of Figures 4.4 and 4.5 are evidently less or more covered by other occlusions with similar reflectance, making accurate road part extraction from images difficult. This phenomenon is due to the objects with the same spectral features, which possibly become visible as a road section in the extracted image. Consequently, OBIA, connected components analysis, and morphological operations were applied along with segmentation and classification method to obtain additional information, such as texture and geometry, and eliminate irrelevant road components and noises to improve the accuracy. As shown in Figures 4.4 and 4.5, the

proposed integration of KNN, DT, and SVM methods with connected components could generally extract accurate road section from orthophoto images. However, the three proposed classification methods demonstrated better performance for extracting road from images in Figure 4.5 with parameters values of scale=20, shape=0.2 and compactness=0.6 than those in Figure 4.4 with parameters values of scale=50, shape=0.5 and compactness=0.3. In both figures, the proposed SVM method could produce better qualitative results for road extraction with less false positive (FPs) prediction (shown as blue color) than other methods while KNN method predicted more FPs and less false negative (FNs) (shown as yellow color) and generated low-quality visualization results compared to other approaches.

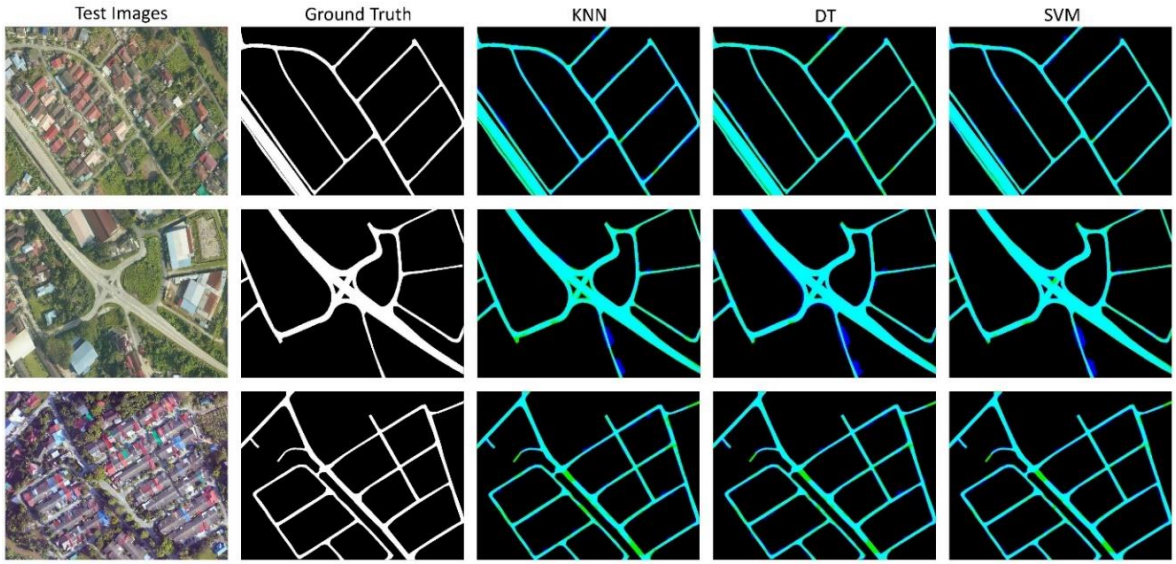

Figure 4.5. Extracted road class from orthophoto images with scale=20, shape=0.2 and compactness=0.6. First and second columns show the original image road label, respectively while third, fourth and fifth columns show the results of road detection by KNN, DT and SVM approaches, respectively.

A confusion matrix with four main factors (TN, FN, TP, and FP) was used for assessing the accuracy of the suggested approach because road part extraction from remote sensing image is a binary classification [55]. Several main metrics, such as recall, F1-score, and precision, were considered based on the parameters of the confusion matrix to evaluate the

capability of the introduced approach in road network extraction from orthophoto images. Table 4.4 demonstrates the quantitative results achieved by the proposed methods for Figure 4.4 and those for Figure 4.5 are presented in Table 4.5.

Table 4.4. Evaluated metrics for different methods (Figure 4.4). Best values are in bold and second-best values are underlined.

		KNN	DT	SVM
Image1	Recall	0.8833	0.8305	0.8485
	Precision	0.8112	0.8957	0.8765
	F1-score	0.8457	0.8619	0.8623
Image2	Recall	0.8881	0.9025	0.9326
	Precision	0.9095	0.9161	0.9044
	F1-score	0.8987	0.9092	0.9182
Image3	Recall	0.7851	0.8058	0.8547
	Precision	0.8998	0.8967	0.8823
	F1-score	0.8386	0.8488	0.8683
Average	Recall	0.8522	0.8463	**0.8786**
	Precision	0.8735	**0.9028**	0.8877
	F1-score	0.8610	0.8733	**0.8829**

Table 4.5. Evaluated metrics for different methods (Figure 4.5). Best values are in bold and second-best values are underlined.

		KNN	DT	SVM
Image1	Recall	0.8966	0.8492	0.8922
	Precision	0.8442	0.9167	0.8982
	F1-score	0.8696	0.8817	0.8952
Image2	Recall	0.8952	0.9318	0.9218
	Precision	0.9144	0.9023	0.9223
	F1-score	0.9047	0.9168	0.9220
Image3	Recall	0.8064	0.8475	0.8809
	Precision	0.8865	0.8722	0.8651
	F1-score	0.8446	0.8597	0.8730
Average	Recall	0.8661	0.8762	**0.8983**
	Precision	0.8817	**0.8971**	0.8952
	F1-score	0.8730	0.8861	**0.8967**

4.2.2.1. Discussion

Based on Table 4.4, the average percentage of F1 score metric is 86.10%, 87.33%, and 88.29% for KNN, DT and SVM methods, respectively. Meanwhile, the percentage of such metric presented in Table 4.5 is 87.30%, 88.61%, and 89.67% for KNN, DT and SVM, respectively. The suggested approaches evidently showed satisfactory performance in terms of road extraction from orthophoto images. However, the accuracy of specific measurements is slightly higher for all the methods in Figure 4.5 (with scale=20, shape=0.2 and compactness=0.6) than those in Figure 4.4 (with scale=50, shape=0.5 and compactness=0.3). As illustrated in Tables 4.4 and 4.5, the precision factor percentage is high for the DT model compared with that of the two other methods. However, the SVM model achieved a higher percentage in recall and F1 score than that of the two other methods, which demonstrates the effectiveness of the model for road extraction. In both tables, the KNN method was ranked the least in road detection. The poor road extraction performance of the KNN technique is related to its prediction of a large number of FPs and a smaller number of FNs, which results in poor accuracy. In contrast, the SVM model was ranked the number-one in road extraction in both. In fact, the SVM model could improve the results of F1 score to 2.19% and 0.96% compared to KNN and DT, respectively for Figure 4.4 and 2.37% and 1.06%, respectively for Figure 4.5. Figure 4.6 illustrates the average accuracy of the metrics achieved using the proposed road extraction methods for Figures 4.4 and 4.5. The vertical and horizontal axes shows the average percentage of accuracy and the three accuracy assessment metrics, respectively. As displayed in Figure 4.6, SVM model could achieve better quantitative results than KNN and DT. However, all the three proposed models showed a deficiency in road extraction when road parts are covered by occlusions, such as vehicles, shadows, vegetation, and buildings, and predicted more FP pixels. In addition, I measured the computational time

of the proposed methods applied on the three images, which the average running time among the approaches is shown in Table 4.6. As it is obvious, KNN method takes more time than DT and SVM for training with the average running time of 147.33. The reason is that we have to ascertain the value of parameter K (number of nearest neighbors) and the type of distance to be utilized. Therefore, the computation time is much as the model requires measuring the distance of every query instance to all training samples.

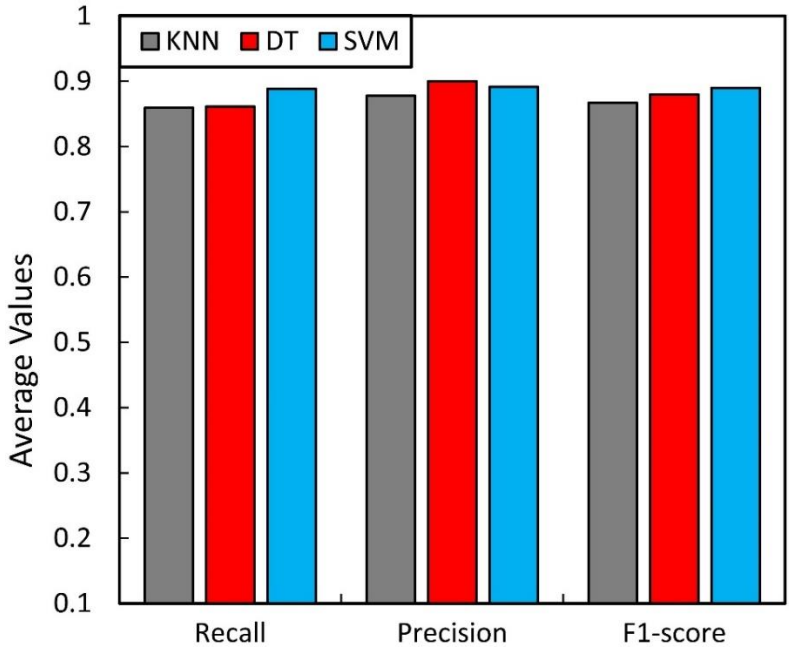

Figure 4.6. Comparison of average performance metrics achieved by the proposed methods for road extraction.

Table 4.6. Computational time comparison of various approaches. Here, the time is measured in second.

Methods	**Images**			
	Image1	Image2	Image3	Average
DT	140	104	139	127.66
KNN	142	105	195	147.33
SVM	141	106	172	139.66

In addition, the efficiency of the introduced approaches was compared with that of other works to demonstrate the effectiveness of the model for road extraction from orthophoto imagery. The average percentage of recall, precision and F1 score metrics were considered

for comparison. A method for road extraction from Ziyuan-3 satellite images based on spectral–spatial classification and shape features was introduced by [167]. Recall, precision and F1 score metrics were calculated for the accuracy assessment, in which the average values were obtained caught for comparison. Miao, et al. [49] extracted road sections from remotely sensed images according to a fusion method of geodesic, kernel density, and tensor voting techniques. They evaluated recall, precision and F1 score measures to assess the performance, in which the average amount is obtained for comparison with the suggested techniques in this paper. A technique for road extraction from different high-resolution remote sensing images was also introduced by [117], in which the average percentage of recall, precision and F1 score factors are obtained for comparison. Table 4.7 depicts the average amount of performance metrics for the proposed methods in this study and other prior studies.

Table 4.7. Performance factors of different proposed methods compared with various previous studies. Best values are in bold.

Methods	Recall	Precision	F1 score
Proposed DT	0.8762	0.8971	0.8861
Proposed KNN	0.8661	0.8817	0.8730
Proposed SVM	**0.8983**	0.8952	**0.8967**
[167]	0.79	0.77	0.7798
[49]	0.87	**0.92**	0.8943
[117]	0.86	0.91	0.8842

Table 4.7 shows that the three proposed SVM method in this study demonstrated a higher percentage in F1 score factor compared with that from previous works. The DT method is ranked third with 88.61%, while SVM is ranked first with 89.67%. By contrast, the average value of F1 score for the second-best method (Miao, et al. [49]) is 89.43%, which could achieve better results than the proposed KNN and DT methods. Miao, et al. [49] also achieved a high precision amount with 92%, which is more than the average percentage of precision for the three proposed methods with 89.52%, 89.71%, and 88.17% for SVM, DT, and KNN. The decreasing accuracy for the proposed methods is due to the high FP amount prediction, which affected the percentage of precision. Also, Shi, et al. [167] obtained the lowest amount of F1 score with 77.98%, indicating that their method was ineffective in road extraction. By comparing the quantitative results, it can be seen that the three proposed classification methods integrated with connected components analysis demonstrated efficiency in road extraction from orthophoto images**.**

4.3. Results of DCNN methods for road segmentation (Objective 1)

The results of the proposed DCNN approaches such as MCG-UNet, BCD-UNet, VNet, and GAN+MUNet for road segmentation from different HRSI data based on the first objective are described in this section. The proposed techniques could generate high-quality road segmentation maps even under complex environments compared to the traditional methods discussed in the previous section and other comparative DL methods that are discussed in this part.

4.3.1. Results of GAN+MUNet

Figure 4.7 visually illustrates the results obtained with the proposed MUNet and GAN models for some images with varied characteristics, specifically including non-complex and

complex backgrounds, shadows, and occlusions due to trees and buildings. From the results in the figure, one can observe that, while both the proposed approaches could extract and detect roads in the images with good accuracy, the GAN framework offered several advantages over the MUNet approach. The MUNet approach was sensitive to occlusion by trees and to shadows and predicts few FN pixels (depicted in blue box in Figure 4.7) but its accuracy compromised due to a number of FP pixels (depicted in yellow box in Figure 4.7). Given that the textural and spectral characteristics of parking lots, shadows, and buildings frequently match those of roads, the proposed MUNet model could not reliably distinguish roads from these other elements, resulting in incorrect classification for several small patches. Moreover, some of the extracted road parts are not continuous; lack of connectivity is observed between the roads at junction regions where roads connect. For complex images, extracting road parts can be challenging for the proposed MUNet model. The proposed GAN model offered a significant improvement over the MUNet approach and generated more coherent high-resolution road segmentation maps with better preservation of the road borders and mitigation of the effects of occlusions and shadows. Compared to the MUNet approach the GAN approach predicted fewer FP pixels, which is a key contributor to the improved accuracy.

The accuracy of the proposed MUNet and GAN models was also evaluated numerically in terms of the five metrics and the results are summarized in Table 4.8. The numerical results in Table 4.8 reinforce the findings from the visually presented results in Figure 4.7; compared with the MUNet model the GAN model provided significantly higher precision but slightly lower recall, indicating that the MUNet model predicted more false positive and less false negative pixels than the GAN model. For the combined F1 score and MCC accuracy metrics, the GAN model achieved scores of 92.20%, and 91.13% compared with scores of 90.18%,

and 88.92% for the MUNet, respectively. The improvements of 2.02% and 2.21%, respectively, for F1 score and MCC demonstrate the superiority of the proposed GAN approach for road extraction. Although the proposed GAN approach offered state of the art performance, it is also impacted by the complicated backgrounds and occlusions, as well as the challenge of common spatial and spectral characteristics of roads with other regions, such as parking lots, and buildings. I also conducted some experiments to check the effect of different hyper-parameters on the performance of the model for road extraction. I changed the Adam optimizer to Stochastic gradient descent (SGD) with a learning rate of 0.001 and ReLU activation function used in the encoder part of the model to Exponential linear unit (ELU). I then performed the Prop-GAN with these hyper-parameters (Pro-GAN+ELU+SGD) on the dataset. I measured the evaluation metrics for the same test images after adding the SGD and ELU parameters. I achieved an average accuracy of 88.01% for Precision, 92.02% for F1 score, 90.99% for MCC, and 87.25% for MIOU. As it is shown, the Prop-GAN approach with ReLU and Adam parameters (Prop-GAN+ReLU+Adam) obtained better accuracy and improved the results by 3.53%, 0.18%, 0.14%, and 0.18% for Precision, F1 score, MCC, and MIOU, respectively. In contrast, the Prop-GAN+ELU+SGD method obtained 96.43% for Recall compared to the Pro-GAN+ReLU+Adam with 92.92%, which shows that more FPs and fewer FNs were predicted by the method. Furthermore, I depicted some qualitative results of the Prop-GAN+ELU+SGD method in Figure 4.7 (e). As it can be seen, compared with the same test images in Figure 4.7 achieved with Prop-GAN+ReLU+Adam, more non-road pixels were predicted by the Prop-GAN+ELU+SGD, which leads to obtaining less accurate qualitative results compared to the Prop-GAN+ReLU+Adam.

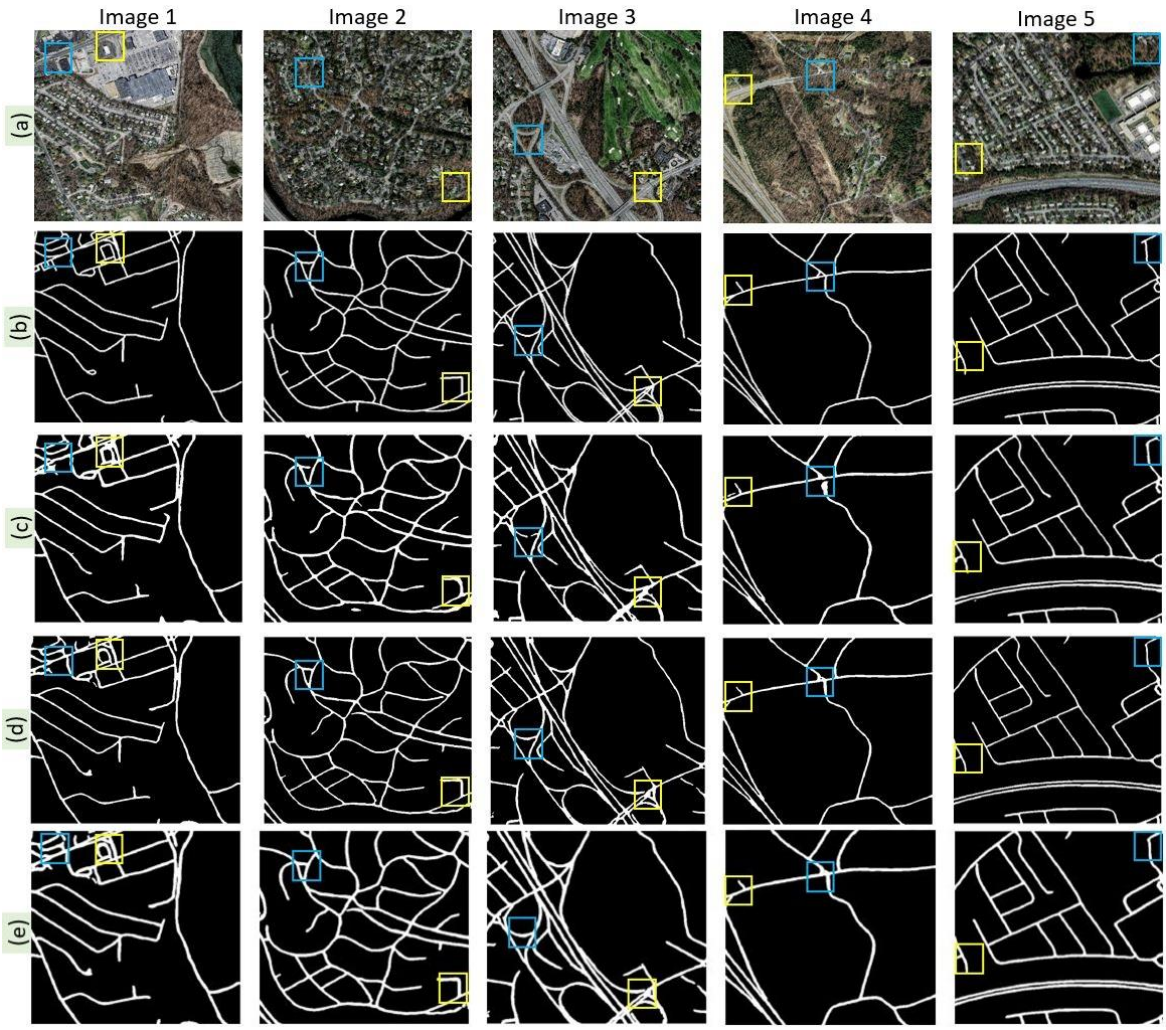

Figure 4.7. Sample image blocks and corresponding extracted road regions using alternative techniques: (a) image block, (b) ground truth road segmentation, (c) road segmentation obtained with the proposed modified U-Net model (Prop-MUNet), (d) road segmentation obtained with the proposed GAN approach (Prop-GAN+ReLU+Adam), and (e) road segmentation obtained with the proposed GAN approach with new parameters (Prop-GAN+ELU+SGD). The blue and yellow boxes present the FNs and FPs, respectively.

Table 4.8. Quantitative accuracy metrics for the proposed approaches for the individual images in the Massachusetts road dataset. values are reported in percentage, and the best metrics are indicated by bold font.

		Prop-MUNet	Prop-GAN
Image 1	Recall	95.80	93.45

	Precision	79.51	86.83
	F1 score	86.90	90.02
	MCC	85.38	88.65
	MIOU	80.02	84.24
Image 2	Recall	94.30	91.25
	Precision	79.66	89.19
	F1 score	86.36	90.21
	MCC	84.29	88.57
	MIOU	79.91	84.91
Image 3	Recall	93.50	91.28
	Precision	84.53	89.70
	F1 score	88.79	90.48
	MCC	86.98	88.91
	MIOU	83.02	85.27
Image 4	Recall	96.88	95.24
	Precision	87.12	92.91
	F1 score	91.74	94.06
	MCC	91.02	93.47
	MIOU	86.24	89.86
Image 5	Recall	94.29	91.86
	Precision	88.99	93.37
	F1 score	91.56	92.61
	MCC	90.38	91.58
	MIOU	86.55	88.04
Image 6	Recall	96.15	93.34
	Precision	89.36	93.63
	F1 score	92.63	93.48
	MCC	91.74	92.66
	MIOU	87.97	89.22
Image 7	Recall	94.15	91.94
	Precision	91.41	95.28
	F1 score	92.76	93.58
	MCC	91.98	92.92
	MIOU	87.87	89.13
Image 8	Recall	95.03	95.02
	Precision	86.83	91.42
	F1 score	90.74	93.19
	MCC	89.60	92.31
	MIOU	88.43	88.82
Average	Recall	**95.01**	92.92
	Precision	85.92	**91.54**
	F1 score	90.18	**92.20**
	MCC	88.92	**91.13**
	MIOU	85.00	**87.43**

4.3.1.1. Comparison and discussion

The performance of the proposed MUNet and GAN approaches over the Massachusetts road dataset was also compared against six state-of-the-art prior approaches for road extraction from high resolution aerial imagery: (1) The SEEDS-MCNN proposed recently by Lv, et al. [168], which used super-pixels extracted via energy-driven sampling (SEEDS) followed by a CNN classifier, (2) The CNN approach of Zhong, et al. [37], (3) The RSRCNN approach of Wei, et al. [55] which used road structure-refined CNN model that is provided with road geometric information and spatial correlation, (4) The Road-RCF technique proposed by Hong, et al. [44] which used richer convolutional features (RCFs) for road extraction, (5) the RDRCNN approach proposed by Gao, et al. [79] which used a novel architecture called the refined deep residual CNN composed of dilated perception and residual connected units, and (6) the RDRCNN+Postprocessing approach of Gao, et al. [79] which performed a post-processing step on the RDRCNN output using mathematical morphology and a tensor-voting method to incorporate split roads.

Table 4.9. Average precision, recall, and F1 score metrics over the Massachusetts road dataset for the proposed approach and alternative techniques. for each metric, the best value obtained across the different methods is indicated by bold font.

	Average Percentage		
	Recall	Precision	F1 score
CNN [37]	68.6	43.5	53.2
SEEDS-MCNN [168]	80.4	78.0	79.0
RSRCNN [55]	72.9	60.6	66.2
Road-RCF [44]	**98.5**	85.8	91.5
RDRCNN [79]	75.33	84.64	79.72
RDRCNN + post-processing [79]	75.75	85.35	80.31
Prop-MUNet	95.01	85.92	90.18
Prop-GAN	92.92	**91.54**	**92.20**

The referenced publications for these prior methods reported precision, recall, and F1 score on the Massachusetts road dataset and those values are compared in Table 4.9 against the corresponding metrics for the MUNet and GAN approaches proposed in this paper.

The results in Table 4.9 demonstrate the effectiveness of the proposed GAN approach, which provided the highest F1 score among all the methods compared, which at 92.20% is 0.7% better than the next best performing Road-RCF technique and 2.02% better than the proposed MUNet approach. The proposed GAN approach yielded the highest precision metric, which at 91.54% is almost 5.74% better than the next best Road-RCF [44] technique and 5.62% better than the proposed MUNet approach, which has the third best precision value. The proposed GAN approach also has a high recall metric, which at 92.92% is only superseded by the 98.5% value for the Road-RCF [44] technique but is better than all other prior methods and only slightly worse than the 95.01% value for the proposed MUNet. Among the prior methods, the Road-RCF [44] technique offered performance that is clearly superior to other methods in all three reported metrics. The CNN and RSRCNN achieved the lowest accuracy compared with the other methods and our proposed methods in this paper. In addition to the numerical results presented in Table 4.9, I also presented a sample set of images and extracted road regions for the images to highlight and compare the performance of the alternative techniques that are depicted in Figure 4.8. The first and second columns present the test and ground truth images, whereas the third column depict the results achieved by the state-of-the-art SEEDS-MCNN Lv, et al. [168], CNN [37] and RDRCNN [79] methods. The fourth column shows the results achieved by the state-of-the-art Road-RCF [44], RSRCNN [55] and RDRCNN [79]. Finally, the fifth and sixth columns illustrate the extracted road parts using proposed MUNet and GAN models, respectively. These images further highlighted the

effectiveness of the proposed GAN approach, which is particularly effective in preserving the edges of the roads while maintaining high fidelity with the ground truth labels.

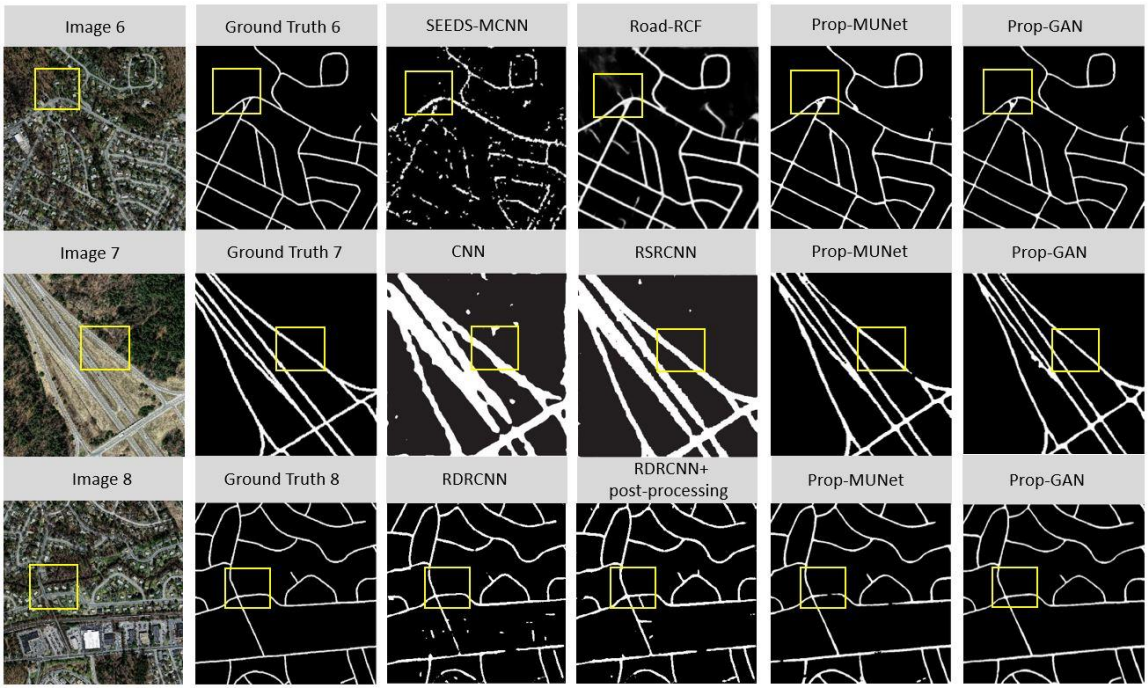

Figure 4.8. Comparison of road segmentation obtained with the proposed method (GAN) against other techniques illustrated on the three images from the Massachusetts road dataset. The yellow boxes highlight regions with the FP and FN pixel predictions by the models.

