



**AN INNOVATIVE FRAMEWORK INTEGRATING OBJECT-BASED IMAGE
ANALYSIS AND DEEP LEARNING FOR URBAN LANDUSE/LANDCOVER
CLASSIFICATION USING WORLDVIEW 3 IMAGES**

By

OMER SAUD AZEEZ AL-GBURI

**Thesis Submitted to the School of Graduate Studies,
Universiti Putra Malaysia, in Fulfilment of the Requirements for the
Degree of Doctor of Philosophy**

March 2024

FK 2024 41

All material contained within the thesis, including without limitation text, logos, icons, photographs, and all other artwork, is copyright material of Universiti Putra Malaysia unless otherwise stated. Use may be made of any material contained within the thesis for non-commercial purposes from the copyright holder. Commercial use of material may only be made with the express, prior, written permission of Universiti Putra Malaysia.

Copyright © Universiti Putra Malaysia



Abstract of thesis presented to the Senate of Universiti Putra Malaysia in
fulfilment of the requirement for the degree of Doctor of Philosophy

**AN INNOVATIVE FRAMEWORK INTEGRATING OBJECT-BASED IMAGE
ANALYSIS AND DEEP LEARNING FOR URBAN LANDUSE/LANDCOVER
CLASSIFICATION USING WORLDVIEW 3 IMAGES**

By

OMER SAUD AZEEZ AL-GBURI

March 2024

Chairman : Associate Professor Helmi Zulhaidi bin Mohd Shafri, PhD
Faculty : Engineering

In recent years, deep learning-based image classification has become widespread, especially in remote sensing applications, due to its automatic and strong feature extraction capability. However, as deep learning methods operate on rectangular-shaped image patches, they cannot accurately extract objects' boundaries, especially in complex urban settings. As a result, combining deep learning and object-based image analysis (OBIA) has become a new avenue in remote sensing studies. Therefore, the purpose of this study is to assess the current (DL-OBIA) integration frameworks for remote sensing image classification and then innovate a new framework to optimize this integration in terms of accuracy and computational efficiency. On the other hand, the proposed framework is validated for land cover mapping and road extraction applications. The proposed approach combines convolutional neural networks (CNN) with OBIA based on joint optimization of segmentation parameters and deep feature extraction. A Bayesian technique

was used to find the best parameters for the multiresolution segmentation (MRS) algorithm while the CNN model learned the image features at different layers, achieving joint optimization. The proposed classification framework achieved the best accuracy, with 0.96 OA, 0.95 Kappa, and 0.96 mIoU in the training area and 0.97 OA, 0.96 Kappa, and 0.97 mIoU in the test area, outperforming several benchmark methods including Patch CNN, Center OCN, Random OCN, and Decision Fusion. The analysis of CNN variants within the proposed classification workflow showed that the HybridSN model achieved the best results compared to 2D and 3D CNNs. The 3D CNN layers and combining 3D and 2D CNN layers (HybridSN) yielded slightly better accuracies than the 2D CNN layers regarding geometric fidelity, object boundary extraction, and separation of adjacent objects. The Bayesian optimization could find comparable optimal MRS parameters for the training and test areas, with excellent quality measured by AFI (0.046, -0.037) and QR (0.945, 0.932). In the proposed framework, higher accuracies could be obtained with larger patch sizes (e.g., 9×9 compared to 3×3). Moreover, the proposed framework is computationally efficient, with the longest training being fewer than 25 s considering all the subprocesses and a single training epoch. The validation process shows the reliability of the proposed model for accurate classification from high-resolution satellite images. On the other hand, the framework's validation on road extraction achieved an overall accuracy of 0.8637.

The presented framework provides a new insight into OBIA-deep learning integration through the application of a Bayesian optimization technique, which

offers a mutual optimization between segmentation and classification processes. Furthermore, it facilitates the intelligent selection of segmentation parameters. This can have a positive impact on future studies focusing on OBIA-deep learning integration by enhancing classification accuracy, preserving precise object boundaries, and reducing the number of iterations. This makes it more suitable for studies that require precise ground objects and accurate classification, such as building extraction, road detection, cloud detection, tree classification, shoreline, and damage assessment, which typically use very high-resolution imagery sources.

Keywords: OBIA, deep learning, Bayesian, joint optimization

SDG: GOAL 9: Industry, Innovation and Infrastructure

Abstrak tesis yang dikemukakan kepada Senat Universiti Putra Malaysia
sebagai memenuhi keperluan untuk ijazah Doktor Falsafah

**RANGKA KERJA INOVATIF MENINGTEGRASIKAN OBJECT-BASED
IMAGE ANALYSIS DAN DEEP LEARNING UNTUK
PENGKLASIFIKASIAN GUNA TANAH/LITUPAN TANAH BANDAR
MENGUNAKAN IMEJ WORLDVIEW 3**

Oleh

OMER SAUD AZEEZ AL-GBURI

Mac 2024

Pengerusi : Profesor Madya Helmi Zulhaidi bin Mohd Shafri, PhD
Fakulti : Kejuruteraan

Dalam tahun-tahun kebelakangan ini, klasifikasi imej berasaskan deep learning telah meluas, terutamanya dalam aplikasi penderiaan jauh, disebabkan keupayaan pengekstrakan ciri automatik dan kuatnya. Walau bagaimanapun, kerana kaedah deep learning beroperasi pada tompok imej berbentuk segi empat tepat, kaedah tersebut tidak dapat mengekstrak sempadan objek dengan tepat, terutamanya dalam tetapan bandar yang kompleks. Hasilnya, menggabungkan deep learning dan analisis imej berasaskan objek (OBIA) telah menjadi jalan baharu dalam kajian penderiaan jauh. Oleh itu, tujuan kajian ini adalah untuk menilai rangka kerja integrasi semasa (DL-OBIA) untuk pengelasan imej penginderaan jauh dan kemudian mencipta rangka kerja baru untuk mengoptimumkan integrasi ini dari segi ketepatan dan kecekapan pengiraan. Sebaliknya, rangka kerja yang dicadangkan disahkan untuk pemetaan tutupan tanah dan aplikasi pengekstrakan jalan. Pendekatan yang dicadangkan adalah gabungan

rangkaian neural convolutional (CNN) dengan OBIA berdasarkan pengoptimuman bersama parameter segmentasi dan pengekstrakan ciri dalam. Teknik Bayesian digunakan untuk mencari parameter terbaik untuk algoritma segmentasi berbilang resolusi (MRS) manakala model CNN mempelajari ciri imej pada lapisan berbeza, mencapai pengoptimuman bersama. Model pengelasan yang dicadangkan mencapai ketepatan terbaik, dengan 0.96 OA, 0.95 Kappa, dan 0.96 mIoU dalam kawasan latihan dan 0.97 OA, 0.96 Kappa, dan 0.97 mIoU dalam kawasan ujian, mengatasi beberapa kaedah penanda aras termasuk Patch CNN, Center OCNN, Random OCNN dan Decision Fusion. Analisis varian CNN dalam aliran kerja klasifikasi yang dicadangkan menunjukkan bahawa model HybridSN mencapai hasil terbaik berbanding CNN 2D dan 3D. Lapisan CNN 3D dan gabungan lapisan CNN 3D dan 2D (HybridSN) menghasilkan ketepatan yang lebih baik sedikit daripada lapisan CNN 2D mengenai kesetiaan geometri, pengekstrakan sempadan objek dan pemisahan objek bersebelahan. Pengoptimuman Bayesian boleh menemui parameter MRS optimum yang setanding untuk kawasan latihan dan ujian, dengan kualiti cemerlang yang diukur oleh AFI (0.046, -0.037) dan QR (0.945, 0.932). Dalam model yang dicadangkan, ketepatan yang lebih tinggi boleh diperoleh dengan saiz tampalan yang lebih besar (cth., 9×9 berbanding 3×3). Selain itu, model yang dicadangkan adalah cekap dari segi pengiraan, dengan latihan paling lama adalah kurang daripada 25 s dengan mengambil kira semua subproses dan satu zaman Latihan. Proses pengesahan menunjukkan kebolehpercayaan model yang dicadangkan untuk pengelasan yang tepat daripada imej satelit beresolusi tinggi. Sebaliknya, pengesahan rangka kerja untuk pengekstrakan jalan mencapai ketepatan keseluruhan

sebanyak 0.8637. Kerangka yang dipersembahkan memberikan pandangan baru mengenai integrasi OBIA-deep learning melalui aplikasi teknik pengoptimuman Bayesian, yang menawarkan pengoptimuman bersama antara proses segmentasi dan klasifikasi. Selain itu, ia memudahkan pemilihan parameter segmentasi yang bijak. Ini boleh memberi impak positif kepada kajian masa depan yang memfokuskan pada integrasi OBIA-deep learning dengan meningkatkan ketepatan pengelasan, mengekalkan sempadan objek yang tepat, dan mengurangkan bilangan iterasi. Ini menjadikannya lebih sesuai untuk kajian yang memerlukan objek tanah yang tepat dan pengelasan yang tepat, seperti pengambilan bangunan, pengesanan jalan, pengesanan awan, pengelasan pokok, garis pantai, dan penilaian kerosakan, yang biasanya menggunakan sumber imej resolusi sangat tinggi.

Kata Kunci: OBIA, deep learning, Bayesian, pengoptimuman Bersama

SDG: MATLAMAT 9: Industri, Inovasi dan Infrastruktur

ACKNOWLEDGEMENTS

First and foremost, I am truly grateful to Allah the Almighty for all his help and endless blessings. I would like to express my deep sense of gratitude to **Prof. Helmi Zulhaidi Mohd Shafri**, my supervisor, for his insightful comments and suggestions. His sharp critical ideas were an immense help to me in completing this study. Without his encouragement and guidance, this thesis would merely turn into a heap of dreams. I would also like to thank my co-supervisors, **Associate Professor Dr. Nuzul Azam Bin Haron** and **Associate Professor Dr. Aidi Hizami Bin Alias**, for their valuable guidance and assistance throughout my PhD research.

Most of all, I would like to express my deep sense of gratitude to my family for their endless support and encouragement, whom I shall be indebted to forever. May almighty Allah bless them.

OMER SAUD AZEEZ AL-GBURI

This thesis was submitted to the Senate of the Universiti Putra Malaysia and has been accepted as fulfilment of the requirement for the degree of Doctor of Philosophy. The members of the Supervisory Committee were as follows:

Helmi Zulhaidi bin Mohd Shafri, PhD

Associate Professor
Faculty of Engineering
Universiti Putra Malaysia
(Chairman)

Aidi Hizami bin Alias, PhD

Associate Professor
Faculty of Engineering
Universiti Putra Malaysia
(Member)

Nuzul Azam bin Haron, PhD

Associate Professor
Faculty of Engineering
Universiti Putra Malaysia
(Member)

ZALILAH MOHD SHARIFF, PhD

Professor and Dean
School of Graduate Studies
Universiti Putra Malaysia

Date: 7 November 2024

TABLE OF CONTENTS

	Page
ABSTRACT	i
ABSTRAK	iv
ACKNOWLEDGEMENTS	vii
APPROVAL	viii
DECLARATION	x
LIST OF TABLES	xiv
LIST OF FIGURES	xv
LIST OF ABBREVIATIONS	xvii
CHAPTER	
1 INTRODUCTION	1
1.1 Background of Study	1
1.2 Problem Statement	3
1.3 Research Objectives	5
1.4 Research Novelty and Contributions	6
1.5 Scope and Limitations of the Study	7
1.6 Thesis Organization	7
2 LITERATURE REVIEW	9
2.1 Introduction	9
2.2 Remote Sensing Image Classification	10
2.3 Remote Sensing LULC Mapping	14
2.3.1 LULC Definition and Scheme	14
2.3.2 Very High-Resolution Satellite Images for LULC Extraction	17
2.4 Classification Models	20
2.4.1 Deep Learning	20
2.4.2 OBIA	30
2.5 Integration	41
2.5.1 Deep Learning Based on Object-Level Features	44
2.5.2 Feature-Level Fusion	52
2.5.3 Decision-Level Fusion	55
2.5.4 Patch Filtering (Deep Learning with Filtered Patches)	61
2.5.5 Patch Filtering (Deep Learning with Context Patches)	63
2.6 Discussions and Research Gaps	73
3 METHODOLOGY	80
3.1 Study Area	80
3.2 Dataset and Tools	81
3.3 Data Pre-Processing and Preparing	85
3.4 Assessment of OBIA-Deep Learning Integration	

Frameworks	86
3.4.1 OBIA	86
3.4.2 CNN and Deep Neural Network	89
3.4.3 Evaluated Integration Frameworks	92
3.5 The Innovative OCNN Framework	95
3.5.1 The Proposed CNN	95
3.5.2 The OBIA-CNN with Joint Optimization (OCNN-JO)	97
3.6 Accuracy Assessment	107
3.6.1 Classification Accuracy Assessment	107
3.6.2 Segmentation Quality Assessment	108
3.7 Framework's Validation on the Road Extraction	109
4 RESULTS AND DISCUSSION	111
4.1 Results of Assessing OBIA and Deep Learning Integration Frameworks	111
4.1.1 Assessment by Overall Accuracy	111
4.1.2 Assessment by Per-Class Accuracies	113
4.1.3 Results of Image Classification Maps	114
4.2 Results of the Innovative Classification Framework	118
4.2.1 Segmentation Results	118
4.2.2 The Results of the OCNN-JO Framework	120
4.2.3 Performance Comparison with Benchmark Models	126
4.2.4 Sensitivity Analysis	132
4.2.5 Computational Efficiency	135
4.2.6 Results of the Validation on Land Use/Land Cover Classification	137
4.3 Results of Validation on Road Extraction	141
4.4 Discussions	143
5 CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORKS	150
5.1 Conclusion	150
5.2 Recommendations for Future Work	154
REFERENCES	155
APPENDICES	174
BIODATA OF STUDENT	184
LIST OF PUBLICATIONS	185

LIST OF TABLES

Table	Page
2.1 LULC Schemes	15
2.2 Multispectral VHR Satellites	19
2.3 The Summary of Common Segmentation Methods	39
2.4 The Difference between Deep Learning and OBIA Classification	41
2.5 The Summary of Literature Review	66
2.6 The Summary of OBIA-DL Integration Methods	79
3.1 Training and Test Data Class Distribution (i.e., Number of Pixels and Area Percentage) in the Ground-Truth Dataset	83
3.2 Parameters Used to Train the Base Models	90
3.3 Hyperparameters Used in the Base Models	91
4.1 Accuracy Assessment of the Classification Methods Used in this Research for the Training and Test Areas	112
4.2 Per-Class Accuracies Obtained for Each Classification Method Using the Training Area Dataset	114
4.3 Per-Class Accuracies Obtained for Each Classification Method Using the Test Area Dataset	114
4.4 Search Space, Initial Values, and Best Values of MRS Segmentation Parameters for the Training and Test Areas	120
4.5 Segmentation Quality Metrics for the Training and test Area Datasets	120
4.6 Classification Assessment of Different CNN Methods Used with the Proposed Model for the Training and Test Areas	123
4.7 Per-Class Accuracies of the Proposed and Benchmark Classification Methods Based on Samples from the Training Area	129
4.8 The Computational Efficiency of the Proposed and Benchmark Classification Models Including Their Subprocesses	137
4.9 The Result of Image Registration of Training Area	138
4.10 The Result of Image Registration of Testing Area	138

LIST OF FIGURES

Figure	Page
2.1 ML and DL Techniques	22
2.2 The Difference between Pixel-Based and Object-Based Classification Methods (A) Illustrates the Pixel Based and (C) Illustrates OBIA (B) the Original RS Image	31
2.3 An Example of the Integration between OBIA and CNN for Satellite Image Classification	44
3.1 The Overall Methodology	82
3.2 The Study Area	83
3.3 The training and test areas are indicated in the Worldview-3 true color composite of the study area	84
3.4 The Ground-Truth Maps of the (a) Training and (b) Test Areas	85
3.5 CNN Architecture Used in integrated OBIA-Deep Learning Frameworks	91
3.6 ANN Architecture Used in Integrated OBIA-Deep Learning Frameworks	92
3.7 Typical Architectures of the OBIA-CNN Integration Frameworks Identified in This Research	94
3.8 An Illustration of the CNN Architecture Used in This Study	97
3.9 Workflow of the Bayesian Optimization Procedure to Jointly Optimize Segmentation and Feature Learning for Image Classification	105
3.10 The Selection of Optimal Segmentation Parameters Using Bayesian Optimization	106
3.11 Workflow of Road Extraction Based on the Innovative Framework (OCNN-JO)	110
4.1 Classification Maps Obtained for the Training Area with the Pixel ANN, Patch CNN and the Four OBIA-CNN Integration Frameworks Assessed in the Current Research	116
4.2 Classification Maps Obtained for the Test Area with the Pixel ANN, Patch CNN and the Four OBIA-CNN Integration Frameworks Assessed in the Current Research	117

4.3	The Results of MRS Segmentation with Optimal Parameters Were Found by the Proposed Bayesian Optimization Method for the (Left) Training and (Right) Test Areas	119
4.4	Classification Maps of Different CNN Methods for the Training Area: (a) 2D CNN, (b) 3D CNN, and (c) HybridSN	122
4.5	Classification Maps of Different CNN Methods for the Test Area	123
4.6	The Learning Curve of the Proposed Model with HybridSN Backbone Model: (a) Training Dataset, (b) Test Dataset	124
4.7	Convergence Plots Showing the Progress of Bayesian Optimization of MRS Parameters with 2D CNN, 3D CNN, and HybridSN Models	125
4.8	The Overall Accuracy of the Proposed and Benchmark Classification Methods Based on Samples from the Training Area	128
4.9	The Overall Accuracy of the Proposed and Benchmark Classification Methods Based on Samples from the Testing Area	128
4.10	Classification Maps of Different Methods Used In this Research for the Training Area	130
4.11	Classification Maps of Different Methods Used in this Research for the Test Area	131
4.12	Sensitivity Analysis on MRS Segmentation Parameters	134
4.13	Accuracy Assessment Based on mIoU of the Proposed Classification under Different CNN Models Using Patch Sizes from 3×3 to 9×9	135
4.14	The Validation Results for the Training Area	139
4.15	The Validation Results for the Testing Area	140
4.16	Results of Road Extraction Based on the Proposed Model, (a) the Satellite Image (RGB True Color Composite), (b) Ground Truth Map, (c) Segmentation Map, and (d) Road Extraction Map of the Study Area	142
4.17	Accuracy Assessment of the Proposed Framework for Road Extraction	142
4.18	The Improved Land-Cover Extraction Depending on the Proposed OCNN Method Compared to Traditional Methods	149

LIST OF ABBREVIATIONS

ADAM	adaptive moment estimation
AEs	Autoencoders
AFI	area of fit
AHP	Analytic Hierarchy Process
AI	Artificial Intelligence systems
ANNs	Artificial Neural Networks
BNN	Bayesian convolutional neural network
CDF	cumulative distribution function
CHM	Canopy Height Model
CNNs	Convolutional Neural Networks
CRF	Conditional Random Field
DBN	Deep Belief Networks
DEM	Digital Elevation Model
DL	Deep Learning
DNNs	Deep Neural Networks
DOOG	Difference Of Offset Gaussian
DT	Decision Trees
ENVI	Exelis Visual Information Solutions
FCNNs	Fully Connected Neural Network
FN	false negative
FP	false positive
GC	Graph-cut segmentation
GCPs	Ground control points
GEE	Google Earth Engine
GF	Gaofen

GIS	Geographic Information System
GPs	Gaussian processes
GPU	Graphics processing unit
GSD	Ground Sampling Distance
HBC-SEG	hard-boundary constrained segmentation
HD	High-Definition
I	Hyperspectral Image
HRRSI	High-resolutions RS image
IDE	integrated development environment
IOCNN	Improved Object-based CNN
ISODATA	Iterative Self-Organizing Data Analysis
JDL	Joint Deep Learning
K	Kappa coefficient
KNN	k-nearest neighbors
LiDAR	Light Detection And Ranging
LSTM	Long short-term memory
LULC	Land Use & Land Cover
MCPS	maximum classification probability scale
mIoU	mean intersection over union
MLH	Maximum Likelihood
mIoU	mean intersection over union
ML	Machine Learning
MODIS	Moderate Resolution Imaging Spectroradiometer
MRS	Multiresolution Segmentation
Nadam	Nesterov-accelerated Adaptive Moment Estimation
NDVI	Normalized difference vegetation index

NIR	Near Infrared Band
NOAA	National Oceanic and Atmospheric Administration
NODE	Neural Oblivious Decision Ensemble
OA	Overall Accuracy
OBIA	Object Based Image Analysis
OCNN	Object-Based Convolutional Neural Network
OCNN-JO	OBIA and CNN and Joint optimization
OHSF	Object-Based Heterogeneous Filter
OSA	Object-Scale Adaptive
OSVM	Object based support Vector Machine
ODUN	Object-based U-net-DenseNet-coupled network
PCA	Principal Component Analysis
PDF	probability density function
PM2.5	Fine particulate matter
QR	quality rate
QUAC	quick atmospheric correction module
RADAR	Radio Detection and Ranging
RBM	Restricted Boltzmann Machines
ReLU	Rectified Linear Unit
ResNet	Residual Network
RF	Random Forest
RMV	Region-based Majority Voting
RNNs	Recurrent Neural Networks
RS	Remote Sensing
SAE	Stacked Autoencoders
SAR	Synthetic aperture radar

SGD	stochastic gradient descent
SLIC	Simple Linear Iterative Clustering
SOD-CNN	Standardized object-based dual CNN
SS-JDL	Scale Sequence Joint Deep Learning
SS-OCNN	Scale Sequence Object-based CNN
SVM	Support Vector Machine
TASSEL	attentive weakly Supervised Satellite image time series classifier
TP	true positive
3DCNN	Three Dimensional Convolutional Neural Networks
2DCNN	Two Dimensional Convolutional Neural Networks
UAV	Unmanned Aerial vehicle
VHR	Very High Resolution
WV-3	Worldview-3

CHAPTER 1

INTRODUCTION

This chapter presents the context and background of the study, including an overview of the problem statement, the research aim, and objectives. Additionally, it provides a description of the study area, highlighting its geographical settings. It also explains the contributions of the research, along with an outline of the organization of the thesis.

1.1 Background of Study

Image classification is one of the basic tasks in remote sensing. In this operation, image pixels or image objects are clustered and labelled via automatic-learned or hand-crafted features. In general, several methods are employed for remote sensing image classification. In the last decade, deep learning techniques have attracted experts in the field of remote sensing due to their sophisticated architecture, including several as Convolutional Neural Networks (CNNs). Deep learning methods, with their hierarchical architecture, learn deep and abstract image features, which are useful for the classification task. CNNs have shown great results in computer vision, including pattern recognition. CNNs have revealed high efficiency for learning spatial, contextual, and textural information from remotely sensed images. However, the patch-wise CNN (Patch CNN) obtains artefacts on the boundary of the classified patches and usually results in blurred boundaries among ground surface objects (Zhang et al., 2018a, 2018b). CNNs have excellent capabilities in extracting useful abstract features from remote sensing images; however,

they are not directly applicable for performing classification at the object level. CNNs require image patches as input data to make inferences on a pixel's category label. This data structure is prone to errors when classifying the boundaries of ground objects (Pan et al., 2019). Furthermore, the rectangular patch is scarcely consistent with the non-homogenous segment obtained from an object-based image analysis (OBIA) process. Therefore, when applying CNN classifier to a segment at the boundary of a specified land cover, a patch CNN that includes the segment that is to be classified may also contain other segments belonging to a different type of land cover. These segments that belonged to other land cover categories may exhibit higher band values or notable patterns and textures. Consequently, they will significantly influence the decision of CNNs. As a result of this inconsistency, classification errors will occur when we directly train CNNs on segments resulting from the OBIA method. This kind of error is called jagged errors, which will be seen between land cover borders as a shrink or over-expand forms in the resulting image. This issue has severely restricted the application of CNNs in OBIA classification. Generally, patch CNN or pixel-level CNNs (Pixel CNN) methods are not effectively designed respecting computational efficiency or precision to handle the complicated challenges of the classification of remotely sensed images (Zhang et al., 2018c). Unlike using deep learning on image pixels or patches, its use with OBIA is relatively more complex because of the data transformation requirements to specific data structures that deep learning models can be trained on.

1.2 Problem Statement

The combined OBIA and deep learning frameworks demonstrated significant success in remote sensing due to their ability to extract deep features as well as allow the image objects to be defined spatially in the real world (Blaschke 2010). However, deep learning models such as convolutional neural networks (CNNs) are not directly applicable for performing classification at the object level. They require image patches as input data to make inferences on a pixel's category label. This data structure is prone to errors when classifying the boundaries of ground objects (Pan et al., 2019). Furthermore, the non-homogenous segment from the OBIA technique rarely matches the rectangular patch.

A specific type of error known as jagged errors occurs when applying a deep learning classifier (e.g., CNN) to a segment at the boundary of a specified land cover; other segments belonging to a different type of land cover may also appear in an image patch, and these segments may exhibit higher band values or notable patterns and textures. This inconsistency, which clearly rises between land cover borders as a shrink or over-expand form in the resulting image, significantly impacts CNN's decision-making process. This problem has severely restricted the use of CNNs in OBIA classification. As a result, several methods have emerged to enhance the integration between OBIA and deep learning methods (Giles et al., 2023; Ubben et al., 2024; Lu et al., 2024).

There are five main frameworks in the literature that combine deep learning and OBIA: (1) deep learning based on object-level features, (2) feature-level

fusion, (3) decision-level fusion, and (4) deep learning with context patches (5) deep learning with filtered patches. While there are many research papers on the integration of deep learning and OBIA for remote sensing image classification, there is a significant lack of understanding of how these integration frameworks compare and what is the best way to achieve such integration for high resolution satellite image classification. Furthermore, the majority of these methods implement optimization processes separately, one for OBIA and another deep learning. This often results in a knowledge gap between the segmentation and feature learning processes; consequently, there is a lack of joint optimization between OBIA and CNN, which is necessary to achieve the best performance in the classification of remotely sensed data. On the other hand, despite the robust performance of combining OBIA and DL methods, there is a noticeable lack of implementation in transportation studies (e.g., road extraction from remotely sensed data).

Therefore, this study presents an innovative framework that integrates OBIA and deep learning with methods regarding integration strategies and joint optimization to address the limitations mentioned above. The proposed methodology provides more accurate classification of high resolution satellite images which contributes to applications of urban studies, and other related tasks that require land use and land cover as a base information.

1.3 Research Objectives

The main objective of this study is to develop an innovative object-based CNN (OCNN) framework for extracting land use and land cover from high spatial resolution satellite imagery using a joint Bayesian optimization approach for learning the best segmentation parameters and classifying the image data employing deep learning methods.

The following sub-objectives are addressed to achieve the main objective:

1. **Assess several benchmarks of OBIA-deep learning integration frameworks for land cover classification using Worldview-3 satellite imagery data.** To the best of our knowledge, this is the first study that compares the common integration frameworks and provides an assessment of each framework by using a high-resolution satellite image dataset (worldview3) that has training and test areas, which could be a guideline for researchers in the field of study. A custom-made computational framework is developed in Python programming to combine the two main image processing frameworks i.e., (1) OBIA and (2) deep learning, which was necessary to avoid using multiple software that is considered complex and time-consuming.
2. **Construct an integrated OBIA-deep learning framework (OCNN) by employing a joint Bayesian optimization to find the best segmentation parameters and updating the CNN's network weights by transfer learning.** The proposed framework employs the joint Bayesian optimization to find the best segmentation parameters and updating the CNN's network weights by transfer learning. This framework applied the concept of the mutual optimization to find the optimal integration, to reduce the training time by minimizing the iteration process, and improving the handling of knowledge gap between the two models.