Also, I compared the performance of the proposed GAN+MUNet approach with other GAN-based road extraction approaches reported in the literature such as GAN+FCN [42], GAN+SegNet [88], E-WGAN [90], MsGAN [92], and McGAN [94] to test the efficacy of the presented model in road extraction. For comparison purpose, that the statistical measure such as the accuracy, recall, and F1 scores reported in the referenced papers vs. our proposed Prop-GAN approach are shown in Table 4.10. The quantitative results indicate that the presented GAN+MUNet model attained the highest F1 score value with 92.20%, which could improve the earlier methods by 2.57% compared to the second highest approach called GAN+SegNet. Also, the model could improve the F1 score value

compared to the other GAN-based road extraction methods such as GAN+FCN, E-WGAN, MsGAN, and McGAN to 5.2%, 7.2%, 6%, and 7.3%, respectively, assert the GAN+MUNet model's ability to extract roads from aerial imagery. Also, I estimated the runtime of the suggested approach applied on the dataset, which took 78.6s per epoch and 71ms per step for training and testing process, respectively. The model was trained for 100 epochs and tested on 28 images; thus, it took 131 minutes for training and 2s for testing. Overall, the proposed model does not require high computational time and a large training dataset and still achieved the best performance among other comparative models in term of both quantitative and qualitative results.

Table 4.10. Average precision, recall, and F1 score metrics for the proposed GAN+MUNet and alternative GAN-based road detection approaches. bold font indicates the best value.

	Average Percentage		
	Recall	**Precision**	**F1 score**
GAN+FCN	82	93	87
GAN+SegNet	91.01	88.31	89.63
E-WGAN	85	86	85
MsGAN	87.1	85.3	86.2
McGAN	85.8	84.1	84.9
Prop-GAN	92.92	91.54	**92.20**

4.3.2. Results of VNet

In this section, the results achieved by the proposed VNet approach based on CE, DL and CEDL loss functions are highlighted. Figure 4.9 and Figure 4.10 illustrate the obtained results via the suggested technique based on CE loss function, DL and CEDL for Massachusetts road dataset and Ottawa dataset, respectively. The figures are represented in six columns and five rows. The main RGB images, the ground truth labels, the results achieved by the VNet+CE, VNet+DL and VNet+CEDL are presented in the first, second, third, fourth and last row, respectively. Also, the second, fourth and sixth columns show the zoomed outcomes. Based on the figures, the suggested VNet network with all loss

functions could generally segment road class from high-resolution remote sensing data precisely. However, the results achieved by VNet+CEDL is more accurate than VNet+CE and VNet+DL. In fact, VNet+CE and VNet+DL predicted more false positive pixels (FPs) (shown as blue pixels) and less false negative pixels (FNs) (shown as red pixels) in both datasets that lead to achieving lower accuracy compare to VNet+CEDL for road extraction. The proposed VNet+CE and VNet+DL models could not segment road part from remote sensing data where the road network is covered by shadows or in the junction parts. Therefore, by using new CEDL loss function that consider both local and global information and solve the issue of lessening the influence of class imbalance, the proposed VNet plus CEDL could improve the results.

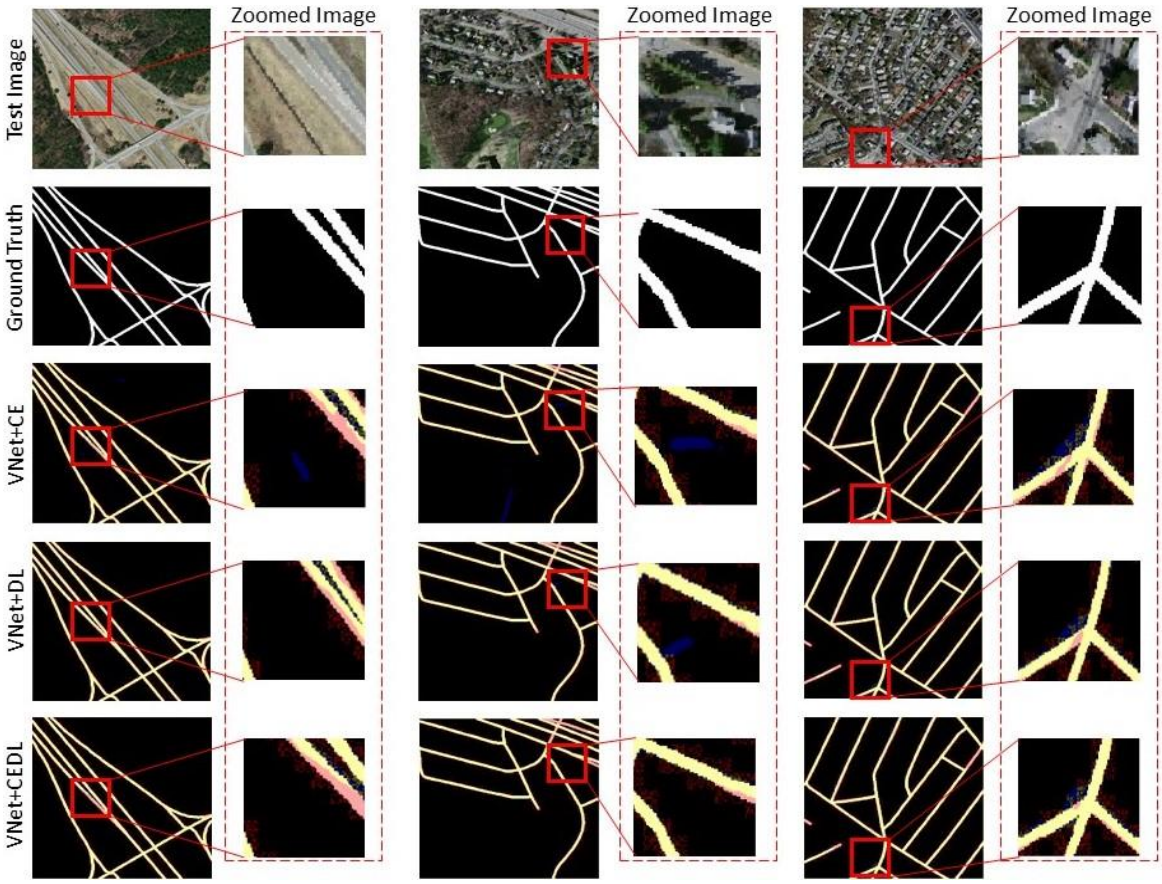

Figure 4.9. The achieved outcomes using the proposed VNet+CE, VNet+DL and VNet+CEDL from Massachusetts road dataset. The second, fourth and sixth columns present the zoomed outcomes of the prior column. The black, yellow, blue, and red colors show the TNs, TPs, FPs, and FNs, respectively.

Moreover, I assessed the accuracy measurements of VNet+CE, VNet+DL, and VNet+CEDL for Massachusetts and Ottawa datasets to probe the capability of the proposed network for road extraction. Table 4.11 and Table 4.12 depict the accuracy of each defined metric for the Massachusetts and Ottawa road datasets, respectively. As it can be seen from both Tables, the proposed VNet+CEDL model could achieve higher average accuracy than VNet+CE and VNet_DL for F1 score, MCC and IOU with 90.11%, 88.30% and 82.07%, respectively for Massachusetts dataset; and 93.54%, 89.89% and 87.93%, respectively for Ottawa dataset.

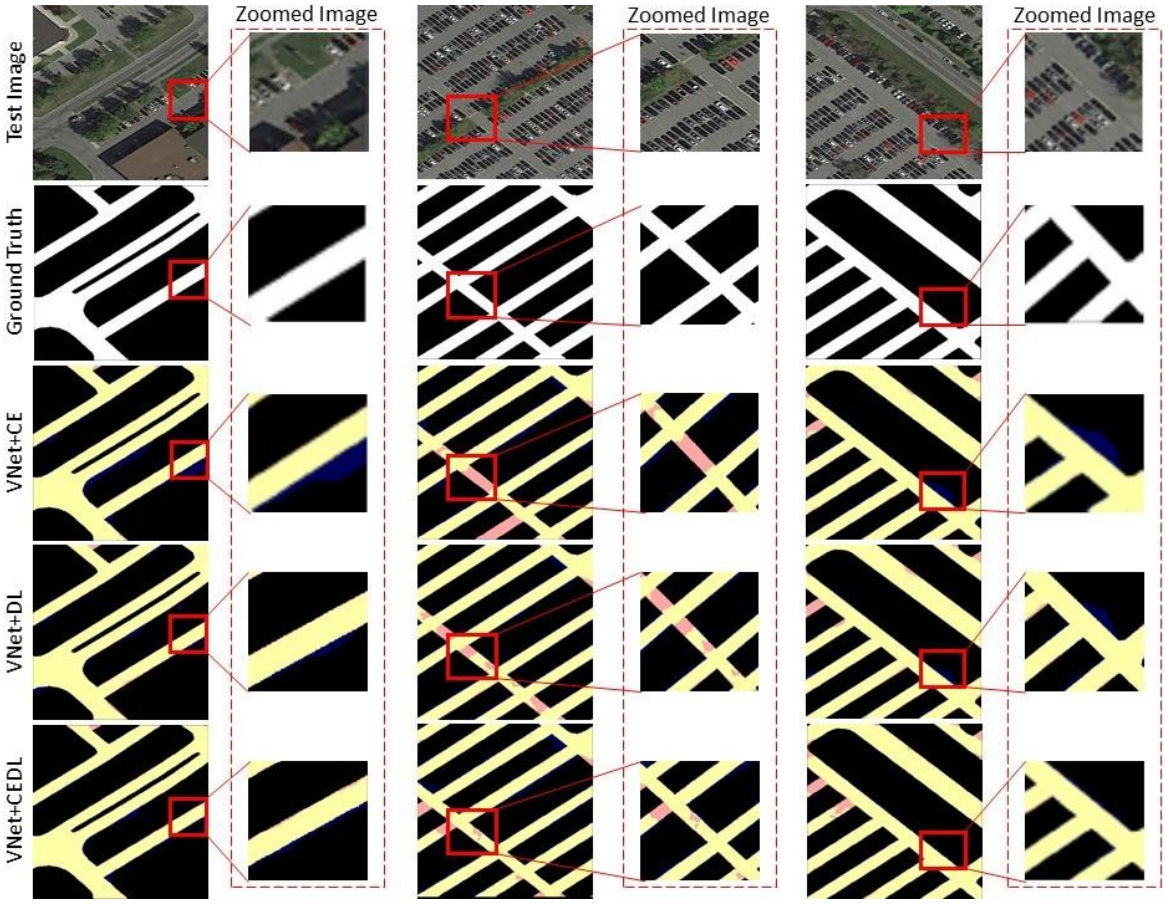

Figure 4.10. The achieved outcomes using the suggested VNet+CE, VNet+DL and VNet+CEDL from Ottawa road dataset. The second, fourth and sixth columns present the zoomed outcomes of the prior column. The black, yellow, blue, and red colors show the TNs, TPs, FPs, and FNs, respectively.

Note that the proposed VNet+CEDL network achieves good results for the road segmentation from both datasets and determines that the segmented road sections are close to labels, verifying the effectiveness of our approach in road extraction. Furthermore, the suggested VNet+CEDL network could maintain edge information and achieve higher precision on the segmentation boundary than the other comparative approaches.

Table 4.11. Comparing VNet model with CE, DL and CEDL loss functions for road extraction form Massachusetts dataset.

		F1 score	MCC	IOU
VNet+CE	Image1	0.9242	0.9103	0.8590
	Image2	0.8825	0.8633	0.7897
	Image3	0.8616	0.8347	0.7568
	Average	0.8894	0.8694	0.8018
VNet+DL	Image1	0.9268	0.9133	0.8635
	Image2	0.8973	0.8801	0.8138
	Image3	0.8705	0.8454	0.7706
	Average	0.8982	0.8796	0.8159
VNet+CEDL	Image1	0.9289	0.9159	0.8672
	Image2	0.9027	0.8865	0.8225
	Image3	0.8717	0.8466	0.7725
	Average	0.9011	0.8830	0.8207

Table 4.12. Comparing VNet model with CE, DL and CEDL loss functions for road extraction form Ottawa dataset.

		F1 score	MCC	IOU
VNet+CE	Image1	0.9361	0.9060	0.8799
	Image2	0.8814	0.8099	0.7878
	Image3	0.9391	0.9030	0.8852
	Average	0.9189	0.8730	0.8510
VNet+DL	Image1	0.9551	0.9338	0.9141
	Image2	0.9022	0.8437	0.8218
	Image3	0.9416	0.9074	0.8896
	Average	0.9329	0.8949	0.8751
VNet+CEDL	Image1	0.9563	0.9356	0.9162
	Image2	0.9069	0.8509	0.8295
	Image3	0.9431	0.9102	0.8922
	Average	0.9354	0.8989	0.8793

4.3.2.1. Discussion

The obtained measurement factors in the current work and in other studies were compared to further explore the benefit of the suggested approach for road network extraction from high-resolution remote sensing imagery. For comparison, I used the results achieved by

VNet+CEDL for both Massachusetts and Ottawa datasets as it shows better results in road extraction compared to VNet+CE. Particularly, the proposed approach was compared with some deep learning-based neural networks such as UNet framework introduced by [95], FCN proposed by [31] for image semantic segmentation and SegNet architecture applied by [96] for sematic pixel-wise segmentation. The quantitative results achieved by the proposed technique and other comparisons approaches for both Massachusetts and Ottawa road datasets are illustrated in Table 4.13 and Table 4.14. By comparing the results achieved for each metric, the difference between the precision for road extraction can be observed.

Table 4.13. Quantitative outcomes achieved by the VNet+CEDL and other techniques for Massachusetts dataset.

		FCN	**SegNet**	**UNet**	**VNet_CEDL**
Image1	F1 score	0.9120	0.8916	0.9283	0.9314
	MCC	0.8993	0.8759	0.9180	0.9216
	IOU	0.8381	0.8043	0.8662	0.8716
Image2	F1 score	0.9058	0.8933	0.9076	0.9087
	MCC	0.8891	0.8742	0.8910	0.8924
	IOU	0.8278	0.8072	0.8308	0.8327
Image3	F1 score	0.9048	0.9083	0.9087	0.9152
	MCC	0.8887	0.8928	0.8933	0.9008
	IOU	0.8261	0.8320	0.8327	0.8437
Image4	F1 score	0.8785	0.8392	0.8864	0.8933
	MCC	0.8489	0.8022	0.8588	0.8675
	IOU	0.7832	0.7229	0.7959	0.8072
Image5	F1 score	0.9032	0.9041	0.9084	0.9105
	MCC	0.8860	0.8874	0.8922	0.8947
	IOU	0.8235	0.8249	0.8322	0.8356
Average	F1 score	0.9009	0.8873	0.9079	0.9118
	MCC	0.8824	0.8665	0.8907	0.8954
	IOU	0.8197	0.7983	0.8316	0.8382

As illustrated in Table 4.13 and 4.14, the proposed VNet+CEDL model could achieve higher average accuracy for the whole three evaluation metrics (F1 score, MCC and IOU) than other cutting-edge deep learning-based techniques for both datasets. In fact, the model predicts less FPs and more FNs than other methods, leading to the improve in the results for IOU factor with almost 0.66%, 3.99% and 1.85% compared to UNet, SegNet and FCN for Massachusetts dataset respectively and 4.38%, 3.09% and 11.17% for Ottawa dataset, respectively.

Table 4.14. Quantitative outcomes achieved by the VNet+CEDL and other techniques for Ottawa dataset.

		FCN	**SegNet**	**UNet**	**VNet+CEDL**
Image1	F1 score	0.8665	0.907	0.8867	0.9236
	MCC	0.799	0.8501	0.8181	0.8793
	IOU	0.7643	0.8297	0.7965	0.858
Image2	F1 score	0.7886	0.9005	0.8602	0.9093
	MCC	0.7193	0.8538	0.7864	0.8622
	IOU	0.6509	0.8189	0.7547	0.8336
Image3	F1 score	0.9046	0.887	0.91	0.9178
	MCC	0.8455	0.8202	0.8565	0.8662
	IOU	0.8258	0.797	0.8347	0.848
Image4	F1 score	0.8034	0.8826	0.8876	0.9009
	MCC	0.7268	0.8388	0.8415	0.8608
	IOU	0.6714	0.7899	0.7979	0.8197
Average	F1 score	0.8408	0.8943	0.8861	0.9129
	MCC	0.7727	0.8407	0.8256	0.8671
	IOU	0.7281	0.8089	0.7960	0.8398

Moreover, Figures 4.11 and 4.12 depict the visual results obtained by the introduced VNet+CEDL model and other state-of-the-art deep learning-based techniques for Massachusetts road dataset and Ottawa dataset, respectively to show the proficiency of the proposed model in road extraction. The results demonstrate that the proposed deep learning-based models could generally reduce the effect of obstacles to a particular degree as they are using spatial information for segmentation. However, the other methods such

as UNet, SegNet and FCN could anticipate more FNs than the proposed VNet+CEDL model that could predict less FPs and consequently could achieve better results. This is because this technique could obtain and preserve boundary information that leads to anticipating less FPs and achieving high-resolution and smooth segmentation maps compared to the other deep learning-based methods.

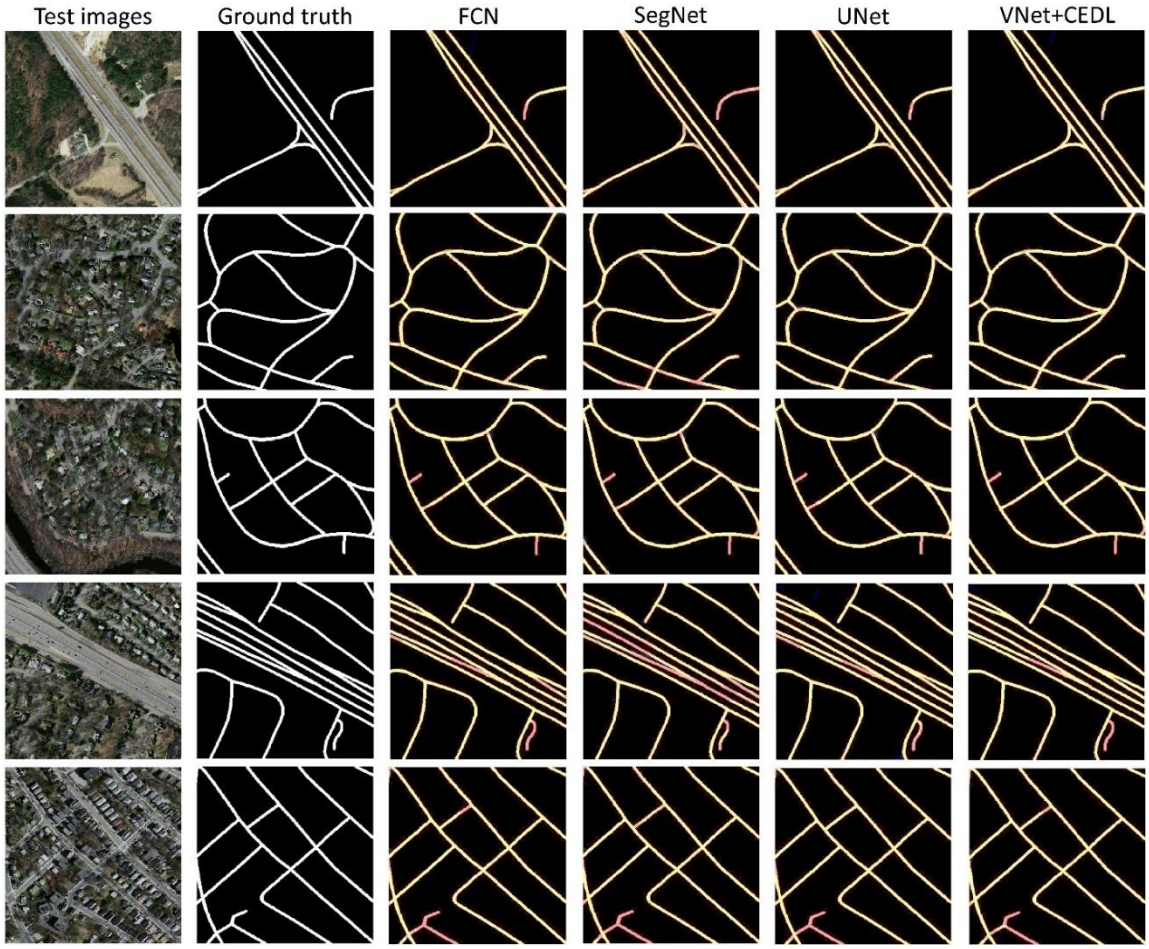

Figure 4.11. Road segmentation results obtained by the proposed VNet+CEDL against other comparison approaches from the Massachusetts road dataset. The yellow, blue, and red colors show the TPs, FPs and FNs, respectively.

In addition, I compared the results achieved by the proposed model with more several deep learning-based networks such as CNN [37] and road structure-refined CNN model

(RSRCNN) [55] for Massachusetts dataset and CasNet [71] and RoadNet [163] for Ottawa dataset.

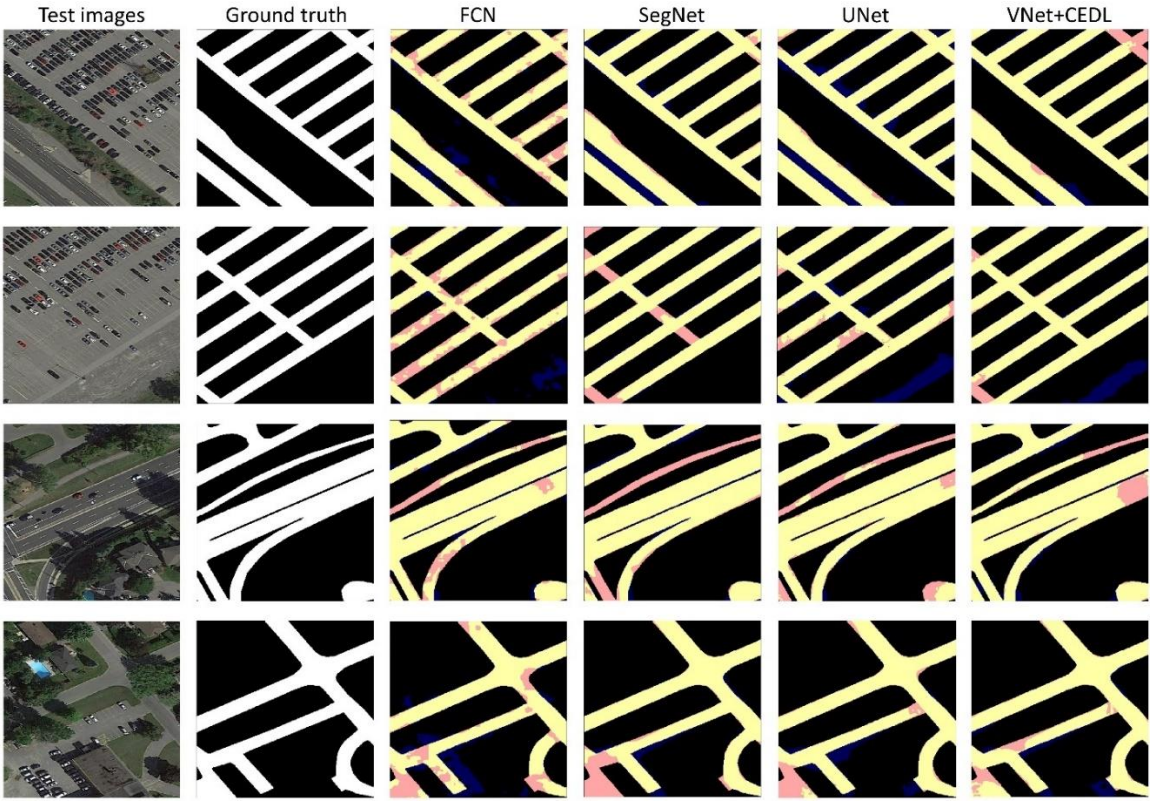

Figure 4.12. Road segmentation results obtained by the proposed VNet+CEDL against other comparison approaches from the Ottawa road dataset. The yellow color, blue and red colors depict the TPs, FPs, and FNs, respectively.

Note that the outcomes for the other methods were chosen from the main published papers, whereas the suggested approach has been performed and tested on the experimental datasets. The visualization and quantitative results for the proposed network and other comparative methods are shown in Figure 4.13, Table 4.15 and Table 4.16, respectively. In terms of F1 score, the outcomes illustrate that our suggested technique was superior to all other approaches. Our suggested technique achieved higher F1 score than those of [37] and [55], at 39.69% and 26.69% for Massachusetts and higher F1 score than those of [71] and [163], at 0.67% and 0.17% for Ottawa dataset, respectively.

Table 4.15. Quantitative values on the testing data of Massachusetts dataset in terms of F1 score.

Method	CNN	RSRCNN	VNet+CEDL
F1 score	0.5320	0.6620	0.9289

Table 4.16. Quantitative values on the testing data of Ottawa dataset in terms of F1 score.

Method	CasNet	RoadNet	VNet+CEDL
F1 score	0.9340	0.9390	0.9407

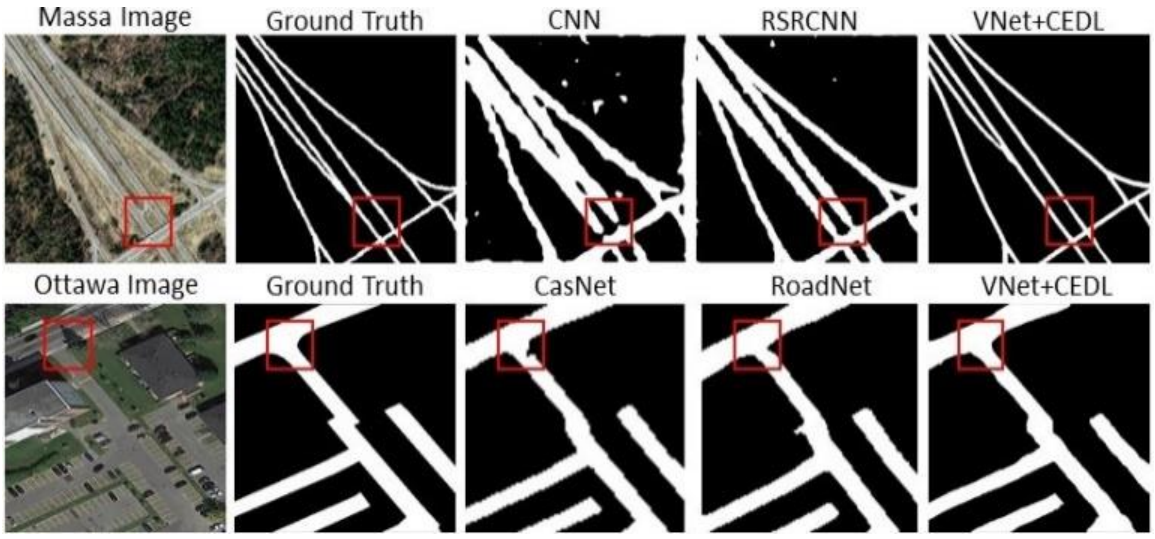

Figure 4.13. Comparison of road segmentation achieved with the suggested approach (VNet+CEDL) against other techniques for Massachusetts and Ottawa datasets.