3. **Validate the proposed framework for land cover mapping and road extraction applications.** The proposed study is validated based the most common applications in remote sensing, land use and land cover mapping and road extraction.

1.4 Research Novelty and Contributions

This thesis makes significant contributions to the field of satellite image classification. It provides a comprehensive overview of the different ways to integrate deep learning with OBIA and innovates a framework that achieves state-of-the-art results. The research suggests that the combination of deep learning and OBIA is a promising approach for improving the accuracy of satellite image classification.

This research makes the following contributions to the field of remote sensing image classification based on OBIA and deep learning integration models:

1. Assesses four different integration frameworks for combining deep learning and OBIA for satellite image classification.
2. Develops an innovative framework a novel approach for combining deep learning and OBIA for satellite image classification.
3. Uses a Bayesian optimization technique to find the best parameters for the Multiresolution Segmentation (MRS) algorithm.
4. Uses a CNN and transfer learning to learn the image features at different layers.

1.5 Scope and Limitations of the Study

The main goal of this study is to enhance the classification of worldview3 imagery by improving the performance of the OBIA-DL integration framework. the improvement of OBIA-DL integration was performed by using Bayesian optimization and transfer learning techniques. In this study, urban LULC and road extraction were considered. The scope of this study was limited to the spatial coverage of high-resolution remote sensing data. Therefore, the transferability of the proposed method was not performed for medium- and low-resolution imagery. On the other hand, this study did not cover the detection of railways or parking lots that are classified as roads. However, with additional compatible modifications the proposed framework can be incorporated for these more specific applications.

1.6 Thesis Organization

This thesis is split into five chapters as follows:

1. The first chapter (**Introduction**) introduces the research subject, the problem statement, the research aim, and objectives and briefly describes the research novelty/contributions.
2. The second chapter (**Literature Review**) provides an overview and discussions on previous studies on remote sensing image classification using integrated OBIA-deep learning methods.
3. The third chapter (**Methods and Materials**) describes the research data and materials. It also presents the methodology used for image segmentation, deep learning model, and integration frameworks for remote sensing image classification using OBIA and deep learning. In addition, accuracy assessment and validation methods were also

presented in this chapter.

4. The fourth chapter (**Results and Discussion**) presents the image classification results based on several integrated OBIA and deep learning models, including the proposed novel models; also, it provides detailed discussions on several experiments conducted in this research.
5. The last chapter (**Conclusion and Recommendations**) summarizes the major findings of this study and offers recommendations for future works after explaining the limitations of the current work.

CHAPTER 2

LITERATURE REVIEW

This chapter reviews and discusses previous studies related to integrating deep learning and OBIA for remote sensing image classification and assessment.

2.1 Introduction

In remote sensing imagery, urban land-use/land-cover (LULC) information is illustrated as high-level semantic functions or geospatial patterns (Zhu et al., 2022; Karimi et al., 2024). As a result, urban LULC extraction from remotely sensed sources requires methods with efficient spectral, spatial, and contextual feature learning capabilities. In recent years, deep learning methods such as CNNs have shown outstanding success in feature extraction from remotely sensed images. However, there is a problem with these methods for the accurate extraction of object boundary information due to the use of rectangularly shaped image patches (Zhang et al., 2018c).

OBIA is another widely used image classification technique that extracts features from remotely sensed data and performs LULC classification at the object level. In addition, several new research works combined the two methods into a single approach, commonly known as OCNN. The major advantage of such an integrated approach is that it effectively learns image features while accurately characterizing the boundaries of objects through image segmentation (Pan et al., 2019). There are several ways to integrate

OBIA and deep learning methods for LULC classification, which will be discussed in previous works. They can be categorized into five categories: deep learning based on object-level features, feature-level fusion, decision-level fusion, deep learning with context patches, and deep learning with filtered patches. This chapter explains the methodologies and principles of remote sensing image classification based on integrated OBIA-deep learning techniques.

2.2 Remote Sensing Image Classification

Pixels are the basic units of remote sensing images that have been adopted since the 1980s for related applications (Li et al., 2014). In remote sensing, image classification is the operation of clustering and classifying the image pixels (pixel-based classification) or objects based on spectral, spatial, geometric, and textural information (OBIA classification). Since the early developments of remote sensing, image classification has attracted scientists due to its outputs, which have become the basis for several socioeconomic and environmental applications (Khatami et al., 2016). Several techniques have been developed for remote sensing image classification based on pixel units or image objects. In pixel-based methods, remote sensing image pixels are clustered into several groups that refer to specific land use, land cover, or other categorical types (e.g., material type) (Wilkinson 2005).

The common methods of pixel-based classification include:

- Unsupervised classification (i.e., ISODATA and K-means algorithms).
- Supervised classification (i.e., Maximum Likelihood, Neural Network,

Support Vector Machine, Random Forest algorithms).

- Hybrid techniques (i.e., integration of supervised and unsupervised classification).

Generally, pixel-based classification has drawbacks for these techniques, as the image pixel may include more than a single LULC type (Li et al., 2014). As a result, with the rapid development of remote sensing sensors (e.g., unmanned aerial vehicles and very high-resolution satellite images), experts have developed spectral mixture analysis and fuzzy classification methods to address the mixing pixel problem (Wang 1990). Pixel-level methods are solely based on spectral information. They can classify land cover, but their use in urban settings is limited. These methods induce uncertainty and “salt-and-pepper” effects in classification results, especially in high-spectral heterogeneity regions (Shendryk et al., 2019). The problem with this method is that it requires the pre-definition of arbitrary image structures, whereas actual objects in the real world may be irregularly shaped.

As a result, object-level methods are preferable to moving kernel windows because they allow image objects to be defined spatially in the real world (Blaschke 2010). Objects are created by segmenting image pixels based on spectral, spatial, and contextual information. Object-based methods make use of both within-object (spectral, texture, and shape) and between-object (connectivity, contiguity, and distances) information.

Considering the two available classification frameworks, i.e., pixel-based or object-based methods, remote sensing image classification is considered a complex process that requires the consideration of several factors as follows (Lu and Weng 2007):

Selection of suitable remote sensing data by considering several factors like a user's need, remote sensing data availability, the required characteristics of data such as spatial, spectral, radiometric, and temporal resolutions, the scale of the study area, the analyst's experience in using the selected image, and the atmospheric condition. Time and cost constraints (Lefsky and Cohen 2003; Quattrochi and Goodchild 1997).

After selecting suitable datasets and accounting for the factors listed above, the second step includes selecting a suitable classification technique and sufficient training samples. The selection of suitable classification techniques usually relies on the data used, compatibility with previous studies, available classification algorithms, and time constraints (Landgrebe 2005). The selection of optimal training samples, on the other hand, is usually dependent on the availability of reference data, the spatial resolution of the data used, and the complexity of the landscapes in the study area (Chen and Stow 2002).

Then, the selected images need to be pre-processed and prepared for analysis. This step usually includes the detection and restoration of bad lines, geometric correction or image registration, radiometric calibration, atmospheric correction, and data conversion among different sources or formats (Zitova and Flusser 2003; Coll et al. 2009). The fourth step is the

classification process, feature selection, and extraction based on the selected classification technique (Bhandari et al., 2012; Lin et al., 2011). Finally, post-classification processing is applied by filtering and adding ancillary data to reduce 'salt and pepper' and spectral confusion effects in pixel-based methods (Dharmarajan and Kannan 2010). The final step also includes the evaluation of the classification process based on the qualitative technique through expert knowledge or quantitative, depending on sampling techniques. In general, six criteria have been proposed to evaluate classification performance; reproducibility, accuracy, robustness, ability to fully use the information content of the data, uniform applicability, and objectiveness.

Nowadays, with the rapid development of computation technologies, a new framework has been developed based on artificial neural networks (ANNs) and other advanced deep learning models. In general, a deep learning framework includes a hierarchical architecture based on many layers, each of which constitutes a non-linear information processing unit. Deep Neural Networks (DNNs) employ deep architectures in a neural network, which can represent functions with higher complexity if the number of layers and units in a single layer is increased. Given enough labeled training datasets and suitable models, deep learning approaches can help humans establish mapping functions for operation convenience (Ma et al., 2019).

Examples of deep learning architectures include CNNs, recurrent neural networks (RNNs), autoencoders (AEs), restricted Boltzmann machines (RBM), deep belief networks (DBN), and generative adversarial networks

(GAN). Deep learning models can be used with both pixel-based and OBIA frameworks. However, their application with OBIA is relatively more complex due to the need for data transformation into specific structures (Pan et al., 2019).

Hence, researchers have been developing approaches and tools specific for integrating deep learning models with OBIA, including methods depending on (1) deep learning based on object-level features, (2) feature-level fusion, (3) decision-level fusion, (4) deep learning with context patches, and (5) deep learning with filtered patches.

2.3 Remote Sensing LULC Mapping

2.3.1 LULC Definition and Scheme

Land Cover's notion refers to any type of surface covering the ground, such as waterbodies, vegetation, built-up areas, and bare soil. Land Use's notion indicates the purpose that land serves, for example, agriculture, residential, and recreation. When used both within the expression, LULC refers to any kind of human activities and the distribution of natural elements on the land at a specified time frame that is classified based on scientific methods. LULC information has become widely used in many fields, such as disaster management, agriculture, land development, and environmental applications (Anderson 1976). Table 2.1 shows LULC schemes.

Table 2.1 : LULC Schemes

LULC type	Description
Urban or Built-up land	Residential, commercial and services, industrial, socio-economic infrastructure, mixed urban and other urban, transportation, roads, and airports.
Agricultural land	All cultivated and uncultivated agricultural land areas include farmlands, crop fields including fallow lands/plots, and Horticultural lands.
Rangeland	Herbaceous rangelands, Shrub-Brushland Rangeland, Mixed Rangeland
Forest Land	Protected forests, plantations, deciduous forests, mixed forest lands, and forests on customary land.
Water	Rivers, permanent open water, lakes, ponds, reservoirs
Wetland	Permanent and seasonal grasslands along the lake, river, streams, marshy land, and swamps
Barren land	Areas around and within forest-protected regions with no or very little vegetation cover, including exposed soils, stock quarry, rocks, landfill sites, and active excavation areas.
Tundra	Shrub, Brush tundra, bare ground tundra, wet tundra, and mixed tundra
Perennial Snow or Ice	Perennial snowfields and Glaciers

(Source: Jensen 1996)

LULC information can be used in different applications, such as disaster management, urban planning, change detection, and environmental contamination. Numerous studies have been conducted to extract LULC from various remote sensing sources. Many researchers took advantage of open-source satellite imagery (e.g., Landsat 8 images), such as Atef et al. (2023), who proposed a framework based on GEE by integrating SVM, RF, and MLH models with GIS techniques for the monitoring of LULC changes in El-Fayoum province, Egypt. Their results showed that the SVM model outperformed the RF and MLH models. With the advent of Sentinel-2 sensors that provide a spatial accuracy finer than Landsat imagery, Arrechea-Castillo et al. (2023) presented an approach that aims to reduce the computational complexity by using the LeNet-CNN model for Sentinel-2 imagery classification in Colombia. Their method achieved an accuracy higher than traditional machine learning

algorithms and lower computational resources than other CNN architectures, with the emergence of sophisticated sensors (e.g., very high-resolution satellites). Experts have developed several methods to improve the quality of LULC extraction. For instance, Pal et al. (2023) analyzed urban LULC changes using a series of worldview3 images and a hybrid approach containing a multi-class SVM and ISODATA segmentation to automatically detect building changes in Barasat City, West Bengal, India.

On the other hand, LULC information is necessary for risk assessment, as stated in flood, earthquake, and tsunami studies. Waghwal and Agnihotri (2019) used LULC data to conduct a flood risk assessment in Surat City, India; their method combined flood susceptibility maps, LULC data, and social information to produce risk maps that showed an increase in flood losses in built-up areas compared to agricultural lands. In another study, Jena and Pradhan (2020) used LULC as a primary parameter to develop an earthquake risk analysis. Their analysis used ANN and AHP methods to generate earthquake risk maps. According to Superczynski and Christopher (2011), LULC data are often used as important parameters in environmental studies. They looked at how PM_{2.5} concentrations changed with LULC changes in Birmingham, UK, using GIS tools, ground observations, and LULC from Landsat images to find the link between LULC changes and PM_{2.5} concentrations. They found a strong link between PM_{2.5} concentrations and built-up areas.

2.3.2 Very High-Resolution Satellite Images for LULC Extraction

Satellite sensors are mainly classified into three groups in terms of spatial resolution: low-resolution sensors with a resolution over 30 m per pixel, such as NOAA AVHRR and MODIS; medium-resolution sensors with a spatial resolution between 5 and 30 m per pixel, like the Landsat satellite series; and high-resolution sensors with a resolution ranging from 1 to 5 m per pixel, such as the Rapid Eye Satellite system (Franklin 2001). With the rapid advancements in satellite sensors, a new kind of sensors called a very high-resolution sensor (VHR) has emerged (Table 2.2 shows the recent VHR satellites and their specifications).

VHR satellites provide panchromatic and multispectral imagery with finer spatial resolution than in prior decades. IKONOS sensors are considered the first commercial VHR sensors launched in 1999 with 0.8 m spatial resolution, followed by satellites such as Pleiades and Quickbird that broke the 0.8 mark. In 2009, a revolutionary development occurred in satellite sensors. Worldview 2 satellites have been launched with a spatial resolution of 0.4 m and eight spectral bands ranging between visible and near infrared. In this context, recent VHR imagery and advanced image processing algorithms demonstrate the feasibility of image analysis in terms of objectivity, cost, and mapping fidelity (Eugenio González and Marcello Ruiz 2019).

VHR satellite images are distinguished by covering vast areas with detailed information (e.g., spatial, contextual, and spectral information). As a result, numerous studies have been conducted using this data. LULC mapping is

considered a dominant application. For example, Mohammadi and Sharifi (2021) compared deep learning and machine learning techniques concerning VHR image classification. They employed a Pleiades image with a spatial resolution of 50 cm. Their result showed that the deep learning method outperformed machine learning methods. Over the last decade, VHR images have gained widespread use in agricultural applications.

To extract tree crowns, Lassalle et al. (2022) used worldview3 and worldview4 images with the combined deep learning and OBIA paradigms. The experimental results revealed that the proposed method outperformed benchmark approaches regarding precision. VHR images are also employed for damage assessment during disasters and conflicts. Berezina and Liu (2022) performed a building damage assessment caused by Hurricane Michael in 2018 in Panama. Their approach included several steps. The first step is the building extraction using VHR images, including GeoEye-1 and Worldview2, and damages are classified into four categories (undamaged, lightly damaged, severely damaged, and destroyed buildings). Their approach has outperformed traditional machine learning and achieved 86.3%. On the other hand, VHR images became critical for humanitarian studies, such as refugee monitoring during conflicts or disasters. Ghorbanzadeh et al. (2022a) employed worldview3 images and the CNN method for extracting refugee dwellings; their approach has improved the mapping of refugee dwellings with an accuracy range of 86% to 99%.

Nowadays, some agencies (e.g., MAXAR Technologies) announce new images with a spatial resolution of 15 cm obtained by post-processing images with a 30 cm ground sampling distance (GSD). Therefore, experts used these data to improve the image classification process. Yalcin et al. (2022) used support vector machines and the random forest method to classify satellite images with a spatial resolution of 15 cm, considering two areas in Serbia. The final findings indicated that using images can provide objects with sharper edges, which leads to improved LULC extraction. Consequently, the significant evolution of satellite sensors and image processing techniques has greatly influenced the classification of RS images. Therefore, we anticipate an increasing reliance on VHR images in the foreseeable future.

Table 2.2 : Multispectral VHR Satellites

Satellite	Spatial Resolution	Number of bands	Temporal Resolution (Days)	Year of Launch	
IKONOS	0.82 m	4	3-5	1999	*
Resurs-DK1	0.8 m	3	6	2006	*
GaoFen-2	0.81 m	4	5	2014	
GaoFen-7	0.8 m	4	5	2019	
TripleSat	0.8 m	4	1	2015	
Pleiades 1A/B	0.7 m	4	4	2012	
Satelloic	0.7 m	4	7	2023	
KOMPSAT-3	0.7 m	4	3	2012	*
Quickbird	0.61 m	4	1-3	2001	*
KOMPSAT-3A	0.55 m	5	3-4	2015	*
Worldview-1	0.46 m	1	5-4	2007	
Worldview-2	0.46 m	8	3-4	2009	
GeoEye-1	0.46 m	4	2-3	2008	
Worldview-3	0.31 m	8	4-5	2014	
Worldview-4	0.31 m	4	4-5	2016	
SuperView Neo	0.30 m	4	1	2022	
Pleiades Neo	0.30 m	6	1	2021	

* : Decommissioned satellites

2.4 Classification Models

2.4.1 Deep Learning

2.4.1.1 Overview of Artificial Neural Networks

ANNs are popular machine learning algorithms. ANNs are designed based on the architecture of biological neurons to model the relationship between input and output data. Like biological neurons, ANNs receive several inputs and then process them as matrix calculations to produce a single value with an activation function (Jain et al., 1996). In general, the architecture of an artificial neural network consists of input layers, hidden layers, and output layers.

In the modeling-based feed-forward neural network algorithm, neurons in the hidden layers get information from neurons that came before them and send out a signal based on their activation function (Bebis and Georgiopoulos 1994). In comparison, in Fully Connected Neural Networks (FCNNs), each neuron in each layer is connected with both the next and previous layer neurons. On the other hand, with a fully connected feed-forward neural network, input layers receive the input data and propagate them to the subsequent layers. Hidden layers are where the extensive calculations take place in a neural network. The output layer extracts results such as classification and predictions (Agatonovic-Kustrin and Beresford 2000).

2.4.1.2 Overview of Deep Learning

Deep learning (DL) technique is considered one of the essential components of artificial intelligence (AI). DL, in general terms, is a sophisticated approach

derived from machine learning (ML) techniques. DL framework has been designed on the basis of ANNs architecture, a more complicated network that comprises a hierarchical architecture with many layers, each of which constitutes a non-linear information processing unit (Du et al., 2016).

DNNs employ deep architectures in neural networks, which can represent functions with higher complexity if the number of layers and units in a single layer is increased. Given enough labeled training datasets and suitable models, deep learning approaches can help humans establish mapping functions for operation convenience (Ma et al., 2019). Although the DL concept dates back to the end of the 20th century, it was implemented in 2006. It presented a novel architecture called the Deep Belief Network (DBN) aimed at optimizing the prediction of hand-written digits (Hinton et al., 2006). Subsequently, the development of deep learning models proceeded at a consistent pace. The main advantages of DL methods over traditional ML methods are as follows: automatic extraction of deep features; no human interference for feature selection; large and complex data; and non-linear relationships. While DL methods have advantages, their drawbacks include high computational cost, overfitting, a lack of interpretability, and dependence on data quality. Figure 2.1 shows the differences between ML and DL methods.

A deep feature is the consistent response of a node or layer within a hierarchical model to an input that gives a response that's relevant to the model's final output. One feature is considered “deeper” than another,

depending on how early in the decision tree or other framework the response is activated. A neural network that's designed for image classification is trained on a set of natural images and learns filters (features) from earlier layers, such as image edge and contour detectors. The "deeper" layers can respond and create their own feature filters for more complicated patterns in the input, such as textures, shapes, or variations of previously processed features.

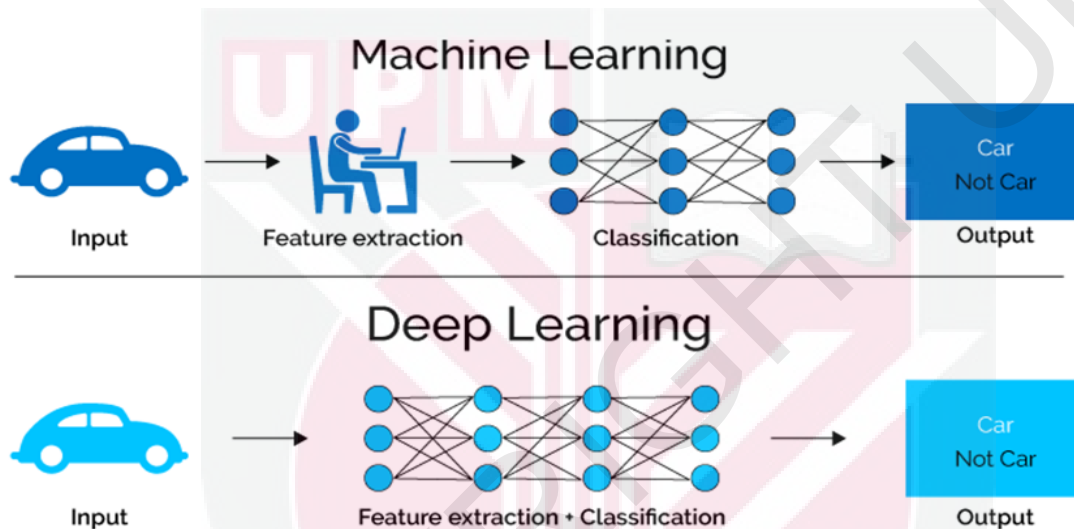


Figure 2.1 : ML and DL Techniques
(Source: www.softwaretestinghelp.com)

2.4.1.3 Deep Learning Methods

Several DL models have been developed for various applications across multiple fields and industries (e.g., computer vision, speech recognition, and robotics). Subsequently, DL models received greater attention in the GIS and remote sensing fields. DL models showed high efficiency in RS image classification. On this basis, numerous DL models have been developed to classify remote sensing imagery (e.g., CNNs, RNN, unsupervised AE, DBN, and GAN).

CNNs, the most considerably used models, were primarily designed to arrange and process imagery data as multiple arrays (LeCun et al., 2015). Due to this characteristic, CNN models are most suitable for multi-band image classification, in which pixels are regularly arranged. CNN is the most popular type of neural network that has significantly succeeded in computer vision tasks, including image classification (Hongtao and Qinchuan 2016, Srinivas et al., 2016). It also achieved excellent results for remote sensing image classification (Sharma et al., 2017, Shakya et al., 2021). CNN was applied to solve remote sensing problems utilizing aerial photographs, multispectral images, and hyperspectral images. CNNs combine three basic architectural ideas, including local receptive fields, sharing weights, and subsampling, to achieve shift, scale, and distortion invariant properties that are critical for image feature extraction. In general, local receptive fields provide each neuron within a specified connected convolutional layer with the only nearby region belonging to the previous layer, which assists the network in deriving the basic visual characteristics. The concept of sharing weights refers to the stability of the convolutional kernel's weight when generating a feature map at a specified layer. Consequently, the number of trained parameters in CNNs will significantly decrease in comparison with the ANN method. Finally, the process of subsampling decreases the feature map's resolution, which integrates with a convolution operator to obtain translation invariance.

In general, experts have developed many CNN architectures. For example, Maggiori et al. (2016) proposed a fully connected CNNs to classify an aerial photo. Their method outperformed the traditional ANN model, particularly in

terms of overall accuracy (OA) and execution time. Later, some CNN architectures were introduced. Joshaghani et al. (2023) investigated Bayesian CNNs for hyperspectral imagery with limited training samples. Bayesian CNNs demonstrated a satisfactory ability to measure uncertainty. As a result, Bayesian CNNs outperformed the traditional CNN and random forest (RF). With the rise of 3DCNN models that can automatically analyze multiple bands and multi-temporal imagery, Firat et al. (2023a) introduced a hybrid model that combines two architectures (3DCNN and ResNet). Their proposed model was characterized by extracting deeper spectral, textural, and spatial features than traditional CNN models.

Later, hybrid CNN models are developed to optimize 3D CNN models in terms of the analysis complexity and the processing time, as introduced by Ma et al. (2023) who proposed a hybrid 1D- 3D CNN model for the extraction of features from hyperspectral imagery in a complex urban area located in Houston, United States. The OA of the hybrid CNN was higher than the OA of 3D CNN by (0.42%). This study indicated that hybrid models can efficiently extract spectral and spatial features from hyperspectral imagery. Firat et al. (2023b) used a hybrid model (2D-3D CNN) for hyperspectral image classification in different areas; their model has improved the performance of 3D CNN architecture in terms of the processing complexity of hyperspectral imagery, where 2DCNN used to extract the spatial features while spectral features extracted by 3D CNN model. Their model managed to minimize the computation time and improve the OA.

RNNs are a type of supervised learning model. They are specifically designed for analyzing discrete sequences, such as predicting the next letter in a text. Over time, applications of RNNs have expanded to include RS image classification. For instance, Lyu et al. (2016) conducted a novel study utilizing the RNN model to detect changes in multitemporal images. This groundbreaking research marks the first application of the RNN model in this context; RNN has optimized change detection compared with traditional change detection methods. Their model has a proficient change rule, offering more realistic information.

Moreover, the change rule allows for the seamless transfer of change information to subsequent images. In a different study, Lakhal et al. (2018) suggested a framework to classify RS images by combining CNN and RNN models. They used CNN to extract deep features, taking into account their spatial relationships. Then, RNN was utilized for class prediction.

Additional popular models include AEs and stacked autoencoders (SAE), traditionally utilized to organize and represent compressed and distributed data. As a result, AE and SAE models are highly favoured for dimensionality reduction and the processing of feature hierarchy (Masci et al., 2011). In remote sensing, AEs and SAE are standard models used for hyperspectral imagery classification due to their capability for dimensionality reduction. For example, Özdemir et al. (2014) applied the SAE model to classify hyperspectral imagery at Pavia University, Italy. Their approach achieved satisfactory accuracy compared to state-of-the-art methods. The SAE model

was employed by Li et al. (2016) to get a large-scale LULC map from Landsat images. Their findings showed that the SAE model was more accurate and time-effective in terms of predictions than other machine learning models (SVM and RF). The work of Zhao et al. (2021), on the other hand, combined SAE and 3D CNN models to improve the classification of a hyperspectral image. Their model used SAE to reduce the time and data size, and it worked better than other models (like SVM, CNN, and KNN), with an OA of 99.67% using only 5% training samples. Singh et al. (2022), in a new study, provided a framework that uses SAE and hybrid 2D–3D CNN models to sort joined LiDAR and hyperspectral data into different categories. Their framework took advantage of SAE's dimensionality reduction power and hybrid CNN's classification power; modeling results demonstrated the framework's efficiency with an OA of 98%.

Restricted Boltzmann Machine (RBM) is one of the DL models that can learn a probability distribution over its set of inputs. It's also used for LULC extraction. Lekfuangfu et al. (2016) produced a land cover map based on the integration between RBM and SVM models. The RBM model clustered the image into regions without the need for training samples. In contrast, SVM is employed for the classification. They revealed that texture information can optimize the classification process. Tan et al. (2019) developed a new Gaussian-Bernoulli-restricted Boltzmann machine to pull out deep features from hyperspectral images. They also used a logistic regression model to make the classification process work. The main advantage of the proposed model is the unsupervised deep feature extraction, which can minimize

prediction time with high-quality classification results. DBN is a more advanced network than RBM, which is an unsupervised model that includes a group of RBMs layers. Vignesh et al. (2019) suggested a DBN model to extract deep features from satellite images in Kerala, India.