4.3.3. Results of BCD-UNet and MCG-UNet

To show the ability of the presented BCD-UNet and MCG-UNet models for road object extraction, I measured the accuracy assessment factors. Table 4.17 depict the accuracy of every specified measurement factor for road extraction. The average F1 score achieved by the UNet, BCL-UNet, and MCG-UNet is 86.89%, 87.55%, and 88.74%, respectively, for road extraction. Clearly, the MCG-UNet model worked better than the other approaches in road extraction and could improve the F1 score to 1.19% and 1.85% compared with the BCL-UNet and UNet models, respectively, for road segmentation results.

Table 4.17. Comparison of the MCG-UNet, BCL-UNet, and UNet networks for road segmentation.

	Metrics	UNet	BCL-UNet	MCG-UNet
Image1	Recall	0.8592	0.8604	0.8643
	Precision	0.8757	0.8801	0.9051
	F1 score	0.8674	0.8701	0.8842
	MCC	0.8431	0.8465	0.8637
	IOU	0.7657	0.7701	0.7924
Image2	Recall	0.8277	0.8374	0.8984
	Precision	0.884	0.887	0.8984
	F1 score	0.8549	0.8615	0.8984
	MCC	0.8283	0.8358	0.8797
	IOU	0.7466	0.7567	0.8156
Image3	Recall	0.857	0.8589	0.8672
	Precision	0.9043	0.9165	0.9191
	F1 score	0.88	0.8868	0.8924
	MCC	0.8546	0.8632	0.8699
	IOU	0.7857	0.7965	0.8057
Image4	Recall	0.7787	0.7831	0.7658
	Precision	0.8874	0.8924	0.905
	F1 score	0.8295	0.8342	0.8296
	MCC	0.7943	0.80	0.7969
	IOU	0.7086	0.7154	0.7088
Image5	Recall	0.9026	0.9097	0.9340
	Precision	0.9233	0.9410	0.9312
	F1 score	0.9128	0.9251	0.9326
	MCC	0.9034	0.9171	0.9251
	IOU	0.8396	0.8606	0.8736
Average	Recall	0.8450	0.8499	0.8659
	Precision	0.8949	0.9034	0.9118
	F1 score	0.8689	0.8755	0.8874
	MCC	0.8447	0.8525	0.8670
	IOU	0.7692	0.7799	0.7992

For qualitative results, I showed examples of road segmentation maps achieved by the networks in Figure 4.14, respectively. The figure is presented in three rows and five columns. The first and second columns of the figures depict the RGB and reference images, respectively. The results acquired by UNet, BCL-UNet, and MCG-UNet are depicted in third, fourth, and fifth columns, respectively. All the networks can normally obtain an accurate road segmentation map. However, the road segmentation map produced by the

MCG-UNet is more accurate than those by other methods. In other words, the presented MCG-UNet network could obtain a high-quality segmentation map, preserve the higher accuracy of object boundaries' information on the edge segmentation, and predict fewer FPs (depicted in yellow color) and more FNs (depicted in blue color), which achieved an average F1 score of 88.74% for road compared with other deep learning-based models. This is due to the addition of the BConvLSTM, DC, and SE modules to the network. BConvLSTM mixes the encoded and decoded features that include more local information and more semantic information. Additionally, the DC assist the model to learn more varying features and the SE module can capture the spatial relations between features. Therefore, these modules, which were embedded into the models, could improve the performance in road object segmentation.

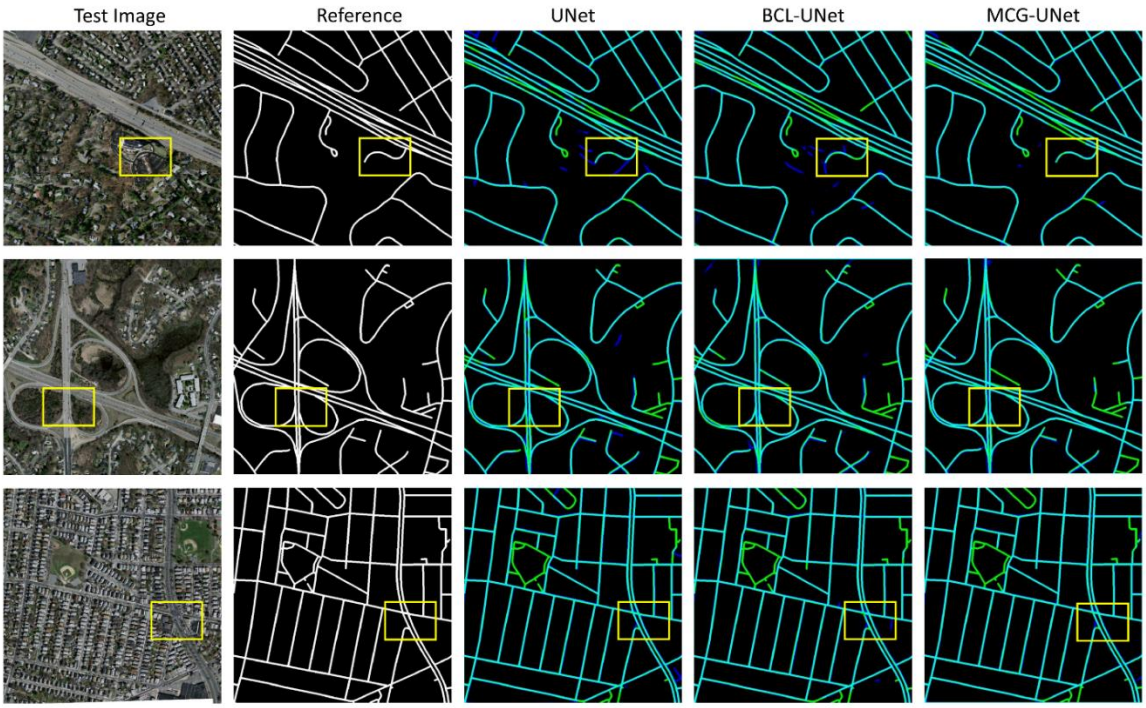

Figure 4.14. Obtained products with the presented UNet, BCL-UNet, and MCG-UNet networks from the Massachusetts road dataset. The yellow, blue, and white colors present the FNs, FPs, and TPs, respectively.

4.3.3.1. Discussion

To further investigate the advantage of the presented techniques in this study for road object extraction from aerial imagery, I compared the F1 score attained by the networks with other comparative deep learning-based networks applied for road segmentation. Note that the findings for other networks are taken from the key published manuscripts, whereas the presented networks were performed on experiential datasets. Specially, the proposed models in the current work were compared with convolutional networks, such as DeeplabV3 [169], BT-RoadNet [170], DLinkNet-34 [74], RoadNet [163], and GL-DenseUNet [69] for road extraction. Table 4.18 provide the average F1 score for the proposed frameworks and other comparative techniques for road extraction, respectively. As indicated in Table 4.18, both the models applied in the current study, such as BCL-UNet and MCG-UNet, worked better than other comparative models for road extraction. The BCL-UNet and MCG-UNet models achieved F1 score of 87.55% and 88.74% for road extraction, respectively, which is higher than other comparative road segmentation methods. This is because the proposed BCL-UNet and MCG-UNet networks use dense connections and BConvLSTM in the skip connections and SE in the expansive part. These functions help the networks learn more various features, learn more discriminative information, extract more valuable information, and improve accuracy.

Table 4.18. Quantitative results generated by the BCL-UNet and MCG-UNet and other deep learning-based techniques for road extraction.

Methods	Precision	Recall	IOU	F1 score
DeeplabV3	74.16	71.82	57.60	72.97
BT-RoadNet	87.98	78.16	74.00	82.77
DLinkNet-34	76.11	70.29	57.77	73.08
RoadNet	64.53	82.73	56.86	72.50
GL-DenseUNet	78.48	70.09	72.73	74.04
BCL-UNet	0.9034	0.8499	0.7799	87.55
MCG-UNet	0.9118	0.8659	0.7992	88.74

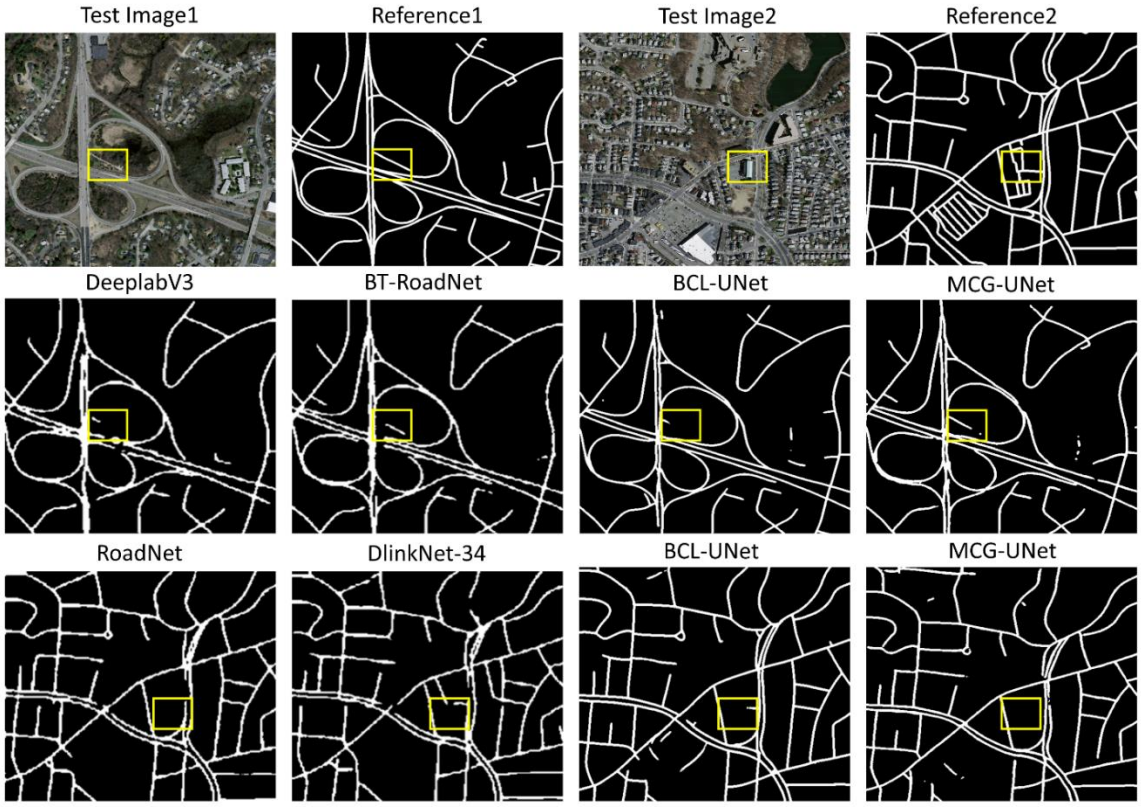

Figure 4.15. Road map comparisons generated by the presented BCL-UNet and MCG-UNet techniques against other deep learning-based networks. The yellow boxes show the predicted FPs and FNs.

Additionally, I portrayed the visual road products achieved by other techniques and the proposed BCL-UNet and MCG-UNet frameworks in Figure 4.15, to evaluate the efficiency of the suggested approaches in road segmentation. The proposed BCL-UNet and MCG-UNet methods could maintain the boundary information of roads and produce a high-resolution segmentation map for road objects compared with other comparative frameworks. By contrast, DeeplabV3 [169], BT-RoadNet [163], DLinkNet-34 [74], and RoadNet [163], which were performed for road segmentation, achieved lower quantitative values for F1 score, could not preserve the boundaries of road object, and identified more FNs and FPs, especially where roads were surrounded by obstructions and located in the

dense and complex areas. As a result, they produced low-resolution segmentation maps for roads.

4.3.3.2. DeepGlobe dataset

Moreover, I implemented our proposed models on another dataset called the DeepGlobe road dataset [151] to prove the effectiveness of the models on the road segmentation from various types of remote sensing images. DeepGlobe dataset includes 7469 training and validation images and 1101 testing images with a spatial resolution of 50 cm and a pixel size of 1024×1024. I compared the results of our methods for roads with other comparative methods, such as DeeplabV3 [169], and LinkNet [171]. Table 4.19 presents the quantitative results, while Figure 4.16 present the visualization outcomes obtained by the proposed models and other methods for road extraction from the dataset. The proposed BCL-UNet and MCG-UNet models could improve the F1 score compared to the comparative techniques and achieved an accuracy of 87.03% and 88.09% for road extraction, respectively. Additionally, according to the qualitative outcomes (Figure 4.16), the proposed models could extract roads from the DeepGlobe dataset accurately and achieve high-quality segmentation maps compared to the other approaches, which confirms the efficiency of the models for road extraction from another remote sensing dataset.

Table 4.19. Quantitative results generated by BCL-UNet and MCG-UNet for road extraction from DeepGlobe dataset.

	Methods	**Recall**	**Precision**	**F1 score**	**MCC**	**IOU**
DeepGlobe Road Dataset	DeeplabV3	0.8115	0.8750	0.8411	0.8139	0.7258
	LinkNet	0.8852	0.8238	0.8486	0.8199	0.7369
	BCL-UNet	0.8408	0.9047	0.8703	0.8482	0.7705
	MCG-UNet	0.8597	0.9044	0.8809	0.8595	0.7870

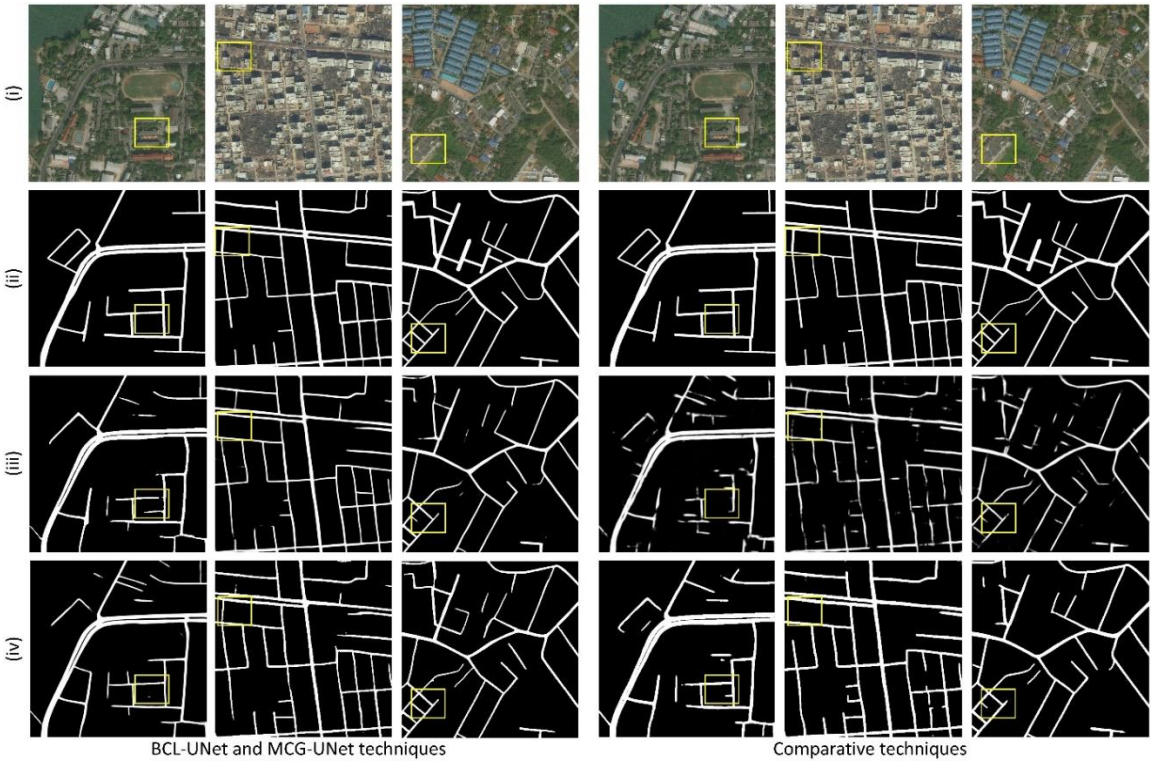

Figure 4.16. Road maps produced by the proposed BCL-UNet and MCG-UNet and comparative techniques from the DeepGlobe dataset. (**i**) Original imagery, (**ii**) ground truth imagery, (**iii**) results of BCL-UNet and DeeplabV3, and (**iv**) results of MCG-UNet and LinkNet. The yellow boxes present the predicted FPs and FNs.

4.4. Results of SC-RoadDeepNet for road shape and connectivity-preserving (Objective 2)

This study was compared with some state-of-the-art techniques, including deep learning approaches, such as LinkNet [171], DeeplabV3+ [169], ResUNet [153], UNet [95], and VNet [136], to examine the applicability of the presented SC-RoadDeepNet method for road segmentation from Google Earth imagery. I tested the proposed RRCNN model by integrating the edge map to the semantic segmentation to see how boundary learning (BL) fine-tunes the road shape via penalizing boundary misclassification. I call this network RRCNN-boundary-learning or RRCNN+BL. Furthermore, I compared the proposed SC-RoadDeepNet with different values of α , such as $\alpha = 0.1, \alpha = 0.3$, $\alpha = 0.5$, $\alpha = 0.7$ and

$\alpha = 0.9$, to show the effect of alpha parameter on the road connectivity and segmentation results. All of the mentioned methods were tried using the same collection of imagery to make the assessments fair and objective.

Table 4.20 demonstrates the quantitative findings obtained by the methods. The accuracy of the methods was calculated using IOU and F1 scores. LinkNet, DeeplabV3+, ResUNet, and UNet achieved the lowest IOU values with 81.52%, 82.53%, 84%, and 85.01%, respectively, when I compare the outcomes of different approaches (Table 4.20). VNet could improve the results to 86.99% compared with the mentioned four methods. By adding BL to the proposed RRCNN method (RRCNN+BL) and the proposed loss function to the model without BL (RRCNN+CP_clDIce), the accuracy of the IOU was also increased to 89.02% and 89.75%, and these methods were the third-best and second-best methods in all approaches, which proved the influence of edge-map and CP_clDice on improving road shape and segmentation results. In contrast, by including BL and connectivity-preserving CP_clDice techniques to the proposed SC-RoadDeepNet, IOU values reached 90.04%, 90.43%, 91.05%, 90.34%, and 89.85% for $\alpha = 0.1$, $\alpha = 0.3$, $\alpha = 0.5$, $\alpha = 0.7$ and $\alpha = 0.9$, respectively. I found that including CP_clDice in any values ($\alpha > 0$) results in improving road connectivity and segmentation. Figures 4.17 and 4.18 also depict the qualitative results obtained using state-of-the-art techniques. According to the findings, all extraction methods can reduce the impact of occlusions to some extent. However, LinkNet, DeeplabV3+, UNet, ResUNet, and VNet approaches are sensitive to noise and introduced some FPs in some parts, such as the shadows, buildings, and trees, and could not extract roads accurately. Benefited from BL and CP_clDice, the proposed RRCNN+BL and RRCNN+CP_clDice methods could reduce boundary misclassification and achieve relatively satisfactory results.

Table 4.20. Quantitative experimental outcomes yielded by the comparative approaches for the Google Earth road dataset.

		Image 1	Image 2	Image 3	Image 4	Image 5	Image 6	Average
LinkNet	F1 score	0.8821	0.9183	0.9149	0.8830	0.8970	0.8930	0.8981
	IOU	0.7890	0.8488	0.8430	0.7905	0.8132	0.8067	0.8152
	MCC	0.7903	0.8474	0.8615	0.8225	0.8341	0.8214	0.8295
	OA	0.8869	0.9231	0.9334	0.9154	0.9219	0.9137	0.9157
ResUNet	F1 score	0.8851	0.9302	0.9404	0.8870	0.9177	0.9157	0.9127
	IOU	0.7938	0.8694	0.8874	0.7969	0.8478	0.8445	0.8400
	MCC	0.7941	0.8662	0.9054	0.8288	0.8661	0.8564	0.8528
	OA	0.8923	0.9331	0.9556	0.9181	0.9364	0.9304	0.9277
UNet	F1 score	0.8901	0.9354	0.9313	0.9064	0.9284	0.9210	0.9188
	IOU	0.8019	0.8785	0.8714	0.8289	0.8663	0.8536	0.8501
	MCC	0.8051	0.8743	0.8906	0.8584	0.8840	0.8639	0.8627
	OA	0.8953	0.9372	0.9487	0.9333	0.9452	0.9328	0.9321
DeeplabV3+	F1 score	0.8626	0.9159	0.9254	0.8901	0.9226	0.9067	0.9039
	IOU	0.7584	0.8448	0.8612	0.8020	0.8563	0.8293	0.8253
	MCC	0.7516	0.8510	0.8791	0.8336	0.8668	0.8479	0.8383
	OA	0.8738	0.9231	0.9426	0.9206	0.9347	0.9274	0.9204
VNet	F1 score	0.9315	0.9390	0.9418	0.9108	0.9312	0.9277	0.9303
	IOU	0.8718	0.8850	0.8899	0.8361	0.8713	0.8650	0.8699
	MCC	0.8784	0.8797	0.9063	0.8647	0.8880	0.8758	0.8822
	OA	0.9382	0.9386	0.9559	0.9370	0.9463	0.9393	0.9426
RRCNN+BL	F1 score	0.9344	0.9517	0.9584	0.9209	0.9455	0.9397	0.9418
	IOU	0.8768	0.9078	0.9202	0.8534	0.8965	0.8862	0.8902
	MCC	0.8833	0.9052	0.9337	0.8811	0.9113	0.8971	0.9020
	OA	0.9414	0.9052	0.9689	0.9435	0.9576	0.9501	0.9445
RRCNN+CP_clDice	F1 score	0.9362	0.9669	0.9628	0.9213	0.9456	0.9418	0.9458
	IOU	0.8800	0.9359	0.9282	0.8541	0.8967	0.8900	0.8975
	MCC	0.8865	0.9352	0.9405	0.8810	0.9117	0.9002	0.9092
	OA	0.9423	0.9673	0.9721	0.9444	0.9578	0.9513	0.9559
SC-RoadDeepNet (α=0.1)	F1 score	0.9399	0.9719	0.9601	0.9247	0.9440	0.9437	0.9474
	IOU	0.8866	0.9453	0.9232	0.8599	0.8939	0.8934	0.9004
	MCC	0.8935	0.9450	0.9361	0.8865	0.9085	0.9034	0.9122
	OA	0.9462	0.9723	0.9700	0.9466	0.9557	0.9527	0.9573
SC-RoadDeepNet (α=0.3)	F1 score	0.9411	0.9726	0.9610	0.9301	0.9459	0.9467	0.9496
	IOU	0.8888	0.9466	0.9248	0.8693	0.8973	0.8988	0.9043
	MCC	0.8956	0.9479	0.9375	0.8945	0.9119	0.9090	0.9161
	OA	0.9472	0.9738	0.9610	0.9509	0.9575	0.9558	0.9577
SC-RoadDeepNet (α=0.5)	F1 score	0.9435	0.9775	0.9677	0.9331	0.9466	0.9493	0.9530
	IOU	0.8929	0.9560	0.9374	0.8746	0.8985	0.9034	0.9105
	MCC	0.8997	0.9561	0.9484	0.8992	0.9132	0.9130	0.9216
	OA	0.9495	0.9781	0.9758	0.9529	0.9581	0.9574	0.9620
SC-RoadDeepNet (α=0.7)	F1 score	0.9398	0.9687	0.9611	0.9283	0.9475	0.9491	0.9491
	IOU	0.8864	0.9392	0.9251	0.8661	0.9002	0.9031	0.9034
	MCC	0.8934	0.9390	0.9373	0.8916	0.9146	0.9129	0.9148
	OA	0.9458	0.9695	0.9705	0.9498	0.9591	0.9575	0.9587
SC-RoadDeepNet (α=0.9)	F1 score	0.9367	0.9711	0.9495	0.9311	0.9457	0.9441	0.9464
	IOU	0.8809	0.9438	0.9039	0.8710	0.8970	0.8941	0.8985
	MCC	0.8876	0.9439	0.9201	0.8956	0.9127	0.9052	0.9109
	OA	0.9441	0.9720	0.9625	0.9521	0.9589	0.9541	0.9573

Furthermore, the proposed SC-RoadDeepNet, which takes advantage of BL and CP_clDice techniques, could obtain fewer FPs (shown in blue) and FNs (shown in red), reduce road discontinuity and produce high-resolution road segmentation maps compared to the other approaches. The presented SC-RoadDeepNet model with $\alpha = 0.5$ improved the results of IOU to 2.03% and 1.3% compared with the RRCNN+BL (third best) and RRCNN+CP_clDice (second best) models, respectively. They all showed that combining the suggested BL and CP_clDice techniques in the shape and connectivity-aware SC-RoadDeepNet model resulted in superior performance than other current approaches.

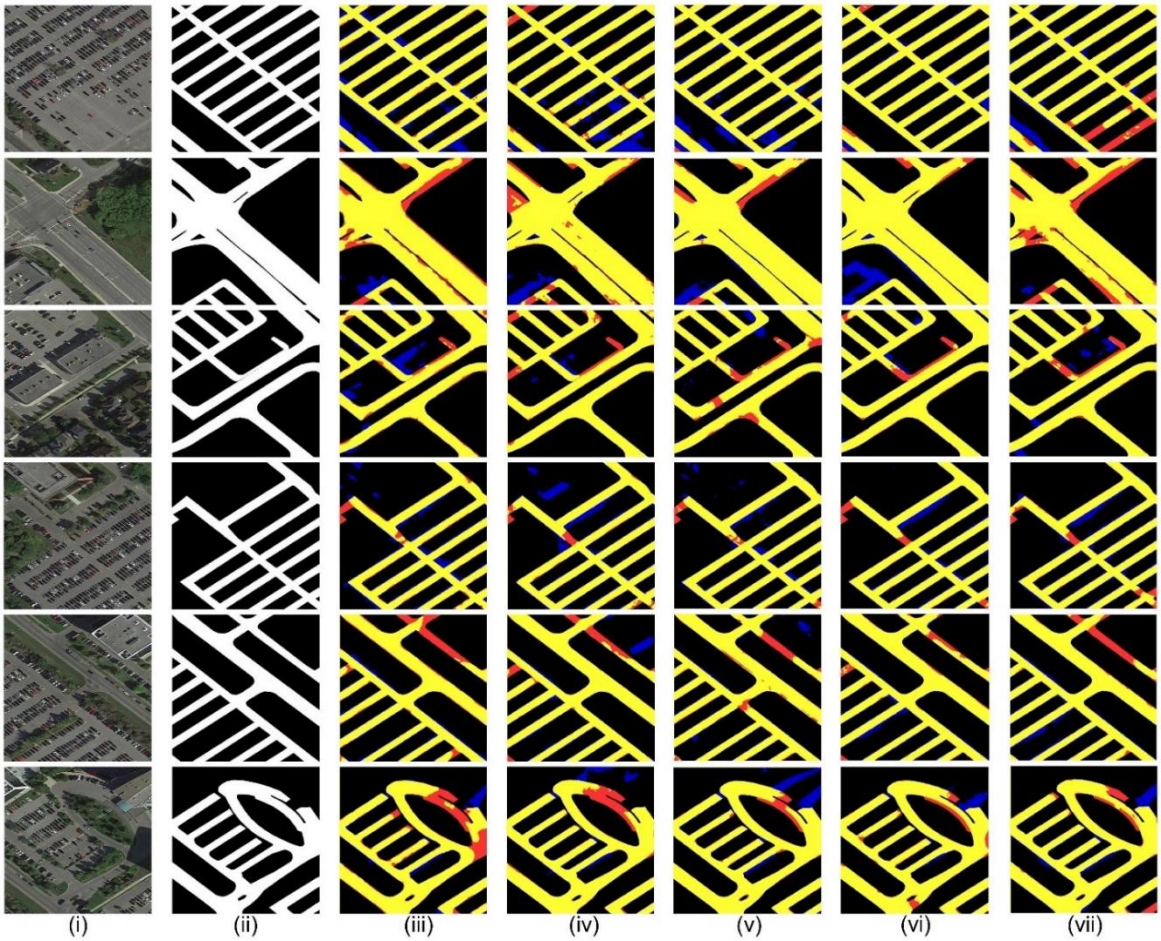

Figure 4.17. Road qualitative results were compared visually using various comparing models: (i) original RGB Google Earth images, (ii) reference images, (iii) LinkNet results, (iv) ResUNet results, (v) UNet results, (vi) VNet results, and (vii) DeeplabV3+ results. TPs, FPs, and FNs are represented by yellow, blue, and red, respectively.

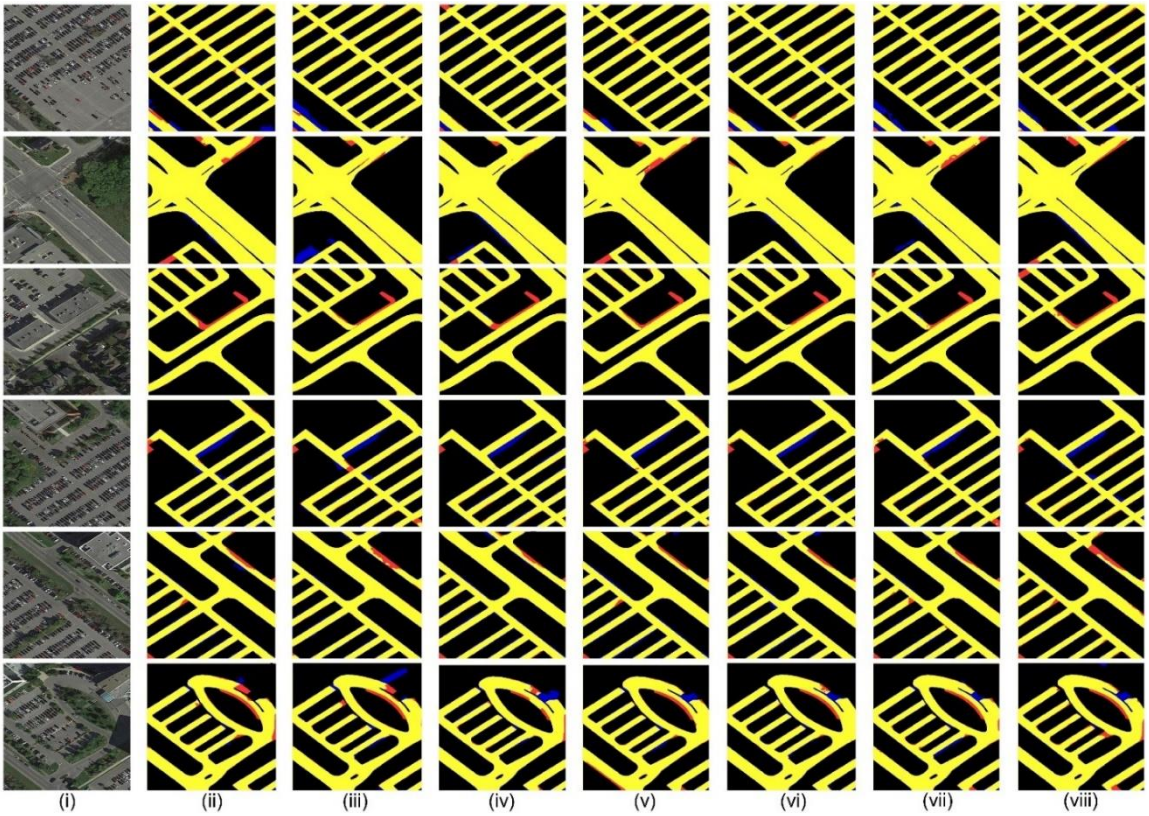

Figure 4.18. Road qualitative results were compared visually using proposed models: (i) original RGB Google Earth images, (ii) RRCNN+BL results, (iii) RRCNN+CP_clDice results, (iv) SC-RoadDeepNet results (α=0.1), (v) SC-RoadDeepNet results (α=0.3), (vi) SC-RoadDeepNet results (α=0.5), (vii) SC-RoadDeepNet results (α=0.7), and (viii) SC-RoadDeepNet results, (α=0.9). The TPs, FPs, and FNs are represented by yellow, blue, and red, respectively.

4.4.1. Discussion

In this section, I evaluated the performance of the proposed framework by analyzing the ablation study and testing the model on another road datasets.

4.4.1.1. Ablation study

To assess the efficiency of the proposed shape and connectivity-preserving SC-RoadDeepNet model's ability in improving road discontinuity and road shape segmentation, I conducted an ablation study in this work. In this case, I applied the

proposed RRCNN model with the primary binary cross-entropy loss function and without BL and CP_clDice techniques to see the influence of these methods on fine-tuning road shape and preserving road connectivity. I obtained the quantitative and visualization findings by the model in road segmentation from the Google Earth dataset. Table 4.21 contains the quantitative results, whereas Figure 4.19 depicts the visualization results. After adjusting various variables and removing those crucial techniques, the IOU's accuracy of the proposed RRCNN model was reduced to 87.85%, as shown in Table 4.21. Furthermore, as shown in Figure 4.19, the suggested approach introduced spurs and generated more FPs and FNs in homogeneous areas, reducing the smoothness and connectedness of the road segmentation network significantly. Therefore, BL and CP_clDice have shown a significant role in preserving road shape and connectivity and producing high-quality road segmentation maps.

Table 4.21. Quantitative experimental outcomes yielded by the RRCNN approach for road extraction without BL and CP_clDice techniques.

		Image 1	Image 2	Image 3	Image 4	Image 5	Image 6	Average
RRCNN	F1 score	0.9350	0.9424	0.9513	0.9140	0.9386	0.9301	0.9352
	IOU	0.8779	0.8909	0.9071	0.8415	0.8842	0.8693	0.8785
	MCC	0.8853	0.8876	0.9227	0.8694	0.9000	0.8807	0.8910
	OA	0.9407	0.9438	0.9637	0.9402	0.9522	0.9421	0.9471

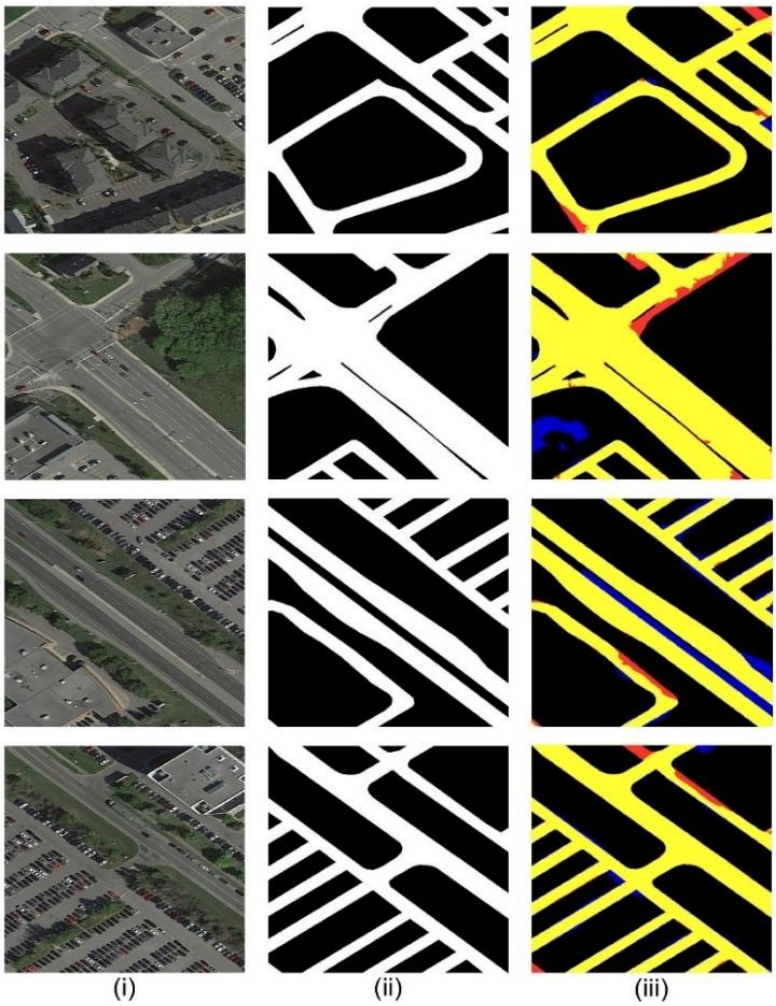

Figure 4.19. Road qualitative results were compared visually using the proposed RRCNN model: (i) original RGB Google Earth images, (ii) reference images, (iii) RRCNN results. TPs, FPs, and FNs are represented by yellow, blue, and red, respectively.

4.4.1.2. DeepGlobe and Massachusetts road datasets

Furthermore, I applied our proposed SC-RoadDeepNet model on more road datasets called DeepGlobe [151] and Massachusetts [135] to show the model's efficiency in road segmentation from various types of remote sensing imagery. The DeepGlobe dataset is captured in India, Indonesia, and Thailand, containing 8570 images with 50 cm per pixel spatial resolution and covering 2220 km^2. Each image is 1024×1024pixels in size. The training and testing datasets consisted of 1006 and 26 images in this study, respectively.

Table 4.22. Quantitative experimental outcomes yielded by the RRCNN, RRCNN+BL, RRCNN+CP_clDice, and SC-RoadDeepNet approaches for road extraction from the DeepGlobe road dataset.

		Image 1	Image 2	Image 3	Image 4	Image 5	Average
RRCNN	F1 score	0.9207	0.9250	0.8927	0.8918	0.9444	0.9149
	IOU	0.8529	0.8604	0.8061	0.8047	0.8947	0.8438
	MCC	0.9078	0.9157	0.8632	0.8763	0.9329	0.8992
	OA	0.9774	0.9835	0.9534	0.9722	0.9808	0.9735
RRCNN+BL	F1 score	0.9327	0.9296	0.8976	0.8973	0.9469	0.9208
	IOU	0.8738	0.8684	0.8141	0.8137	0.8990	0.8538
	MCC	0.9215	0.9209	0.8707	0.8823	0.9360	0.9063
	OA	0.9807	0.9846	0.9563	0.9734	0.9817	0.9753
RRCNN+CP_clDice	F1 score	0.9394	0.9304	0.8983	0.8988	0.9481	0.9230
	IOU	0.8858	0.8698	0.8154	0.8161	0.9013	0.8577
	MCC	0.9294	0.9217	0.8726	0.8834	0.9380	0.9090
	OA	0.9828	0.9846	0.9570	0.9733	0.9824	0.9760
SC-RoadDeepNet	F1 score	0.9416	0.9349	0.9062	0.9065	0.9499	0.9278
	IOU	0.8896	0.8777	0.8285	0.8289	0.9046	0.8659
	MCC	0.9319	0.9271	0.8805	0.8924	0.9394	0.9143
	OA	0.9834	0.9859	0.9593	0.9756	0.9826	0.9774

The Massachusetts dataset that I used contains 1032 training and 32 testing images with a size of 768×768 and spatial resolution of 0.5 m. I obtained quantitative and visualization outcomes yielded by the presented RRCNN, RRCNN+BL, RRCNN+CP_clDice, and SC-RoadDeepNet models for road segmentation from the DeepGlobe and Massachusetts datasets, which are demonstrated in Table 4.22 and Figure 4.20 for DeepGlobe and Table 4.23 and Figure 4.21 for Massachusetts dataset, respectively. Table 4.22 shows that the proposed RRCNN model did not benefit from BL and CP_clDice techniques achieved the lowest F1 score with 91.49% for DeepGlobe and 87.19% for Massachusetts. In contrast,

the proposed RRCNN+BL, RRCNN+CP_clDice, and SC-RoadDeepNet could improve the results of DeepGlobe to 92.08%, 92.30%, and 92.78% for F1 score and the results of Massachusetts to 87.95%, 88.47%, and 89.33%, respectively. According to the visualization results (Figures 4.20 and 4.21), the proposed RRCNN model failed to segment roads in the complex areas, where the road is covered by shadows and trees and brought in more FPs, FNs, and discontinuity.

Table 4.23. Quantitative experimental outcomes yielded by the RRCNN, RRCNN+BL, RRCNN+CP_clDice, and SC-RoadDeepNet approaches for road extraction from the Massachusetts road dataset.

		Image 1	Image 2	Image 3	Image 4	Image 5	Average
RRCNN	F1 score	0.8827	0.8591	0.8785	0.8663	0.8730	0.8719
	IOU	0.8099	0.7729	0.8032	0.7841	0.7946	0.7929
	MCC	0.8586	0.8320	0.8614	0.8490	0.8543	0.8511
	OA	0.9552	0.9534	0.9680	0.9677	0.9642	0.9617
RRCNN+BL	F1 score	0.8964	0.8711	0.8866	0.8700	0.8733	0.8795
	IOU	0.8321	0.7915	0.8162	0.7898	0.7950	0.8049
	MCC	0.8738	0.8477	0.8704	0.8529	0.8538	0.8597
	OA	0.9627	0.9599	0.9716	0.9698	0.9663	0.9661
RRCNN+CP_clDice	F1 score	0.8985	0.8743	0.8898	0.8820	0.8790	0.8847
	IOU	0.8357	0.7966	0.8215	0.8088	0.8040	0.8133
	MCC	0.8765	0.8518	0.8743	0.8665	0.8604	0.8659
	OA	0.9629	0.9611	0.9726	0.9726	0.9678	0.9674
SC-RoadDeepNet	F1 score	0.9037	0.8808	0.9039	0.8899	0.8881	0.8933
	IOU	0.8443	0.8070	0.8446	0.8216	0.8187	0.8272
	MCC	0.8828	0.8581	0.8902	0.8762	0.871	0.8757
	OA	0.9655	0.9617	0.976	0.9752	0.9695	0.9696

On the contrary, the presented SC-RoadDeepNet that benefited from BL and CP_clDice could obtain the segmentation map with fewer FPs and FNs and showed higher extraction accuracy on the boundary and road connectivity than others. In summary, the proposed method could improve road extraction by tackling occlusion-related interruptions. It could solve discontinuity in road extraction results and produce high-resolution results compared with the other methods. Also, I calculated the runtime of the presented method on each

dataset, which took 117s, 388s, and 226s per epoch for the training process for Ottawa, DeepGlobe, and Massachusetts datasets, respectively. The model was trained for 100 epochs; therefore, it took 195 minutes for Ottawa, 646.66 minutes for DeepGlobe, and 376.66 minutes for Massachusetts datasets. It is clear that as the size of images and datasets increases, the training time is also increased. Overall, the suggested method does not need a huge training dataset or a lot of computational effort, yet it still outperformed previous models in terms of statistical outcomes.

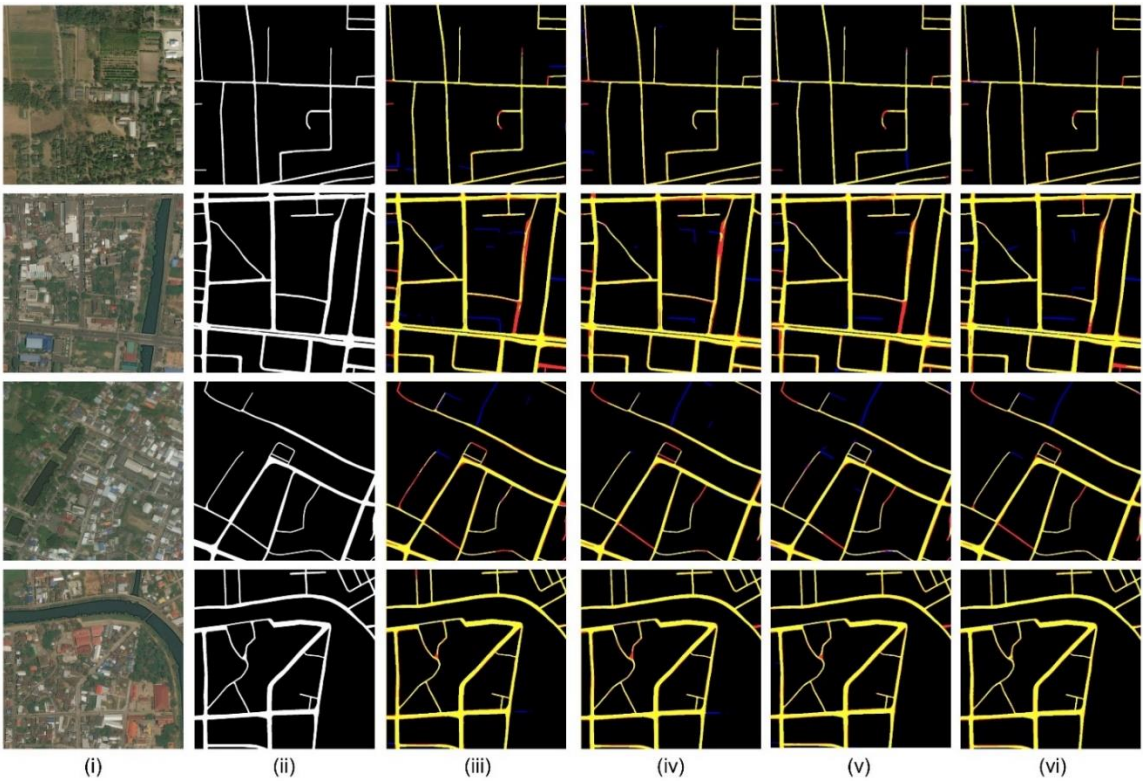

Figure 4.20. Road qualitative results achieved by the models from the DeepGlobe road dataset: (i) original RGB images, (ii) reference images, (iii) RRCNN results, (iv) RRCNN+BL results, (v) RRCNN+CP_clDice results, and (vi) SC-RoadDeepNet results. TPs, FPs, and FNs are represented by yellow, blue, and red, respectively.

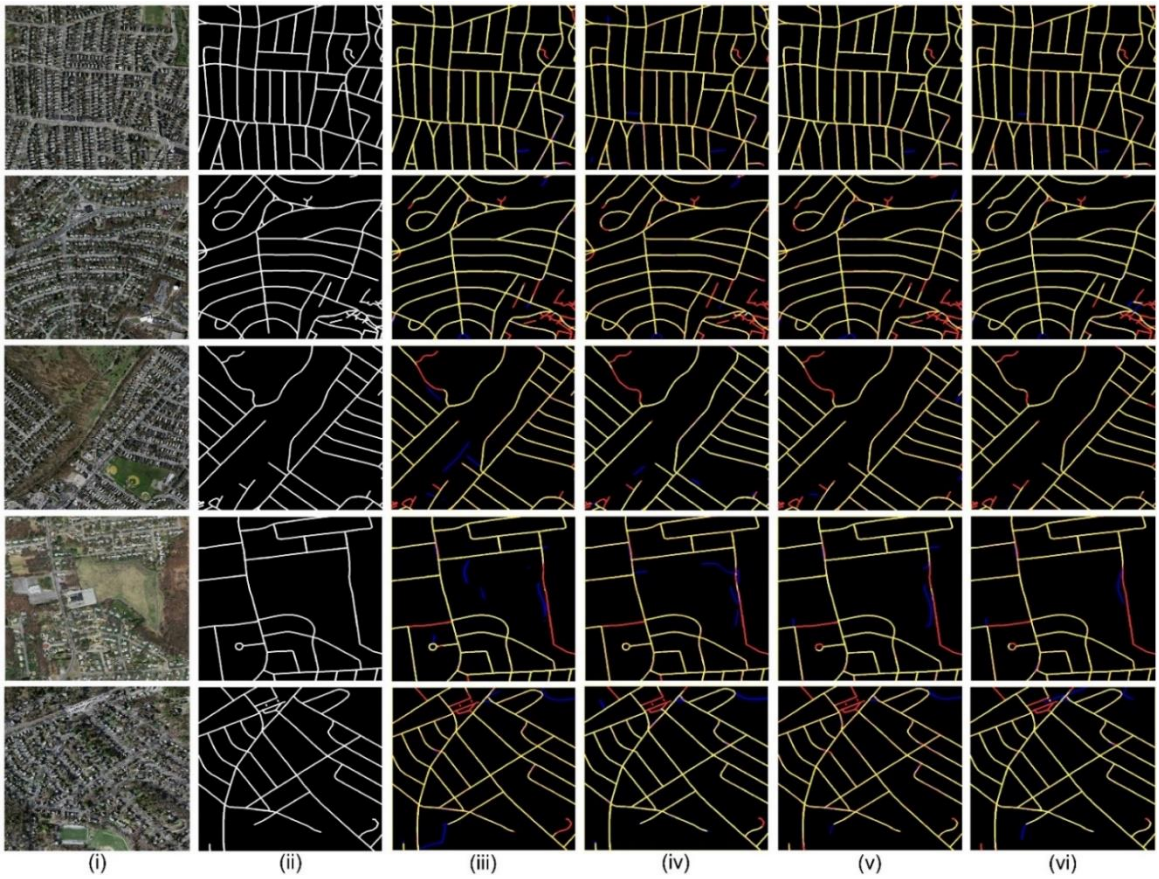

Figure 4.21. Road qualitative results achieved by the models from the Massachusetts road dataset: (i) original RGB images, (ii) reference images, (iii) RRCNN results, (iv) RRCNN+BL results, (v) RRCNN+CP_clDice results, and (vi) SC-RoadDeepNet results. TPs, FPs, and FNs are represented by yellow, blue, and red, respectively.

4.5. Results of road vectorization using RoadVecNet (Objective 3)

In this part, the quantitative and qualitative results achieved by the proposed RoadVecNet for automatic and simultaneous road surface segmentation and vectorization from different high-resolution remote sensing datasets are presented.

4.5.1. Qualitative comparison of road surface segmentation

I compared the presented RoadVecNet architecture with some other state-of-the-art classification-based deep learning networks to investigate the capability of the network in road surface segmentation from HRSI. Examples of these networks are as follows: UNet

architecture provided by [95]; SegNet network implemented by [96]; DeepLabV3 framework performed by [160]; VNet model applied by [136]; ResUNet provided by [153]; and FCN architecture developed by [31]. For denoting segmentation, I utilized the suffix "-S" after each method's name. The visualization outcomes obtained by the presented RoadVecNet architecture and other comparative networks for road surface segmentation from the Massachusetts and Ottawa datasets are demonstrated in Figures 4.22, 4.23, and 4.24. The figures illustrate that the SegNet-S, ResUNet-S, and DeepLabV3-S networks were sensitive to the barriers of trees and shadows and predicted more FN pixels (depicted as blue color) and FP pixels (depicted as green color), thereby producing low-quality road segmentation maps for both datasets. Meanwhile, the FCN-S, UNet-S, and VNet-S architectures could improve the results and generate more coherent and satisfactory road segmentation maps. However, none of the abovementioned models achieved better qualitative results than Ours-S. Ours-S could generate high-resolution road segmentation maps for both datasets by alleviating the effect of obstacles, predicting less FP pixels, and preserving the road border information. The reason is that I used the DDSPP module to create feature pyramids with more denser scale variability and a bigger receptive field. I also utilized the SE module to extract more valuable information by considering the interdependencies between feature channels. In addition, I applied the MFB_FL loss function to overcome highly unbalanced datasets and allow more attention on the hard samples. Therefore, I could obtain more constant and smoother road segmentation and vectorization results.

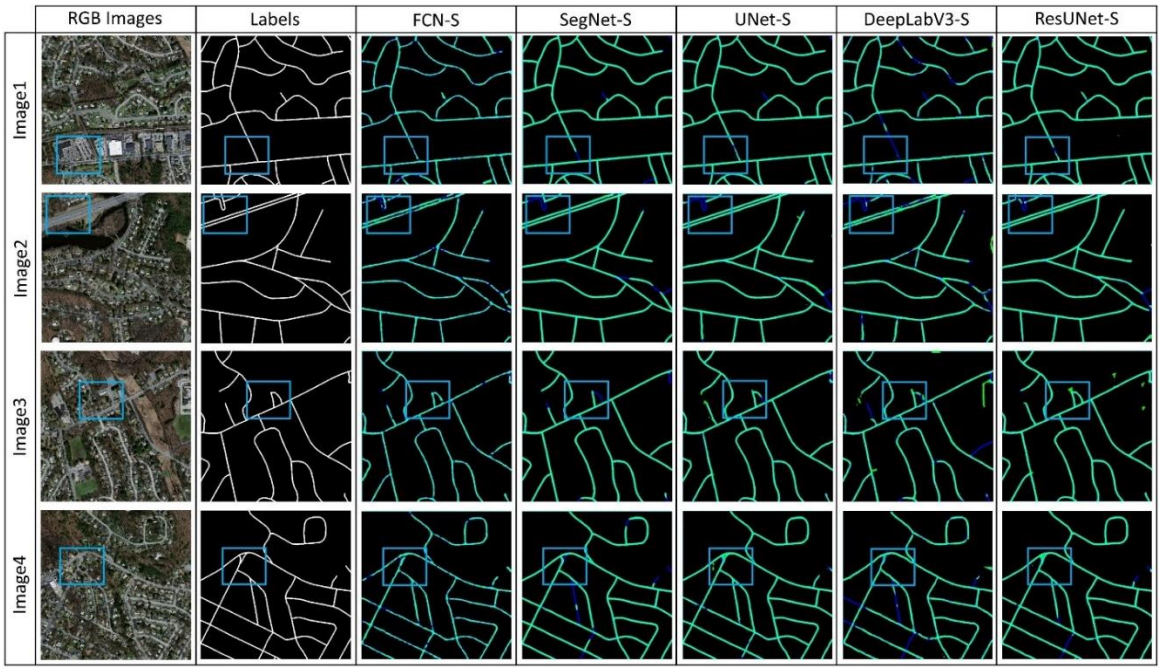

Figure 4.22. Visual performance attained by Ours-S against the other comparative networks for road surface segmentation from the Massachusetts imagery. The cyan, green and blue colors denote the TPs, FPs, and FNs, respectively.