GANs is an unsupervised DL model (Goodfellow et al., 2014). The generative network learns to map from a latent space to a specific data distribution of interest (for example, photos), whereas the discriminative network distinguishes between actual data and generated data created by the generative network. Zhu et al. (2018) applied the GAN model for hyperspectral image classification. Their findings showed that the proposed models outperformed state-of-the-art approaches in terms of performance. Kong et al. (2023) proposed a method that includes dual GAN models with super-resolution models to simulate long-term time series of vegetation cover. This study presents a new approach for resolving sub-pixel spatial information in Landsat images.

2.4.1.4 Hyperparameter Optimization Techniques

Tuning the hyperparameters of models is crucial for a successful learning process. Because there are several types of hyperparameters, like discrete, categorical, and continuous, the hyperparameter tuning process varies among models such as deep learning algorithms. Moreover, it is considered a very expensive approach due to the vast number of possible combinations to analyze and the resources required to do the computations (Elshaw et al., 2019). In general, there are three main techniques for tuning hyperparameters:

the grid search, the random search, and the Bayesian search.

The first technique is grid search, which is a generic technique for creating replacement model configurations. In this technique, values to be analyzed are fragmented into every hyperparameter of interest, and models are trained and tested using all hyperparameters and value combinations. Lemenkova et al. (2024) employed the grid search technique for tuning the hyperparameters of a machine learning classifier, which improved the coastal region mapping. On the other hand, Alaieri et al. (2024) applied hyperparameter fine-tuning to several machine learning algorithms using a grid search technique in order to improve the prediction of soil moisture. In a separate study, Shams et al. (2024) used grid search optimization and several machine learning models to improve water quality prediction through hyperparameter tuning of different parameters. The grid search optimization is simple to implement, but it is time-consuming and expensive due to the number of hyperparameters and their distinct levels. This results in an increase in computation costs.

Another technique for hyperparameter optimization is called random search. It performs a random search on a fixed number of combinations in the hyperparameter space (Bergstra and Bengio 2012). KRAYEM et al. (2022) used random search techniques to fine-tune CNN parameters to improve image classification based on studies that employ random search techniques. In their study, Rom et al. (2024) applied random search, which contributes to the growing body of information on hyperparameter tuning for DL models and acts as a resource for practitioners looking to optimize their DL designs.

However, the random search technique is more effective than grid search; it is considered a time-consuming technique and requires high computational resources. Furthermore, it doesn't guarantee the attainment of optimal results.

Optimization methods such as grid and random search are often used with objective functions $f(x)$ that are cheap to evaluate. However, with expensive objective functions, it is important to minimize the number of samples drawn from the black box function. Bayesian optimization is an approach that best suits this kind of problem. It attempts to find the global optimum in a minimum number of iterations. Bayesian optimization incorporates a prior belief about f and updates the prior with samples drawn from f to obtain a posterior that better approximates f .

A surrogate model is used to approximate the objective function. Bayesian optimization also uses an acquisition function that directs sampling to areas where an improvement over the current best observation is likely. Bayesian optimization works based on the following steps: (1) selecting a surrogate model for defining the prior of the objective function; (2) obtaining the posterior using the Bayes rule based on the function evaluations; (3) using an acquisition function to decide the next sample point; and (4) adding newly sampled data to the set of observations and repeating the process from step 2 until convergence or budget elapses.

As a result of Bayesian optimization's success, a variety of studies have been conducted. Sameen et al. (2020) applied a Bayesian optimization technique

to find the best CNN hyperparameters for improving landslide mapping based on remote sensing imagery sources. Another study, Zhang et al. (2021), presented a method for automatically tuning the hyperparameters of a random forest model based on Bayesian optimization in another application in terms of improving land cover classification. In a recent study, Hamza et al. (2023) proposed the integration of deep learning and Bayesian optimization, where the hyperparameters are automatically configured instead of manually, leading to an improvement in feature learning. Furthermore, they are trying to reduce the processing time. Bayesian optimization, in general, makes smart choices depending on previous evaluations. As a result, it improves model performance by finding an optimal set of parameters with minimal search time. However, while Bayesian optimization demonstrated its feasibility in optimizing the black-box function, it does have certain limitations. It's a complicated process that requires a large number of initial evaluations and relies on the surrogate model's accuracy.

2.4.2 OBIA

In OBIA, images are first divided into several image objects (i.e., segments), each given a specific categorical label. OBIA techniques have been developed since the late 1990s. This method uses image objects as the basic unit for analysis rather than image pixels (Blaschke 2010). It is one of the major methods developed and applied to overcome the drawbacks of traditional pixel-based methods, such as salt and pepper errors. OBIA includes two main processing stages, i.e., (1) image segmentation and (2) the extraction of spatial, spectral, contextual, and textural information to determine thematic

classes within remote sensing images. For each image object, a collection of spectral, spatial, and textural features can be extracted and used for the following data processing stage, i.e., classification. The classification in OBIA is often performed with any statistical or machine learning method, including SVM, ANN, and Decision Trees (DT). Figure 2.2 illustrates the difference between pixel-based and OBIA classification methods.

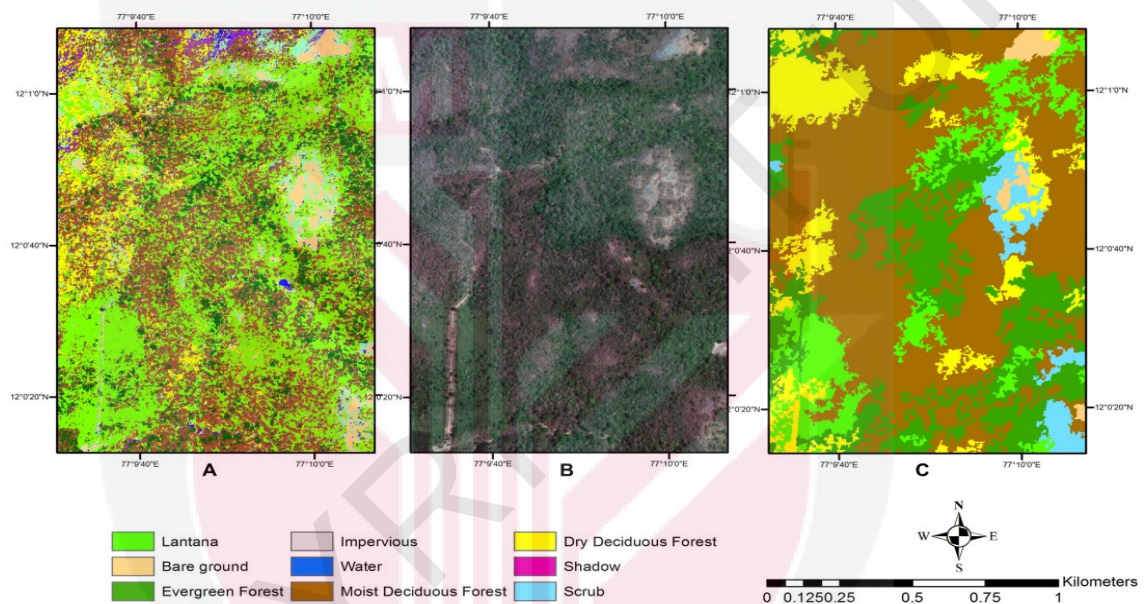


Figure 2.2 : The Difference between Pixel-Based and Object-Based Classification Methods (A) Illustrates the Pixel Based and (C) Illustrates OBIA (B) the Original RS Image
(Source: Niphadkar et. al, 2017)

2.4.2.1 Segmentation

The segmentation stage divides remote sensing images into relatively homogeneous objects (Baatz 2000). It is based on the selected segmentation algorithm but commonly depends on several parameters, like shape and scale, that can be extracted from the nature of features. The second step includes feature extraction and classification operations. Several image

features are calculated for each image object, and objects are classified based on the computed features. In the classification operation, all pixels contained in one object are assigned to a single class. As a result, eliminating mixed-pixel errors is typically achieved using traditional pixel-based methods (Blaschke 2010). There are several segmentation algorithms applied in remote sensing, including multiresolution segmentation (Atik et al., 2021), mean shift (Lourenço et al., 2021, Zhou et al., 2021), watershed methods (Xue et al., 2021, Dawei et al., 2021), and simple linear iterative clustering (SLIC) (Ameslek et al., 2023). However, MRS first proposed by Baatz in 2000 (Baatz 2000) is the most common approach used for fine spatial resolution image segmentation (Hossain et al., 2019 a).

Mean shift algorithm is one of the clustering algorithms that work in the feature space and considers the segmentation problem a clustering problem by determining the density function's mode and allocating points for these modes (Cheng 1995). For each point, the scheme converges to a mode of the density function. Then, for every point that converges to the same model, we assign them to a cluster. Thus, we get the final segmented image. Many applications have adopted mean shift segmentation, following (Wang et al., 2004) introduction of this method, which adapts the shape, scale, and orientation of the kernels to the local structure of the image or video. On the other hand, Ming et al. (2015) developed a segmentation-based mean shift segmentation to select the scale parameter for object-based information extraction from high-resolution remote sensing images. Another common application is object tracking. For example, Hardini et al. (2020) presented a mean-shift-based

object tracking system with faster processing time performance to reach the converging point of the object target on each video frame.

Lilay et al. (2023) conducted a study on how to classify a sentinel2 image using the meanshift algorithm for semantic segmentation and SVM and RF machine learning classifiers for LULC mapping. They discovered that semantic segmentation improved the classification's performance. In another study, Ismayilova et al. (2023) presented an approach based on mean shift segmentation and the Normalized Difference Vegetation Index (NDVI) to extract urban green spaces from satellite images (Sentinel 2). Their approach managed to identify 90% of green areas within the study area. Although the mean shift algorithm has been widely used in many applications, a rigid proof for the algorithm's convergence using a general kernel in a high-dimensional space is still unknown (Ghassabeh et al., 2015).

Another type of segmentation, called a graph-cut (GC), represents the image's pixel as a specific node in a graph; each node is joined to its surrounding neighbouring nodes (4 nodes) based on particular links called n-links. Pixel nodes are also connected to other members by t-links, such as terminal nodes (2 nodes), target, and source. All links are assigned non-negative weights. GC finds a cut through the graph, resulting in a bipartite graph with minimal cumulative costs (Boykov 2006). GC models were applied for many geospatial applications; for example, Reza et al. (2019) optimized the accuracy of rice estimation from UAV imagery by integrating k-means and GC segmentation. Their findings revealed that the determination coefficient between the

proposed method and the actual data was 0.98. Gandhimathi et al. (2018) proposed a hybrid model that combines the Difference of Offset Gaussian (DOOG) filters and GC segmentation to detect land cover changes using multi-temporal Landsat imagery. The experimental findings show that in the hybrid model, the mean value of the changed area achieved a 47.2% reduction compared to the conventional models. The main limitation of this method is that when a 4-connected lattice represents an image, graph cut methods can exhibit unwanted "blockiness" artifacts; multiple labels: GC is only able to find a global optimum for binary labeling (i.e., two labels) problems, such as foreground/background image segmentation. Memory: the memory usage of graph cuts increases quickly as the image size increases.

Watershed technique is a segmentation algorithm that was developed based on the notion of "immersion." Watershed segmentation conceptually considers local minima belonging to the Gray scale of the image as a surface submerged in water. After that, the water will gradually fill catchment basins within the image (surface), starting from the minima of the lowest intensity value of the image. Theoretically, the watershed algorithm creates a dam to avoid water coming from multiple local minima, which would be merged. As a result, any local minima within the image will be enclosed by dams that correspond to the image's watersheds (surface) (Rafael et al., 2002).

Derivaux et al. (2006) presented watershed segmentation and fuzzy pixel-based classification for a VHR image (Quickbird). Their results indicated that performing a watershed on a membership space is less prone to over-

segmentation, which leads to optimizing the classification process. Tang and Wang (2012) innovated an approach that includes combined watershed and vegetation-impervious surface-soil to improve urban LULC extraction from Landsat images; they achieved substantial accuracy compared to the conventional maximum likelihood method. López-Fandiño et al. (2019) presented a GPU framework that includes watershed segmentation and spectral angle mapper to detect changes from multi-temporal hyperspectral images. Their results indicated that the GPU framework had accelerated the processing time and achieved real-time execution.

The most common segmentation in the remote sensing field is multiresolution segmentation (MRS). MRS is a bottom-up multi-scale segmentation that generates the objects using an iterative algorithm, which minimizes the average heterogeneity of image objects weighted by size. It has three critical parameters that need to be set to control the growth of the generated objects: scale, shape, and compactness. The scale and shape parameters are the maximum allowed heterogeneity and textural homogeneity in the resultant image segments. Likewise, the last parameter is used to optimize segments based on their compactness, with a focus on relatively compact segments (Chen et al. 2017, Definiens 2006). The scale parameter controls the size of the generated image objects. For a certain scale value, the size of the image object is larger for homogeneous data, whereas it is smaller for heterogeneous data. The shape parameter value influences the relationship between the color and shape criteria. The color standard can be adjusted and set by selecting a suitable value for shape criteria. The compactness of an image object can be

defined as the product of the width and the length over pixel numbers.

Several studies employed MRS for image classification. Aguilar et al. (2016) optimized the extraction of plastic greenhouses using MRS and a VHR image (Worldview-2). Cai et al. (2019) presented an approach that depends on MRS and RF models to extract urban LULC from medium and VHR satellite imagery. Their method achieved an accuracy of 0.87 and 0.89. To identify and map desert landforms, Garajeh et al. (2022) employed MRS segmentation, the Digital Elevation Model (DEM), and Sentinel-2 images. The results demonstrated the high potential of the rule-based OBIA remote sensing technique for detecting and delineating arid landforms. In terms of mangrove classification based on VHR and medium-resolution images, Sanam et al. (2023) conducted a comparison between OBIA and pixel-based classification techniques. The final results showed that the integration between MRS and RF achieved better accuracy than the pixel-based technique.

Another segmentation type is simple linear iterative clustering (SLIC). This segmentation works by clustering pixels based on their proximity and color similarity in the scene. In 2017, Xie et al. suggested a method to improve the classification of hyperspectral images with few ground truth samples by using the superpixel technique (SLIC) and adaptive multiscale segmentation (MSGLAMS). Zi et al. (2018) used a novel framework to optimize the cloud detection application; their framework includes the refinement of detected clouds resulting from SLIC segmentation using conditional random field (CRF). The results indicate that our proposed method can accurately detect

clouds under different conditions, which is more effective and robust than the compared state-of-the-art methods. On the other hand, Zhao et al. (2022) proposed a hybrid model that includes the application of SLIC at different scales to classify hyperspectral imagery that suffers from a limited number of training samples. Experimental results revealed that the applied model outperformed state-of-the-art models. Table 2.3 shows a summary of the common segmentation methods.

2.4.2.2 Feature Extraction

Feature extraction in OBIA is a second significant step that is often applied after we obtain image segments. There are a number of features that are commonly handcrafted and grouped into spatial, spectral, and textural categories. The spatial features are computed based on the image object's geometry. The following are common features: rectangular fit, roundness, asymmetry, area, length-to-width ratio, relative border length, connectivity with other objects, and proximity to other objects. The spectral features are computed based on the average statistics of the pixels within each image object. The common features are standard deviation, mean brightness, and band ratios. In addition to spatial and spectral features, there are also textural features such as smoothness and local homogeneity that are frequently included in image analysis.

Those features are handcrafted based on the application at hand. For example, in urban LULC mapping, the focus is on the spatial features, while in other applications, such as forest mapping, spectral and textural features

may play more significant roles. In any case, the issue with feature extraction in OBIA is subjective and handcrafted. This process is time consuming and the results are not always effective. In general, Table 2.4 shows the main differences between the OBIA and DL classifications.



Table 2.3 : The Summary of Common Segmentation Methods

Method	Working Principle	Parameters	Advantages/Drawbacks	Common Applications
Mean shift	Clustering problem by determining the density function's mode (a non-parametric kernel density estimation) and allocating points for these modes.	- Spatial resolution - Range resolution - Size of smallest segment	- It is capable of modeling complex clusters with a non-convex shape; the user only needs to set one parameter, the bandwidth, which has an intuitive physical meaning of local scale and no local minima. - There is still no rigid proof for the algorithm's convergence using a general kernel in a high-dimensional space.	Image classification Multivalued regression and inversion video segmentation object tracking
Graph cut	The graph represents each pixel of the image as a distinct node. Each node is connected to its surrounding neighboring nodes (4 nodes) through specific links, known as n-links. On the other hand, pixel nodes are also connected with other members, such as terminal nodes (2 nodes), targets, and sources, based on t-links. All links are assigned non-negative weights. Graph-cut finds a cut through the graph, which results in a bipartite graph and has minimal cumulative costs (Boykov 2006).	- Intensity - Features - Gradient - Direction - Features - Boundary Corner - Feature - Texture Feature	- The graph-cut based method is efficient and widely accepted worldwide since it can achieve a globally optimal result for the energy function. It is not only promising for specific images with known information, but it is also effective for natural images without any pre-existing information. - When an image is represented by a 4-connected lattice, graph cuts methods can exhibit unwanted "blockiness" artifacts. Multiple labels: Graph cuts can only identify a global optimum for binary labeling problems, which involve two labels, such as the segmentation of foreground and background images. Memory: Graph cuts quickly increase in memory usage as the size of the image increases.	Medical Image Segmentation Change detection Land cover classification

Table 2.3 : Continued

Watershed	The watershed technique is a segmentation algorithm that was developed based on the notion of "immersion." Watershed segmentation conceptually considers local minima belonging to the gray scale of the image as a surface submerged in water. After that, the water will gradually fill the catchment basins within the image (surface), starting from the minima of the lowest intensity value of the image.	the minimum threshold value for the watershed and the watershed level gradient values at each pixel	- The watershed algorithm is widely used in many applications, such as the segmentation of medical images, due to its advantages. It is considered a simple method, fast, and could generate complete division of images in isolated regions even when the contrast is often poor, hence avoiding the need for contour joining techniques. - The watershed method has the disadvantage that it is highly sensitive to local minima, since at each minima, a watershed is created. If you have an image with noise, this will influence the segmentation.	Medical Image Segmentation Morphological studies Land cover classification
Multiscale (multiresolution)	The iterative process in the multiresolution algorithm aims to decrease the mean heterogeneity within each segment, resulting in smaller segments merging into larger ones.	Level Name Image Layer Weights Thematic Layer usage Scale Parameter Shape – Colour Compactness	Multiresolution segmentation has the ability to analyze multiple layers and cluster the image's regions based on spectral and spatial heterogeneity reduction.	Remote Sensing image classification Geographic object detection.
Simple linear iterative clustering	The super-pixel technique groups pixels based on color and spatial distances.	Pixel value Colour Spatial distance	Minimize the computation time; uniform size of the super-pixel block; and regular contour are widely used in color images.	Medical applications, Land cover extraction.

Table 2.4 : The Difference between Deep Learning and OBIA Classification

OBIA	Deep learning
A better spatial representation of ground objects • Subjective: manual selection of features • It necessitates optimizing the selected features • Aggregated to single values • Lacks transferability Any model can be used as a classification	The representation of ground objects lacks both spatial and semantic information • Optimized features that are hierarchal from low-level to more abstract • Does not require aggregation • Better transferability Any model can be used as a classification

2.5 Integration

Recently, OBIA classification has paid more attention to DL algorithms due to their potential for extracting deep features. In 2016, Längkvist proposed the earliest framework that integrates segmented objects and deep features, where SLIC segmentation was applied to refine the CNN classification outputs. DL algorithms, such as CNN algorithms, can be applied to derive the spatial features of each image's pixel, which can be used as auxiliary information to improve the object-based classification accuracy of multilayer perceptron (Zhao et al., 2017). With the help of the segmentations' geometric characteristics and classification rules, segmentation techniques and deep learning decisions can be integrated to realize object-based and deep learning classification (Zhang et al., 2018c). Although deep learning algorithms (i.e., CNNs) have excellent image recognition capabilities, they are usually not directly applied to classify segments; accordingly.

The main problem arises from the integration between OBIA and DL. First, let's examine the OBIA-CNN integration issue, which arises from the incompatibility between the CNN input and the OBIA segments. In general, the rectangular shape of CNN patch will never be compatible with the non-homogenous shape of the segment (S) that resulted from OBIA method. For example, when performing OBIA classification based on patch-based CNN, the non-homogenous segment is clipped based on a rectangular geometry of CNN patch, then CNN method is used to assign a specific category to this patch, and then all pixels within the segment S will be assigned to the same category. Figure 2.4 shows the standard process of the integration between OBIA and CNN methods; two ideal situations may occur during the process of OBIA-CNN integration; the first situation arises when the segment (S) is larger than CNN-patch, the patch is fully contained inside the segment (S), while the second situation occurs when the CNN-patch is larger than the segment (S).

The patch contains multiple segments that represent the same land cover type; in these situations, the CNN-patch typically aligns with segment (S), thereby improving classification accuracy. The problem arises when the CNN-patch covers multiple segments with different land cover categories. For example, a segment belongs to a specific type of land cover called (A); however, the CNN decision is mainly influenced by all of the pixels throughout the CNN-patch, and the final output will represent the dominant land cover called (B) depending on several factors such as the dominant band value, larger areas, and structured textures. In this case, CNN classifier will make a wrong decision and obtain an incorrect classification for this segment (Pan et

al., 2019).

Few research achievements in the field of traditional structured CNNs applied to object-based remote sensing image classification have been reported. In general, there are five common frameworks for integrating object-based image analysis and deep learning algorithms; deep learning based on object-level features, feature-level fusion, decision-level fusion, deep learning with context patches , and deep learning with filtered patches. Table 2.5 describes the summary of the literature review. This section describes almost all related works. Figure 2.4 illustrates the concept of integration between OBIA and CNN.

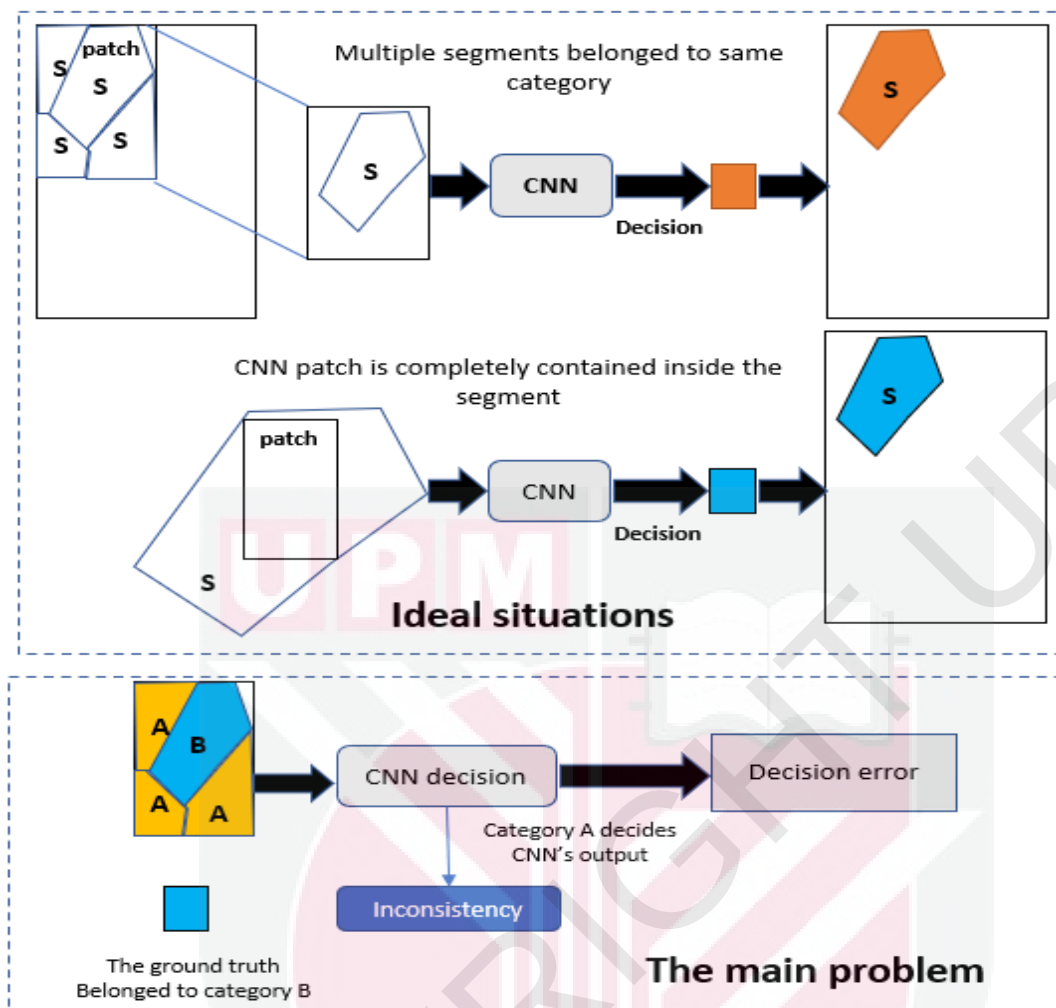


Figure 2.3 : An Example of the Integration between OBIA and CNN for Satellite Image Classification

2.5.1 Deep Learning Based on Object-Level Features

This technique is applied to train CNNs on object-based features. Furthermore, the remote sensing image is initially segmented. Then, several textural, geometric, contextual, and spectral features are calculated for each image object, and the extracted information is transformed into tabular form to be prepared for the classification operation. Finally, CNNs are trained with these features, and a classified image is obtained (Liu and Abd-Elrahman 2018a, Bycroft et. al, 2019). In a number of studies, it has been used to train

CNN on OBIA features for remote sensing image classification. For example, Zhang et al. (2017) suggested a new framework that uses graph-based segmentation and stacked denoising autoencoders to make land cover mapping-based UAV imagery sources better in China. Their framework's main disadvantage is that the extracted features may not be as meaningful or useful as expected. It also requires significant computational resources. Given the effectiveness of recurrent neural networks (RNNs) in sequential learning applications, Ienco et al. (2017) proposed a model for landcover detection that integrates recurrent neural networks (RNN) and OBIA classification, applied to time series satellite images in France. Their model was computationally intensive. With the CNNs' exceptional performance in remote sensing image classification, Liu et al. (2018a) developed a Fully Convolutional Networks (FCN-OBIA) model based on multi-view orthophotos to improve land cover classification in the United States. Although multi-view imagery has advantages in terms of information's enrichment, it is considered expensive and time-consuming to process. The extraction of urban LULC is a complex task in remote sensing due to the heterogeneity of LULC types. Therefore, it requires effective approaches to improve the accuracy of LULC extraction. Zhang et al. (2018c) classified urban LULC based on the OCNN model, which includes the integration between Meanshift segmentation and CNN models. Experimental results exhibited that the OCNN model is efficient for urban LULC. The main drawback of this model is that the Meanshift algorithm may make it challenging to accurately delineate objects with complex shapes and overlapped features, which can lead to jagged errors. Liu et al. (2018b), on the other hand, created a new model to classify multi-view images. It was based

on the combined CNN-OBIA model and the conditional random field model. Their results showed that the proposed model improved the process of classifying land cover by adding more training data from multi-view images. However, the conditional random field algorithm has minimized the processing time. This method is still expensive because of data collection. In another study, Cui et al. (2018) presented a novel approach by training the CNN-OBIA model on a multispectral image and then training it on a panchromatic band. Their method improved the classification process by increasing the amount of information in high-resolution images.

Given the high capabilities of super-pixel segmentation for minimizing computational costs, Lv et al. (2018) proposed a model integrating super-pixel segmentation and CNN classification for land cover classification in the United States. Their results optimized the traditional super-pixel classification by reducing the number of processing units at various scales. This type of segmentation is responsive to the characteristics of the used image; therefore, finding optimal segmentation parameters can be a challenging task. On the topic of agricultural applications, Li et al. (2019) combined the OSVM model and the OCNN model to make crop classification from RS imagery better. Their results showed that the proposed method can be used in other study areas. On the other hand, Liu et al. (2019) proposed a novel approach for improving change detection performance in the United States by integrating CNN and OBIA classification based on multi-temporal images. In a complex urban environment in the United States, Jozdani et al. (2019) compared machine learning models and the CNN-OBIA model for urban LULC mapping;

their experimental results revealed that CNN-OBIA outperformed the traditional machine learning model. Mboga et al. (2019a) proposed a semi-automatic method for image classification in the Democratic Republic of Congo, utilizing fully CNN and OBIA classification. Jin et al. (2019) suggested using a method that combines object-oriented MRS segmentation with deep CNNs (COCNN) to classify land cover in China. Their results showed that this method worked better than traditional object-oriented models. On the other hand, the existing research in general rule set construction and deep learning structure design remains imperfect and requires further improvement. However, this technique has been used in many studies and has achieved satisfactory results. In another study, Ienco et al. (2020) presented a new deep learning framework named “aTtentive weAkly Supervised Satellite Image Time sEries cLassifier” (TASSEL) to deal with the weak supervision provided by the coarse granularity labels. In general, the CNN algorithm lacks the ability to learn the spatial characteristics of the image content (pixels and objects). Geiß et al. (2020) proposed a novel post-classification relearning strategy for CNNs to enhance classification accuracy. Another study was conducted to optimize LULC classification; Zhang et al. (2020a) developed a joint deep learning framework as a novel framework for urban LULC classification. Three state-of-the-art methods were compared under the framework: JDL, SS-JDL, and a newly proposed cross-GAN. Many applications have thought about combining OBIA and DL. For example, Merchant et al. (2020) classified open water features using sentinel data, multiresolution segmentation, and the CNN algorithm with eCognition and TensorFlow. However, this framework is considered complex and requires careful design and tuning of architectures

and hyperparameters.

Li et al., 2020 Segmentation: The experimental VHR image is segmented into different less-segmented object image patches based on geographic object-based image analysis. Then, using superpixel segmentation algorithm segments, the less-segmented object is over-segmented into patches with high homogeneity. Finally, a less-segmented object and one of its subsets (less-segmented objects) are used as a spatial processing unit. Standardization and SOD-CNN training: Four standardization methods are developed to limit the number of pixels around the segmented object. To train SOD-CNN, corresponding standardized less-segmented objects and over-segmented objects are simultaneously fed into two CNN networks with different model structures and window sizes. In addition, a feature fusion in the fully connected layer is undertaken to determine the final classification results. Classification: Using the trained SOD-CNN, the test data of all processing units of the original image is classified. The decision about the category of over-segmented objects in the processing unit determines the final classification map. The superiority of OCNN models has encouraged researchers to experience them in other applications such as road extraction, agriculture, and disaster management. For instance, Abdollahi et al. (2021) claim that the primary contribution of this study involves combining PCA and OBIA with the CNN model for the purpose of classifying orthophoto images into road and non-road parts. They then apply a trivial opening to create a binary image and extract road parts, a technique not previously used in the literature. As a result, the computation and training time were reduced, and

the proposed model was trained by some samples (road and non-road segments) for only a few seconds, which is significantly less compared to the aforementioned deep learning methods in this study while still achieving excellent results. For agricultural applications, as reported by Lam et al. (2021), This work has shown the potential of combining OBIA and CNN to develop and implement a semi- or fully automated system for weed species identification and detection in a challenging grassland environment. The proposed open-source workflow aimed to support small and medium-sized farmers and nature conservation organizations in site-specific weed management, particularly weed monitoring, mapping, and weeding. In another study, Li et al. (2022a) presented an approach to improving crop classification that depends on RS images. This study classified multi-temporal satellite images based on the OCNN approach, including the integrated MRS and CNN. They achieved high accuracy in comparison with state-of-the-art methods.

Regarding disaster management techniques that rely on RS imagery, Qing et al. (2022) used UAV photos to conduct an earthquake assessment in Yunnan, China; they proposed an approach based on integrating the superpixel technique (Extra Feature Enhancement Bands) and CNN model. The experimental results revealed that their method achieved higher accuracy than traditional techniques in terms of damage detection. In a separate study, Wan et al. (2023) detected flooded areas from SAR data in Mongolia, China; their approach combined the segmentation technique (SLIC) and CNN algorithm. The proposed method outperformed traditional ML algorithms in terms of

water detection precision, which increased by 3%. SLICS segmentation typically generates artifacts within object boundaries. Furthermore, they have limited contextual information, whereas SLIC relies primarily on spatial proximity and local color for segmentation. Detka et al. (2023) proposed a methodology to extract coastal shrubs from UAV photos in California, USA; their model depends on the combined MRS and CNN models. The final findings indicated that the proposed model optimized the traditional methods of coastal shrub extraction. Because OBIA-deep learning methods are excellent at getting spectral and spatial features from RS imagery, Wiskes et al. (2023) used OBIA-CNN to find abandoned logging roads in California, USA, using VHR aerial photos and LiDAR data. The proposed framework contains the MRS-CNN combination, and their results showed a significant improvement in terms of accuracy. Dewali et al. (2023) used a framework integrating OBIA, CNN, and UAV images for avalanche deposit detection and characterization. The proposed framework has outperformed the state-of-the-art in terms of overall accuracy. In a different study, Giles et al. (2023) suggested a model that combined the MRS and U-Net models. This model made it easier to find and classify coral reefs by using both detailed object boundaries and high-level features. In the most recent studies conducted by Hossain et al. (2024a), they proposed an OCNN framework to improve the building extraction process by using aerial photos compared with the traditional OBIA. In the same way, Hossain et al. (2024b) showed a new OCNN model that used watershed segmentation and Neural Oblivious Decision Ensemble (NODE) for tabular data. This model made building extraction from HR imagery more accurate. Its superiority over traditional

methods has led experts to use this framework in various applications. However, this framework has the ability to merge spatial information with spectral and textural information to enhance building extraction and classification. The main limitation of this study is the complexity of transforming tabular information into imagery, because the proposed model lacks the ability to capture the spatial relationship between the table's cells. Bai et al. (2024), who scored outstanding results in the change detection study using an advanced method based on the combined MRS and Unet3 models. This framework This framework is known for its simplicity in learning contextual relationships among OBIA attributes for change detection applications. However, it lacks the learning of the spatial characteristics of the image content, such as pixels and objects. It also disregards the powerful capabilities of deep learning for extracting spatial and abstract features from image data. Xie et al. (2024) did another study that used a combination of advanced CNN (DeepLabV3+) and MRS models to better find waterlogged areas in satellite images. Their framework made extraction 6% more accurate than traditional OBIA methods. In another application, Sussi et al. (2024) proposed a model to extract roads from very high-resolution aerial photos. Their model integrated SLIC and advanced DeepLab V3+ models, and the extraction of deep (spatial, contextual, and spectral) features as road boundaries led to an improvement in the extraction of roads.

Despite the fine results provided by the technique of the deep learning classification based on object-level features, this method lacks the ability to learn the spatial characteristics of the image content, including pixels and

objects. It also disregards the powerful capabilities of deep learning for extracting spatial and abstract features from image data.

2.5.2 Feature-Level Fusion

In the second technique, object-based features are combined with deep features (feature fusion). The features extracted from OBIA can be combined with deep features learned by CNN. The combined set of features can then be used as input for a classifier, such as tree-based models or support vector machines (Zhao 2017; Majd 2019).

Several works have been conducted considering the feature fusion technique. Li et al. (2016) conducted the first study and suggested a way to improve the change detection map in China by using a model that combines mean shift segmentation and a stacked autoencoder (SAE) algorithm. This framework is considered complicated in terms of hyperparameter tuning, which requires a high number of iterations. On the other hand, Zhao et al. (2017) combined semantic-free segmentation and the CNN algorithm to extract image objects and semantic attributes that were extracted using a conditional random field model (CRF) to refine the land cover map; their results showed a significant improvement in classification accuracy, and the overall accuracy increased from 82% to 96%. The complexity of the framework stems from the duplication of certain features extracted by OBIA and CNN, which increases the model's training time. Another related work was presented by Attaf et al. (2018) they conducted a fusion between OBIA and CNN features to optimize the feature selection process. They enhanced the LULC outcomes in a specific region,

Florida, USA. Their approach presented satisfactory results compared to traditional feature selection methods. Both studies produce redundant information that requires extra steps to refine the final result. On the other hand, Fu et al. (2019) improved the automatic mapping of impervious surfaces by using the OCNN technique through the integration of MRS, CNN-ResNet-V2 and VHR satellite images (GF2). Furthermore, Wu 2020 presented a new methodology that combines the CNN-OBIA model with ancillary data, such as historical digital maps, to update a land cover map in China. The main challenge from this study is that the ancillary data may not perfectly match the image, which may negatively impact the classification results. Rajesh et al. (2020) proposed a new system for classifying LISS IV images by combining the CNN model with OBIA classification. To improve detection accuracy, Sutha et al. 2020 proposed object-based classification (OBC) of high-resolution RS images (HRRSI) using the HRSVM-CNN classifier. Hong et al. 2020 applied the multiscale segmentation algorithm to obtain multiscale image objects, and multiscale low-level features of objects were extracted. Secondly, CNN is devoted to extracting deep features from low-level object features at each scale. Thirdly, the obtained deep features at all scales are stacked and fed to one fully connected layer to extract the multiscale deep learning features for classification. Finally, the logistic regression classifier is applied to hyperspectral image (HSI) classification based on object-oriented multiscale deep features. But shallow classifiers, in general, often lack the ability to learn complex relationships and patterns among data. In a bid to improve the extraction of plants from remote sensing imagery, Tang et al. (2020) used object-based CNNs to extract tea plantations from VHR images. Image

segmentation was performed to obtain image objects, while a finely tuned CNN model was used to extract deep image features. To reduce the dimensionality of deep features, they conducted feature selection based on the Gini index, and the selected features were then used to classify tea objects via a random forest. The hyperparameter optimization for feature learning and segmentation has been separately implemented, resulting in a knowledge gap between segmentation and feature learning. Guirado et al. (2021) explored and analyzed the accuracy of object-based image analysis (OBIA) and mask region-based CNNs (Mask R-CNN), as well as the fusion of both methods in the segmentation of scattered vegetation in a dryland ecosystem. Zaabar et al. (2022) improved the conventional LULC extraction methods from sentinel2 and Pleiades images. They implemented the proposed approach by combining MRS and CNN. Their results have achieved 93.5%. In China, Zhu et al. (2022) integrated MRS and CNN models to extract roads based on high-resolution RS images; the proposed method achieved a comprehensive evaluation index value of 0.8. Zhang et al. (2022) applied the integration between multi-scale superpixel and CNN algorithm to perform land use classification from VHR images in Lagos and Kano, Nigeria. The feature-level fusion framework attracted experts to explore more advanced applications. The feature-level fusion framework attracted experts to explore more advanced applications. Li et al. (2023) proposed an advanced model that includes the integration between MRS and U-Net models to identify check dam systems. Their model has increased the accuracy of check dam system identification using remote sensing imagery. And for agricultural applications based on remote sensing imagery, Ubben et al. (2024) proposed a model

based on the integration between SLIC and CNN models and UAV photos for weed mapping. Their model improves the accuracy of weed (i.e., soybean) detection and delineation compared to traditional weed mapping methods.

The problem with this technique is that it increases the complexity of the problem and requires more time for training, validation, and hardware resources. Also, there could be duplication in some features extracted from GOBIA and CNN, such as shape, texture, and color. Also, the combined set of features is represented in tabular format, which restricts the use of deep learning models for the final classification task. The same problems as in the first method are highly likely to occur in this method as well.

2.5.3 Decision-Level Fusion

The third technique refines deep learning results based on image objects and majority statistics. This method uses deep learning models, such as CNN, to obtain a classified image. Additionally, it uses OBIA to obtain image objects for the same area. Then, using statistical methods such as majority filtering, the CNN results are refined based on the image objects. This approach is a simple post-classification refinement aiming to improve deep learning results (Lv et., 2018, Liu et al., 2019). Due to the convenience of using a decision-level framework, numerous research studies have been conducted for various applications. Mboga et. al (2019b) proposed a model that refines CNN through the segmentation process. The final results demonstrated an improvement in building extraction efficiency through the integration of the CNN algorithm and segmentation process. Liu et al. (2019) presented a novel method by refining

CNN classification results based on the object-based image analysis model and majority voting algorithm to classify Sentinel, SAR, and hyperspectral data. The proposed method demonstrated the feasibility of detecting small ground features like roads. Papadomanolaki et al. (2019) developed a model by combining anisotropic morphological leveling simplification and CNN models, and then refined their result based on OBIA classification; their results indicated that this method could force pixels belonging to the same object to be classified on the corresponding dominant class, retaining spectral and spatial characteristics. The manual selection of segmentation parameters complicates the model. Lv et al. (2018) developed a new method called region-based majority voting CNN (RMV-CNN) and OBIA classification. Their results showed that using a new majority voting strategy to get around the problem of region shape for CNNs could lead to better multi-scale CNN image classification with reasonable boundary fitting segmentation. Their results achieved 4% higher accuracy than traditional OBIA. This model is highly dependent on the segmentation algorithm; the quality of segmentation impacts the classification. In the same way, Ge et al. (2017) showed a new way to classify radar images (PolSAR) using Deep Belief Networks (DBN) and OBIA based on simple linear iterative clustering (SLIC) super-pixel segmentation. Deep learning results were refined based on super-pixel segmentation. Their findings revealed a significant reduction in the radar image's speckle noise. Their model usually requires treatment through the fine-tuning process because it is sensitive to initial hyperparameters. In a separate study, Abdi et al. (2018) proposed a method by combining CNN-logistic regression algorithms, and classification results were modified by using OBIA

classification; their framework showed high accuracy compared to traditional classifiers. This method is considered complex and time-consuming because it requires extra processing to remove duplications. Robson et al. (2020) The entire image classification procedure was carried out using eCognition 9.5, which included training the CNN, applying it to the entire imagery, and performing OBIA reshaping. Using CNNs together with OBIA is likely to provide a novel and powerful technique to automatically identify rock glaciers in mountainous landscapes. This procedure is considered time-consuming due to the implementation of multiple programs and platforms. As for the application of urban tree extraction, Timilsina et al. (2020) effectively map urban tree cover changes and model the relationship of such changes with socioeconomic variables. The OCNN method is illustrated by mapping urban tree cover changes between 2005 and 2015 using satellite, Google Earth imageries, and Light Detection and Ranging (LiDAR) datasets. The training sample for the CNN model was generated by OBIA using thresholds in the canopy height model (CHM) and the NDVI. Finally, the tree's heatmap was refined using OBIA segments. Many authors attempted to develop frameworks to optimize decision-level fusion techniques. For example, the OUDN model was developed by He et al. (2020). It is made up of three parts: the first part combines the improved U-net and Dense-Net architectures; the network is then trained using labeled data sets to sort the land use information in the study area; the last part combines object boundary information from multiresolution segmentation into the classification layer; and a voting method is used to find the best classification results. Correspondingly, Bengoufa et al. (2021) applied the OCNN approach to extract shoreline from VHR imagery;

their approach offered an immediate insight in terms of the extraction of rocky shoreline. Ghorbanzadeh et al. (2021) prepared a manually labeled dataset using various VHR images collected over two years in a dynamic refugee camp (2015 and 2016). They then established a CNN model based on the input window size patches of the applied RS images. Next, they applied the OBIA classification, a rule set that integrates expert knowledge, to the probability values of various dwellings and non-target classes, taking into account the CNN results. Finally, they applied an object-based accuracy assessment to validate the model's performance. Also, Ghorbanzadeh et al. (2022b) tried to improve OCNN modeling by filtering deep features derived from the CNN model using objects that had been segmented by MRS. This model aimed to enrich spatial features with elevation information extracted from DEM and others (i.e., spectral and spatial features). Their method provided a more accurate LULC map than conventional methods. Urban planning is considered a typical application that employs RS imagery. For instance, Zhang et al. (2022) developed a novel framework to optimize the accuracy of urban functional zone identification from VHR satellite images. The first step in their framework is to improve the OCNN model using the equal-area dividing algorithm. The second step is to classify urban functional zones using the new Distance Weight-Graph Attention Model. And the third step is to improve the result using MRS segmentation and the majority filter technique. In general, the proposed framework optimizes the accuracy of urban functional zone identification compared to state-of-the-art methods. The main challenge of this model is its computational complexity. Thomas et al. (2023) used multiple RS data (i.e., Sentinel 1, Sentinel 2, LandSat 8, DEM)

and the OCNN model for mapping debris-covered glaciers in Nepal, according to glacier ice investigations. The experimental findings showed that the proposed approach was more robust and flexible in terms of the detection of debris-covered glaciers compared to the existing methods. Their method didn't fully exploit OBIA's capabilities during the extraction of debris-covered ice areas. In terms of flood studies, Wang et al. (2023) proposed a hybrid OCNN model for flood-affected area detection using multiple RS images, such as GF2 and GF3; Their model integrates MRS segmentation and the CNN model to extract LULC, and then filters the classification results using Markov random field segmentation. According to their findings, it is evident that the hybrid model has a higher capability of discriminating between non-water and water areas compared to traditional thresholding methods. The hybrid model enhanced the accuracy by 3–5%. This framework requires careful initialization of parameters that usually impact the final classification. A different study by Ghorbanzadeh et al. (2023) about the disaster management model tried to improve the accuracy of finding landslides by combining MRS with an advanced CNN model called ResU-Net and using imagery sources like the Sentinel-2 image and DEM. Their results indicated that the MRS-ResU-Net model has enhanced the accuracy of landslide mapping from the traditional ResU-Net model by more than 22%. This method does not effectively employ OBIA facilities, as no features are used for classification. For another application, Ameslek et al. (2023) proposed an OCNN model that includes the integration between SLIC and CNN for tree detection and classification. Their model's high performance, achieved by filtering deep learning classification based on segmentation, produces an accuracy higher than the existing

methods. Super-pixels usually do not align perfectly with irregular-shaped objects in the images. Recently, decision-level fusion models have become widespread due to their ease of application and their remarkable results. Due to the flexibility of the decision-level fusion technique, experts developed advanced frameworks that include multiple OBIA-DL frameworks, as shown by Vinod et al. (2023), who proposed a framework that runs the classification based on OBIA features, then the classification's result is filtered using the decision-level fusion framework. Their method has improved the classification of trees outside forests compared to traditional OBIA by 9%. In a separate study, Lu et al. (2024) used a framework that includes multiple OBIA-CNN stages; their framework aimed to improve LULC classification by enhancing classification results based on OBIA segmentation. Chen et al. (2024) explore the application of the OCNN model in the classification of wetland plants from very high-resolution aerial photos. Through the integration of MRS and DeepLab3 models, the integration between spatial and spectral information has enhanced the extraction and classification of wetland plants. On the other hand, Abbasi et al. (2024) presented a framework that includes the integration between MRS and two algorithms (DenseNet-LSTM and DenseNet-Conv1D) for the improvement of multi-temporal change detection. Their results showed that the integration of MRS-DenseNet-Conv1D outperformed the integration of MRS-DenseNet-LSTM in terms of accuracy. Although the decision-level fusion framework is simple in terms of construction and application, the main problem with this method is that it does not fully utilize the OBIA method, as no features are used for classification. Furthermore, feature learning and segmentation optimization are conducted individually.

2.5.4 Patch Filtering (Deep Learning with Filtered Patches)

The fourth technique is the heterogeneous segment filter. In a traditional CNN, the image patch not only includes the land cover category to which it belongs but also contains additional land cover information. This additional information affects CNN's decision, as the output will reflect the more obvious land cover types. Studies have attempted to filter and refine the image objects within each image patch to achieve heterogeneous segments before classification tasks (Pan et al., 2019, Fu et al., 2018). Liu et al. (2018 a) presented a novel approach to improving the classification of wetlands in the United States using multi-view UAV orthophotos, CNN-based adaptive patch windows, and OBIA classification. Despite their well-performing framework for wetland detection, multi-view data are considered time-consuming and expensive. Similarly, Fu et al. (2018) created a model that combined multi-resolution segmentation, the center of gravity, and CNN algorithms. Their results made it easier to identify objects that were segmented unevenly by a very high-resolution remote sensing imager. In addition, their approach successfully reduced the uncertainty associated with OBIA during classification in China. The challenge remains to best map image objects into image patches. Liu et al. (2018 b) evaluated the implementation and compared the CNN-OBIA model and traditional machine learning models (i.e., support vector machines (SVM), neural networks (ANN), and random forests (RF)) in wetland classification and mapping in the United States. They found that the CNN-OBIA integrated model achieved an accuracy higher than traditional models. In 2019, Pan et al. presented a novel methodology known as object-based heterogeneous filter and deep CNN (OHSF-CNN), which aims to enhance the integration process

of CNN, OBIA, and fuzzy logic classification. The experimental results demonstrate that this method not only fully utilizes the recognition capabilities of deep learning methods but also effectively addresses issues such as jagged errors at boundaries and the expansion or shrinkage of land cover areas that originate from traditional CNN structures. Moreover, the proposed OHSFCNN can achieve higher classification accuracy compared with the traditional methods. However, this approach is more complex than others. Fu et al. (2019) developed an approach based on integrating FCNN and OBIA classification (i.e., multiresolution segmentation). The CNN algorithm was pre-trained based on transfer learning techniques. They selected the optimal patch size to be integrated with image segments, and then they labelled segments using the majority overlapping region method. Then feature extraction was implemented. Ji et al. (2019), on the other hand, developed a methodology that integrates R-CNN-OBIA and R-CNN-Pixel-based classification methods to improve OBIA-Pixel-based classification techniques for change detection. Wang et al. (2020) developed an object-scale adaptive CNN (OSA-CNN) model, which is divided into two main modules. The first one is the image segmentation and classification module. The object primitives at a set of scales are obtained through HBC-SEG. Then, the object primitives are mapped into image patches using adaptive patch sampling along the object primitive axes, and the results are classified using CNN. These processes are conducted at all scales to obtain classification results. The second one is the multiscale classification fusion module. Each image patch's maximum classification probability scale (MCPS) is determined according to the classification probability comparisons at different scales, and the multiscale

classifications are fused into one result. Finally, they classified each object primitive using a majority vote based on the image patch classification results, which allowed them to obtain the overall image classification result. Object primitive/image patch conversion, CNN network modification, and multiscale classification fusion are the critical links for fulfilling the proposed method. While improving traditional CNN methods, this technique does not use OBIA features, which may add additional information and boost classification accuracy. Generally, this framework aims to improve the model's ability to classify the precise edge of ground objects correctly with some filtering methods that can be applied to image patches or image objects.

2.5.5 Patch Filtering (Deep Learning with Context Patches)

Traditional deep learning models (for example, CNN) operate on rectangular image patches of a fixed size (e.g., 24×24). While these techniques outperform traditional pixel-based methods and OBIA alone, they are ineffective at accurately extracting object boundaries. As a result, several studies have proposed using context patches created by object centres, random points within image objects, object skeletons (Martins et al., 2020), or, more recently, region-based voting methods. The idea behind these new methods is to use image objects generated by a segmentation algorithm to create image patches from which a deep learning model can extract features and then perform classification. Studies have shown that such methods can more accurately classify data at object boundaries. Furthermore, such methods have a lower computational cost than traditional rectangular patch-based methods. The segmentation step, on the other hand, has a notable

effect on the accuracy of the preceding methods. In most cases, the size of objects delineated from remote sensing imagery varies widely, resulting in large object representations that fail to capture small ground objects such as urban trees, small buildings, and bridges on the water. Martins et al. (2020) tested integration between CNN and multiscale object-based methods for image classification at the regional level and in heterogeneous landscapes. For CNN predictions, they used a skeleton-based algorithm to extract convolutional positions. Iowa-Net, the method on their newly developed datasets, presented a classification accuracy of 87.2%, which is considered better than other methods such as fixed-input (OCNN) and patch-based CNN, which achieved accuracy of 81.6% and 82%, respectively. Misclassification was detected in some classes, such as shadow versus lake or road versus buildings. The multi-OCNN approach's main limitations are that it is influenced by the number of image bands (i.e., aerial photos) and the quality of segmentation. In another study, Li et al. (2021) proposed a scale sequence object-based CNN (SS-OCNN) that classifies images at the object level. These segmented objects were subsequently classified using a CNN model integrated with an automatically generated scale sequence of input patch sizes. This scale sequence can effectively fuse the features learned at different scales by progressively transforming the information extracted at small scales to larger scales. Experimental results revealed that the SS-OCNN consistently achieved the most accurate classification results. This framework is sensitive to segmentation. In a related topic, Lv et al. (2021) created a model using the improved object-based convolutional neural network (IOCNN) to sort very high-resolution image sources using convolutional position sampling and

zone-division methods. This model is able to classify objects that have irregular shapes. The final result indicated that the IOCNN model is considerably more accurate than state-of-the-art models. The IOCNN model achieved classification accuracy of 91.65% and 93.49% on two different images. This framework often has a lower computational cost compared to patch filtering. At the object level, there are also a number of options for extracting patches, such as the center, multipoints, and object skeletons. However, the main limitation of these methods is the impact of segmentation quality on extracting image patches and, ultimately, the accuracy of the classification result.

Table 2.5 : The Summary of Literature Review

Title	Ref.	OBIA-DL Framework	Dataset	Advantage	Drawbacks
Using a VGG-16 Network for Individual Tree Species Detection with an Object-Based Approach	Rezaee et al., 2018	Deep Learning Based on Object-Level Features (MRS/VGG-16)	WV3	Improving the detection of trees by combining deep spectral and contextual features and tree boundaries.	Lacks the training data from other species of trees
Target-based Building Extraction from High-Resolution RGB Imagery using GEOBIA Framework and Tabular Deep Learning Model	Hossain et al., 2024	Deep Learning Based on Object-Level Features (Watershed / Neural Oblivious Decision Ensemble)	UAV	Merging spatial, spectral, and textural information enhanced the building's extraction and classification.	The complexity of transforming tabular information into imagery is because the proposed model lacks the ability to capture the spatial relationship between the table's cells.
Tradeoffs among multi-source remote sensing images, spatial resolution, and accuracy for the classification of wetland plant species and surface objects based on the MRS_DeepLabV3+ model	Chen et al., 2024	Decision-Level Fusion (MRS / DeepLabV3)	UAV, GF-1 ZY3	Providing a novel framework for wetland plant classification depends on filtering the deeply extracted features using the segmentation technique.	This method didn't extensively employ OBIA features (i.e., feature extraction and optimization).

Table 2.5 : Continued

Title	Ref.	OBIA-DL Framework	Dataset	Advantage	Drawbacks
Assessment of Trees Outside Forest (TOF) in Urban Landscape Using High-Resolution Satellite Images and Deep Learning Techniques	Vinod et al., 2023	Hybrid-(Deep Learning Based on Object-Level Features And Decision-Level Fusion) (MRS / U-Net)	Cartosat-2 LISS-IV	Optimizing the extraction of tress outer borders through multiple refinements of objects using OBIA segments	The optimization of segmentation and classification is implemented separately, causing problems during the matching process between the classified feature and the counterpart segment.
An integrated deep learning and object-based image analysis approach for mapping debris-covered glaciers	Thomas et al., 2023	Decision-Level Fusion (MRS / CNN)	Landsat sentinel 1 DEM	The combining of precise edges extracted from OBIA technique and deep spectral, textural, and contextual features led to a notable improvement of the accuracy of the debris-covered ice's mapping.	- This method didn't fully exploit OBIA's capabilities during the extraction of debris-covered ice areas. - Limited spatial resolution
Combining Drones and Deep Learning to Automate Coral Reef Assessment with RGB Imagery	Giles et al., 2023	Deep Learning Based on Object-Level Features (MRS / U-Net)	UAV	This method easily learns contextual relationships among the OBIA attributes. This led to improvements in coral reef detection and classification.	This method lacks the ability to learn the spatial characteristics of the image content (pixels and objects).