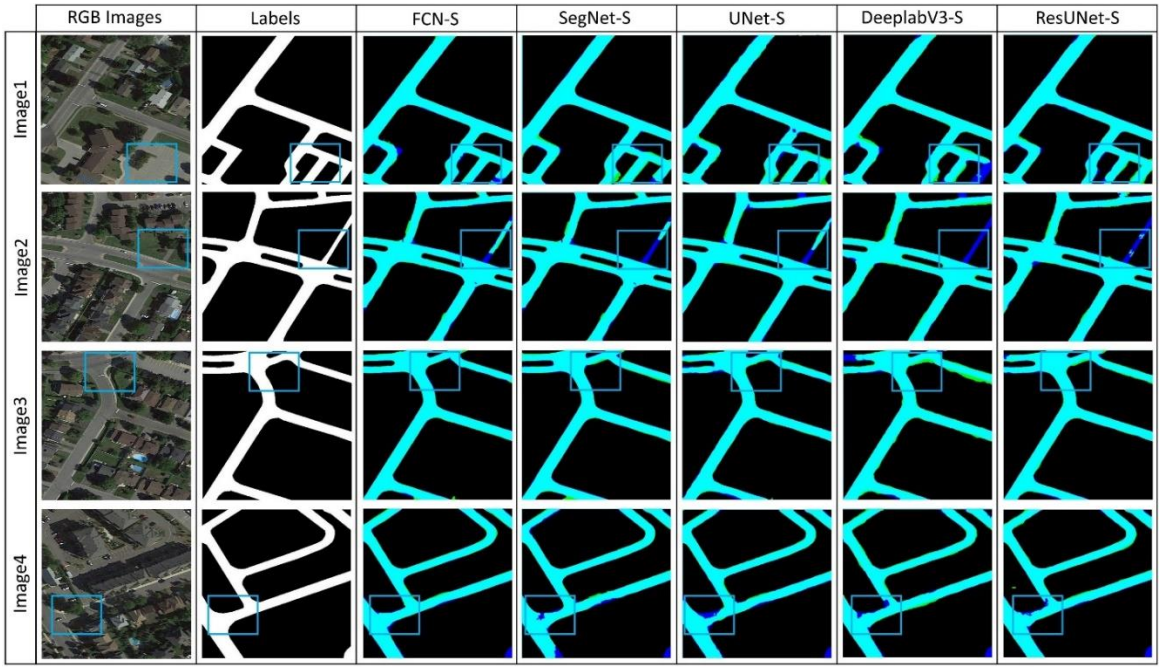

Figure 4.23. Visual performance attained by the comparative networks for road surface segmentation from the Ottawa imagery. The cyan green, and blue colors denote the TPs, FPs, and FNs, respectively.

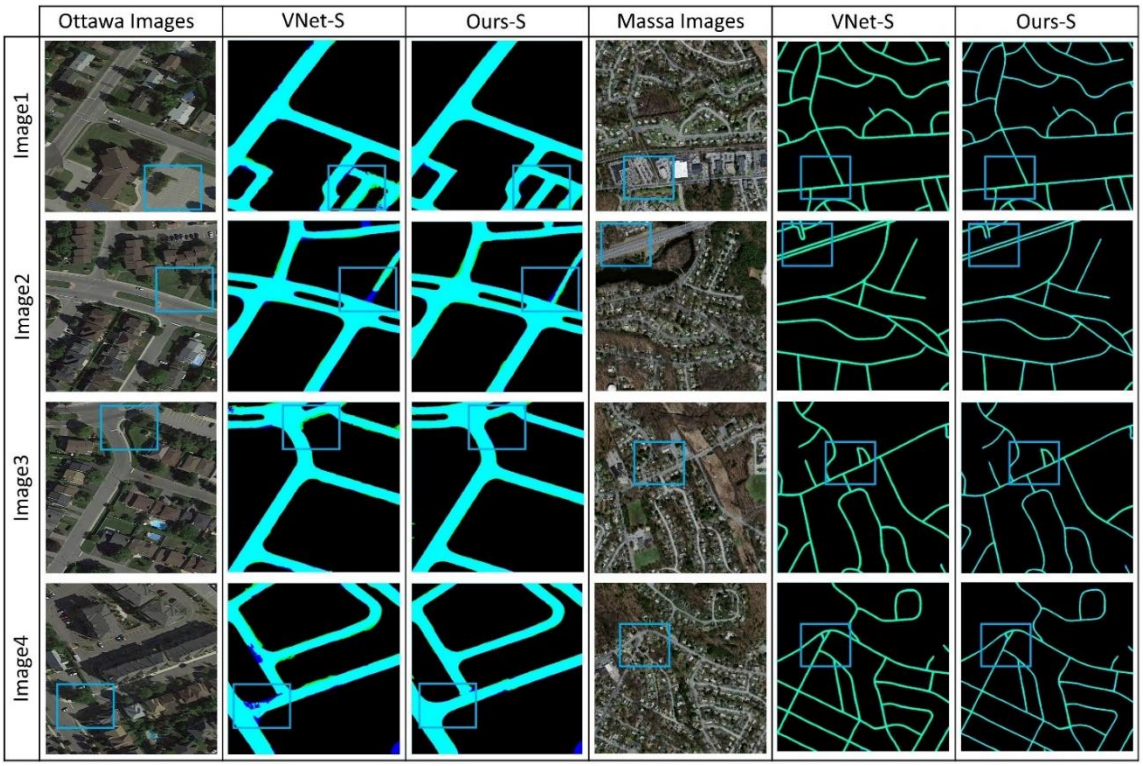

Figure 4.24. Visual performance attained by Ours-S against VNet-S network for road surface segmentation from the Ottawa and Massachusetts imagery. The cyan, green, and blue colors denote the TPs, FPs, and FNs, respectively.

4.5.2. Qualitative comparison of road vectorization

Here, I compared the results attained by the presented RoadVecNet architecture for road vectorization from the Massachusetts and Ottawa datasets with the same comparative deep learning methods applied in the road surface segmentation part, such as UNet architecture [95], DeepLabV3 framework [160], SegNet network [96], VNet applied by [136], ResUNet provided by [153], and FCN [31]. I utilized the suffix "-V" after every approach's name to denote road vectorization. Figures 4.25 and 4.26 demonstrate the comparison outcomes of various approaches and the presented RoadVecNet for road vectorization in visual performance for Ottawa imagery. The vectorized road ground truth map is also included in the second column of the figure to better display the contrast influences. I also used blue rectangular boxes in the figures to show the FP and FN pixels for facilitating comparison.

Figures 4.25 and 4.26 illustrate that although the FCN-V, SegNet-V ResUNet-V, and DeepLabV3-V architectures could generate relatively complete road vectorization network, they brought in spurs and produced some FPs in the homogenous regions where the road was covered by occlusions and around the intersections, reducing the correctness and smoothness of the road vectorization network. The UNet-V and VNet-V methods could improve the results and generate a complete network of the road vectorization; however, it failed to vectorize the road in the intersection parts and brought in some discontinuity and FPs. Figures 4.27 and 4.28 demonstrate the visual performance of the comparative models for Massachusetts imagery. In this dataset, the complexity of obstacles and backgrounds are more, and the road width is less than those in the Ottawa dataset. Accordingly, all the above-mentioned comparative models, including VNet-V, could not accurately vectorize the road, resulting in non-complete and non-smooth vectorized road network, especially for complex backgrounds and intersection areas where they brought in more discontinuity and FPs. By contrast, Ours-V could detect complete and non-spur vectorized road network even from the Massachusetts dataset with narrow road width and complex backgrounds. Our vectorized road map is more similar to the actual ground truth vectorized road than the other comparative models.

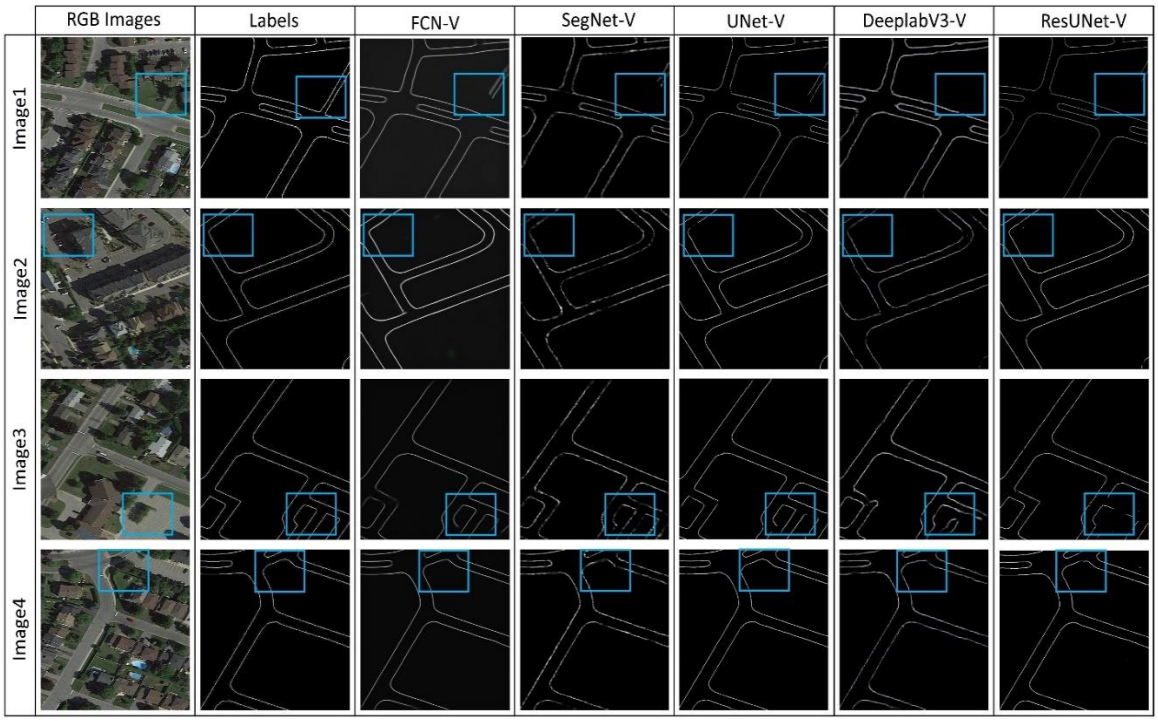

Figure 4.25. Comparison outcomes of various approaches for road vectorization in visual performance for Ottawa imagery. The first and second columns demonstrate the original RGB and corresponding reference imagery, respectively. The third, fourth, fifth, sixth, and last columns demonstrate the results of FCN-V, SegNet-V, UNet-V, DeepLabV3-V, and ResUNet-V. More details can be seen in the zoomed-in view.

4.5.3. Quantitative comparison of road segmentation

I obtained the quantitative calculations for the presented technique and other comparative networks applied to the Massachusetts and Ottawa datasets for road segmentation, which are summarized in Table 4.24 and 4.25, respectively. The first four columns in both tables are the performance of four test sample imagery, and the final column is the average accuracy of the whole test imagery. The bold value is the best in the F1 score metric, while the underlined values are the second-best.

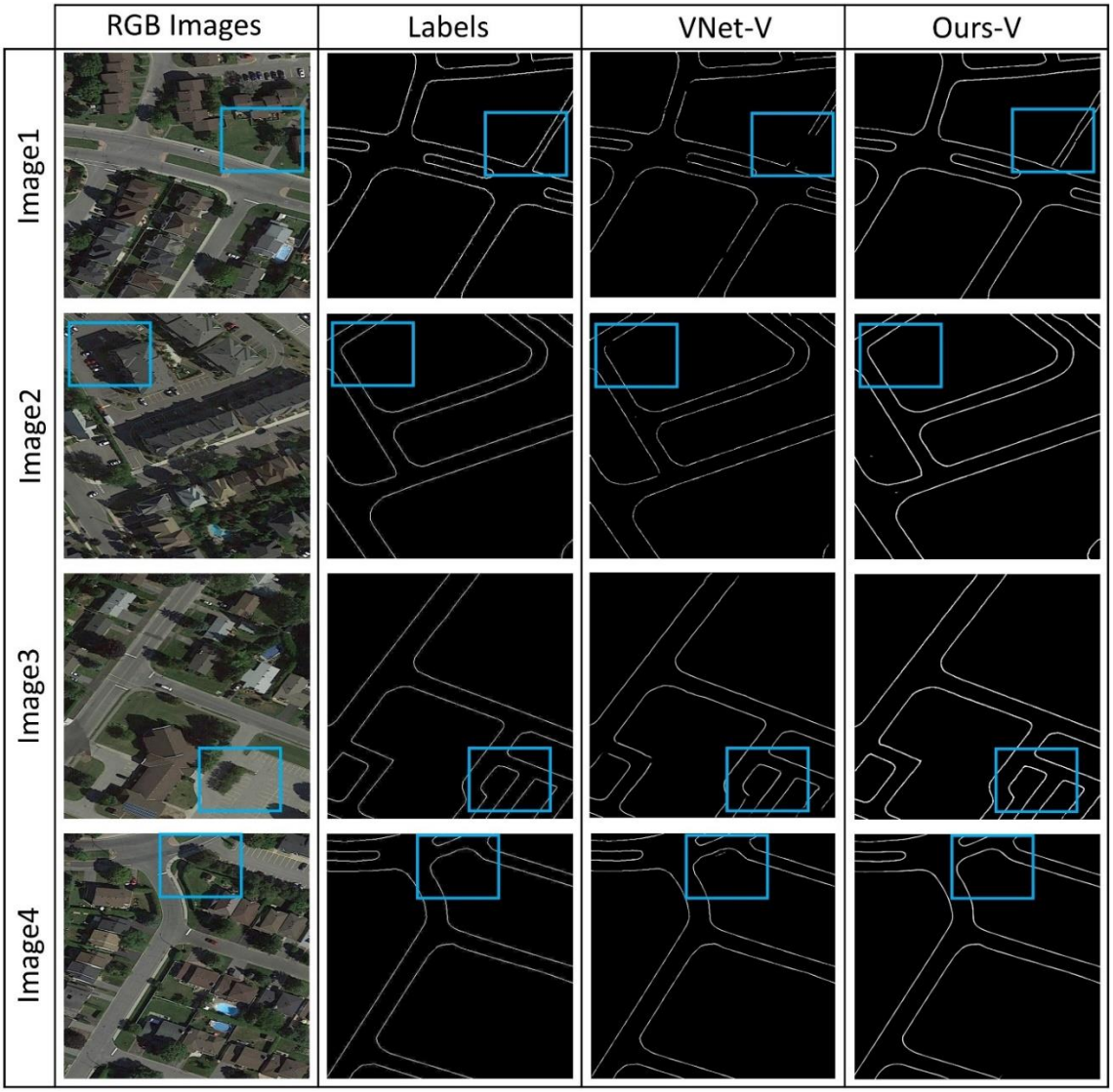

Figure 4.26. Comparison of the outcomes of the VNet-V approach and Ours-V for road vectorization in terms of visual performance for Ottawa imagery. The first and second columns demonstrate the original RGB and corresponding reference imagery, respectively. The third and fourth columns demonstrate the results of VNet-V and Ours-V. More details can be seen in the zoomed-in view.

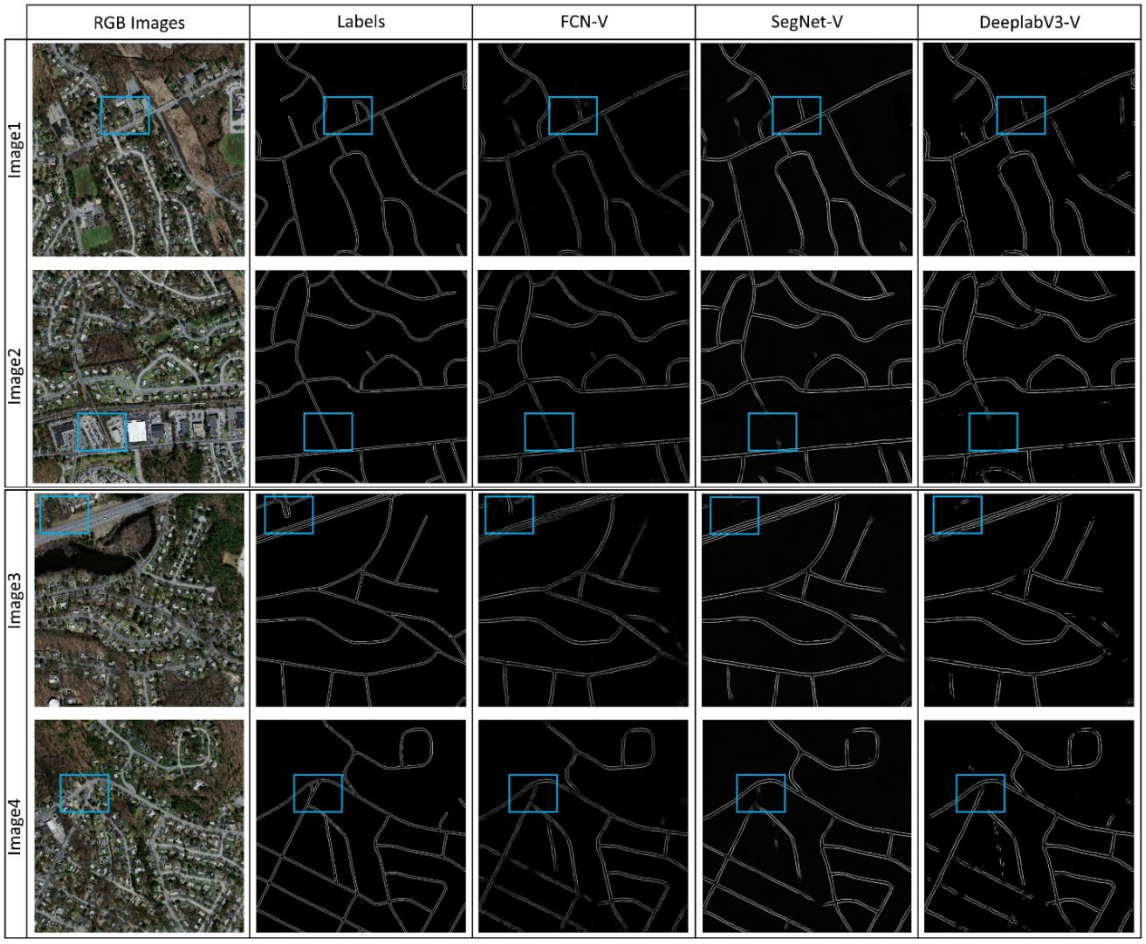

Figure 4.27. Comparison of the outcomes of various approaches for road vectorization in terms of visual performance for Massachusetts imagery. The first and second columns demonstrate the original RGB and corresponding reference imagery, respectively. The third, fourth, and fifth columns demonstrate the results of FCN-V, SegNet-V, and DeepLabV3-V, respectively. More details can be seen in the zoomed-in view.

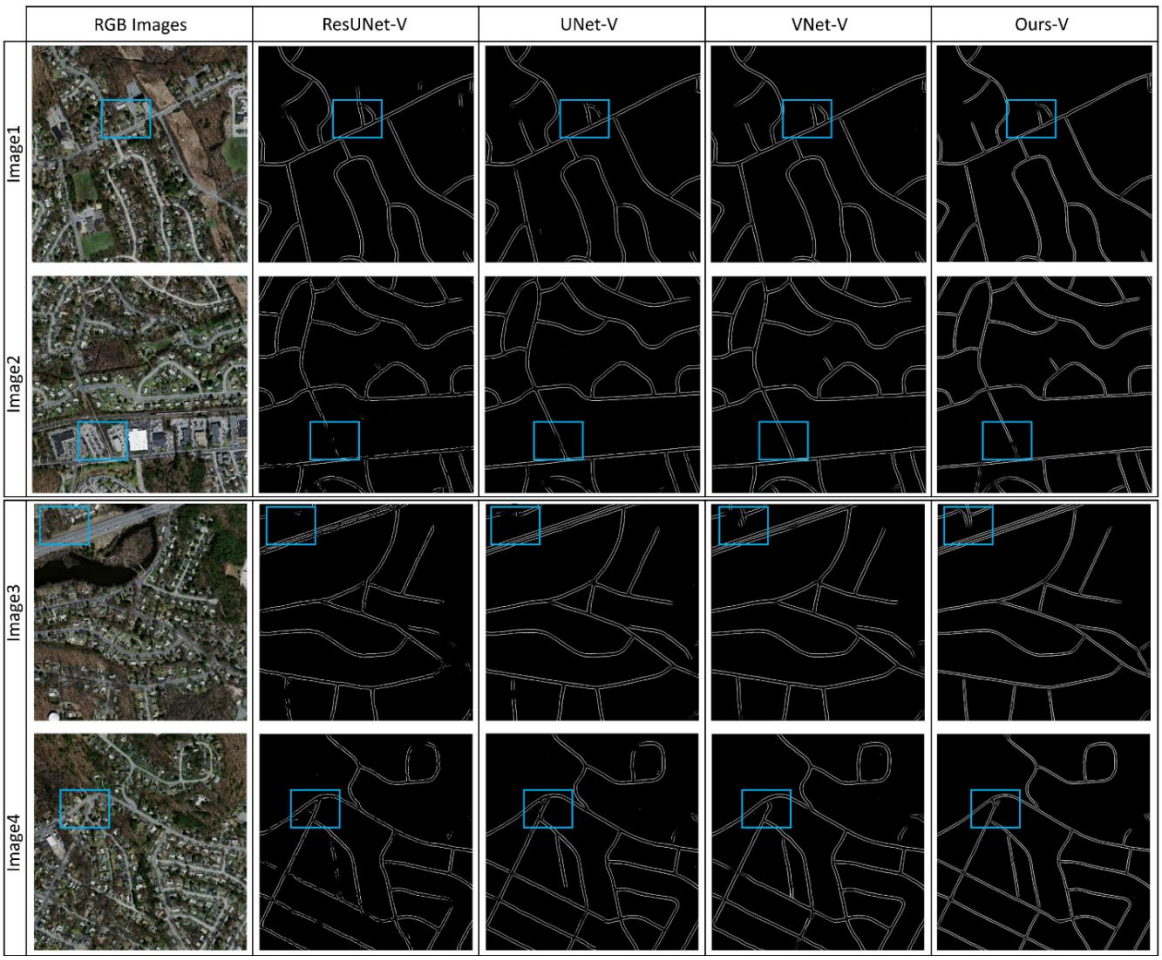

Figure 4.28. Comparison outcomes of our approach and the other comparative models for road vectorization in visual performance for Massachusetts imagery. The first column demonstrates the original RGB imagery. The second, third, fourth, and last columns demonstrate the results of ResUNet-V, UNet-V, VNet-V, and Ours-V, respectively. More details can be seen in the zoomed-in view.

Table 4.24 and 4.25 illustrate that Ours-S, along with other comparative convolutional networks, could attain satisfactory outcomes for road segmentation from both datasets. However, the DeepLabV3-S, ResUNet-S, and SegNet-S architectures achieved the lowest F1 score accuracy with 85.83%. 86.97%, and 87% for Massachusetts and 90.54%, 90.72%, and 91.48% for Ottawa. The SegNet-S model could slightly improve the accuracy because it utilizes the max-pooling indices at the encoder and corresponding decoder paths to upsample the layers in the decoding process. The model does not need to learn the

upsampling weights again because this function makes the training process more straightforward.

Table 4.24. Percentage of F1 score**,** MCC, and IOU attained by Ours-S and other comparative networks for road segmentation from Massachusetts imagery. The bold and underline F1 scores demonstrate the best and second-best, respectively.

		Image1	**Image2**	**Image3**	**Image4**	**Average**
FCN-S	F1 score	0.9104	0.9117	0.9028	0.9007	0.9064
	MCC	0.9008	0.9037	0.8910	0.8901	0.8964
	IOU	0.8338	0.8360	0.8212	0.8176	0.8272
SegNet-S	F1 score	0.8680	0.8909	0.8701	0.8511	0.8700
	MCC	0.8554	0.8838	0.8573	0.8324	0.8572
	IOU	0.7654	0.8017	0.7686	0.7394	0.7688
UNet-S	F1 score	0.9128	0.9141	0.9075	0.9073	0.9104
	MCC	0.9017	0.9057	0.8984	0.8942	0.9000
	IOU	0.8378	0.8570	0.8289	0.8286	0.8381
VNet-S	F1 score	0.9122	0.9192	0.9084	0.9173	<u>0.9145</u>
	MCC	0.9023	0.9108	0.8965	0.9067	0.9040
	IOU	0.8385	0.8504	0.8322	0.8473	0.8421
ResUNet-S	F1 score	0.8632	0.8882	0.8668	0.8609	0.8697
	MCC	0.8493	0.8806	0.8539	0.8453	0.8572
	IOU	0.7593	0.7988	0.7649	0.7557	0.7696
DeeplabV3-S	F1 score	0.8564	0.8798	0.8468	0.8503	0.8583
	MCC	0.8383	0.8693	0.8294	0.8303	0.8418
	IOU	0.7475	0.7839	0.7330	0.7382	0.7507
Ours-S	F1 score	0.9243	0.9239	0.9282	0.9240	**0.9251**
	MCC	0.9143	0.9168	0.9190	0.9128	0.9157
	IOU	0.8574	0.8740	0.8641	0.8568	0.8631

Table 4.25. Percentage of F1 score**,** MCC, and IOU attained by Ours-S and other comparative networks for road segmentation from Ottawa imagery. The bold and underline values demonstrate the best and second-best, respectively.