Table 2.5 : Continued

Title	Ref.	OBIA-DL Framework	Dataset	Advantage	Drawbacks
Combining Deep Learning and Hydrological Analysis for Identifying Check Dam Systems from Remote Sensing Images and DEMs in the Yellow River Basin	Li et al., 2023	Feature-Level Fusion (MRS / U-Net)	WV2 DEM 5 m	The presented advanced framework utilizes additional deep features that have advantages over the traditional OBIA-DL integration technique. It was successful in pinpointing check dam systems through satellite imagery.	- It's a complicated framework due to the duplication of some features extracted by OBIA and CNN, such as shape, texture, and color, which required additional steps for the classification process. - Heavy computation and hardware resources are necessary.
Performance Comparison of Deep Learning (DL)-Based Tabular Models for Building Mapping Using High-Resolution Red, Green, and Blue Imagery and the Geographic Object-Based Image Analysis Framework	Hossain et al., 2024	Deep Learning Based on Object-Level Features (MRS /CNN)	UAV	This method easily learns contextual relationships among the OBIA attributes compared to shallow learning methods in terms of the detection of urban buildings.	This method lacks the ability to learn the spatial characteristics of the image content (pixels and objects).

Table 2.5 : Continued

Title	Ref.	OBIA-DL Framework	Dataset	Advantage	Drawbacks
Object-Based Semi-Supervised Spatial Attention Residual U-Net for Urban High-Resolution Remote Sensing Image Classification	Lu et al., 2024	Hybrid-(Deep Learning Based on Object-Level Features And Decision-Level Fusion) (MRS / U-Net)	Aerial photo	The implementation of multiple filtering on classified objects resulted in enhanced urban LULC classification.	In this method, the optimization process for both OBIA segmentation and CNN feature learning is individually implemented, which results in a knowledge gap between the segmentation and the feature learning.
Spatial Resolution as a Factor for Efficient UAV-Based Weed Mapping—A Soybean Field Case Study	Ubben et al., 2024	Feature-Level Fusion (SLIC / CNN)	UAV	This framework reinforced the feature learning level via the fusion process, with features resulting from OBIA techniques.	- The duplication of some features extracted from OBIA and CNN makes this framework time-consuming and requires heavy computation and hardware resources. - The segmentation is unrealistic.
A Novel UNet 3+ Change Detection Method Considering Scale Uncertainty in High-Resolution Imagery	Bai et al., 2024	Deep Learning Based on Object-Level Features (MRS / UNet3)	GF-2	In terms of change detection application, it is considered a simple framework that easily learns contextual relationships among the OBIA attributes.	This method fails to learn the spatial characteristics of the image content, including pixels and objects. Furthermore, it neglects the powerful capability of deep learning for extracting spatial and abstract features from the image data.

Table 2.5 : Continued

Title	Ref.	OBIA-DL Framework	Dataset	Advantage	Drawbacks
Waterlogged Area Identification Models Based on Object-Oriented Image Analysis and Deep Learning Methods in Sloping Croplands of Northeast China	Xie et al., 2024	Deep Learning Based on Object-Level Features (MRS / DeepLabV3+)	Jilin-1	The model's ability to analyse the ambiguity at edges, minimized image noise, and optimized the continuity between terrain objects.	This method lacks learning spatial characteristics of the image content (pixels and objects).
The application of ResU-net and OBIA for landslide detection from multi-temporal Sentinel-2 images	Ghorbanzadeh et al., 2023	Decision-Level Fusion (MRS/ ResU-Net)	Sentinel2DEM	It is a simple method in terms of construction and application.	This method does not fully utilize the OBIA method, as no features are used for classification.
Utilization of Improved Annotations From Object-Based Image Analysis as Training Data for DeepLab V3+ Model: A Focus on Road Extraction in Very High Resolution Orthophotos	Sussi et al., 2024	Deep Learning Based on Object-Level Features (SLIC / DeepLab V3+)	Aerial photo	This framework provides a simple OCNN framework that efficiently extracts roads from remote sensing data.	This framework does not learn the spatial characteristics of the image content (pixels and objects).

Table 2.5 : Continued

Title	Ref.	OBIA-DL Framework	Dataset	Advantage	Drawbacks
Identification and mapping of Individual Trees from Unmanned Aerial Vehicle Imagery Using an object oriented Convolutional Neural Network	Ameslek et al., 2023	Decision-Level Fusion (SLIC / CNN)	UAV	The system provides a rapid OCNN framework to detect trees from VHR imagery sources.	• SLIC segmentation is very delicate in terms of the number of superpixels. - SLIC creates a uniform size of segments that may mismatch some objects in reality.
Multi-temporal change detection of asbestos roofing: A hybrid object-based deep learning framework with post-classification structure	Abbasi et al., 2024	Decision-Level Fusion (MRS / DenseNet-LSTM DenseNet-Conv1D)	Aerial photo	This framework improved the change detection process due to its ability to learn long-term dependencies in sequential data in addition to the precise spatial information derived from the OBIA technique.	It's a complicated method due to the manual selection of segmentation parameters.
An object-based and heterogeneous segment filter convolutional neural network for high-resolution remote sensing image classification	Pan et al., 2019	Deep Learning with Filtered Patches (Quickshift / CNN and Fuzzy neighborhood vector)	Aerial photos	The application of the fuzzy model improved pixel grouping in a segment, resulting in precise ground edges.	The challenge remains to best map image objects into image patches.

Table 2.5 : Continued

Title	Ref.	OBIA-DL Framework	Dataset	Advantage	Drawbacks
Object-scale adaptive convolutional neural networks for high-spatial resolution remote sensing image classification	Wang et al., 2020	Deep Learning with Filtered Patches (HBC-SEG /CNN)	Aerial photos	It aims to improve the model's ability to classify the precise edge of ground objects correctly based on the application of multi-sized patches.	The challenge remains to best map image objects into image patches.
A Scale Sequence Object-based Convolutional Neural Network (SS-OCNN) for crop classification from fine spatial resolution remotely sensed imagery	Li et al., 2021	Deep Learning with Context Patches (Scale Sequence Object-based (MRS) /CNN)	Rapideye SAR data	This framework applies a scale sequence of input patch sizes, which can effectively fuse the features learned at different scales by progressively transforming the information extracted at small scales to larger scales.	Individual optimization can cause classification accuracy problems due to the knowledge gap between segmentation and feature learning.
Improved object-based convolutional neural network (IOCNN) to classify very high-resolution remote sensing images	Lv et al., 2021	Deep Learning with Context Patches (MRS /CNN)	GF2	The models often have lower computational costs compared to patch filtering. There are also a number of options to extract patches at object-level including center, multi-points, and object skeletons.	The major limitation of these methods is the impact of segmentation quality on extracting image patches and ultimately the accuracy of classification result.

2.6 Discussions and Research Gaps

As discussed in the literature, the integrated OBIA-deep learning frameworks can have advantages over traditional methods in classifying remote sensing images, in particular the semantics of the objects in the output classification maps. A number of frameworks and methods have been developed to optimize the integration process of OBIA and deep learning. These methods include deep learning that relies on object-based features, feature-level fusion, decision-level fusion, filtered patches, and context patches. Table 2.4 shows the summary of OBIA-DL integration frameworks.

The first framework is training a deep learning model on OBIA features; this framework is considered a simple method that easily learns contextual relationships among the OBIA attributes. The first method's problem lies in representing the OBIA features as tabular data; deep learning methods were previously thought to be ineffective for tabular data (Mayr et al., 2018). Deep learning models like CNNs lack the ability to capture the spatial relationship between the table's cells (Hossain et al., 2024a). The problem with integrating deep learning with object-level features represented as tabular data is that (1) OBIA features are extracted manually, many of which may be irrelevant to the classification task at hand; (2) under- and over-segmentation have a significant impact on the features calculated; and (3) the spatial characteristics of image pixels are not effectively represented in the classification model. Using these methods, it is also difficult to handcraft abstract features (edges, textures) that may be useful for classifying image data.

The second framework, feature fusion, can be performed to make use of both OBIA-based features and deep features, which may help to improve the classification results.

It is frequently implemented as a two-branch computational network, which includes a processing chain for segmentation and OBIA feature extraction, as well as a network for learning deep and abstract features from the data. After combining the two feature sets, a classifier such as tree-based models or Support Vector Machines (SVM) is used to obtain class labels for the image pixels (Zhao et al., 2017). However, this framework introduces an advanced approach that utilizes additional deep features, offering advantages over the previous framework. This framework's basic problem is heavy computation (Guirado et al., 2021). Other issues with this integration method include duplication in some features extracted by OBIA and CNN, such as shape, texture, and color, which makes this framework complicated and time-consuming.

The third framework is the decision-level fusion, which is implemented as a deep learning model to establish a classification map of the study area first. The results are then refined by majority voting based on segmentation. Each object contains several pixels with class labels predicted at the pixel or patch level by a deep learning model. Finally, each object is assigned a single label depending on the most frequently occurring class labels within that object (Abbasi et al., 2024). While studies have highlighted the significance of decision-level fusion techniques, which considered a simple framework in

terms of construction and application. this framework does not fully utilize the OBIA method, as no features are used for classification (Thomas et. al., 2023).

Another framework, known as patch filtering, allows for the filtering of image patches created to train deep learning models based on the statistics of OBIA segmentation, before they are fed into the models. This framework aims to improve the model's ability to classify the precise edge of ground objects correctly based on the application of multi-sized patches (Wang et al., 2020). The problem with these methods is not taking advantage of the advantages of the geometric and spatial properties of OBIA segments. The other problem lies in the computational cost of these methods, as they filter each patch individually based on segmentation objects (Wang et al., 2020). Other studies have attempted more efficient methods, such as context patches. These methods generate image patches based on OBIA segmentation under different settings, which include the center of the objects, random points within the objects, and patches over different points around the object skeleton (Lv et al., 2021). According to the literature review, such methods can more accurately classify data at object boundaries. Furthermore, such methods have a lower computational cost than traditional rectangular patch-based methods. The segmentation step, on the other hand, has a notable effect on the accuracy of the preceding methods. Objects delineated from remote sensing imagery vary widely in size in most cases, resulting in large object representations failing to capture small ground objects (e.g., urban trees, small buildings, bridges on the water).

The pixel-based CNN algorithm that depends on image patches often weakens the boundary information of the land use, somewhat like the Gabor filter or morphological methods. In these methods, blurred boundaries occur between the classified objects, resulting in a loss of useful land features. This problem can be overcome with object-based CNNs. Object-based deep learning classification demonstrates a strong ability for classifying complex land uses through deep feature representation and maintaining the fine spatial details of image objects (Wang et al., 2020, Zaabar et al., 2022).

Accurate segmentation is very important for learning effective spatial features that can easily distinguish different land-use classes, especially for methods that are based on context patches, which are extracted based on the object center or random points within objects. The way these objects are created determines the image patches that are used for feature extraction by the deep learning models. In optimal segmentation, there is a need to ensure that small objects are segmented while also preventing large objects from being over-segmented. Furthermore, the MRS algorithm's parameters can be used to control the boundary of image objects through shape and compactness. However, object-based CNNs require innovative use of appropriate functional units and convolutional processes based on image segmentation. Despite this, the model's computation efficiency can be significantly improved. Some studies have attempted to overcome this and related issues by using the appropriate distribution of voting points within image objects with geometry conditions or advanced analysis of convolutional processes.

In general, this study detects some research gaps:

- There have been no assessment studies that compare the effectiveness of each of the methods listed above. Nevertheless, different streams of study are being pursued to improve the performance of each method.
- To our best knowledge, all integrated OBIA-deep learning frameworks have optimized the feature learning process and the segmentation parameters separately. The knowledge gap between the feature learning and segmentation processes frequently leads to misclassification in some areas. Therefore, the joint optimization technique can provide appropriate solutions by bridging the knowledge gap among CNN and OBIA models. Furthermore, almost all these studies employed a grid and random search for the optimization of parameters or hyperparameters, where the grid search is very time consuming and required a heavy computational resources and the random search doesn't guarantee the attainment of optimal results. The Bayesian technique can provides solution to find the optimal parameters/hyperparameters with less time and efforts.
- Rare studies have implemented the combination of OBIA and deep learning in road extraction from remotely sensed data, particularly in dense urban areas, which presents a significant challenge in examining these frameworks due to the difficulty of delineating road boundaries within the complex area.

To that end, this study presents three parts. Based on the existing research gaps, this study offers the first comparative evaluation of the most common integrated OBIA and deep learning frameworks. Furthermore, this study presents an innovative classification framework based on decision-level fusion to resolve the fundamental issue with methods in this category, namely, that the “segmentation step is independent of feature extraction and classification.”

For accurate image classification, the proposed model learns the best segmentation parameters and high-level features jointly. On the other hand, the innovated framework has been validated on the road extraction in dense urban areas. Table 2.6 shows the summary of OBIA-DL integration methods.



Table 2.6 : The Summary of OBIA-DL Integration Methods

OBIA-DL Framework	Pros	Cons	References
Training Deep Learning Models on OBIA Features	It is considered a simple method that easily learns contextual relationships among the OBIA attributes.	This method lacks the ability to learn the spatial characteristics of the image content, including pixels and objects. Moreover, it neglects the powerful capability of deep learning for extracting spatial and abstract features from the image data.	Jozdani et al. (2019) Abdollahi et al. (2021), Lam et al. (2020)
OBIA-Deep Learning Features Fusion	This method presents an advanced framework that utilized additional deep features that offer advantages over the first method.	This method requires heavy computation and hardware resources. Furthermore, it includes duplication in some features extracted by OBIA and CNN, such as shape, texture, and color.	Zhao et al. (2017), Majd et al. (2019), Sutha et al. (2020), Hong et al. (2020), Tang et al. (2020)
Decision Level Fusion	It is a simple method in terms of construction and application.	This method does not fully utilize the OBIA method, as no features are used for classification.	Guirado et al. (2021) Lv et al. (2018), Liu et al. (2019), Abdi et al. (2018), Robson et al. (2020), Timilsina et al. (2020), He et al. (2020).
Deep Learning with Filtered Patches	It aims to improve the model's ability to classify the precise edge of ground objects correctly with some filtering methods that can be applied to image patches or image objects. The models often have lower	The challenge remains to best map image objects into image patches.	Pan et al. (2019), Fu et al. (2019), Liu et al. (2018), Liu et al. (2018), Fu et al. (2019), Ji et al. (2019), Wang et al. (2020)
Deep Learning with Context Patches	computational costs compared to patch filtering. There are also a number of options for extracting patches at the object level, such as the center, multi-points, and object skeletons.	The major limitation of these methods is the impact of segmentation quality on extracting image patches and, ultimately, the accuracy of the classification result.	(Martins et al.2020), Li et al. (2021), Lv et al. (2021)

CHAPTER 3

METHODOLOGY

This chapter presents a description of the study area, datasets, pre-processing, the proposed methodology and benchmark models for land cover mapping using very high-resolution satellite imagery (WV-3), and the evaluation process. Figure 3.1 illustrates the structure of the overall methodology, which primarily comprises three sections to achieve the study's objectives. The first section of this chapter describes the comparative assessment of the most common OBIA-DL integration frameworks; the second section describes the innovative framework that includes the integration of OBIA-DL and Bayesian optimization techniques; and the third section describes the framework's validity for a specified application (i.e., road extraction).

3.1 Study Area

The satellite image used in this study was obtained over the University Putra Malaysia (UPM) campus in Selangor, Malaysia. The area has various land cover types, including bare lands, grasslands, water bodies, roads, buildings, and dense vegetation and trees. Roads and buildings are the most dominant land-cover classes in the area (Figure 3.2). The training and test areas encompass 0.369 km² and 0.254 km², respectively. Figure 3.3 illustrates the training and test areas selected from the study area image. In addition, it depicts some examples of these land-cover types within the study area.

The ground-truth data was created based on visual inspection and manual digitization. Figure 3.4 depicts the ground-truth data for the training and test areas. The area has six land-cover types: bare land, grassland, roads, buildings, dense vegetation/trees, and water bodies. Table 3.1 shows the ground-truth dataset's training and test data class distribution (i.e., number of pixels and area percentage). The total number of pixels within the training and test areas is 1,055,425 and 603,597, respectively.

3.2 Dataset and Tools

The Digital Globe captured the Worldview 3 (WV3) image in November 2014. The spatial resolution of WV-3 image data is 0.31 m for the panchromatic band and 1.24 m for the multispectral bands. WV-3 includes eight spectral bands with a radiometric resolution of 11 bits each: coastal, yellow, green, blue, red, red edge, near-infrared 1 (NIR1), and near-infrared 2 (NIR2). WorldView-3 Satellite Sensor Specifications are provided in Appendix (B).

The proposed methodology was implemented using a variety of software and platforms, including the Exelis Visual Information Solutions (ENVI) software for the pre-processing step and the Python-based platform (PyCharm) with various libraries such as Keras, Scikit-learn, and SciPy for the comparative assessment of OBIA deep learning frameworks and the development of an innovative framework. On the other hand, the software ArcGIS Pro is used for producing maps and the image registration process.

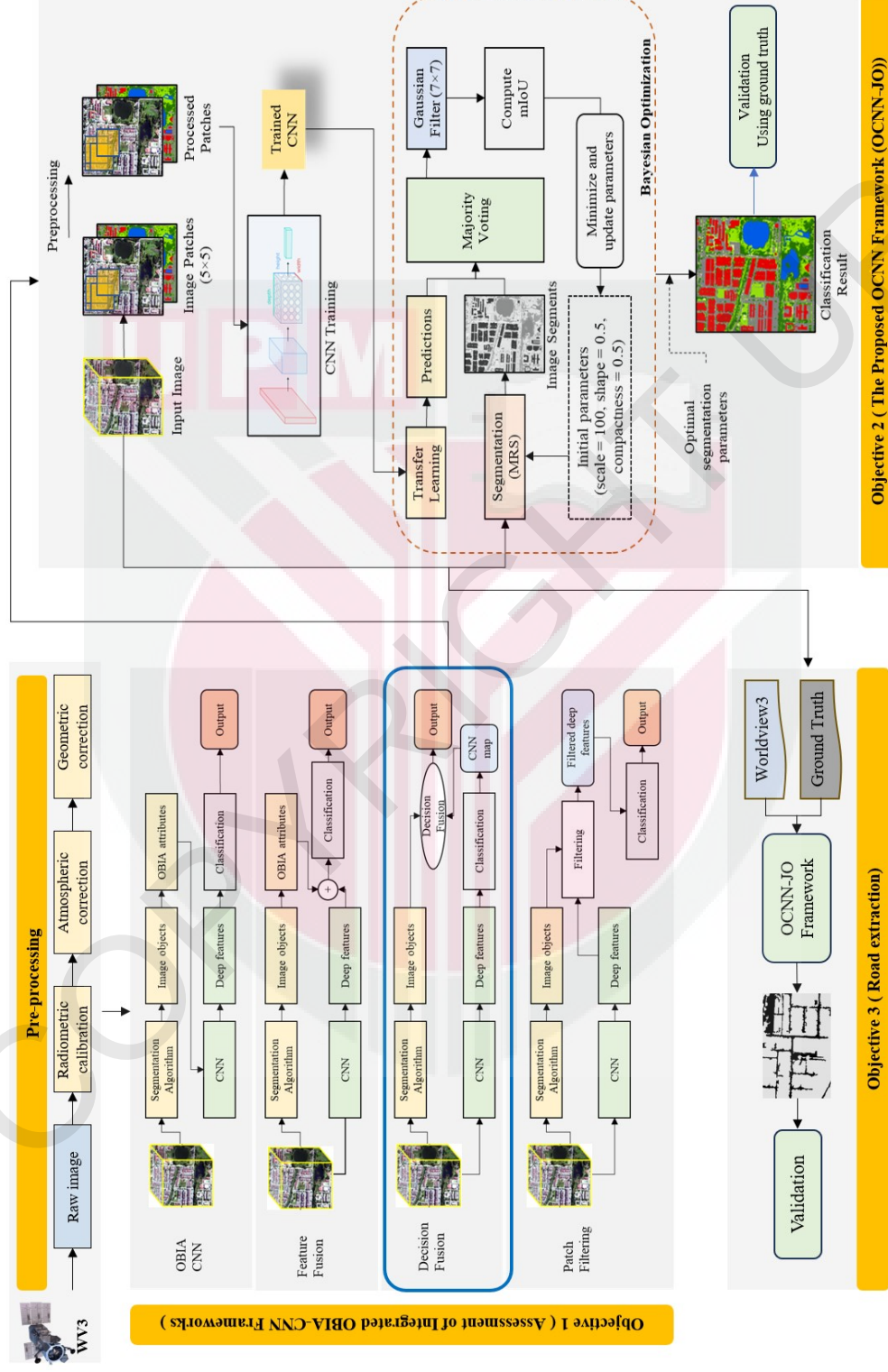


Figure 3.1 : The Overall Methodology

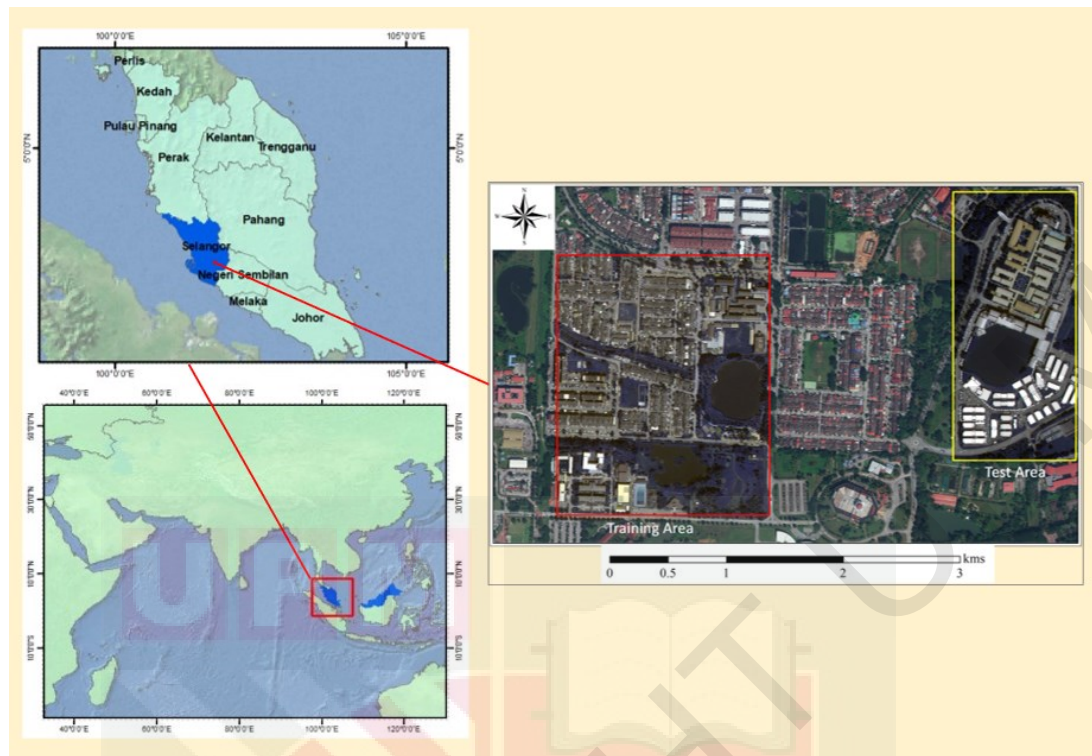


Figure 3.2 : The Study Area

Table 3.1 : Training and Test Data Class Distribution (i.e., Number of Pixels and Area Percentage) in the Ground-Truth Dataset

Land-Cover Class	Pixels		Percentage	
	Training Area	Test Area	Training Area	Test Area
Buildings	275,441	144,260	26.10%	23.90%
Roads	288,816	191,106	27.36%	31.66%
Grassland	168,018	87,707	15.92%	14.53%
Dense Vegetation/Trees	215,104	111,872	20.38%	18.53%
Water Body	85,762	54,183	8.13%	8.98%
Bare Land	22,284	14,469	2.11%	2.40%
SUM	1,055,425	603,597	100.00%	100.00%



Figure 3.3 : The training and test areas are indicated in the Worldview-3 true color composite of the study area

According to the WV3 image, the high cost posed a significant barrier to this research. As a result, we recommended an archival image for this study. Furthermore, using WV3 in dense urban areas presents unique challenges, such as spectral heterogeneity and shadow and occlusion effects. Conversely, the ground truth derived from the image has its own set of limitations. Compared to advanced ground truthing methods, the image's ground truth suffers from certain limitations. Several factors, such as atmospheric conditions and sensor calibration, could influence the spectral resolution of the image.

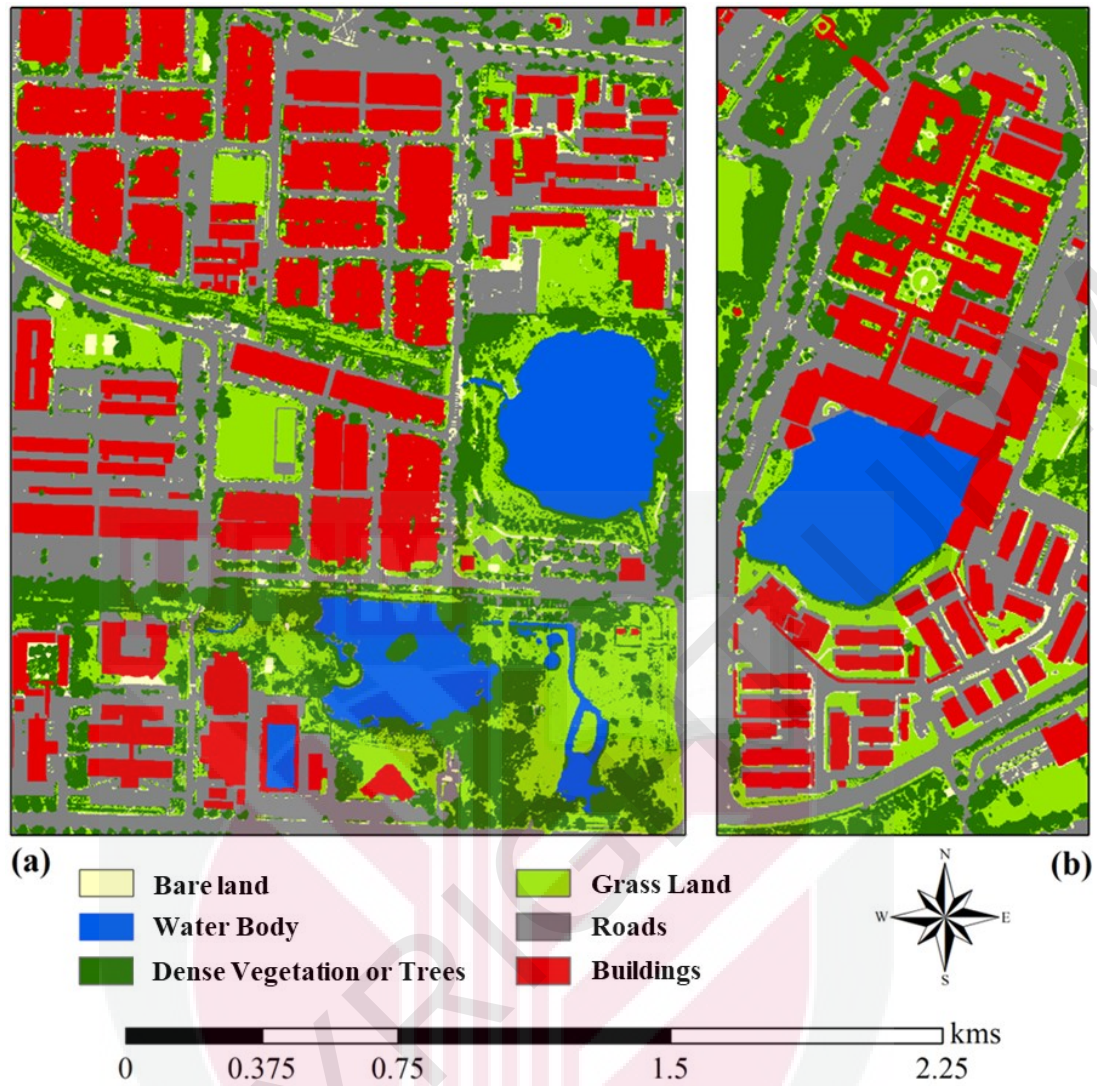


Figure 3.4 : The Ground-Truth Maps of the (a) Training and (b) Test Areas

3.3 Data Pre-Processing and Preparing

Generally, we subjected WV-3 image data to standard pre-processing steps like radiometric and atmospheric corrections (Russ 2006), but the data provider already performed geometric correction and registration. We employed the software Exelis Visual Information Solutions (ENVI) to perform both radiometric calibration and atmospheric correction. ENVI includes several tools that specialize in processing various types of remotely sensed data like multispectral, hyperspectral, LiDAR, and RADAR.