		Image1	**Image2**	**Image3**	**Image4**	**Average**
FCN-S	F1 score	0.8829	0.9150	0.9375	0.9282	0.9159
	MCC	0.8453	0.8887	0.9103	0.8796	0.8810
	IOU	0.7888	0.8415	0.8803	0.8641	0.8437
SegNet-S	F1 score	0.8816	0.9302	0.9371	0.9103	0.9148
	MCC	0.8432	0.9053	0.9102	0.8572	0.8790
	IOU	0.7867	0.8676	0.8797	0.8336	0.8419
UNet-S	F1 score	0.8849	0.9329	0.9321	0.9231	0.9183
	MCC	0.8477	0.9108	0.9028	0.8711	0.8831
	IOU	0.7921	0.8722	0.8709	0.8554	0.8477
VNet-S	F1 score	0.8933	0.9294	0.9390	0.9191	<u>0.9202</u>
	MCC	0.8597	0.9070	0.9137	0.8678	0.8870
	IOU	0.8072	0.8681	0.8850	0.8502	0.8526
ResUNet-S	F1 score	0.8761	0.9160	0.9372	0.8995	0.9072
	MCC	0.8137	0.8887	0.9110	0.8159	0.8573
	IOU	0.7484	0.8450	0.8818	0.7949	0.8175
DeeplabV3-S	F1 score	0.8731	0.9274	0.9330	0.8884	0.9054
	MCC	0.8101	0.9016	0.9027	0.8229	0.8593
	IOU	0.7427	0.8627	0.8725	0.7759	0.8135
Ours-S	F1 score	0.8992	0.9412	0.9434	0.9520	**0.9340**
	MCC	0.8666	0.9202	0.9186	0.9190	0.9061
	IOU	0.8152	0.8869	0.8909	0.9062	0.8748

Table 4.24 and 4.25 also show that the VNet-S framework was the second-best approach in road surface segmentation, with 91.45% for Massachusetts and 92.02% for Ottawa. By contrast, the accuracy of the F1 score metric for Ours-S was higher than all the comparative approaches. In fact, the presented model could improve the F1 score accuracy by 1.06%

for Massachusetts and 1.38% for Ottawa compared with the VNet-S network, which was the second-best model. Furthermore, I compared the quantitative results achieved by the proposed model with more deep learning-based models, such as CNN-based segmentation method [66], road structure-refined CNN (RSRCNN) technique [55], and FCNs approach [37] applied for road segmentation from Massachusetts imagery. The presented method was built and evaluated on an experimental dataset, while the outcomes for the other three works were taken from a previously published study. The F1 score accuracy achieved by the CNN-based approach, RSRCNN, and FCNs were 82%, 66.2%, and 68%, respectively, while that of Ours-S approach is 92.51%. The results confirmed that the more supervised information in the presented model obtains better outcomes against the other pre-existing deep learning approaches in road surface segmentation from the HRSI.

4.5.4. Quantitative comparison of road vectorization

I calculated the F1 score and MCC metrics to better probe the capability of Ours-V and other comparative models in road vectorization. The qualitative outcomes for the Ottawa (Google Earth) and Massachusetts (aerial) imagery are demonstrated in Tables 4.26 and Table 4.27, respectively. Tables 4.26 and 4.27 show that the VNet-V could achieve satisfactory results for road vectorization from the Ottawa imagery with 91.27% F1 score accuracy, which could improve the results of other comparative models, such as FCN-V, DeepLabV3-V, ResUNet-V, UNet-V, and SegNet-V, and it was ranked as the second-best model. Nevertheless, this method could not perform well in road vectorization using the Massachusetts imagery (Table 4.27) and predicted more FPs and less FNs, resulting in less F1 score accuracy with 83.73%, which is not very good. This phenomenon is attributed to the aerial images that have more complex backgrounds and occlusions, and the road width is narrow. The other methods that could not achieve a higher F1 score accuracy than VNet-

V for Ottawa images could not obtain a higher accuracy for Massachusetts images as well. By contrast, Ours-V was able to achieve better results than others for both datasets. Ours-V achieved F1 score accuracy rates of 92.41% and 89.24% for Ottawa and Massachusetts imagery, respectively. Ours-V could improve the results of VNet-V (second-best method) to 1.14% for Ottawa and 5.51% for Massachusetts, which confirmed its validity for road vectorization from Google Earth and Arial imagery.

Table 4.26. Percentage of F1 score and MCC attained by Ours-V and other comparative networks for road vectorization from the Ottawa imagery. The bold and underline values denote the best and second-best, respectively.

		Image1	**Image2**	**Image3**	**Image4**	**Average**
FCN-V	F1 score	0.8643	0.9017	0.8893	0.8893	0.8862
	MCC	0.8551	0.8953	0.8825	0.8821	0.8788
SegNet-V	F1 score	0.8622	0.8702	0.8820	0.8776	0.8730
	MCC	0.8563	0.8658	0.8782	0.8722	0.8681
UNet-V	F1 score	0.8999	0.9134	0.9072	0.9288	0.9123
	MCC	0.8931	0.9076	0.9015	0.9241	0.9066
DeeplabV3-V	F1 score	0.8566	0.8699	0.8804	0.8742	0.8703
	MCC	0.8513	0.8650	0.8763	0.8695	0.8655
VNet-V	F1 score	0.9045	0.9129	0.9038	0.9297	<u>0.9127</u>
	MCC	0.8985	0.9071	0.8973	0.9254	0.9070
ResUNet-V	F1 score	0.8614	0.8734	0.8839	0.8874	0.8765
	MCC	0.8529	0.8659	0.8771	0.8807	0.8691
Ours-V	F1 score	0.9203	0.9237	0.9164	0.9358	**0.9241**
	MCC	0.9149	0.9187	0.9110	0.9315	0.9190

Table 4.27. Percentage of F1 score and MCC attained by Ours-V and other comparative networks for road vectorization from Massachusetts imagery. The bold and underline values denote the best and second-best, respectively.

		Image1	Image2	Image3	Image4	Average
FCN-V	F1 score	0.7982	0.8350	0.8204	0.8503	0.8260
	MCC	0.8004	0.8273	0.8095	0.8510	0.8221
SegNet-V	F1 score	0.7917	0.8326	0.8047	0.8424	0.8179
	MCC	0.7763	0.8232	0.7911	0.8333	0.8060
UNet-V	F1 score	0.8129	0.8458	0.8237	0.8514	0.8335
	MCC	0.7994	0.8478	0.8244	0.8421	0.8284
DeeplabV3-V	F1 score	0.7539	0.8263	0.7749	0.8237	0.7947
	MCC	0.7350	0.8176	0.7586	0.8114	0.7807
VNet-V	F1 score	0.8206	0.8470	0.8208	0.8619	<u>0.8373</u>
	MCC	0.8069	0.8388	0.8083	0.8535	0.8268
ResUNet-V	F1 score	0.7771	0.8307	0.7814	0.8298	0.8047
	MCC	0.7596	0.8218	0.7653	0.8180	0.7911
Ours-V	F1 score	0.8854	0.8834	0.8878	0.9129	**0.8924**
	MCC	0.8762	0.8754	0.8794	0.9066	0.8844

The average F1 score accuracy attained by our approach and other comparative approaches in road surface segmentation and road vectorization from both datasets is plotted in Figures 4.29(a) and 4.29(b), respectively. The approaches and the average percentage of the F1 score metric are shown in the horizontal and vertical axes, respectively. Figure 4.29 depicts that the Ours-S and Ours-V methods achieved the highest F1 score, affirming the superiority of the proposed technique for road vectorization from Google Earth and Arial imagery. Figures 4.30(a) and 4.30(b) display the training and validation losses of the presented approach over 100 epochs for Ottawa and Massachusetts imagery, respectively. Based on the decrease in model loss, the method has learned efficient features for road

surface segmentation and vectorization. The training and validation losses are close together in the learning curve for both datasets. The model reduced over-fitting, and the variance of the method is negligible.

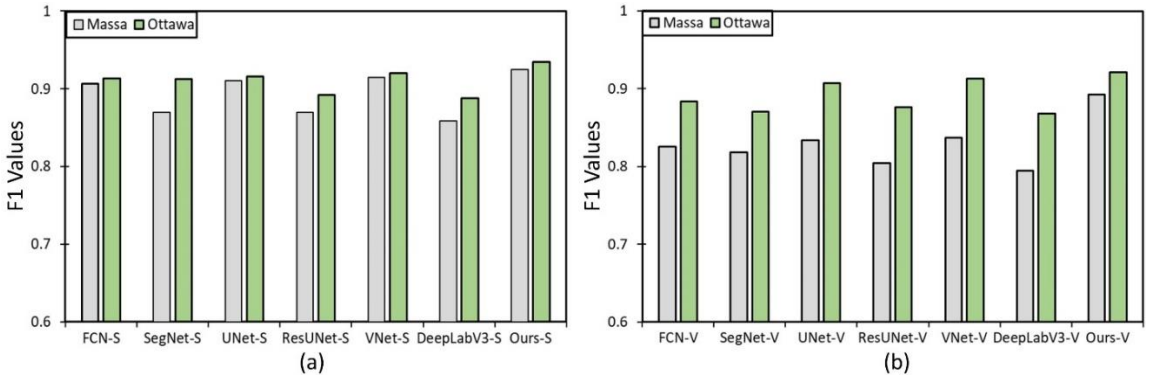

Figure 4.29. Average percentage of the F1 score metric of our method and other methods for road surface segmentation (a) and road vectorization (b) from Ottawa and Massachusetts imagery.

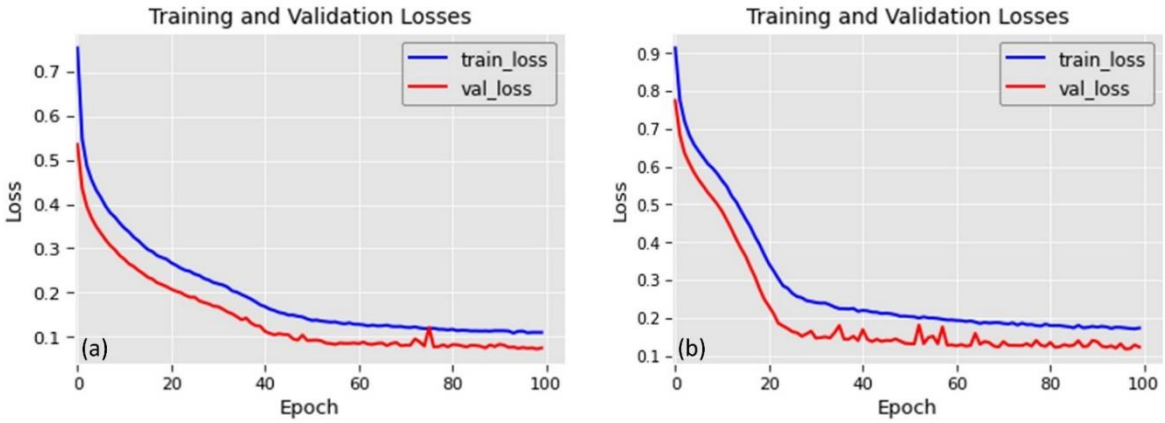

Figure 4.30. Performance of the proposed model for road segmentation and vectorization through training epochs: training and validation losses for the (a) Ottawa and (b) Massachusetts datasets.

4.5.5. Ablation study

I conducted some tests to see how different settings affected the model's performance in road surface segmentation and vectorization. In this case, I used PSPNet backbone [172], stochastic gradient descent with a 0.01 learning rate, and batch size of 4. The quantitative results for both tasks are shown in Table 4.28. Meanwhile, the visualization results for

road segmentation and vectorization tasks are depicted in Figures 4.31 and 4.32, respectively. Table 4.28 illustrates that the accuracy of the F1 score decreased to 91.77% and 87.75% for road segmentation and 90.02% and 85.32% for road vectorization for the Massachusetts and Ottawa images, respectively, after changing some settings. Figures 4.31 and 4.32 also show that the proposed model brought in spurs and produced some FPs in the homogenous regions, thereby considerably decreasing the smoothness of the road vectorization network.

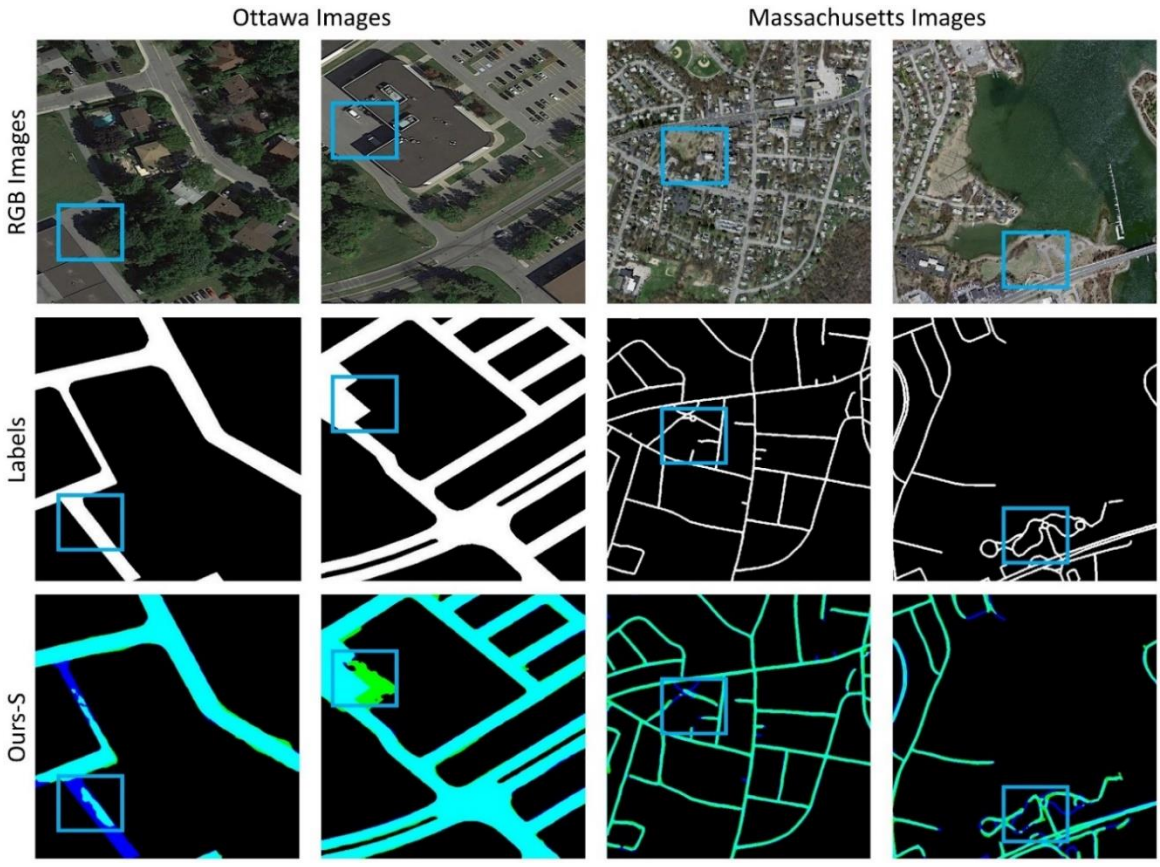

Figure 4.31. Visual performance attained by Ours-S network for road surface segmentation from the Ottawa and Massachusetts imagery after changing several settings. The cyan, green, and blue colors denote the TPs, FPs, and FNs, respectively.

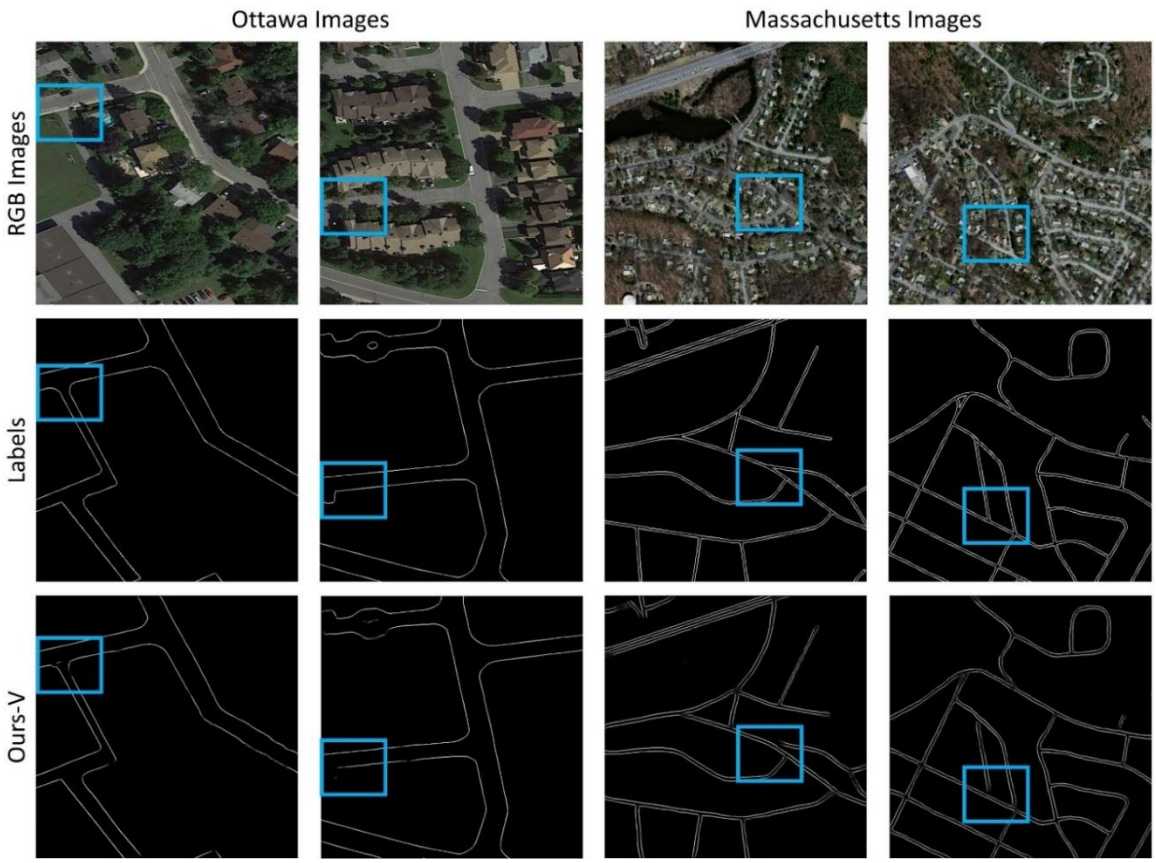

Figure 4.32. Visual performance attained by Ours-V for road vectorization from the Massachusetts and Ottawa imagery after changing several settings. The blue rectangle shows the predicted FPs and FNs. More details can be seen in the zoomed-in view.

4.5.6. Failure case analysis

In this case, I conducted some failure case analysis by reducing the size of images to 256×256 to check the model's performance on road segmentation and vectorization. Table 4.29 shows the quantitative results for both tasks. Meanwhile, Figures 4.33 and 4.34 illustrate the visualization results for road segmentation and vectorization tasks, respectively. In Table 4.29, when the size of the image was halved, the accuracy of the F1 score was decreased for road segmentation to 88.87% and 84.44% and road vectorization to 87.02% 81.61% for both Massachusetts and Ottawa imagery, respectively. In addition, Figures 4.33 and 4.34 depict that the proposed model showed more noise and confused lanes with each other for road segmentation from both datasets. Moreover, the model

produced a non-complete vectorized road network for road vectorization, especially for complicated and intersection areas wherein they brought in more FPs when I decreased the image size. The model could learn considerably less, and the images were distinguished as failure due to overfitting when the image size was reduced. Accordingly, the detection accuracy was greatly diminished. Therefore, reducing the image input size was ineffective for producing high-quality road segmentation and vectorization maps.

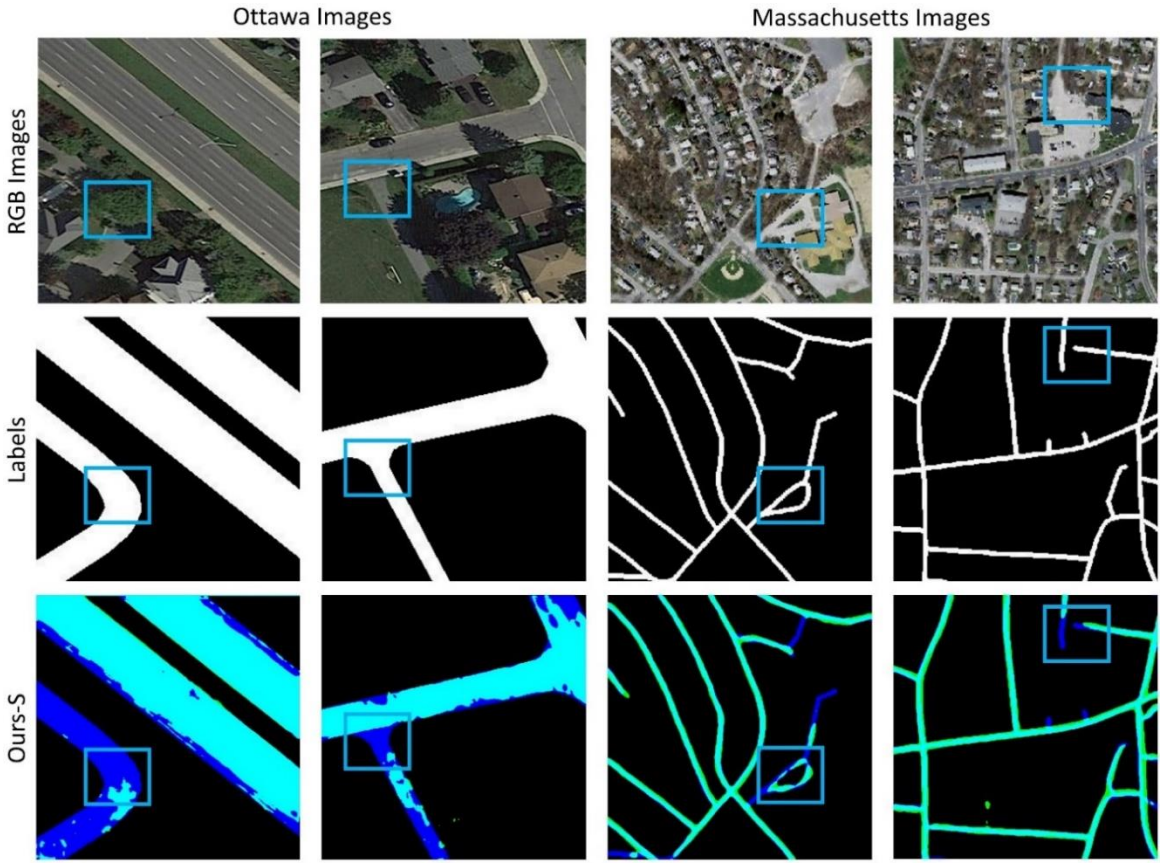

Figure 4.33. Visual performance attained by Ours-S network for road surface segmentation from the Ottawa and Massachusetts imagery after analyzing a failure case. The cyan, green, and blue colors denote the TPs, FPs, and FNs, respectively.

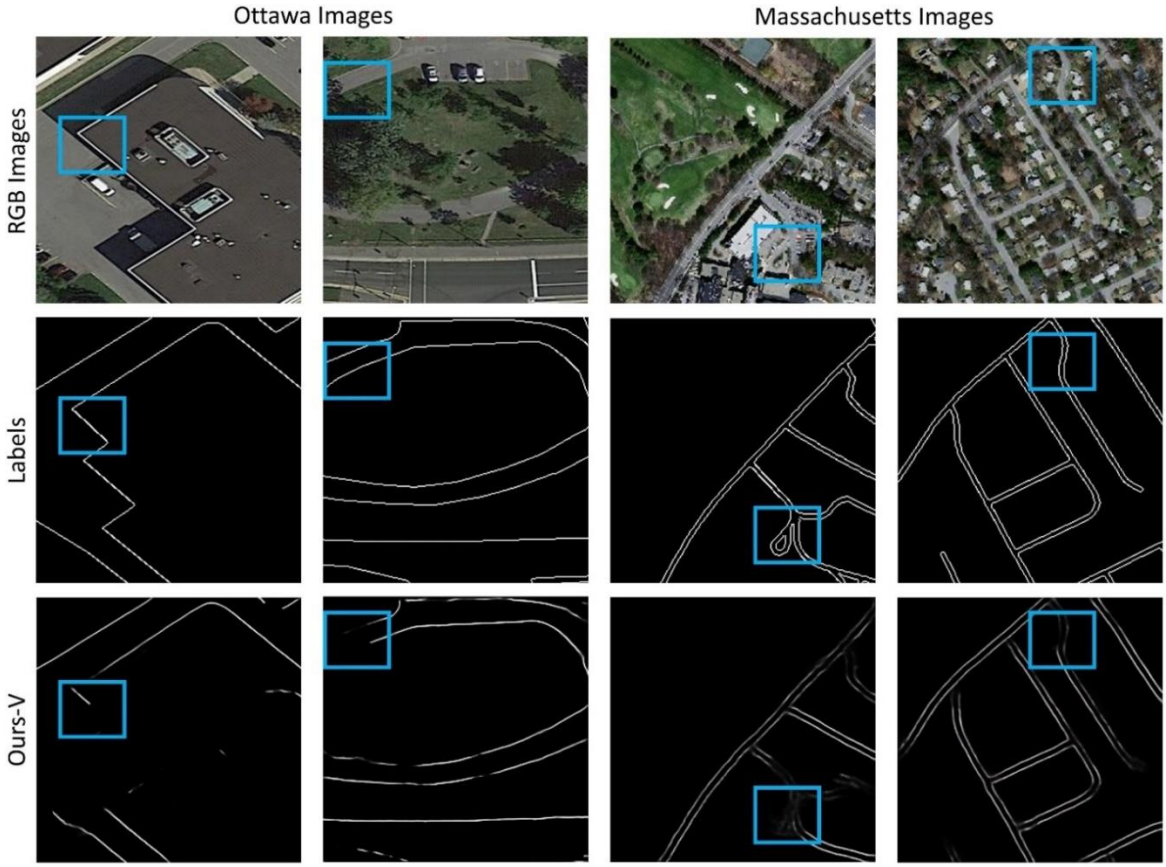

Figure 4.34. Visual performance attained by Ours-V for road vectorization from the Massachusetts and Ottawa imagery after analyzing a failure case. The blue rectangle shows the predicted FPs and FNs. More details can be seen in the zoomed-in view.

Table 4.28. Percentage of the F1 score, IOU, and MCC attained by Ours-V network for road segmentation and vectorization from the Massachusetts and Ottawa imagery after changing several settings.

Road Segmentation	Ottawa	F1 score	0.9177
		MCC	0.8984
		IOU	0.8684
	Massachusetts	F1 score	0.8775
		MCC	0.8662
		IOU	0.7818
Road Vectorization	Ottawa	F1 score	0.9002
		MCC	0.9048
	Massachusetts	F1 score	0.8532
		MCC	0.8460

Table 4.29. Percentage of the F1 score, IOU, and MCC attained by Ours-V network for road segmentation and vectorization from the Massachusetts and Ottawa imagery after analyzing a failure case.