To correct errors caused by sensors or satellites, radiometric correction is necessary. This study used ENVI's radiometric calibration tools to import a WV3 image with auxiliary data, such as the metadata file, and then select radiance as the calibration type with other default options. This process transforms each pixel's digital number into a radiance value. On the other hand, to compensate for atmospheric degradation, this study employed the atmospheric correction process by using ENVI's quick atmospheric correction module (QUAC), which converts image pixel values to surface reflectance.

The geometric correction task is necessary to correct geometric distortions in the image, including the earth's rotation and curvature during the image capture. The data provider previously performed the geometric correction and image registration for this study. This study projected the data to the Universal Transverse Mercator World Geodetic System 84 datum.

3.4 Assessment of OBIA-Deep Learning Integration Frameworks

3.4.1 OBIA

3.4.1.1 Image Segmentation

In this study, the image segmentation was performed based on multiresolution segmentation (MRS) because it's the most widely used segmentation algorithm for classifying remote sensing images. The MRS algorithm begins with individual pixels and groups them until it satisfies the predefined parameters or reaches a stopping criterion (Munyati et al.,2018). The merging cost function, as shown in Equation (3.1), integrates spectral and shapes

heterogeneity.

$$f = w \times h_{color} + (1-w) \times h_{shape} \quad (3.1)$$

Where f is the fusion factor, w belongs to the weight for spectral heterogeneity with the interval 0–1, and h_{shape} and h_{color} refer to shape and color parameters, respectively.

$$h_{color} = \sum_i w_c \left(n_{obj3} \cdot \sigma_c^{obj3} (n_{obj1} \cdot \sigma_c^{obj1} - n_{obj2} \cdot \sigma_c^{obj2}) \right) \quad (3.2)$$

$$h_{shape} = w_{cmpct} \cdot h_{cmpct} + (1 - w_{cmpct}) \cdot h_{smooth} \quad (3.3)$$

$$h_{cmpct} = n_{obj3} \cdot \frac{l_{obj3}}{\sqrt{n_{obj3}}} - \left(n_{obj1} \cdot \frac{l_{obj1}}{\sqrt{n_{obj1}}} \right) + n_{obj2} \cdot \frac{l_{obj2}}{\sqrt{n_{obj2}}} \quad (3.4)$$

$$h_{smooth} = n_{obj3} \cdot \frac{l_{obj3}}{\sqrt{b_{obj3}}} - \left(n_{obj1} \cdot \frac{l_{obj1}}{\sqrt{b_{obj1}}} \right) + n_{obj2} \cdot \frac{l_{obj2}}{\sqrt{b_{obj2}}} \quad (3.5)$$

According to the above equations, $obj1$ refers to segment 1 and $obj2$ refers to segment 2; $obj3$ represents the result of merging segments 1 and 2; c is the index of the spectral band; w_c represents an arbitrary weight set for band c ; σ is the standard deviation; h_{cmpct} represents the component relative to compactness; h_{smooth} refers to the smoothness; l refers to the perimeter of the segment obj ; b refers to the perimeter of the corresponding minimum bounding box for $i = 1, 2, 3 \dots etc.$

MRS parameters are determined experimentally based on a trial-and-error approach (Shahabi et al., 2019). However, the algorithm's three parameters significantly impact the quality of output segmentation results. As a result, it is

critical to find the optimal or suboptimal values for these parameters with a systematic approach to ensure improved classification results.

The OBIA model, which included segmentation and feature calculation, was implemented using the PyCharm app, an integrated development environment (IDE) primarily designed for Python development and python libraries such as SciPy and scikit-image. The initial MRS parameter values were chosen based on a few primary experiments. The optimal parameters are scale, shape, and compactness (100, 0.5, 0.5), respectively.

3.4.1.2 The Feature Calculation

This research uses the standard features described above in the OBIA-related experiments. The feature extraction tool was implemented in Python based on libraries such as SciPy (<https://www.scipy.org>) and scikit-image (<https://scikit-image.org>). Several spectral, spatial, geometric, and textural features were calculated for image objects and used in the OBIA model.

The spatial features, including area, perimeter, elongation index, shape index, density, orientation, average width, average height, and rectangular fit, were computed using the scikit-image library. on the other hand, the spectral features like the number of pixels, minimum pixel's value, maximum pixel's value, mean pixel's value, the standard deviation, kurtosis, skewness, and the spectral value range were computed using the SciPy library. Additionally, we computed the textural features using the scikit-image library, which encompasses contrast, dissimilarity, homogeneity, energy, correlation, and

angular second moment. Appendix (A) provides the feature calculation process using Python coding.

3.4.2 CNN and Deep Neural Network

The standard CNN and ANN architecture employed to assess the integration of OBIA-CNN methods is shown in Figure 3.5 and Figure 3.6 which comprises seven layers, including the input and output layers. The image layer is used as input to the two-dimensional convolutional layers in order to learn the feature map. The extracted features are subjected to the next step, called the 2D max-pooling layer, which aims to repress unwanted information to optimize the computational efficiency of the framework. The extracted features from the learning process are then transformed into an individual feature vector using the flattening layer. After that, a dense (fully connected) layer is used to learn the contextual information from the features and assist the classification layer. The classification is performed with a softmax layer, and the output predictions are used to classify the given image.

CNN's modeling is performed using the Python libraries Keras and Scikit-learn. Training deep learning models such as deep ANN and CNN requires several hyperparameters that need to be configured carefully. In this research, the training parameters are set based on empirical experiments conducted on a subset taken from the whole dataset available for this research. The analyses revealed that some values were suboptimal for the proposed classification models and datasets used in the research. Table 3.2 summarizes the parameters used to train the ANN and CNN base models. For

the ANN, the optimizer is Nadam, which is Adam (Kingma and Ba 2014) with Nesterov momentum. The learning rate and learning rate decay were set at 0.001 and 0.001/100, respectively. The ANN models were trained for 500 epochs with early stopping criteria set to the patience of 15 epochs monitoring the validation loss. For the CNN, the Adam optimizer was found to be best with the same learning rate and learning rate decay as used for the ANN. The CNN models were also trained for 500 epochs with early stopping.

Table 3.2 : Parameters Used to Train the Base Models

Model	Optimization	Learning Rate	Decay	Epochs
ANN	Nadam	0.001	0.001/100	500
CNN	Adam	0.001	0.001/100	500

In addition, Table 3.3 presents the hyperparameters of the ANN and CNN models used in this research. For the ANN, a dropout with a rate of 0.5 was used after the first dense layer to control the overfitting in the network. We set the activation of the hidden layer to ReLU (Rectified Linear Unit), a common activation function in many deep learning models for remote sensing applications. The loss function that was used to train the models was categorical cross-entropy. A softmax layer was used for the classification layer. On the other hand, the CNN model had a few additional hyperparameters that needed to be configured to achieve the best classification results. The patch size of 5×5 was used as a sliding window size to extract image patches. The pool size in the max-pooling layers was set to 2×2 . The size of the convolutional kernels was 3×3 . The other configurations are the same as those for the ANN models.

Table 3.3 : Hyperparameters Used in the Base Models

Hyperparameter	Selected Value
Patch size	5×5
Pool size	2×2
Size of kernel filters	3×3
Dropout rate	0.5
Hidden layer activation	ReLU
Loss function	Categorical crossentropy
Early stopping patience	15 epochs
Classification layer	Softmax

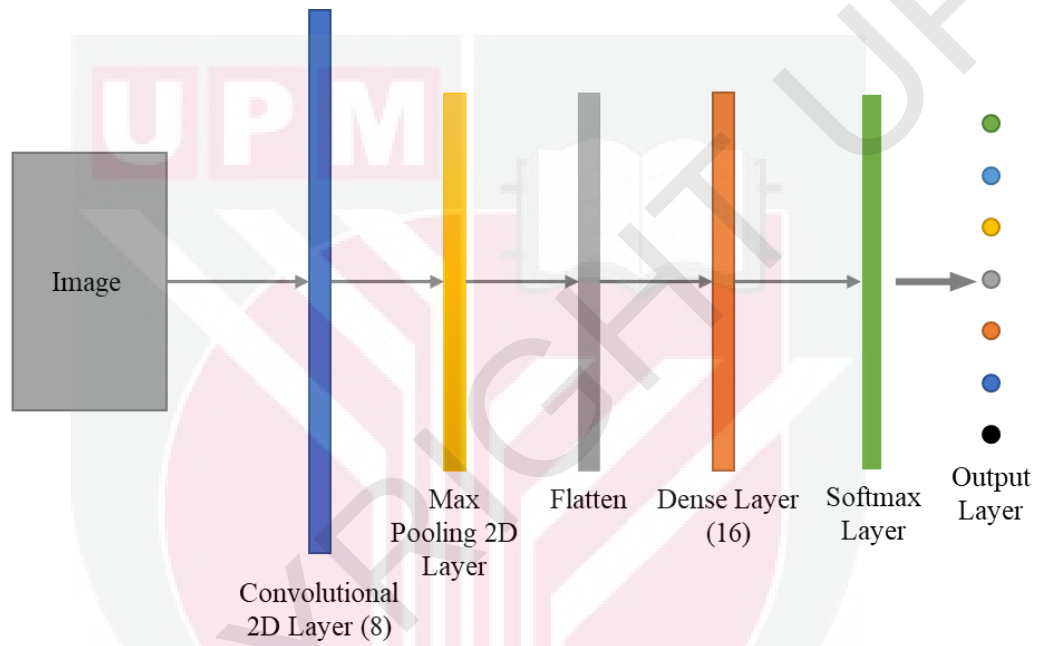


Figure 3.5 : CNN Architecture Used in integrated OBIA-Deep Learning Frameworks

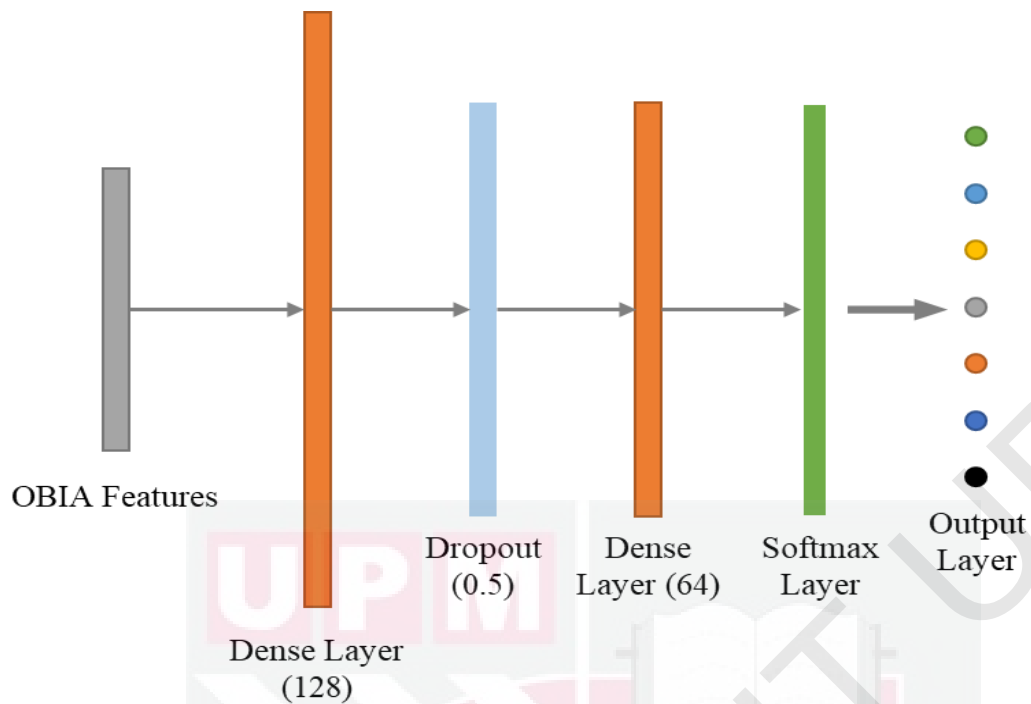


Figure 3.6 : ANN Architecture Used in Integrated OBIA-Deep Learning Frameworks

3.4.3 Evaluated Integration Frameworks

The literature review for this research identified four OBIA-DL frameworks (Figure 3.7). The following subsections provide a brief description of each framework :

3.4.3.1 OBIA CNN

This framework is the most basic integration of OBIA and deep learning. The current research uses the CNN algorithm to extract spatial, spectral, and textural features from OBIA features, which are calculated using a typical OBIA model that includes image segmentation and spectral-spatial-textural feature calculations. After that, OBIA features were organized in a tabular data structure, and the deep neural network is used to learn contextual information

that may be present among the features. Finally, a set of dense (fully connected) layers, followed by a SoftMax layer, performs the classification.

3.4.3.2 Feature Fusion

Another common framework for remote sensing classification is feature-level fusion for OBIA-CNN integration. This method uses OBIA to segment the image and calculate several spectral, spatial, and textural features. Similarly, CNN is used to abstract features from the data. The two obtained feature sets are then combined into a single feature vector. A classification layer obtains the class labels using the feature vector that combines the OBIA and CNN features. This method may produce redundant features, such as shape and textural features, as they can be shared among the OBIA and CNN features. However, the problem can be tackled with network regularization or dimensionality reduction techniques such as PCA.

3.4.3.3 Decision Fusion

This type of integration refines a classification map produced with a CNN or any other classification method based on image segmentation results. The majority filter technique is applied to make sure that each pixel in the image that belongs to the same image object is classified with the same label. This method is highly dependent on the accuracy of the CNN classification map and the results of the image segmentation. Other types of filtering may also be used, such as median and mean filters, instead of the majority filtering technique.

3.4.3.4 Patch Filtering

Patch-filtering frameworks aim to filter image patches based on OBIA segmentation before using them in CNNs. For instance, some filtering methods, such as variance filtering, can filter each image patch based on the dominant image object that covers or contains it. CNNs then use the filtered image patches, just as they do with traditional patch-based CNN methods.

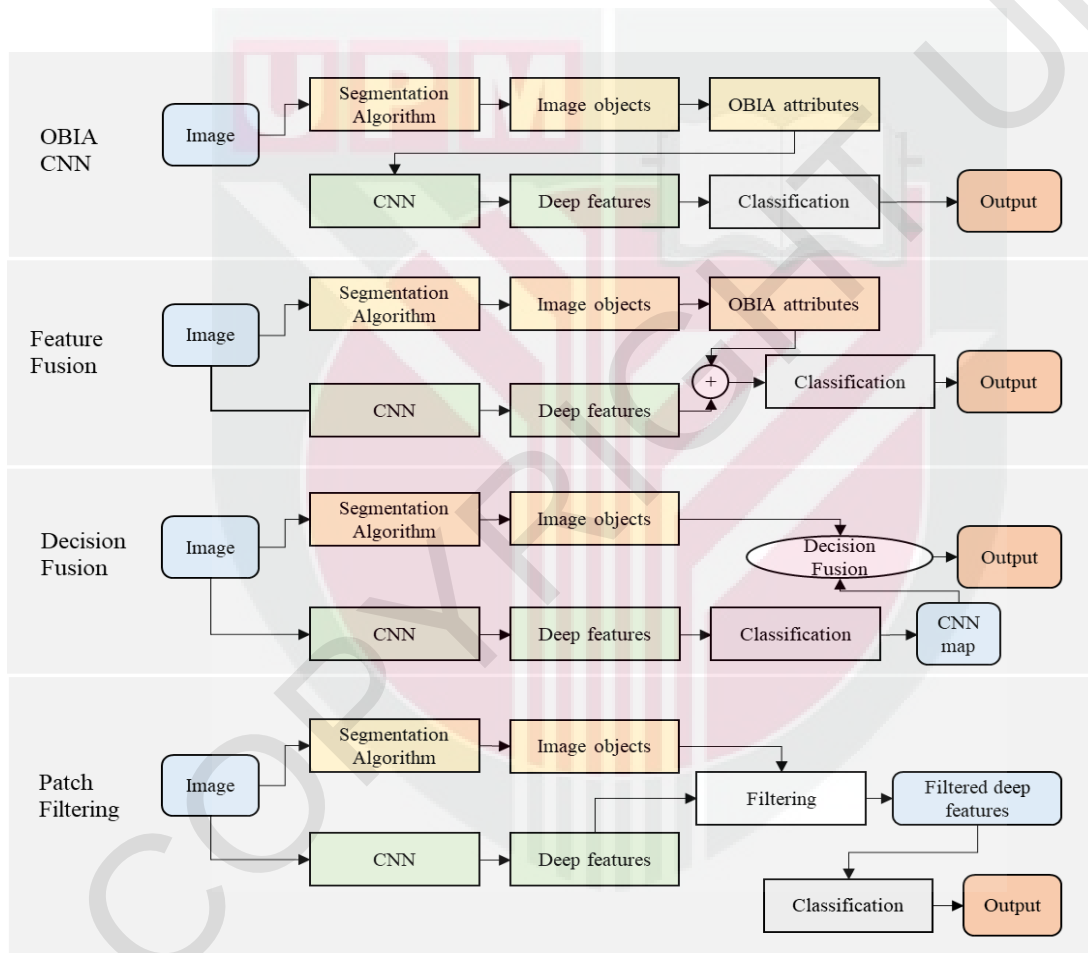


Figure 3.7 : Typical Architectures of the OBIA-CNN Integration Frameworks Identified in This Research

3.5 The Innovative OCNN Framework

3.5.1 The Proposed CNN

The innovative framework employs a specified CNN model consisting of several layers. The first step is to implement a gaussian noise filter to reduce the image's noise before feeding it into the CNN. The second step involves the creation of several patches in various sizes, including 3x3, 5x5, 7x7, and 9x9. These patches scan the entire image to capture various feature attributes at both local and global scales. Subsequently, three convolutional layers (2D CNN, 3D , and hybrid SN) were applied that apply filters to images to extract low- and high-level features. The CNN framework then generalizes feature maps by pooling layers until it produces high-level features (Schmidhuber et al., 2015). This framework uses batch normalization instead of the pooling layer to reduce training time and maintain the quality of image objects. The flattening technique then transformed the data from 2D or 3D form to 1D, making it easier to feed the data to the fully connected layers. These layers then transformed the feature maps into image feature vectors, which the Softmax classifier classified. Figure 3.8 illustrates the CNN architecture used in this study.

The proposed CNN architecture was established and trained using python library TensorFlow, Keras, and Scikit-learn. CNNs, like other deep learning models, are trained using backpropagation and stochastic gradient descent (SGD), which is based on differentiation or chain rules. Concerning the model's parameters, SGD efficiently minimizes a differentiable objective

function (e.g., categorical cross-entropy). Several enhancements and new optimization strategies for training CNNs have recently been proposed. The adaptive moment estimation (Adam) method was employed in this study, which is an optimization method that computes adaptive learning rates for each parameter of a network model. Adam's technical details can be found in the original paper (Kingma and Ba 2014). It was used in this study to minimize categorical cross-entropy loss. The weights are all initialized with a zero-mean Gaussian distribution with a standard deviation of 0.01, whereas the biases are initialized with a constant of 1. The learning rate is initially set at 0.001, and the learning rate policy is sigmoidal decay. The models are trained for 500 iterations before being terminated. If no improvement in validation accuracy is observed after 15 iterations, the training process is immediately halted, as shown in Equation (3.6).

$$loss(\hat{y}, y) = -\frac{1}{m} \sum_{i=1}^m y^{(i)} \log \hat{y}^{(i)} + (1 - y^{(i)}) \log (1 - \hat{y}^{(i)}) \quad (3.6)$$

Where $\hat{y}$ refers to the predicted value, y refers to the actual value , and m denotes the total number of training data.

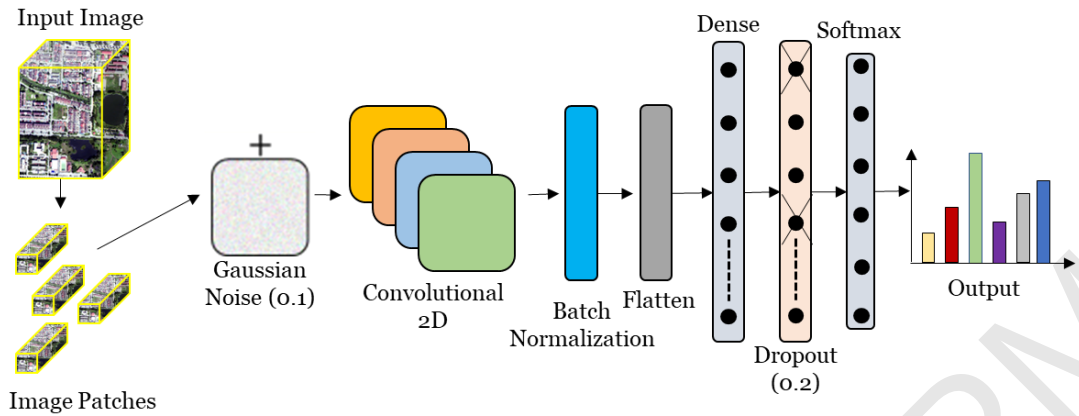


Figure 3.8 : An Illustration of the CNN Architecture Used in This Study

3.5.2 The OBIA-CNN with Joint Optimization (OCNN-JO)

The proposed object-based CNN framework is shown in Figure 3.9. It is based on learning deep features from very high-resolution satellite imagery and optimizing image segmentation at the same time. The framework consists of two main stages: (1) training a CNN on processed image patches; and (2) employing joint Bayesian optimization to find the best segmentation parameters and updating the CNN's network weights by transfer learning.

After establishing image data and ground-truth pairs, image patches extracted from the raw image are processed in four steps: balancing class samples, dividing the data into training, validation, and test sets, encoding class labels, and normalizing the image data into a standard value range.

In remote sensing, data with imbalanced classes is a prevalent issue (Lee et al. 2016). They affect the classification algorithm by correctly predicting classes that make up the majority while misclassifying classes that make up the minority. This study used random oversampling to balance the amount of

data within various classes in the dataset because deep learning methods require rather large training datasets. By choosing random samples from the class dataset with replacement, the approach increases the number of data points in a minority class. The samples are then split into three groups: training (70%), validation (15%), and test (15%), which are used to train, validate, and test the models, respectively. The target data is then encoded. The image data are finally normalized using the min-max technique.

A CNN classification model is trained based on patch-based samples. Image segmentation is utilized in decision-fusion approaches to refine the classification results produced by CNN with majority voting. However, the choice of segmentation parameters has a major impact on the outcome. While optimizing segmentation parameters alone may result in accurate segmentation, not sharing knowledge between segmentation and feature learning can have a major influence on classification performance. This study employed a Bayesian optimization strategy to jointly optimize the image segmentation process and train the CNN model for image classification.

Figure 3.10 illustrates the Bayesian optimization workflow in this study. The approach begins by taking the pretrained CNN and updating the network's weights via transfer learning. The MRS, on the other hand, segments the input image using the initial segmentation parameters scale = 100, shape = 0.1, and compactness = 0.5. To obtain the classification map, a majority voting method is applied to the CNN predictions based on image segmentation. A Gaussian filter with a 7×7 kernel is additionally used to smooth and reduce the noise in

the results. Finally, the classification accuracy is assessed at the pixel level using the mean intersection over union (mIoU) metric. The entire procedure, from transfer learning to accuracy measurement, is regarded as an objective function for Bayesian optimization. Consequently, it is anticipated that the Bayesian optimization will identify the best segmentation parameters that result in the best feature learning and classification. After transfer learning, the optimum segmentation parameters and a trained CNN are used to create the study area's final classification map.

3.5.2.1 Bayesian Optimization

Bayesian optimization has the ability to optimize expensive-to-evaluate functions, such as black box functions. This study utilized Bayesian optimization to refine the segmentation's parameters (i.e., scale, shape, and compactness) and identify the optimal parameter values. Bayesian optimization works based on three main concepts: (1) defining the objective function; (2) selecting the surrogate model; and (3) selecting the acquisition function (Figure 3.10).

Depending on the innovative framework:

- i. OCNN-JO framework represents the objective function, which is considered a complicated function due to the internal interaction of different functions such as segmentation, deep feature learning, filtering, and accuracy assessment.
- ii. Gaussian processes (GPs) are used as a surrogate model for Bayesian optimization and incorporate prior beliefs about the objective function. They are iteratively updated to capture the objective function's posterior

probability distribution. A GP is parametrized by a mean function μ and a covariance or kernel function k . Snoek et al. (2012) recommended using the ARD Matérn kernel (Rathbun et al. 2000), because the squared exponential kernel is unrealistically smooth for practical optimization problems.

- iii. (3) An acquisition function is used to explore and detect the optimal values of the MRS parameters. In general, the acquisition function propose new x combinations in the domain space to evaluate with $f(x)$ by making use of the GP posterior probability distribution. Even though there are many acquisition functions, expected improvement (EI) (Jones et. al 1998) is the most commonly used (Shahriari et. al 2015). EI helps to choose the next query sample as the one which has the highest EI over the current $\max f(x^+)$, where $x^+ = \operatorname{argmax}_{x_i \in x_{1:t}} f(x_i)$ and x_i is the location queried at i th time step.

In this acquisition function, $t + 1^{\text{th}}$ query point, x_{t+1} , is selected according to the Equation (3.7).

$$x_{t+1} = \operatorname{argmin}_x E(\|h_{t+1}(x) - f(x^*)\| | D_t) \quad (3.7)$$

where f is the actual ground-truth function, h_{t+1} is the posterior mean of the surrogate at $t + 1^{\text{th}}$ timestep, D_t is the training data, and x^* is the actual position where f takes the maximum value.

The expected improvement can be measured using the below functions :-

$$EI(x) = \begin{cases} (\mu_t(x) - f(x^*) - \epsilon) \Phi(z) + \sigma_t(x) \phi(z), & \text{If } \sigma_t(x) > 0 \\ 0, & \text{If } \sigma_t(x) = 0 \end{cases} \quad (3.8)$$

$$z = \frac{\mu_t(x) - f(x^*) - \epsilon}{\sigma_t(x)} \quad (3.9)$$

$\mu_t(x)$ denotes the mean function of the gaussian process at (x) value; $f(x^*)$ denotes the maximum value that has been encountered so far, ϵ denotes a variable value, $\sigma_t(x)$ denotes the standard deviation at (x) value, $\Phi(z)$ denotes the cumulative distribution function (CDF), and $\phi(z)$ denotes the probability density function (PDF).

Equation 3.7 indicated that if $\sigma_t(x) > 0$ this indicates there is a probable improvement based on the surrogate model and x value. Which is calculated using the formula $(\mu_t(x) - f(x^*) - \epsilon) \Phi(z) + \sigma_t(x) \phi(z)$. Contrarily, If $\sigma_t(x) = 0$ this implies there is no a probability of improvement. Hence, the samples (i.e., segmentation parameters) with higher expected improvement values would receive priority in the evaluation process.

According to the proposed methodology, our objective function $f(x)$ takes a parameter combination (x) from the domain space and applies a series of data processing steps to obtain the accuracy of classification results using mIoU. Those steps include transfer learning, image segmentation, majority voting, and Gaussian filtering, which are described below.

This study implements Bayesian optimization based on the Python library (GPyOpt) (<http://sheffieldml.github.io/GPyOpt/>) (15 September 2022). GPyOpt is a Bayesian optimization library that builds on the GPy library. This contains the parameter space of each MRS parameter, from which the Bayesian optimization routine has to identify the optimal parameter combination. The scale ranged from 25 to 5000, shape from 0 to 0.9, and compactness from 0

to 1. Figure 3.11 shows the application of the Bayesian optimization process.

3.5.2.2 Transfer Learning

In the proposed framework, the optimal parameters of the MRS algorithm (scale, shape, and compactness) cannot be effectively determined independently. There should be a knowledge-sharing mechanism between classification and segmentation to achieve the best possible optimization. For this reason, the transfer learning technique was employed to produce classification maps based on given MRS parameters. Transfer learning aimed to lessen the computational burden, as using the original classification model would have resulted in significantly longer optimization iterations. Moreover, it also aimed to effectively exploit learned weights with large samples to produce classification maps with small samples in new situations.

The innovative framework implements transfer learning by using Python tool libraries (TensorFlow and Keras) to obtain the CNN model's only dense network and Softmax layer. Additionally, it updates the network's weights with any changes to segmentation parameters and re-runs the classification process.

3.5.2.3 Majority Voting

Deep learning classification results are improved further by the majority voting of class categories by pixels in each object segmented by MRS. Each pixel in the deep learning classification map contains a class label predicted by the model. Calculating the most common class occurring within the objects yields

a single class label for each image object. In the final classification map, each pixel in the object takes the new class label of the object it contains. The Python library NumPy is used to implement majority voting, which is the most powerful of Python's libraries in terms of numerical computation. This tool refined the classified pixels based on their corresponding segments and the majority pixel value.

3.5.2.4 Gaussian Filtering

The Gaussian filter is a non-uniform, low-pass filter often used to blur images or reduce noise in images. It is a 2D convolution operator; however, different kernels can be used, such as bell-shaped, based on the application type. In this study, this technique was implemented using python library (NumPy), and the Gaussian filter was applied with 7×7 kernel size to the input images to smooth out the noise.

3.5.2.5 Benchmark Methods

The patch-based CNN, center-point OCNN, region majority voting OCNN (RMA-OCNN), and decision-fusion OCNN were used as benchmarks to assess the effectiveness of the proposed classification framework (OCNN-JO). To ensure a fair comparison, the underlying CNN in each approach was identical in terms of network parameters and hyperparameters. The methods that use image segments were implemented based on the same segmented objects acquired from MRS. The descriptions and parameter settings of these benchmarks are detailed as follows:

Patch CNN: Experimentally, this model is based on densely overlapping patches with the size of a 9×9 set. The number of convolutional layers and their corresponding number of filters are also set experimentally to 2 and 64, respectively.

Center-Point OCNN: In contrast to pixel-wise CNN, this model uses image segmentation to extract patches at the objects' centers. The segmentation parameter scale, shape, and compactness were set experimentally to 4500, 0.15, and 0.1, respectively.

RMV-OCNN: This model works principally as the CPOCNN; however, it randomly generates N convolutional positions within each image segment and trains the CNN. The same segmentation parameters are chosen for this model. The network's architecture and hyperparameters are also identical to the CPOCNN.

Decision-Fusion OCNN: This model generates predictions for each image pixel within image segments using the PCNN and then fuses them with a majority voting strategy to achieve the final classification results.

In general, many challenges were faced during the research process, including designing and implementing the innovative OBIA-deep learning framework using Python codes of several models such as MRS, CNN, ANN, Bayesian optimization, and accuracy assessment functions, which requires programming skills as well as advanced computation resources. Therefore, several steps have been tackled, including advanced Python coding training

and reviewing several resources on Python programming.

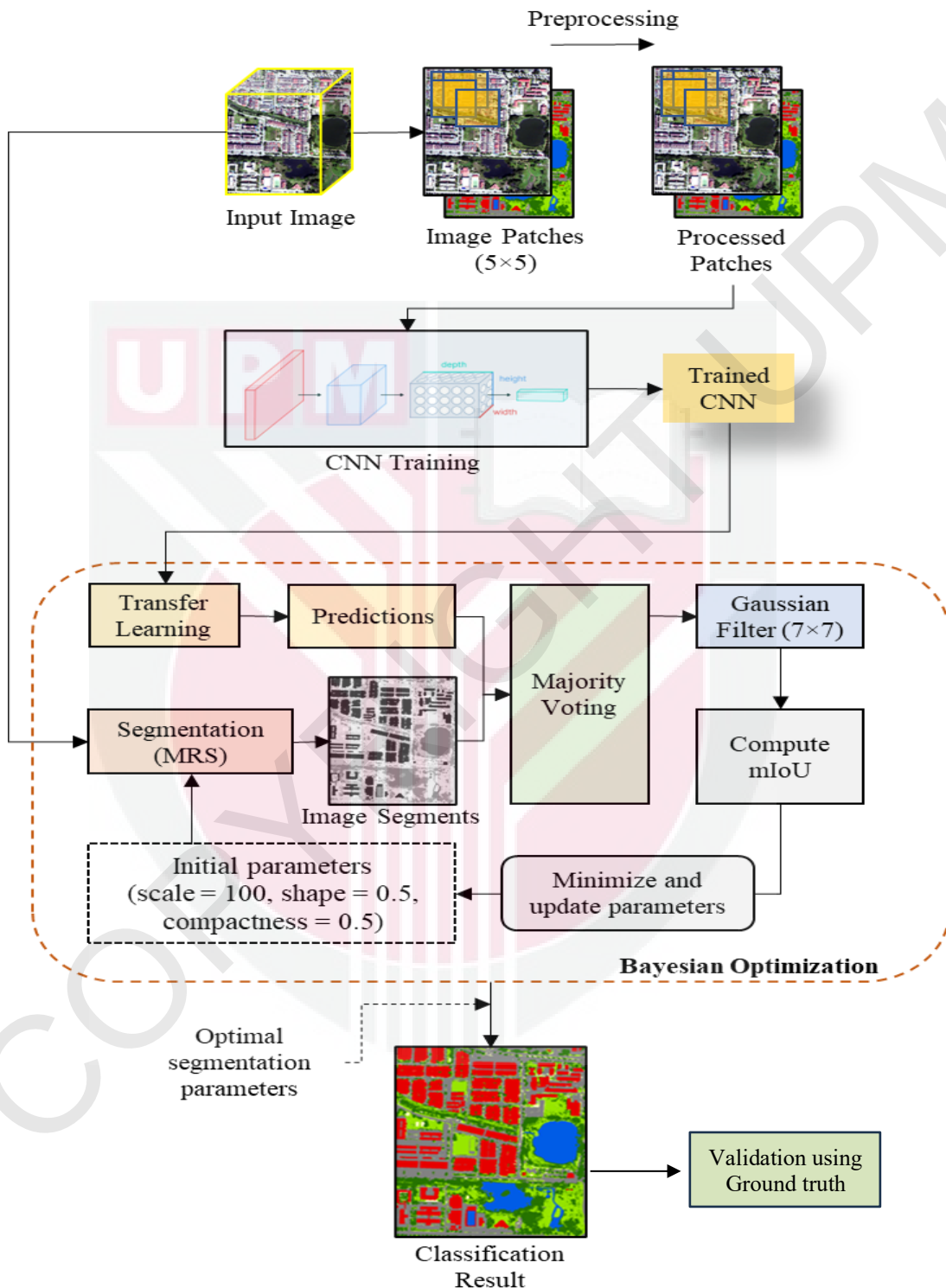


Figure 3.9 : Workflow of the Bayesian Optimization Procedure to Jointly Optimize Segmentation and Feature Learning for Image Classification

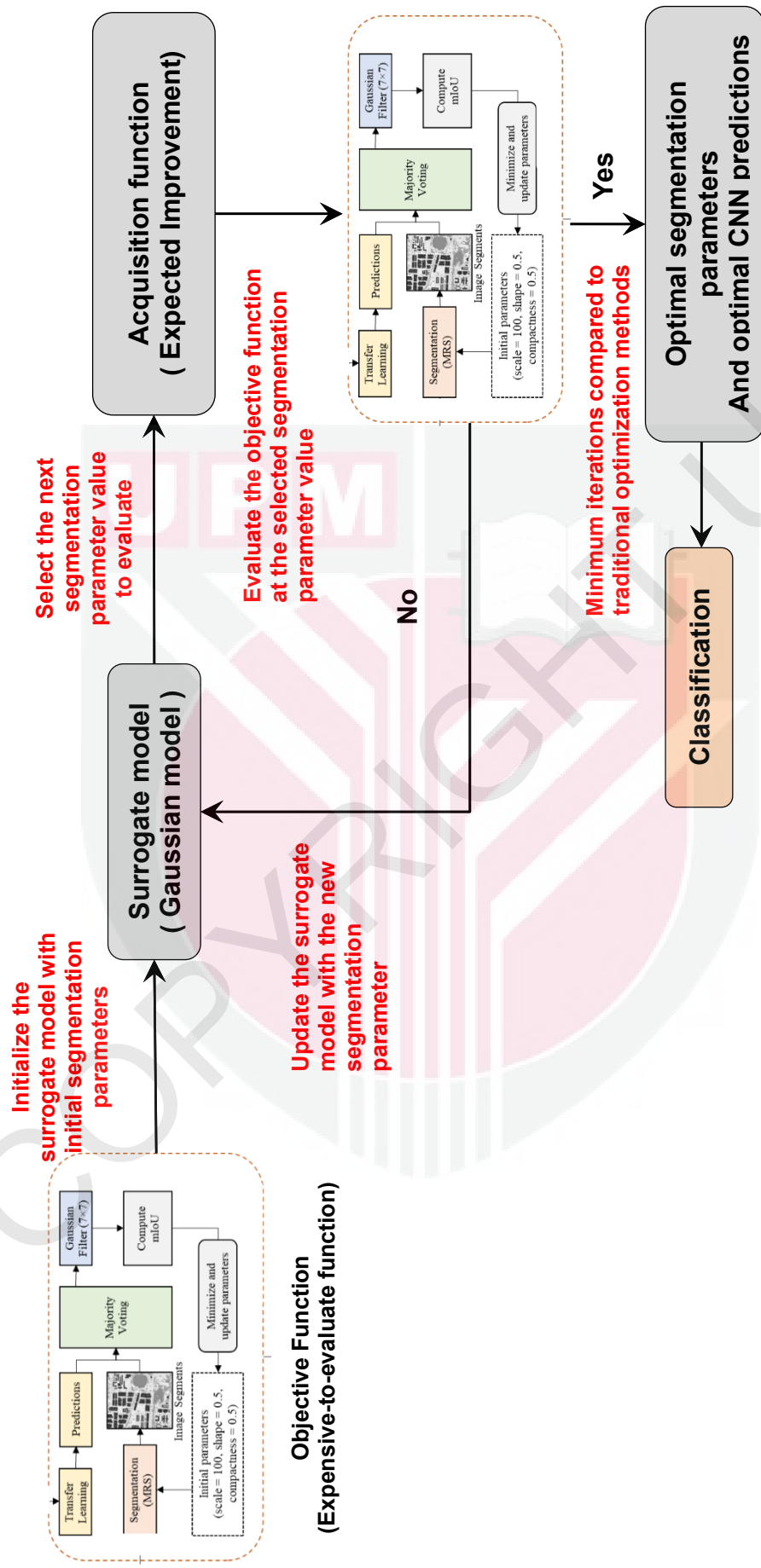


Figure 3.10 : The Selection of Optimal Segmentation Parameters Using Bayesian Optimization

3.6 Accuracy Assessment

3.6.1 Classification Accuracy Assessment

This study uses four common classification accuracy assessment methods, namely, overall accuracy (**OA**), Kappa coefficient (**K**), F1-score, and mean intersection over union (**mIoU**). **OA** and **K** are notably two accuracy measures used in traditional remote sensing studies. These measures can be computed at the pixel or object level. This study uses them at the pixel level due to the nature of the validation data prepared for this research. **OA** refers to the specific value of the total number of all correct classifications and that of samplings and reflects the degree of correctness of all categories in the classification results of images (Equation 3.10). The Kappa coefficient (**K**) is an assessment index used to judge the degree of coincidence between two images and ranges from 0 to 1. It indicates how much the classification method selected is better than the method where the single pixel is randomly assigned to any category (Equation 3.11). On the other hand, **mIoU** (also known as the Jaccard index) is reported occasionally in semantic segmentation applications as well as classification problems (Zhang et al., 2020). **mIoU** can be calculated based on pixels or bounding boxes. It is the ratio of the intersection between the reference and classified samples with the union of the two groups. The former method is well-suited for classification applications, while the latter is preferable for instant segmentation or object-detection tasks. In this study, **mIoU** is applied directly to pixels in the classified image as a whole. It uses the true positive (TP), false positive (FP), and false negative (FN) classes at the pixel level (Equation 3.12).

Where F1-score is a statistic that considers both recall and precision (3.13).

$$OA = \frac{\sum_{i=1}^n m_{ii}}{\sum_{j=1}^n \sum_{i=1}^n m_{ij}} \quad (3.10)$$

$$K = \frac{N \sum_{i=1}^n m_{ii} - \sum_{i=1}^n m_{i+} m_{+i}}{N^2 - \sum_{i=1}^n m_{i+} m_{+i}} \quad (3.11)$$

$$mIoU = \frac{TP}{TP + FP + FN} \quad (3.12)$$

$$F1 \text{ score} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (3.13)$$

$$\text{precision} = \frac{TP}{TP + FP} \quad (3.14)$$

$$\text{recall} = \frac{TP}{TP + FN} \quad (3.15)$$

where m_{ij} represents the total number of pixels that are assigned to Class j from those subordinate to Class i in the research region and n represents the total number of classes. N represents the total number of samples, and m_{+i} and m_{i+} are, respectively, sums of rows and lines in the confusion matrix.

3.6.2 Segmentation Quality Assessment

The quality of image segmentation can be measured with some reference data based on comparing segmented objects with their actual objects in the reference data. The quality of segmentation degrades due to anomalies such as under-segmentation or over-segmentation. These values can be formulated by metrics such as area of fit (AFI) or quality rate (QR), which can be calculated using the reference object area (A_r) and the object area obtained as a result of segmentation (A_s) (Liu et al., 2019, El-naggar et al., 2018).

In optimal segmentation, AFI is expected to be 0 and the QR value is expected to be 1. Quality metrics were calculated using the following equations:

$$AFI = \frac{A_r - A_s}{A_r} \quad (3.16)$$

$$QR = 1 - \frac{A_r \cap A_s}{A_r \cup A_s} \quad (3.17)$$

3.7 Framework's Validation on the Road Extraction

This study also validated the innovative framework for road extraction applications. It also used a data subset that included roads with complex geometry. The WV3 image was employed as well as the ground truth image, which includes roads and non-road areas). The area covers 0.9 km², of which the total road is 26%. In practical terms, the Python tool is based on SciPy, Scikit-image, Keras, Scikit-learn, and NumPy. This study utilized the same architecture of the framework but modified the final classification layer from 6 nodes to 2 nodes (background, road). The road extraction process involves several steps; first, this study used a CNN model to classify the image into road and non-road areas. Next, it refined the CNN result using the majority filter based on MRS segmentation. Finally, it enhanced the resulting road map using gaussian filtering. Finally, we assess the accuracy using the mIoU model. Figure 3.11 illustrates the road extraction process based on the OCNN-JO framework.

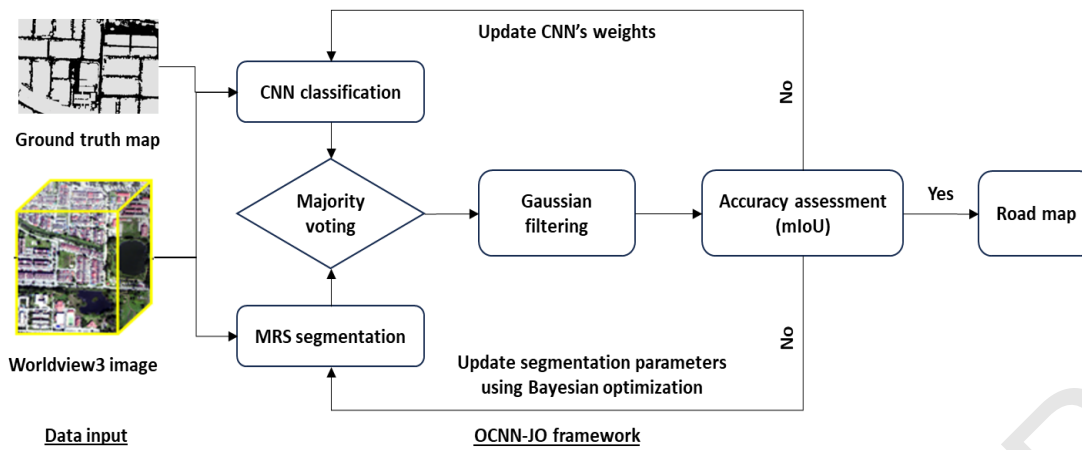


Figure 3.11 : Workflow of Road Extraction Based on the Innovative Framework (OCNN-JO)

CHAPTER 4

RESULTS AND DISCUSSION

This chapter presents results obtained from several experiments conducted in the current research to compare several integration frameworks for the classification of VHR images. The accuracy of each classification framework is presented and used to compare and assess the models. Also, the classification maps are discussed regarding the visual quality and the distribution of the land covers in the study area.

4.1 Results of Assessing OBIA and Deep Learning Integration Frameworks

This research assessed four common OBIA-deep learning integration frameworks, namely, OBIA-CNN, Decision Fusion, Feature Fusion, and Patch Filtering. To evaluate the integration's feasibility, this study compared the integrated frameworks with Pixel ANN and Patch CNN models. This subsection presents the results of the comparative experiments conducted in this study.

4.1.1 Assessment by Overall Accuracy

This research used two models, Pixel ANN and Patch CNN, to assess the integration feasibility. These models were compared based on three accuracy matrices, i.e., OA, Kappa index, and F1-score, for the training and test areas. The results show that the Patch CNN outperformed the Pixel ANN on any accuracy metric, as well as in both the training and test areas. For the training

area, the Patch CNN achieved 0.897 OA, 0.860 Kappa, and 0.900 F1-score, while the Pixel ANN achieved 0.849 OA, 0.818 Kappa, and 0.850 F1-score. The accuracies of the Patch CNN for the test area were 0.898 OA, 0.876 Kappa, and 0.890 F1-score. For the test area, the Pixel ANN achieved accuracies of 0.858 OA, 0.829 Kappa, and 0.860 F1-score.

In addition, Table 4.1 presents a detailed assessment of the four integration frameworks studied in this research. In general, the results indicate that patch filtering achieved the best classification results for both the training and test areas. Based on the OA and Kappa index, patch filtering achieved the best results for both the training (0.919 OA, 0.868 Kappa, and 0.920 F1-score) and the test areas (0.917 OA, 0.872 Kappa, and 0.910 F1-score). OBIA CNN performed slightly worse than Patch CNN and Pixel ANN. It obtained 0.842 OA, 0.784 Kappa, and 0.830 F1-score using the test dataset, whereas Decision Fusion and Feature Fusion achieved 0.862 OA and 0.860 OA, respectively.

Table 4.1 : Accuracy Assessment of the Classification Methods Used in this Research for the Training and Test Areas

Model	Training Area			Test Area		
	OA	Kappa	F1-score	OA	Kappa	F1-score
Pixel ANN	0.849	0.818	0.850	0.858	0.829	0.860
Patch CNN	0.897	0.860	0.900	0.898	0.876	0.890
OBIA CNN	0.838	0.763	0.840	0.842	0.784	0.830
Feature Fusion	0.869	0.807	0.870	0.860	0.788	0.850
Patch Filtering	0.919	0.868	0.920	0.917	0.872	0.910
Decision Fusion	0.864	0.793	0.850	0.862	0.793	0.850

4.1.2 Assessment by Per-Class Accuracies

Tables 4.2 and 4.3 present the per-class accuracy obtained for the assessed integration frameworks, as well as the Pixel ANN and Patch CNN. The Patch CNN classified the grassland in the training area the best (0.93), followed by the Pixel ANN (0.9) and Patch Filtering (0.88). Roads were best identified by the Patch CNN (0.82). Patch Filtering achieved a road classification accuracy of 0.76, which is worse than the methods, i.e., Patch CNN and Feature Fusion. Patch filtering also performed well for the bare land class, with an accuracy of 0.93. For the building class, feature fusion and decision fusion (0.85) outperformed the other methods. Patch CNN achieved the best results for dense vegetation or trees (0.92). On the other hand, the water class was best classified by the Pixel ANN and Feature Fusion (0.94).

The Patch CNN (0.88) and Patch Filtering (0.82) performed the best in classifying the grassland class in the test area. The OBIA CNN (0.55) and the Decision Fusion (0.60) achieved the worst results for the grassland class. OBIA CNN achieved the best results for the building class. In addition, the patch filtering best classified bare land (0.89), while the Pixel ANN and Decision Fusion best identified water (0.98). The Patch CNN (0.91) and Patch Filtering (0.87) performed the best in classifying dense vegetation or trees.

Moreover, certain spectral and spatial features can describe each land cover type. The integrated frameworks depend either on image segmentation, deep feature extraction, or both. That is why some methods may perform better on certain land cover types than others. For example, with the feature fusion

method, no segmentation effect appears in the classification results. This method is therefore not suitable for urban land cover with complex geometry of buildings existing in the area.

Table 4.2 : Per-Class Accuracies Obtained for Each Classification Method Using the Training Area Dataset

Class Name	Average Accuracy					
	Pixel ANN	Patch CNN	OBIA CNN	Feature Fusion	Patch Filtering	Decision Fusion
Building	0.81	0.84	0.81	0.85	0.79	0.85
Road	0.75	0.82	0.72	0.78	0.76	0.75
Grass Land	0.90	0.93	0.75	0.65	0.88	0.70
Dense Vegetation or Trees	0.85	0.92	0.72	0.74	0.89	0.64
Water Body	0.94	0.93	0.88	0.94	0.89	0.93
Bare Land	0.86	0.69	0.70	0.88	0.93	0.89

Table 4.3 : Per-Class Accuracies Obtained for Each Classification Method Using the Test Area Dataset

Class Name	Average Accuracy					
	Pixel ANN	Patch CNN	OBIA CNN	Feature Fusion	Patch Filtering	Decision Fusion
Building	0.89	0.89	0.93	0.85	0.84	0.86
Road	0.80	0.88	0.79	0.74	0.84	0.79
Grass Land	0.78	0.88	0.55	0.71	0.82	0.60
Dense Vegetation or Trees	0.85	0.91	0.81	0.60	0.87	0.71
Water Body	0.98	0.97	0.90	0.93	0.97	0.98
Bare Land	0.84	0.68	0.72	0.86	0.89	0.85

4.1.3 Results of Image Classification Maps

Models like OBIA CNN and Patch CNN, as well as their integration by the four common frameworks, have been utilized to classify the image data of the training and test areas. Figures 4.1 and 4.2 show the classification maps

obtained for the training and test areas using the assessed methods. The classification maps were produced according to six land cover classes, including grassland, roads, buildings, dense vegetation or trees, and water. The training and test areas are mostly urban/built-up areas. The areas also consist of relatively complex road networks due to parking lots. Water bodies in the areas are mostly artificial small lakes. Both the training and test areas contain dense vegetation (or small trees) and grasslands.

Some classification maps display a "salt-and-pepper" noise pattern, while others show less random noise and appear as vector maps. The maps of the Pixel ANN contain random noise and misclassification between building and grassland and dense vegetation or tree classes. The maps of the Patch CNN have less random noise and misclassifications. It can also be observed that the Patch CNN improves the detection of the roads compared to the Pixel ANN. The maps of the OBIA CNN contain very little random noise. However, they contain significant misclassification between building and road classes. It seems that the CNN could not learn useful contextual features from the OBIA features, or the latter is not enough to separate the building and road classes. The maps obtained by the patch filtering appear to have less random noise and misclassification compared to the patch CNN. The maps of the Decision Fusion are better than the OBIA CNN and with less misclassification between building and road classes.