Road Segmentation	Ottawa	F1 score	0.8887
		MCC	0.8291
		IOU	0.7997
	Massachusetts	F1 score	0.8444
		MCC	0.8226
		IOU	0.7308
Road Vectorization	Ottawa	F1 score	0.8702
		MCC	0.8637
	Massachusetts	F1 score	0.8161
		MCC	0.8045

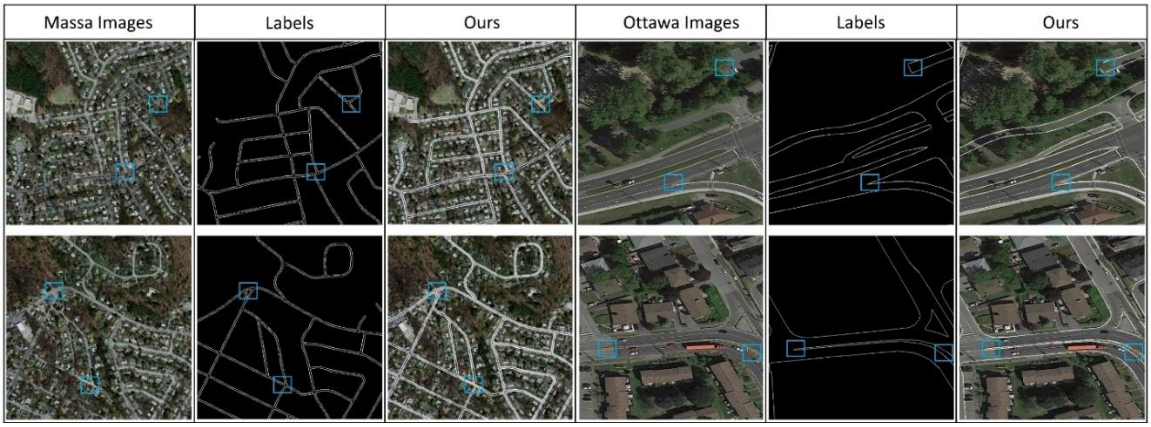

Figure 4.35. The vectorized road is superimposed with the original Aerial (Massachusetts) and Google Earth (Ottawa) imagery to show the overall geometric quality of vectorized outcomes. The first and second rows demonstrate the Aerial images, and the third and last rows illustrate the Google Earth images. The last column also demonstrates the superimposed vectorized road. More details can be seen in the zoomed-in view.

Figure 4.35 demonstrates the vectorized road results overlaid on the original Google Earth and Aerial imagery to prove the overall geometric quality of the road segmentation and vectorization by the model. I calculated the root-mean-square (RMS) of road widths based on the quadratic mean distance between the matched references and extracted widths. The vectorization of the classified outcomes achieved width RMS values of 1.47 and 0.63 m

for Massachusetts and Ottawa images, respectively, proving that the proposed model could achieve precise information about road width. Moreover, the proposed network could extract the precise location of the road network because the vectorized road maps are well superimposed with the original imagery.

4.6. Summary

The application of the developed ML and DCNN approaches for road extraction and vectorization yields the following findings in Chapter 4:

1. The assessment of the proposed models was conducted on different types of high-resolution remote sensing images, and both quantitative and qualitative results were pinpointed.

2. Several traditional ML methods such as Level Set approach and classification methods (e.g.., DT, SVM and KNN) with connected components analysis and segmentation method were first used for road surface segmentation from UAV and Orthophoto images and the results of each were discussed. Although all the mentioned techniques achieved reliable accuracy for road extraction from high-resolution remote sensing imagery, they missed some road segments where there is low visibility of road segments in the images. Also, the proposed techniques faced some errors in road segmentation because of the road-like patterns in some areas of images. Moreover, the traditional ML methods can only be applied to a limited dataset.

3. In the first objective, several robust DCNN models with additional functions and modules such as VNet, GAN+MUNet, MCG-UNet, and BCD-UNet were applied to alleviate the shortcomings of the conventional ML methods in road surface segmentation. The models were tested on various HRSI datasets, including Google Earth and Aerial

images. The proposed techniques were compared with other state-of-the-art approaches that showed better results in road extraction than others. However, the suggested techniques were not effective in accurate road segmentation from areas where the road networks are covered by trees and parking lots, resulting in fragmented and discontinuous road networks.

4. In the second objective, the SC-RoadDeepNet was performed to solve the issue of the suggested DCNN models for road extraction and maintain the road's shape and connectivity. The proposed technique was tested on different remote sensing datasets such as Google Earth images, DeepGlobe, and Massachusetts datasets. The experimental results demonstrated that the suggested approach outperformed other comparative models in terms of maintaining shape and road connectivity while also producing high-resolution segmentation maps.

5. In the third objective, the RoadVecNet was developed for road extraction and vectorization simultaneously. In terms of effectiveness, constraints achieved findings, and validation, the suggested RoadVecNet model differed from pre-existing DL models.

6. The suggested RoadVecNet model demonstrated robustness in simultaneous road segmentation and vectorization, allowing for the achieving of accurate information on the location of the road network and road width.

CHAPTER 5

CONCLUSIONS AND FUTURE WORK RECOMMENDATIONS

5.1. General

This study deployed high-resolution remote sensing images (HRSI) such as Orthophoto, UAV, Aerial, and Google Earth images and complete corresponding ground truth images. The study is aimed at the development of state-of-the-art deep convolutional neural networks (DCNN) for automatic road maps verification and updating. This study applied various types of DCNN methods such as GAN+MUNet, VNet, MCG-UNet, BCD-UNet, SC-RoadDeepNet, and RoadVecNet networks in python. As a result, this research generated high-resolution road segmentation and vectorization maps from the aforementioned datasets using the developed DCNN networks. Introducing novel robust approaches for road network extraction and vectorization from HRSI data would be useful for intelligent transportation systems (ITS) and geospatial information systems (GIS). However, there are some issues that make the process of extracting road parts from high-resolution remote sensing imagery more difficult. For example, high-resolution images are complex and other features such as vehicles on the roads, building on the roadsides, and trees shadows can be observed from these images. This is because these features present similar spectral values as road pixel values, and inadequate context of road parts is similar to these objects in the remote sensing imagery. In addition, road segments are irregular and road networks present complex structures in the remote sensing images. Although many methods, techniques, approaches, and procedures for road networks mapping have been developed, the majority of these methods are sophisticated, conventional, and time-

consuming. Thus, the artificial intelligence (AI) approaches encouraged by the reliable efficiency of DCNN architectures simplify the road networks vectorization and road database updating from HRSI and produce highly accurate results.

5.2. Conclusions of traditional ML methods

In this study, different conventional ML methods were applied for road extraction from HRSI data, which the results and limitations of the methods are concluded in this part.

First, a new interactive approach of TWS and LS methods was introduced for extracting urban and suburban roads. The suggested method consists of steps such as segmentation of approximated road sections from UAV images using TWS, adoption of a LS method for extracting roads from the segmented images and implementation of a morphological operation for eliminating undesirable sections and closing of holes inside the road component. The proposed approach was performed on two UAV images, and results proved its efficiency for road extraction. The achieved results were compared with the manually digitized road layer. Performance factors, such as completeness, correctness and quality, were evaluated in this work; the average values obtained were 93.52%, 85.79% and 84.18% and 81.01%, respectively. These performance findings were compared with those in other works, with plots for illustrative comparison. Comparison results verified that the method is highly efficient for road extraction from UAV images. TWS has great potential for image segmentation, and the LS method is efficient for road extraction from high-resolution remote sensing images. However, road pixel extraction from images needs considerable computation. The approach has several benefits that render it suitable for road extraction from UAV images. One of the most important issues is that the approach is capable of eliciting sections not only of curved roads but also of straight ones. Moreover,

the approach can distinguish obstacles, such as cars, vegetation and buildings, from road class. Limited image transparency restricts road extraction and causes failures during extraction. Another reason for failures in road extraction is the availability of real comparable pixels to roads in images, which can be eliminated using filtering techniques.

Second, a new integrated model of segmentation and classification methods with connected components analysis was introduced to extract road parts from VHR orthophoto images. The introduced model included three main steps. First, the multiresolution segmentation approach was applied to segment orthophoto images. The obtained results were then processed by the classification methods, such as SVM, KNN, and DT, to categorize the image into the road and non-road sections. Training the approaches utilized not only spectral information but also included texture and geometry information to improve the accuracy of the model. Finally, connected components labelling and morphological operations were performed to delete some components that do not belong to the road section, fill the gaps, and enhance the model performance for road extraction. Three different orthophoto images were used for applying the methods, and the final outcomes proved that the suggested models were capable of road extraction with satisfactory results. The roads layer was manually digitized to compare the results achieved by the suggested approaches, and three common accuracy metrics, such as recall, precision, and F1score, were calculated. The average metrics percentage obtained by the suggested methods were 87.62%, 89.71%, and 88.61%, respectively, for DT; 86.61%, 88.17%, and 87.30%, respectively, for KNN; and 89.83%, 89.52%, and 89.67%, respectively, for SVM. The results from different accuracy assessment factors were also compared with those of other previous studies, which showed that the integrated model was still efficient in terms of accurate road region extraction from orthophoto images. The novelty of the proposed

integrated method lies in its capability to distinguish and extract straight and curved road parts. However, some parts of the road in the image are entirely covered by trees and shadows, making accurate road extraction from these parts difficult. Therefore, this difficulty is considered a limitation and deficiency of the integrated approach. In addition, the traditional ML techniques can be applied to a limited number of images. Therefore, as a first objective, I applied robust DCNN models on the large datasets to extract road networks more accurately than ML methods and solve the issues of these methods.

5.3. Conclusions of objective 1

In this study, various types of DCNN approaches were applied for road surface segmentation, which each method's shortcomings and results are highlighted in this section.

In the first work, I proposed a deep learning approach for segmenting road regions from high-resolution images that incorporates two new innovations: a modified UNet (MUNet) architecture for the extraction of road regions and a generative adversarial neural network (GAN) framework for optimizing learning and improving the accuracy of the segmentation map. Experimental results validated the efficacy of the proposed approach. Compared with prior state-of-the-art approaches and GAN-based road detection methods, the proposed GAN framework offered significant improvements in precision and in the F1 score metrics. Visual comparison indicated that the proposed GAN approach yields high-quality segmentation maps where, compared with prior approaches, the edges are particularly well preserved and in agreement with ground truth labels.

In the second work, I applied a new deep convolutional neural network called VNet model to extract road networks from HRSI. Also, I implemented a new loss function named CEDL to decrease the problem of class imbalance in our datasets and improved the result

of road segmentation. I utilized two different remote sensing datasets such as Massachusetts and Ottawa road datasets that contained Aerial imagery and Google Earth imagery, respectively. Also, I calculated different important accuracy measurements such as F1, MCC and IOU to evaluate the performance of the suggested technique in road extraction. The proposed VNet+CEDL model could achieve an average F1 score of 91.18% for Massachusetts dataset and 91.29% for Ottawa dataset confirmed that the model could obtain accurate road results and produce high-resolution segmentation maps. Moreover, the proposed deep convolutional model was compared with other deep learning-based techniques, and the visual and quantitative outcomes proved the superiority of the proposed method in road extraction from HRSI.

In the third work, I used two new deep learning-based networks in this research, namely, BCL-UNet and MCG-UNet, which were inspired by UNet, dense connections, SE, and BConvLSTM, for the segmentation of roads from aerial imagery. The presented networks were tested on the Massachusetts and DeepGlobe road datasets. The results achieved by the presented BCL-UNet framework and MCG-UNet models were firstly compared. The qualitative and quantitative products proved that both frameworks worked better than others and generated an accurate segmentation map for road object. To show the efficiency of the introduced models in road segmentation, I also compared the BCL-UNet and MCG-UNet quantitative and visualization findings to those of other state-of-the-art comparative models used for road segmentation. The empirical consequences affirmed the advantage of the offered techniques for the extraction of road object from aerial imagery. In summary, the proposed techniques could detect roads well even in incessant and prominent regions of closures and could also generate high-resolution and non-noisy road segmentation maps from different datasets.

As it is discussed, the newly developed DCNN models showed substantial improvements in road extraction compared to the comparative DL methods and traditional ML methods, as well as yielded high-quality road segmentation maps. However, the accuracy of the proposed deep learning models is slightly lower, and the method could neither identify roads from complex areas nor extract continuous road parts from these images and produce fragmented results, where the roads are covered by other obstructions. These factors are the main limitations of the proposed methods. Thus, I addressed these limitations and developed a new DCNN model, which uses some topological characteristics like connectivity to improve the accuracy of our proposed approaches for road extraction.

5.4. Conclusions of objective 2

In this work, I introduced SC-RoadDeepNet, a new method for extracting roads from remote sensing imagery based on a shape and connectivity-preserving road segmentation deep learning model. The proposed model consists of a state-of-the-art deep learning model called the RRCNN model, BL, and CP_clDice techniques. The RRCNN model includes convolutional encoder-decoder units similar to the primary UNet model. However, in the encoder-decoder arms, RRCLs were employed instead of standard forward convolutional layers. RRCLs aids in the development of a more effective deeper structure. Furthermore, the suggested model's RRCL units provide an effective feature accumulation mechanism. Concerning distinct time-steps, feature accumulation guarantees stronger and better feature representation. As a result, it aids in the extraction of low-level features that are critical for segmentation tasks. I also employed BL to punish boundary misclassification and fine-tune the road form as a result. I provided CP_clDice for maintaining road connectivity and obtaining correct segmentations. The suggested framework was tested on different HRSI datasets, and the findings demonstrated its usefulness and feasibility in increasing the

performance of road semantic segmentation. Qualitative comparisons were compared with several comparative semantic segmentation algorithms. The presented model outperformed the other models, preserving shape and road connectivity and achieving high-resolution segmentation maps according to the results of the experiments. Compared with the aforementioned semantic segmentation methods, the suggested method could also improve the complete assessment metrics, such as IOU and F1 score. However, most of the pre-existing traditional ML and DCNN models were implemented on HRSI for road surface segmentation and road centerline extraction, not road network vectorization, including accurate road width and location information. Thus, because the semantic segmentation results from the remote sensing images cannot be used for navigation and urban planning, it is essential to extract the vector of roads (location and width/length information) from remote sensing data that can be utilized for updating road database.

5.5. Conclusions of objective 3

In this study, a new interlinked end-to-end UNet framework called RoadVecNet was proposed to implement road surface segmentation and road vectorization simultaneously. The first network in the RoadVecNet architecture was used to produce feature maps. Meanwhile, the second network was performed to formulate road vectorization. The Sobel method was utilized to achieve a complete and smooth vectorized road with accurate road-width information. Two separate datasets, namely, road surface segmentation and road vectorization datasets, were used to train the model. The advantage of the proposed model was verified with rigorous experiments: 1) Two different road datasets imagery called Ottawa (Google Earth) and Massachusetts (Aerial) datasets, which comprise the original RGB images, corresponding ground truth segmentation maps, and corresponding ground truth vector maps, were employed to test the model for road segmentation and

vectorization. 2) In the road surface segmentation tasks, the proposed RoadVecNet could achieve more consistent and smooth road segmentation outcomes than all the comparative models in terms of visual and qualitative performance as well as showed robustness against the obstacle. 3) In the road vectorization task, RoadVecNet also showed better performance than the other comparative state-of-the-art deep convolutional architectures.

In summary, the suggested techniques and the study's findings (high quality and accurate road network data) have great promise for environmental applications including urban land use change detection and emergency tasks. They also have commercial value for navigation and updating road maps.

5.6. Limitations and Future work recommendations

The suggested DCNN approaches were applied in this study, and all three objectives were met. Furthermore, more road network extraction and vectorization works can be done using detailed datasets and the most up-to-date DCNN algorithms. The future work recommendations are listed as follow:

1. More studies can be done to address the constraints of our proposed road extraction and vectorization method by incorporating topological criteria and gap-filling methods into it to improve its accuracy.

2. The developed model for road vectorization and updating road database in this study can be exercised in other remote sensing datasets, e.g., medium-resolution images, LIDAR, and SAR images for the given task.

3. The employed methods in this work can be used to extract and vectorize other urban features like buildings, trees crowns, etc., with the purpose of obtaining complete information on urban features.

4. The current study can be improved further by applying the methods to multi-object segmentation and vectorization from remote sensing data simultaneously. For this, there is a need to prepare datasets including ground truth images with more classes to extract and vectorize the urban objects at the same time.

5. Future research could focus on mapping road vectorization all over the world to see how well the suggested method generalizes on a global scale.

REFERENCE

[1] S. Abdullahi, B. Pradhan, and M. N. Jebur, "GIS-based sustainable city compactness assessment using integration of MCDM, Bayes theorem and RADAR technology," *Geocarto International,* vol. 30, no. 4, pp. 365-387, 2015.

[2] A. M. Youssef, S. A. Sefry, B. Pradhan, E. A. Alfadail, and Risk, "Analysis on causes of flash flood in Jeddah city (Kingdom of Saudi Arabia) of 2009 and 2011 using multi-sensor remote sensing data and GIS," *Geomatics, Natural Hazards,* vol. 7, no. 3, pp. 1018-1042, 2016.

[3] Q. Weng, "Remote sensing of impervious surfaces in the urban areas: Requirements, methods, and trends," *Remote Sensing of Environment,* vol. 117, pp. 34-49, 2012, doi: https://doi.org/10.1016/j.rse.2011.02.030.

[4] I. Kahraman, M. K. Turan, and I. R. Karas, "Road detection from high satellite images using neural networks," *International Journal of Modeling Optimization,* vol. 5, no. 4, pp. 304-307, 2015.

[5] W. Shi, Z. Miao, J. J. I. T. o. G. Debayle, and R. Sensing, "An integrated method for urban main-road centerline extraction from optical remotely sensed imagery," vol. 52, no. 6, pp. 3359-3372, 2014.

[6] J. Hormese and C. Saravanan, "Automated road extraction from high resolution satellite images," *Procedia Technology,* vol. 24, pp. 1460-1467, 2016.

[7] H. R. R. Bakhtiari, A. Abdollahi, and H. Rezaeian, "Semi automatic road extraction from digital images," *The Egyptian Journal of Remote Sensing and Space Science,* vol. 20, no. 1, pp. 117-123, 2017, doi: https://doi.org/10.1016/j.ejrs.2017.03.001.

[8] B. Liu, H. Wu, Y. Wang, and W. Liu, "Main road extraction from zy-3 grayscale imagery based on directional mathematical morphology and vgi prior knowledge in urban areas," *PloS one,* vol. 10, no. 9, pp. 1-16, 2015.

[9] Z. Miao, W. Shi, P. Gamba, and Z. Li, "An object-based method for road network extraction in VHR satellite images," *IEEE journal of selected topics in applied earth observations and remote sensing,* vol. 8, no. 10, pp. 4853-4862, 2015.

[10] I. Grinias, C. Panagiotakis, G. Tziritas, and r. sensing, "MRF-based segmentation and unsupervised classification for building and road detection in peri-urban areas of high-resolution satellite images," *ISPRS journal of photogrammetry,* vol. 122, pp. 145-166, 2016.

[11] M. O. Sghaier and R. Lepage, "Road extraction from very high resolution remote sensing optical images based on texture analysis and beamlet transform," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing,* vol. 9, no. 5, pp. 1946-1958, 2016.

[12] C. He, Z.-x. Liao, F. Yang, X.-p. Deng, and M.-s. Liao, "Road extraction from SAR imagery based on multiscale geometric analysis of detector responses," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing,* vol. 5, no. 5, pp. 1373-1382, 2012.

[13] J. Cheng, W. Ding, X. Ku, and J. Sun, "Road Extraction from High-Resolution SAR Images via Automatic Local Detecting and Human-Guided Global Tracking," *International Journal of Antennas and Propagation,* vol. 2012, pp. 1-10, 2012, doi: http://dx.doi.org/10.1155/2012/989823.

[14] R. Alshehhi and P. R. Marpu, "Hierarchical graph-based segmentation for extracting road networks from high-resolution satellite images," *ISPRS Journal of Photogrammetry and Remote Sensing,* vol. 126, pp. 245-260, 2017, doi: https://doi.org/10.1016/j.isprsjprs.2017.02.008.
[15] Y. Xu, Z. Chen, Z. Xie, and L. Wu, "Quality assessment of building footprint data using a deep autoencoder network," *International Journal of Geographical Information Science,* vol. 31, no. 10, pp. 1929-1951, 2017.
[16] A. Abdollahi, B. Pradhan, and A. M. Alamri, "An Ensemble Architecture of Deep Convolutional Segnet and Unet Networks for Building Semantic Segmentation from High-resolution Aerial Images," *Geocarto International,* pp. 1-13, 2020.
[17] A. Abdollahi and B. Pradhan, "Urban Vegetation Mapping from Aerial Imagery Using Explainable AI (XAI)," *Sensors,* vol. 21, no. 14, p. 4738, 2021.
[18] N. Audebert, B. Le Saux, and S. Lefèvre, "Segment-before-detect: Vehicle detection and classification through semantic segmentation of aerial images," *Remote Sensing,* vol. 9, no. 4, p. 368, 2017.
[19] J. Wang, Q. Qin, Z. Gao, J. Zhao, and X. Ye, "A new approach to urban road extraction using high-resolution aerial image," *ISPRS International Journal of Geo-Information,* vol. 5, no. 7, p. 114, 2016.
[20] F. Hu, G.-S. Xia, J. Hu, and L. Zhang, "Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery," *Remote Sensing,* vol. 7, no. 11, pp. 14680-14707, 2015.
[21] W. Li, G. Wu, F. Zhang, and Q. Du, "Hyperspectral image classification using deep pixel-pair features," *IEEE Transactions on Geoscience Remote Sensing,* vol. 55, no. 2, pp. 844-853, 2017.
[22] J. Senthilnath, A. Dokania, M. Kandukuri, K. Ramesh, G. Anand, and S. Omkar, "Detection of tomatoes using spectral-spatial methods in remotely sensed RGB images captured by UAV," *Biosystems engineering,* vol. 146, pp. 16-32, 2016.
[23] J. D. Wegner, J. A. Montoya-Zegarra, and K. Schindler, "A higher-order CRF model for road network extraction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1698-1705.
[24] R. Maurya, P. Gupta, and A. S. Shukla, "Road extraction using k-means clustering and morphological operations," in *2011 International Conference on Image Information Processing, Shimla, India*, 2011: IEEE, pp. 1-6, doi: https://doi.org/10.1109/ICIIP.2011.6108839.
[25] G. Mattyus, S. Wang, S. Fidler, and R. Urtasun, "Enhancing road maps by parsing aerial images around the world," in *Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile.*, 2015, pp. 1689-1697, doi: https://doi.org/10.1109/ICCV.2015.197.
[26] J. B. Mena and J. A. Malpica, "An automatic method for road extraction in rural and semi-urban areas starting from high resolution satellite imagery," *Pattern recognition letters,* vol. 26, no. 9, pp. 1201-1220, 2005.
[27] C. Zhu, W. Shi, M. Pesaresi, L. Liu, X. Chen**, and B. King, "The recognition of road network from high-resolution satellite remotely sensed data using image morphological characteristics," *International Journal of Remote Sensing,* vol. 26, no. 24, pp. 5493-5508, 2005.
[28] T. Panboonyuen, P. Vateekul, K. Jitkajornwanich, and S. Lawawirojwong, "An enhanced deep convolutional encoder-decoder network for road segmentation on aerial imagery," in *International Conference on Computing and Information*

Technology , Springer, 191-201, 2017. https://doi.org/10.1007/978-3-319-60663-7_18.

[29] S. Tang and Y. Yuan, "Object detection based on convolutional neural network," ed: Stanford University, 2015. http://cs231n.stanford.edu/reports/2015/pdfs/CS231n_final_writeup_ sjtang.pdf.

[30] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," presented at the Proceedings of the 25th International Conference on Neural Information Processing Systems, Lake Tahoe, Nevada, 1, 1097-1105, 2012.

[31] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, vol. 39, no. 4, pp. 3431-3440, doi: https://doi.org/10.1109/TPAMI.2016.2572683.

[32] I. Ševo and A. Avramović, "Convolutional Neural Network Based Automatic Object Detection on Aerial Images," *IEEE Geoscience and Remote Sensing Letters,* vol. 13, no. 5, pp. 740-744, 2016, doi: 10.1109/LGRS.2016.2542358.

[33] M. Volpi and D. Tuia, "Dense Semantic Labeling of Subdecimeter Resolution Images With Convolutional Neural Networks," *IEEE Transactions on Geoscience and Remote Sensing,* vol. 55, no. 2, pp. 881-893, 2017, doi: 10.1109/TGRS.2016.2616585.

[34] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, "Convolutional neural networks for large-scale remote-sensing image classification," *IEEE Transactions on Geoscience Remote Sensing,* vol. 55, no. 2, pp. 645-657, 2017.

[35] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, "Fully convolutional neural networks for remote sensing image classification," in *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Beijing, China.*, 10-15 July 2016 2016, pp. 5071-5074, doi: 10.1109/IGARSS.2016.7730322.

[36] S. Saito, T. Yamashita, and Y. Aoki, "Multiple object extraction from aerial imagery with convolutional neural networks," *Electronic Imaging,* vol. 2016, no. 10, pp. 1-9, 2016.

[37] Z. Zhong, J. Li, W. Cui, and H. Jiang, "Fully convolutional networks for building and road extraction: Preliminary results," in *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, Beijing, China, 2016: IEEE, pp. 1591-1594, doi: https://doi.org/10.1109/IGARSS.2016.7729406.

[38] V. Mnih and G. E. Hinton, "Learning to Detect Roads in High-Resolution Aerial Images," Berlin, Heidelberg, 210-223, 2010. https://doi.org/10.1007/978-3-642-15567-3_16.

[39] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014. [Online]. Available: https://arxiv.org/abs/1409.1556.

[40] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770-778.

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *European Conference on Computer Vision*, 2016: Springer, pp. 630-645.

[42] X. Zhang, X. Han, C. Li, X. Tang, H. Zhou, and L. Jiao, "Aerial image road extraction based on an improved generative adversarial network," *Remote Sensing,* vol. 11, no. 8, p. 930, 2019.

[43] H. Kamangir, M. Momeni, and M. Satari, "Automatic Centerline Extraction of Coverd Roads by Surrounding Objects from High Resolution Satellite Images," *ISPRS-International Archives of the Photogrammetry, Remote Sensing Spatial Information Sciences* pp. 111-116, 2017, doi: https://doi.org/10.5194/isprs-archives-XLII-4-W4-111-2017.

[44] Z. Hong, D. Ming, K. Zhou, Y. Guo, and T. Lu, "Road Extraction From a High Spatial Resolution Remote Sensing Image Based on Richer Convolutional Features," *IEEE Access,* vol. 6, pp. 46988-47000. https://doi.org/10.1109/ACCESS.2018.2867210, 2018.

[45] A. Abdollahi, B. Pradhan, and A. Alamri, "RoadVecNet: a new approach for simultaneous road network segmentation and vectorization from aerial and google earth imagery in a complex urban set-up," *GIScience & Remote Sensing,* pp. 1-24, 2021, doi: 10.1080/15481603.2021.1972713.

[46] F. Casu, M. Manunta, P. Agram, and R. Crippen, "Big Remotely Sensed Data: tools, applications and experiences," *Remote Sensing of Environment,* vol. 202, no. 1, pp. 1-2, 2017.

[47] J. Wang, J. Song, M. Chen, and Z. Yang, "Road network extraction: A neural-dynamic framework based on deep learning and a finite state machine," *International Journal of Remote Sensing,* vol. 36, no. 12, pp. 3144-3169, 2015.

[48] B. Ekim, E. Sertel, and M. E. Kabadayı, "Automatic Road Extraction from Historical Maps Using Deep Learning Techniques: A Regional Case Study of Turkey in a German World War II Map," *ISPRS International Journal of Geo-Information,* vol. 10, no. 8, p. 492, 2021.

[49] Z. Miao, B. Wang, W. Shi, and H. Zhang, "A semi-automatic method for road centerline extraction from VHR images," *IEEE Geoscience and Remote Sensing Letters,* vol. 11, no. 11, pp. 1856-1860, 2014.

[50] C. Unsalan and B. Sirmacek, "Road network detection using probabilistic and graph theoretical methods," *IEEE Transactions on Geoscience and Remote Sensing* vol. 50, no. 11, pp. 4441-4453, 2012, doi: https://doi.org/10.1109/TGRS.2012.2190078.

[51] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature,* vol. 521, no. 7553, pp. 436-444, 2015.

[52] Z. L. Zhang, Qingjie; Wang, Yunhong, "Road Extraction by Deep Residual U-Net," *IEEE Geoscience and Remote Sensing Letters,* vol. 15, no. 5, pp. 749-753, 2018, doi: 10.1109/LGRS.2018.2802944.

[53] A. Abdollahi, B. Pradhan, N. Shukla, S. Chakraborty, and A. Alamri, "Deep Learning Approaches Applied to Remote Sensing Datasets for Road Extraction: A State-Of-The-Art Review," *Remote Sensing,* no. 12, p. 1444, 2020.

[54] P. Luc, C. Couprie, S. Chintala, and J. Verbeek, "Semantic segmentation using adversarial networks," *arXiv preprint arXiv:.08408,* 2016.

[55] Y. Wei, Z. Wang, and M. Xu, "Road Structure Refined CNN for Road Extraction in Aerial Image," *IEEE Geosci. Remote Sensing Lett.,* vol. 14, no. 5, pp. 709-713, 2017.

[56] R. Alshehhi, P. R. Marpu, W. L. Woon, and M. Dalla Mura, "Simultaneous extraction of roads and buildings in remote sensing imagery with convolutional neural networks," *ISPRS Journal of Photogrammetry Remote Sensing,* vol. 130, pp. 139-149, 2017.

[57] R. Liu *et al.*, "Multiscale road centerlines extraction from high-resolution aerial imagery," *Neurocomputing,* vol. 329, pp. 384-396, 2019.

[58] P. Li *et al.*, "Road network extraction via deep learning and line integral convolution," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Beijing, China, 1599-1602, 2016.* https://doi.org/10.1109/IGARSS.2016.7729408.

[59] Z. Chen, C. Wang, J. Li, B. Zhong, J. Du, and W. Fan, "Combined Improved Dirichlet Models and Deep Learning Models for Road Extraction from Remote Sensing Images," *Canadian Journal of Remote Sensing,* pp. 1-20, 2021.

[60] R. Lian and L. Huang, "DeepWindow: Sliding window based on deep learning for road extraction from remote sensing images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing,* vol. 13, pp. 1905-1916, 2020.

[61] N. Varia, A. Dokania, and J. Senthilnath, "DeepExt: A Convolution Neural Network for Road Extraction using RGB images captured by UAV," in *2018 IEEE Symposium Series on Computational Intelligence (SSCI), 1890-1895,*2018. *Bangalore, India.* https://doi.org/10.1109/SSCI.2018.8628717.

[62] T. Moranduzzo and F. Melgani, "Detecting cars in UAV images with a catalog-based approach," *IEEE Transactions on Geoscience Remote Sensing,* vol. 52, no. 10, pp. 6356-6367, 2014.

[63] B. Yang and C. Chen, "Automatic registration of UAV-borne sequent images and LiDAR data," *ISPRS Journal of Photogrammetry Remote Sensing,* vol. 101, pp. 262-274, 2015.

[64] R. Kestur, S. Farooq, R. Abdal, E. Mehraj, O. Narasipura, and M. Mudigere, "UFCN: A fully convolutional neural network for road extraction in RGB imagery acquired by remote sensing from an unmanned aerial vehicle," *Journal of Applied Remote Sensing,* vol. 12, no. 1, p. 016020, 2018.

[65] C. Henry, S. M. Azimi, and N. Merkle, "Road Segmentation in SAR Satellite Images With Deep Fully Convolutional Neural Networks," *IEEE Geoscience Remote Sensing Letters,* no. 99, pp. 1-5, 2018.

[66] Y. Wei, K. Zhang, and S. Ji, "Simultaneous road surface and centerline extraction from large-scale remote sensing images using CNN-based segmentation and tracing," *IEEE Transactions on Geoscience Remote Sensing,* vol. 58, no. 12, pp. 8919-8931, 2020.

[67] T. Panboonyuen, K. Jitkajornwanich, S. Lawawirojwong, P. Srestasathiern, and P. Vateekul, "Road Segmentation of Remotely-Sensed Images Using Deep Convolutional Neural Networks with Landscape Metrics and Conditional Random Fields," *Journal of Remote Sensing,* vol. 9, no. 7, p. 680, 2017.

[68] A. Constantin, J.-J. Ding, and Y.-C. Lee, "Accurate Road Detection from Satellite Images Using Modified U-net," in *2018 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS), Chengdu, China, 423-426,* 2018. *https://doi.org/10.1109/APCCAS.2018.8605652*.

[69] J. Xin, X. Zhang, Z. Zhang, and W. Fang, "Road Extraction of High-Resolution Remote Sensing Images Derived from DenseUNet," *Remote Sensing,* vol. 11, no. 21, p. 2499, 2019.

[70] Y. Li, L. Xu, J. Rao, L. Guo, Z. Yan, and S. Jin, "A Y-Net deep learning method for road segmentation using high-resolution visible remote sensing images," *Remote Sensing Letters,* vol. 10, no. 4, pp. 381-390, 2019.

[71] G. Cheng, Y. Wang, S. Xu, H. Wang, S. Xiang, and C. Pan, "Automatic road detection and centerline extraction via cascaded end-to-end convolutional neural

network," *IEEE Transactions on Geoscience Remote Sensing,* vol. 55, no. 6, pp. 3322-3337, 2017.

[72] Y. Xu, Z. Xie, Y. Feng, and Z. Chen, "Road Extraction from High-Resolution Remote Sensing Imagery Using Deep Learning," *Remote Sensing,* vol. 10, no. 9, p. 1461, 2018, doi: https://doi.org/10.3390/rs10091461.

[73] A. Buslaev, S. Seferbekov, V. Iglovikov, and A. Shvets, "Fully convolutional network for automatic road extraction from satellite imagery, I*EEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 207-210, 2018. https://doi.org/10.1109/CVPRW.2018.00035.

[74] L. Zhou, C. Zhang, and M. Wu, "D-linknet: Linknet with pretrained encoder and dilated convolution for high resolution satellite imagery road extraction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 182-186, 2018. https://doi.org/10.1109/cvprw.2018.00034*.

[75] J. Doshi, "Residual inception skip network for binary segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA,* 216-219, 2018. *https://doi.org/10.1109/cvprw.2018.00037*.

[76] Y. Xu, Y. Feng, Z. Xie, A. Hu, and X. Zhang, "A Research on Extracting Road Network from High Resolution Remote Sensing Imagery," in *2018 26th International Conference on Geoinformatics, Kunming, China,* 1-4, 2018. *https://doi.org/10.1109/GEOINFORMATICS.2018.8557042*.

[77] H. He, D. Yang, S. Wang, S. Wang, and X. Liu, "Road segmentation of cross-modal remote sensing images using deep segmentation network and transfer learning," *Industrial Robot: An International Journal,* 2018. *https://doi.org/10.1108/IR-05-2018-0112*.

[78] W. Xia, Y.-Z. Zhang, J. Liu, L. Luo, and K. Yang, "Road Extraction from High Resolution Image with Deep Convolution Network—A Case Study of GF-2 Image," *Proceedings,* vol. 2, no. 7, p. 325, 2018, doi: https://doi.org/10.3390/ecrs-2-05138.

[79] L. Gao, W. Song, J. Dai, and Y. Chen, "Road Extraction from High-Resolution Remote Sensing Imagery Using Refined Deep Residual Convolutional Neural Network," *Remote Sensing,* vol. 11, no. 5, p. 552, 2019.

[80] Y. Xie, F. Miao, K. Zhou, and J. Peng, "HsgNet: A Road Extraction Network Based on Global Perception of High-Order Spatial Information," *ISPRS International Journal of Geo-Information,* vol. 8, no. 12, p. 571, 2019.

[81] Z. Chen, C. Wang, J. Li, W. Fan, J. Du, and B. Zhong, "Adaboost-like End-to-End multiple lightweight U-nets for road extraction from optical remote sensing images," *International Journal of Applied Earth Observation and Geoinformation,* vol. 100, p. 102341, 2021.

[82] J. Wan, Z. Xie, Y. Xu, S. Chen, and Q. Qiu, "DA-RoadNet: A Dual-Attention Network for Road Extraction from High Resolution Satellite Imagery," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing,* vol. 14, pp. 6302-6315, 2021.

[83] K. Zhou, Y. Xie, Z. Gao, F. Miao, and L. Zhang, "FuNet: A Novel Road Extraction Network with Fusion of Location Data and Remote Sensing Imagery," *ISPRS International Journal of Geo-Information,* vol. 10, no. 1, p. 39, 2021.

[84] Y. Ren, Y. Yu, and H. Guan, "DA-CapsUNet: A dual-attention capsule U-Net for road extraction from remote sensing imagery," *Remote Sensing,* vol. 12, no. 18, p. 2866, 2020.

[85] Z. Shao, Z. Zhou, X. Huang, and Y. Zhang, "Mrenet: Simultaneous extraction of road surface and road centerline in complex urban scenes from very high-resolution images," *Remote Sensing,* vol. 13, no. 2, p. 239, 2021.
[86] S. Wang, X. Mu, D. Yang, H. He, and P. Zhao, "Road Extraction from Remote Sensing Images Using the Inner Convolution Integrated Encoder-Decoder Network and Directional Conditional Random Fields," *Remote Sensing,* vol. 13, no. 3, p. 465, 2021.
[87] S. Wang, H. Yang, Q. Wu, Z. Zheng, Y. Wu, and J. Li, "An improved method for road extraction from high-resolution remote-sensing images that enhances boundary information," *Sensors,* vol. 20, no. 7, p. 2064, 2020.
[88] Q. Shi, X. Liu, and X. Li, "Road detection from remote sensing images by generative adversarial networks," *IEEE Access,* vol. 6, pp. 25486-25494, 2018.
[89] D. Costea, A. Marcu, E. Slusanschi, and M. Leordeanu, "Creating roadmaps in aerial images with generative adversarial networks and smoothing-based optimization," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2017, pp. 2100-2109.
[90] C. Yang and Z. Wang, "An Ensemble Wasserstein Generative Adversarial Network Method for Road Extraction From High Resolution Remote Sensing Images in Rural Areas," *IEEE Access,* vol. 8, pp. 174317-174324, 2020.
[91] C.-I. Cira, M.-Á. Manso-Callejo, R. Alcarria, T. Fernández Pareja, B. Bordel Sánchez, and F. Serradilla, "Generative Learning for Postprocessing Semantic Segmentation Predictions: A Lightweight Conditional Generative Adversarial Network Based on Pix2pix to Improve the Extraction of Road Surface Areas," *Land,* vol. 10, no. 1, p. 79, 2021.
[92] Y. Zhang, Z. Xiong, Y. Zang, C. Wang, J. Li, and X. Li, "Topology-aware road network extraction via Multi-supervised Generative Adversarial Networks," *Remote Sensing,* vol. 11, no. 9, p. 1017, 2019.
[93] J. Senthilnath, N. Varia, A. Dokania, G. Anand, and J. A. Benediktsson, "Deep TEC: Deep transfer learning with ensemble classifier for road extraction from UAV imagery," *Remote Sensing,* vol. 12, no. 2, p. 245, 2020.
[94] Y. Zhang, X. Li, and Q. Zhang, "Road topology refinement via a multi-conditional generative adversarial network," *Sensors,* vol. 19, no. 5, p. 1162, 2019.
[95] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2015: Springer, pp. 234-241.
[96] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Transactions on Pattern Analysis Machine Intelligence,* vol. 39, no. 12, pp. 2481-2495, 2017.
[97] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," *arXiv preprint arXiv:*. 2014.
[98] I. Goodfellow *et al.*, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2014, pp. 2672-2680.
[99] J. Dai, T. Zhu, Y. Wang, R. Ma, and X. Fang, "Road extraction from high-resolution satellite images based on multiple descriptors," *IEEE Journal of Selected Topics in Applied Earth Observations Remote Sensing,* 2020.
[100] N. Ghasemkhani, S. S. Vayghan, A. Abdollahi, B. Pradhan, and A. Alamri, "Urban Development Modeling Using Integrated Fuzzy Systems, Ordered Weighted

Averaging (OWA), and Geospatial Techniques," *Sustainability,* vol. 12, no. 3, p. 809, 2020.
[101] A. K. Gupta, D. J. Bora, and F. A. Khan, "Multispectral satellite color image segmentation using fuzzy based innovative approach," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology,* vol. 3, no. 1, pp. 968-975, 2018.
[102] D. J. Bora and A. K. Gupta, "Clustering approach towards image segmentation: an analytical study," *International Journal of Research in Computer Applications and Robotics,* vol. 2, no. 7, pp. 115-124, 2014.
[103] C. Sujatha and D. Selvathi, "Connected component-based technique for automatic extraction of road centerline in high resolution satellite images," *EURASIP Journal on Image Video Processing,* vol. 2015, no. 1, p. 8, 2015.
[104] H. Yu, J. Wang, Y. Bai, W. Yang, and G.-S. Xia, "Analysis of large-scale UAV images using a multi-scale hierarchical representation," *Geo-spatial information science,* vol. 21, no. 1, pp. 33-44, 2018.
[105] I. Arganda-Carreras *et al.*, "Trainable Weka Segmentation: a machine learning tool for microscopy pixel classification," *Bioinformatics,* vol. 33, no. 15, pp. 2424-2426, 2017.
[106] M. Belgiu and L. Drăguţ, "Random forest in remote sensing: A review of applications and future directions," *ISPRS Journal of Photogrammetry and Remote Sensing,* vol. 114, pp. 24-31, 2016.
[107] Q. Feng, J. Liu, and J. Gong, "UAV remote sensing for urban vegetation mapping using random forest and texture analysis," *Remote Sensing,* vol. 7, no. 1, pp. 1074-1094, 2015.
[108] G. Liasis and S. Stavrou, "Optimizing level set initialization for satellite image segmentation," in *Telecommunications (ICT), 2013 20th International Conference on*, 2013: IEEE, pp. 1-5.
[109] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision,* vol. 1, no. 4, pp. 321-331, 1988.
[110] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations," *J. Comput. Phys.,* vol. 79, no. 1, pp. 12-49, 1988, doi: 10.1016/0021-9991(88)90002-2.
[111] A. Abdollahi and B. Pradhan, "Integrated technique of segmentation and classification methods with connected components analysis for road extraction from orthophoto images," *Expert Systems with Applications,* p. 114908, 2021.
[112] A. Ferraz, C. Mallet, and N. Chehata, "Large-scale road detection in forested mountainous areas using airborne topographic lidar data," *ISPRS Journal of Photogrammetry Remote Sensing,* vol. 112, pp. 23-36, 2016.
[113] H. Aasen, E. Honkavaara, A. Lucieer, and P. Zarco-Tejada, "Quantitative remote sensing at ultra-high resolution with uav spectroscopy: A review of sensor technology, measurement procedures, and data correction workflows," *Remote Sensing,* vol. 10, no. 7, p. 1091, 2018.
[114] A. Grote, C. Heipke, and F. Rottensteiner, "Road Network Extraction in Suburban Areas," *The Photogrammetric Record,* vol. 27, no. 137, pp. 8-28, 2012, doi: 10.1111/j.1477-9730.2011.00670.x.
[115] F. Saba, M. J. Valadan Zoej, and M. Mokhtarzade, "Optimization of Multiresolution Segmentation for Object-Oriented Road Detection from High-Resolution Images,"

Canadian Journal of Remote Sensing, vol. 42, no. 2, pp. 75-84, 2016/03/03 2016, doi: 10.1080/07038992.2016.1160770.

[116] M. Wang and R. Li, "Segmentation of High Spatial Resolution Remote Sensing Imagery Based on Hard-Boundary Constraint and Two-Stage Merging," *IEEE Transactions on Geoscience and Remote Sensing,* vol. 52, no. 9, pp. 5712-5725, 2014, doi: 10.1109/TGRS.2013.2292053.

[117] M. Maboudi, J. Amini, M. Hahn, and M. Saati, "Object-based road extraction from satellite images using ant colony optimization," *International journal of remote sensing,* vol. 38, no. 1, pp. 179-198, 2017.

[118] T. Blaschke, "Object based image analysis for remote sensing," *ISPRS Journal of Photogrammetry and Remote Sensing,* vol. 65, no. 1, pp. 2-16, 2010, doi: https://doi.org/10.1016/j.isprsjprs.2009.06.004.

[119] X. Huang, Q. Lu, and L. Zhang, "A multi-index learning approach for classification of high-resolution remotely sensed images over urban areas," *ISPRS Journal of Photogrammetry Remote Sensing,* vol. 90, pp. 36-48, 2014.

[120] C. J. C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," *Data Mining Knowledge Discovery,* vol. 2, no. 2, pp. 121-167, June 01 1998, doi: 10.1023/a:1009715923555.

[121] K. Huang, S. Li, X. Kang, and L. Fang, "Spectral–spatial hyperspectral image classification based on KNN," *Sensing and Imaging,* vol. 17, no. 1, p. 1, 2016.

[122] Y. Akbulut, A. Sengur, Y. Guo, and F. Smarandache, "NS-k-NN: Neutrosophic set-based k-nearest neighbors classifier," *Symmetry,* vol. 9, no. 9, p. 179, 2017.

[123] Y. Qian, W. Zhou, J. Yan, W. Li, and L. Han, "Comparing machine learning classifiers for object-based land cover classification using very high resolution imagery," *Remote Sensing,* vol. 7, no. 1, pp. 153-168, 2015.

[124] J. R. Otukei and T. Blaschke, "Land cover change assessment using decision trees, support vector machines and maximum likelihood classification algorithms," *International Journal of Applied Earth Observation Geoinformation,* vol. 12, pp. S27-S31, 2010.

[125] P. Mishra, D. Singh, and Y. Yamaguchi, "Land cover classification of PALSAR images by knowledge based decision tree classifier and supervised classifiers based on SAR observables," *Progress In Electromagnetics Research,* vol. 30, pp. 47-70, 2011.

[126] S. C. Vijayan and R. Jyothy, "Histogram Based Connected Component Analysis for Character Segmentation," *International Journal of Scientific Research Publications,* vol. 6, no. 6, pp. 200-202, 2016.

[127] P. Yadav and S. Agrawal, "ROAD NETWORK IDENTIFICATION AND EXTRACTION IN SATELLITE IMAGERY USING OTSU'S METHOD AND CONNECTED COMPONENT ANALYSIS," *The International Archives of the Photogrammetry, Remote Sensing Spatial Information Sciences, Dehradun, India*, 91-98, 2018.

[128] A. Abdollahi, B. Pradhan, G. Sharma, K. N. A. Maulud, and A. Alamri, "Improving Road Semantic Segmentation Using Generative Adversarial Network," *IEEE Access,* 2021.

[129] M. Aubry, S. Paris, S. W. Hasinoff, J. Kautz, and F. Durand, "Fast local laplacian filters: Theory and applications," *ACM Transactions on Graphics,* vol. 33, no. 5, p. 167, 2014.

[130] S. Paris, S. W. Hasinoff, and J. Kautz, "Local Laplacian filters: Edge-aware image processing with a Laplacian pyramid," *ACM Trans. Graph.,* vol. 30, no. 4, p. 68, 2011.
[131] A. Abdollahi, B. Pradhan, S. Gite, and A. Alamri, "Building Footprint Extraction from High Resolution Aerial Images Using Generative Adversarial Network (GAN) Architecture," *IEEE Access,* vol. 8, pp. 209517 - 209527, 2020, doi: https://doi.org/10.1109/ACCESS.2020.3038225.
[132] L. Ding, M. H. Bawany, A. E. Kuriyan, R. S. Ramchandran, C. C. Wykoff, and G. Sharma, "A Novel Deep Learning Pipeline for Retinal Vessel Detection In Fluorescein Angiography," *IEEE Transactions on Image Processing,* 2020.
[133] Y. Enokiya, Y. Iwamoto, Y.-W. Chen, and X.-H. Han, "Automatic Liver Segmentation Using U-Net with Wasserstein GANs," *Journal of Image Graphics,* vol. 6, no. 2, pp. 152-159, 2018.
[134] Q. Zhang, Z. Cui, X. Niu, S. Geng, and Y. Qiao, "Image segmentation with pyramid dilated convolution based on ResNet and U-Net," in *International Conference on Neural Information Processing*, 2017: Springer, pp. 364-372. Doi: https://doi.org/10.1007/978-3-319-70096-0_38.
[135] V. Mnih, *Machine learning for aerial image labeling, Ph.D. dissertation, Dept. Comput. Sci., Univ. Toronto, Toronto, ON, Canada*. Citeseer, 2013.
[136] A. Abdollahi, B. Pradhan, and A. Alamri, "VNet: An end-to-end fully convolutional neural network for road extraction from high-resolution remote sensing data," *IEEE Access,* vol. 8, pp. 179424 - 179436, 2020.
[137] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," *arXiv preprint arXiv:.* pp. 1-14, 2014.
[138] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European Conference on Computer Vision*, 2014: Springer, pp. 818-833, doi: 10.1007/978-3-319-10590-1_53.
[139] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1026-1034.
[140] Y. S. Aurelio, G. M. de Almeida, C. L. de Castro, and A. P. Braga, "Learning from imbalanced data sets with weighted cross-entropy function," *Neural Processing Letters,* vol. 50, no. 2, pp. 1937-1949, 2019.
[141] X. Li, X. Sun, Y. Meng, J. Liang, F. Wu, and J. Li, "Dice Loss for Data-imbalanced NLP Tasks," *arXiv preprint arXiv:.02855,* pp. 1-13, 2019.
[142] A. Abdollahi, B. Pradhan, N. Shukla, S. Chakraborty, and A. Alamri, "Multi-Object Segmentation in Complex Urban Scenes from High-Resolution Remote Sensing Data," *Remote Sensing,* vol. 13, no. 18, p. 3710, 2021.
[143] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4700-4708.
[144] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7132-7141.
[145] H. Song, W. Wang, S. Zhao, J. Shen, and K.-M. Lam, "Pyramid dilated deeper convlstm for video salient object detection," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 715-731.

[146] A. Van Opbroek, M. A. Ikram, M. W. Vernooij, and M. De Bruijne, "Transfer learning improves supervised image segmentation across imaging protocols," *IEEE Transactions on Medical Imaging,* vol. 34, no. 5, pp. 1018-1030, 2014.
[147] A. Abdollahi and B. Pradhan, "Integrating semantic edges and segmentation information for building extraction from aerial images using UNet," *Machine Learning with Applications,* vol. 6, p. 100194, 2021, doi: https://doi.org/10.1016/j.mlwa.2021.100194.
[148] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," pp. 448–456. Available from: https://arxiv.org/abs/1502.03167., 2015.
[149] M. Asadi-Aghbolaghi, R. Azad, M. Fathy, and S. Escalera, "Multi-level Context Gating of Embedded Collective Knowledge for Medical Image Segmentation," pp. 1-10, 2020. Available from: https://arxiv.org/abs/2003.05056.
[150] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Advances in Neural Information Processing Systems*, 2015, pp. 802-810.
[151] I. Demir *et al.*, "Deepglobe 2018: A challenge to parse the earth through satellite images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 172-181.
[152] M. Liang and X. Hu, "Recurrent convolutional neural network for object recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3367-3375.
[153] F. I. Diakogiannis, F. Waldner, P. Caccetta, C. J. I. J. o. P. Wu, and R. Sensing, "Resunet-a: a deep learning framework for semantic segmentation of remotely sensed data," vol. 162, pp. 94-114, 2020.
[154] S. Shit *et al.*, "clDice-a Novel Topology-Preserving Loss Function for Tubular Structure Segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 16560-16569.
[155] K. Palágyi, "A 3-subiteration 3D thinning algorithm for extracting medial surfaces," *Pattern Recognition Letters,* vol. 23, no. 6, pp. 663-675, 2002.
[156] F. Y. Shih and C. C. Pu, "A skeletonization algorithm by maxima tracking on Euclidean distance transform," *Pattern Recognition,* vol. 28, no. 3, pp. 331-341, 1995.
[157] M. Yang, K. Yu, C. Zhang, Z. Li, and K. Yang, "Denseaspp for semantic segmentation in street scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3684-3692.
[158] D. Eigen and R. Fergus, "Nonparametric image parsing using adaptive neighbor sets," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012: IEEE, pp. 2799-2806.
[159] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2980-2988.
[160] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A. L. J. I. t. o. p. a. Yuille, and m. intelligence, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," vol. 40, no. 4, pp. 834-848, 2017.

[161] O. R. Vincent and O. Folorunso, "A descriptive algorithm for sobel image edge detection," in *Proceedings of Informing Science & IT Education Conference (InSITE)*, 2009, vol. 40: Informing Science Institute California, pp. 97-107.
[162] N. Srivastava, G. Hinton, A. Krizhevsky, A. Krizhevsky, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research,* vol. 15, pp. 1929-1958, 2014.
[163] Y. Liu, J. Yao, X. Lu, M. Xia, X. Wang, and Y. Liu, "Roadnet: Learning to comprehensively analyze road networks in complex urban scenes from high-resolution remotely sensed images," *IEEE Transactions on Geoscience Remote Sensing,* vol. 57, no. 4, pp. 2043-2056, 2018.
[164] Y. Lin *et al.*, "Leveraging optical and SAR data with a UU-Net for large-scale road extraction," *International Journal of Applied Earth Observation and Geoinformation,* vol. 103, p. 102498, 2021.
[165] Z. Huang, J. Zhang, L. Wang, and F. Xu, "A feature fusion method for road line extraction from remote sensing image," in *Geoscience and Remote Sensing Symposium (IGARSS), 2012 IEEE International*, 2012: IEEE, pp. 52-55.
[166] Z. Miao, B. Wang, W. Shi, and H. Wu, "A method for accurate road centerline extraction from a classified image," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing,* vol. 7, no. 12, pp. 4762-4771, 2014.
[167] W. Shi, Z. Miao, Q. Wang, and H. Zhang, "Spectral-Spatial Classification and Shape Features for Urban Road Centerline Extraction," *IEEE Geosci. Remote Sensing Lett.,* vol. 11, no. 4, pp. 788-792, 2014.
[168] X. Lv, D. Ming, Y. Chen, and M. Wang, "Very high resolution remote sensing image classification with SEEDS-CNN and scale effect analysis for superpixel CNN classification," *International Journal of Remote Sensing,* vol. 40, no. 2, pp. 506-531, 2019.
[169] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 801-818.
[170] M. Zhou, H. Sui, S. Chen, J. Wang, and X. Chen, "BT-RoadNet: A boundary and topologically-aware neural network for road extraction from high-resolution remote sensing imagery," *ISPRS Journal of Photogrammetry and Remote Sensing,* vol. 168, pp. 288-306, 2020.
[171] A. Chaurasia and E. Culurciello, "Linknet: Exploiting encoder representations for efficient semantic segmentation," in *2017 IEEE Visual Communications and Image Processing (VCIP)*, 2017: IEEE, pp. 1-4.
[172] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2881-2890.