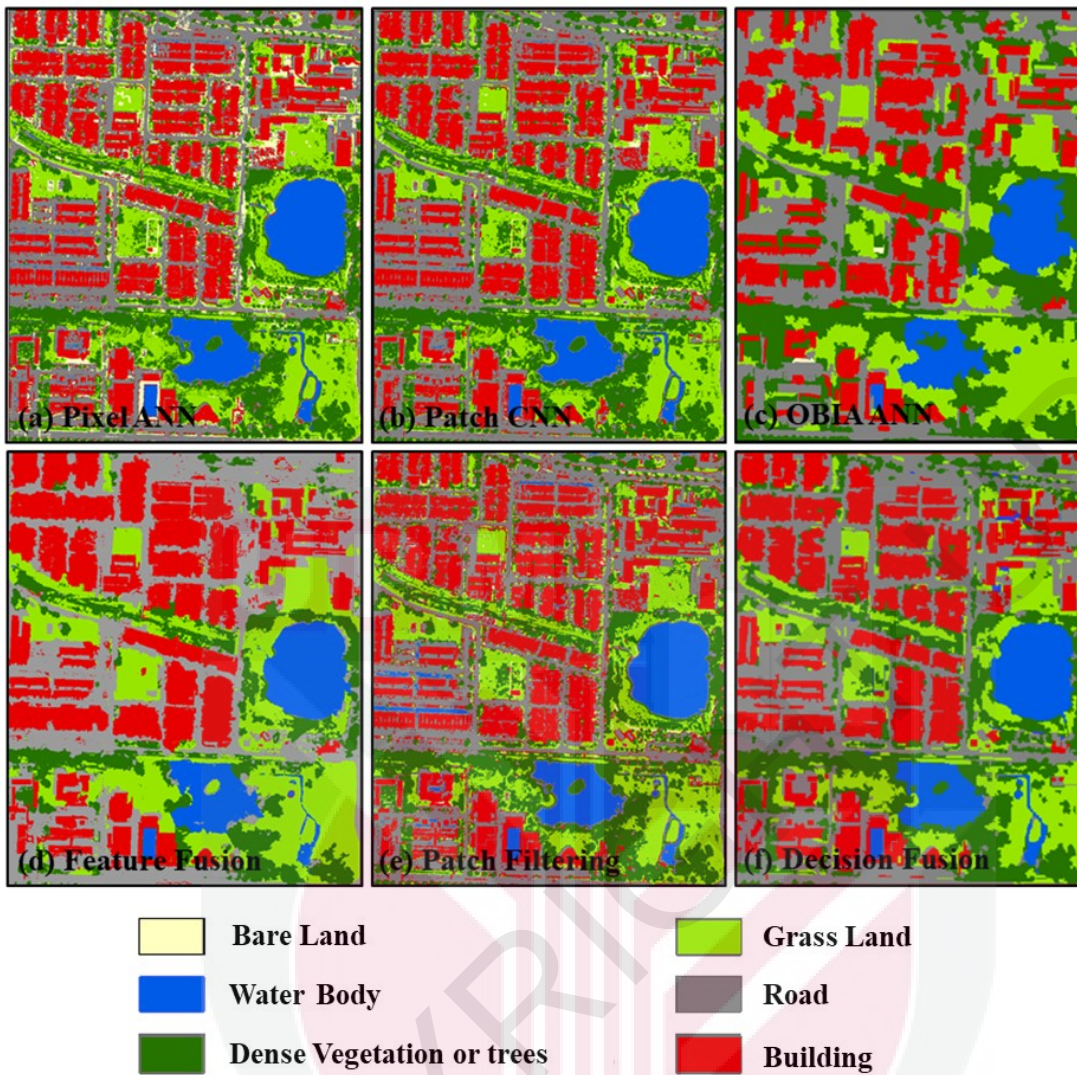


Figure 4.1 : Classification Maps Obtained for the Training Area with the Pixel ANN, Patch CNN and the Four OBIA-CNN Integration Frameworks Assessed in the Current Research

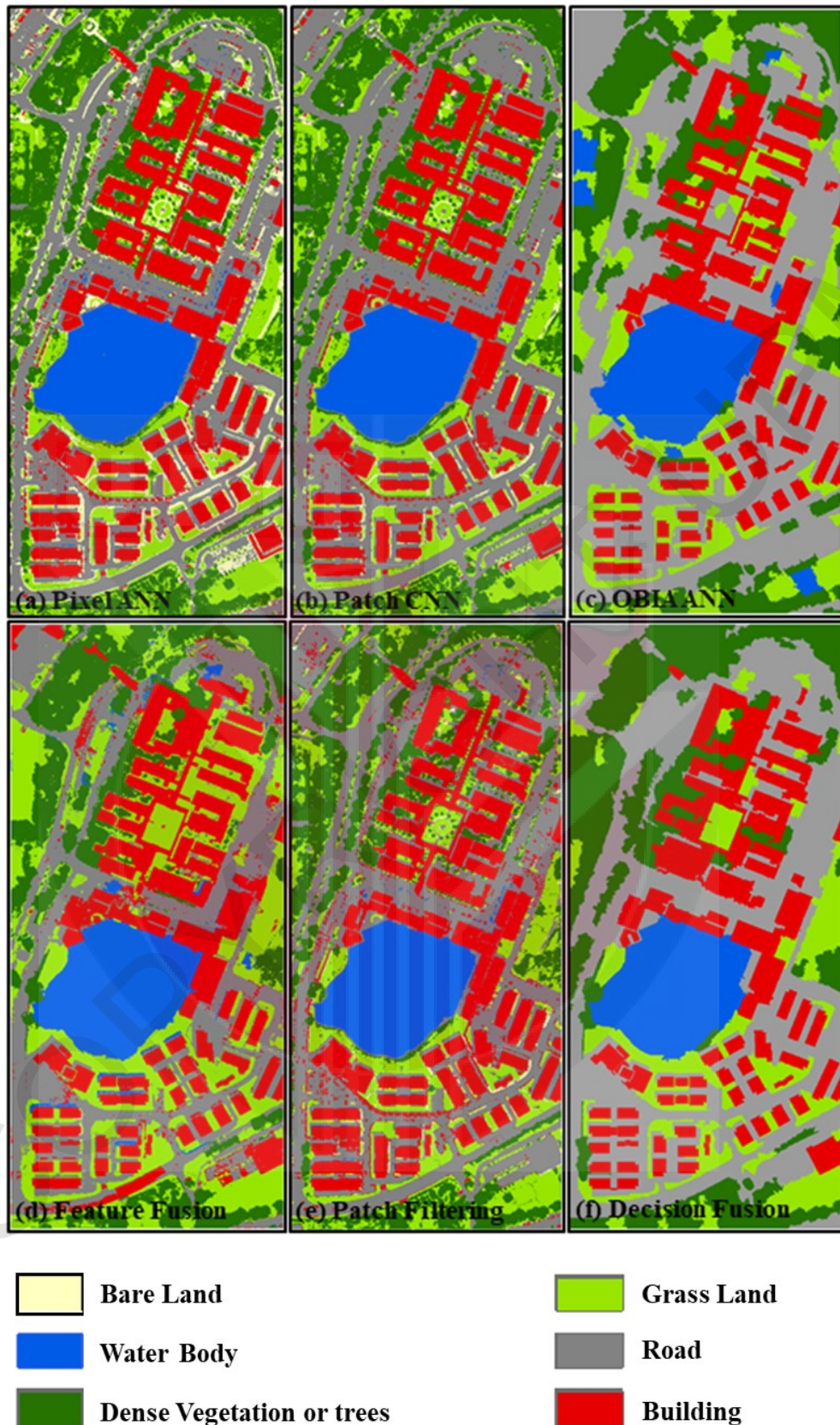


Figure 4.2 : Classification Maps Obtained for the Test Area with the Pixel ANN, Patch CNN and the Four OBIA-CNN Integration Frameworks Assessed in the Current Research

4.2 Results of the Innovative Classification Framework

4.2.1 Segmentation Results

Segmentation is a major processing step in object-based deep learning classification methods, and its parameters can have a significant impact on segmentation results and, ultimately, the classification results. Setting segmentation parameters by experience can achieve reasonable results, but it can only be performed independently from feature learning, which as a result may impact the final classification results as no knowledge is shared between the two segmentation and feature extraction tasks. In this study, segmentation parameters were optimized by the proposed Bayesian optimization workflow discussed in Chapter 3. Figure 4.3 shows the segmentation results for the training and test areas with the optimal parameters. For the training areas, the optimization procedure found the following best parameters, i.e., scale, shape, and compactness. For the test area, the best segmentation parameters were for scale, shape, and compactness, respectively. Table 4.4 illustrates the search space, initial values, and best values of MRS segmentation parameters for the training and test areas. Table 4.5 presents the quality metrics for the segmentation results. The training area was segmented with AFI and QR of 0.046 and 0.945 m, respectively. The metrics for the test area were -0.037 AFI and 0.932 QR. The results indicate good segmentation for both areas.

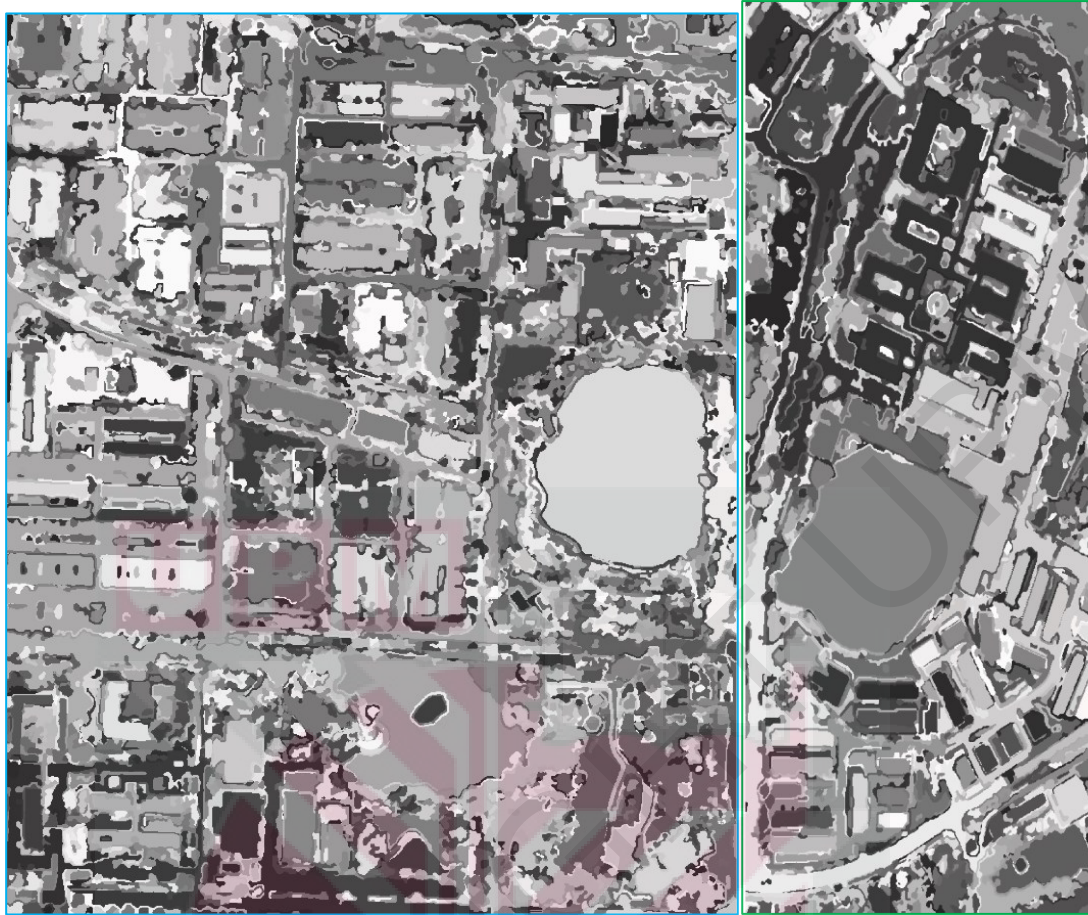


Figure 4.3 : The Results of MRS Segmentation with Optimal Parameters Were Found by the Proposed Bayesian Optimization Method for the (Left) Training and (Right) Test Areas

The three MRS parameters were optimized based on search spaces experimentally set up in this research. For the scale parameter, the low bound and high bound were determined as 25 and 5000, respectively. For the shape and compactness, the search space of the real values between 0.01 and 1 was used in the optimization process. The initial values of the parameters were decided randomly. The proposed Bayesian optimization method then determined the best values for both the training and test areas. The best scales were 4705 and 4617 for the training and test areas, respectively. The best shape and compactness values were 0.2 and 0.1 for the training area and

0.14 and 0.1 for the test area, respectively. The best parameter values for the training area are very close to those for the test area. In both areas, higher scale value and lower shape and compactness are preferred.

Table 4.4 : Search Space, Initial Values, and Best Values of MRS Segmentation Parameters for the Training and Test Areas

Parameter	Search Space	Initial Value	Best Value (Training Area)	Best Value (Test Area)
Scale	Integer [25, 5000]	100	4705	4617
Shape	Real [10^{-2} , 10^0]	0.5	0.2	0.14
Compactness	Real [10^{-2} , 10^0]	0.5	0.1	0.1

Table 4.5 : Segmentation Quality Metrics for the Training and test Area Datasets

Dataset	AFI	QR
Training Area	0.046	0.945
Test Area	-0.037	0.932

4.2.2 The Results of the OCNN-JO Framework

The proposed classification framework experimented with three CNN models, namely, 2D CNN, 3D CNN, and HybridSN. Figures 4.4 and 4.5 present a visual comparison of these methods as classification maps for the training and test areas, respectively. Table 4.6 presents the classification accuracies obtained for the three methods in the training and test areas including the overall accuracies (OA, Kappa, mIoU) as well as per-class accuracies. Generally, The three applied methods produced accurate boundary information and the smoothest visual results. Furthermore, this study identified the semantic contents of buildings and linearly shaped features (roads) with high geometric fidelity and accuracy in comparison to the ground-truth data. The 3D CNN

layers, as well as the combination of 3D and 2D CNN layers (HybridSN), produced slightly higher accuracies than the 2D CNN layers did. CNN models with higher kernel dimensions improved the boundary extraction of the image objects, resulting in better geometric fidelity and classification accuracy. The results also highlight that CNNs with higher kernel dimensions help extract better contextual features by combining the spectral and spatial information of the image data for feature extraction.

In terms of classification accuracy, the HybridSN model achieved the best accuracies in both the training and test areas. The HybridSN model achieved 0.96 OA, mIoU, and 0.95 Kappa in the training area. Slightly better accuracies (0.97 OA and mIoU and 0.96 Kappa) were obtained for this model in the test area due to the less-complex urban features (buildings and roads) compared to the features of the training area. The 3D CNN performed better than the 2D CNN based on the overall accuracy metrics in both the training and test areas. Looking at the per-class accuracies, the results indicate that water bodies were extracted most accurately due to their significant spectral variation compared to other classes. Buildings and roads were also extracted with higher accuracies than dense vegetation and bare lands by the three models due to their accurate geometric representations by the optimized segmentation. As the classes, i.e., buildings and roads are the dominant classes in the training and test areas, the Bayesian optimization could better identify the boundaries of these objects with the optimal segmentation parameters obtained. A generic accuracy metric (mIoU) was used to optimize the segmentation parameters in this study. However, for specific applications, e.g., vegetation studies, a

modified accuracy metric (average vegetation class accuracy) can be used to improve the process of segmentation, which may acquire better accuracies for the required application.

Figure 4.6 illustrates convergence plots showing the progress of Bayesian optimization of MRS parameters with 2D CNN, 3D CNN, and HybridSN models. The results indicate that the 2D CNN requires the least number of iterations (19) to stabilize during optimization, while the 3D CNN and HybridSN required larger numbers of iterations (24 and 25, respectively). On the other hand, Figure 4.7 presents the convergence plots visualizing the progress of optimization. It depicts the classification's accuracy (mIoU) after n calls of Bayesian optimization.

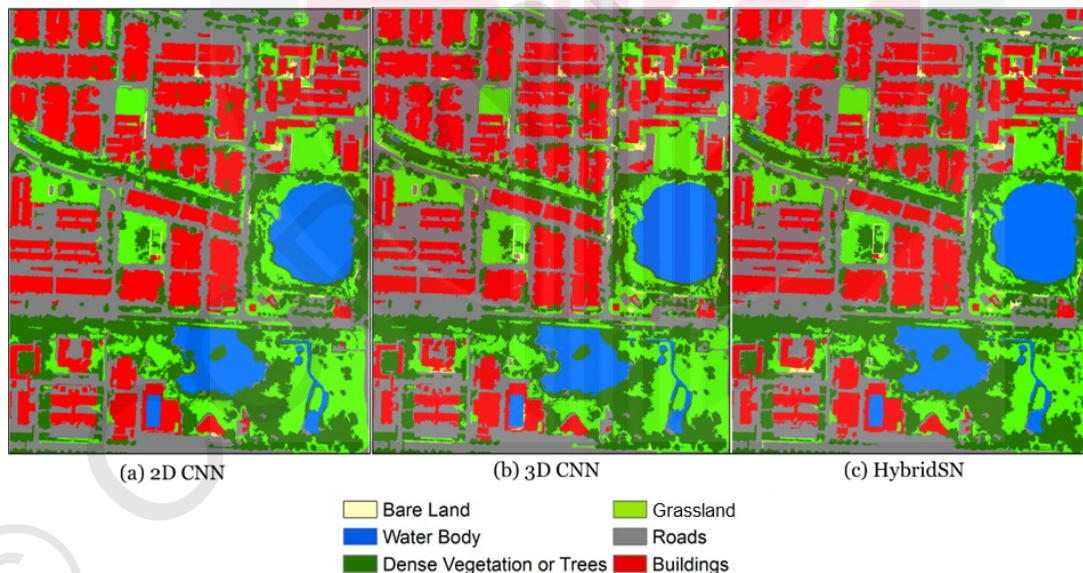


Figure 4.4 : Classification Maps of Different CNN Methods for the Training Area: (a) 2D CNN, (b) 3D CNN, and (c) HybridSN

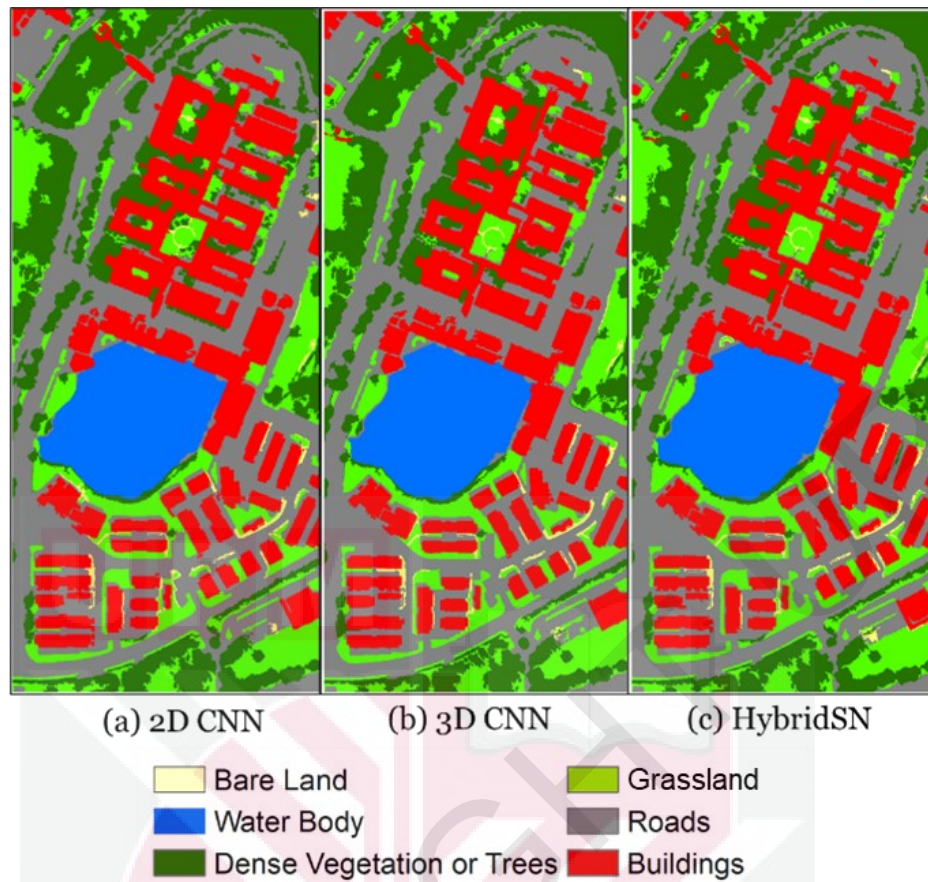


Figure 4.5 : Classification Maps of Different CNN Methods for the Test Area

Table 4.6 : Classification Assessment of Different CNN Methods Used with the Proposed Model for the Training and Test Areas

Class	Training Area			Test Area		
	2D CNN	3D CNN	HybridSN	2D CNN	3D CNN	HybridSN
Buildings	0.94	0.94	0.97	0.95	0.96	0.97
Roads	0.98	0.96	0.94	0.98	0.98	0.98
Grass Land	0.93	0.93	0.95	0.93	0.91	0.94
Dense Vegetation/Trees	0.93	0.96	0.94	0.9	0.91	0.95
Water Body	0.98	0.97	0.98	0.98	0.98	0.99
Bare Land	0.88	0.92	0.98	0.93	0.95	0.98
OA	0.94	0.95	0.96	0.94	0.95	0.97
Kappa	0.92	0.94	0.95	0.93	0.94	0.96
mIoU	0.94	0.95	0.96	0.94	0.95	0.97

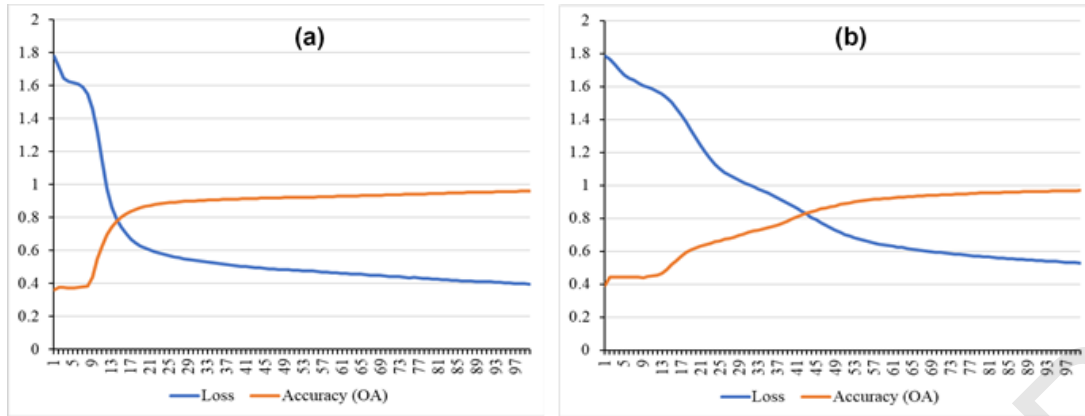


Figure 4.6 : The Learning Curve of the Proposed Model with HybridSN Backbone Model: (a) Training Dataset, (b) Test Dataset

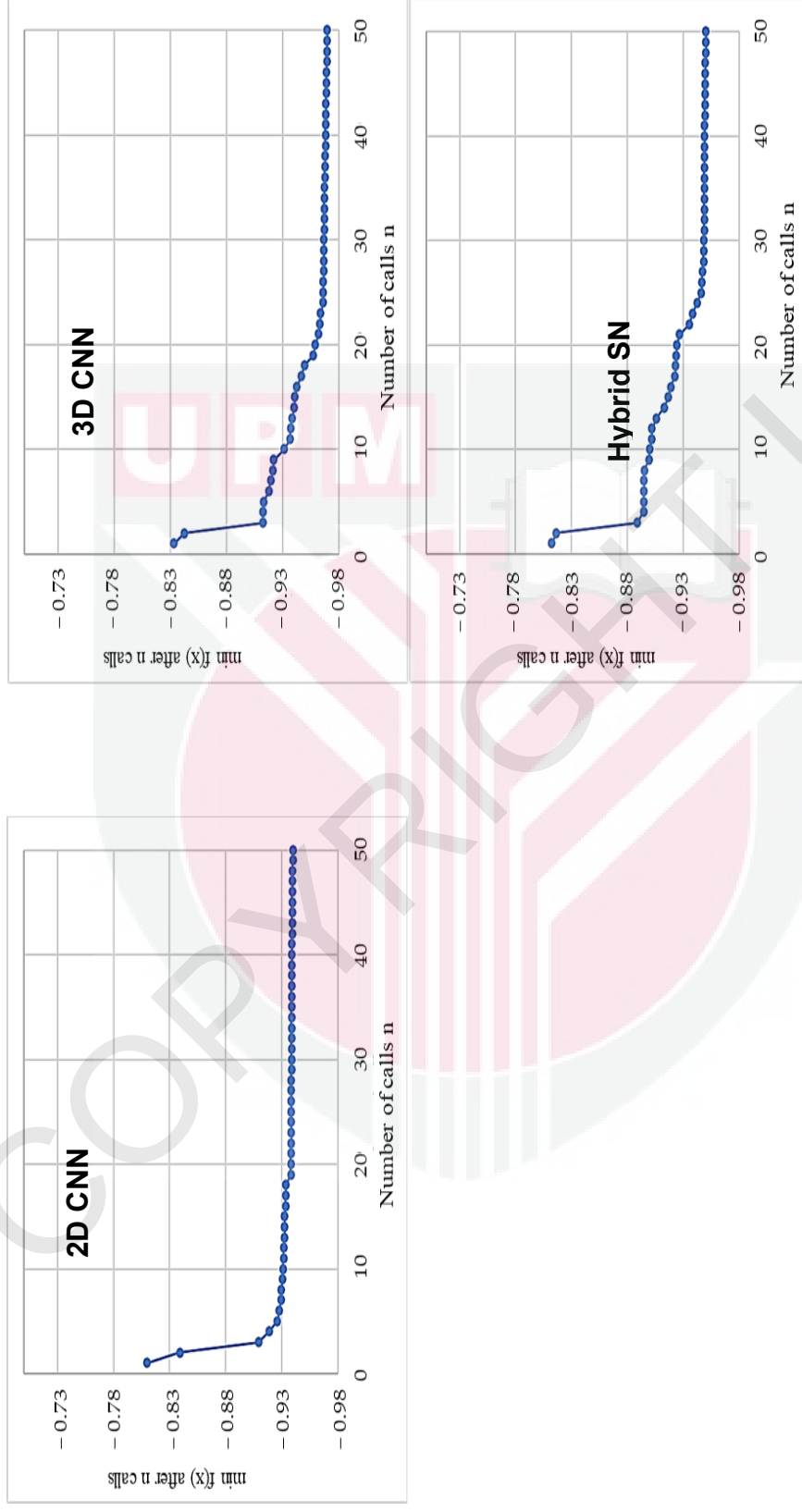


Figure 4.7 : Convergence Plots Showing the Progress of Bayesian Optimization of MRS Parameters with 2D CNN, 3D CNN, and HybridSN Models

4.2.3 Performance Comparison with Benchmark Models

This study assessed the proposed framework's performance by comparing it with existing benchmark approaches such as the pixel-wise Patch CNN, Center OCNN, Random OCNN, and Decision Fusion. The base CNN and network hyperparameters for the object-based methods were the same. The same training data were employed to train the benchmark models and measure their validation performance.

Figure 4.8 and 4.9 show the overall accuracy of the proposed and benchmark classification methods based on samples from the training and testing areas. Table 4.7 summarizes the measured accuracies for different methods in training and testing areas. In the training area, the Patch CNN achieved the lowest accuracy of 0.89 OA and Kappa and 0.86 mIoU, while in the test area, it achieved slightly better accuracy of 0.93 OA and Kappa and 0.91 mIoU. The decision-fusion method performed almost the same as the patch CNN. The methods that rely on context patches show that the Center OCNN achieved better classification accuracy than the Random OCNN in both the training and test areas. For example, based on mIoU, the Center OCNN achieved 0.93 compared to 0.90 for the Random OCNN in the training area. In the test area, the two methods achieved 0.95 and 0.93 mIoU, respectively. The proposed framework, on the other hand, achieved the best accuracy compared to other methods. In the training area, it achieved 0.96 OA, 0.95 Kappa, and 0.96 mIoU. In the test area, the accuracy was slightly better at 0.97 OA, 0.96 Kappa, and 0.97 mIoU.

Figures 4.10 and 4.11 show the classification maps obtained for the training and test areas using different classification models. The result of Patch CNN contains a “salt-and-pepper effect” more than the other methods because it processes the image data at the pixel level. This method also misclassified buildings as roads more than the other methods, indicating that object-level features help to separate these two classes more effectively than using pixel-level features. Image segmentation and the use of majority voting for classification significantly reduce salt-and-pepper noise in object-based methods. However, over-segmentation and using small scales may also lead to classification results with salt-and-pepper noise. As a result, segmentation optimization can help to keep this type of noise to a minimum. Both Center OCNN and Random OCNN, which use majority voting to obtain the class labels of objects, produced classification maps with less speckle noise, particularly within building objects. However, the former method is more accurate in separating adjacent buildings. The Decision-Fusion method obtained results with less noise; however, it has worse smoothness in the boundaries of buildings and roads, especially for complex objects. The proposed framework combines the advantages of low salt-and-pepper noise, accurate separation of adjacent buildings, and accurate boundary identification. OA, Kappa, and mIoU measured the accuracy of classification maps at the pixel level.

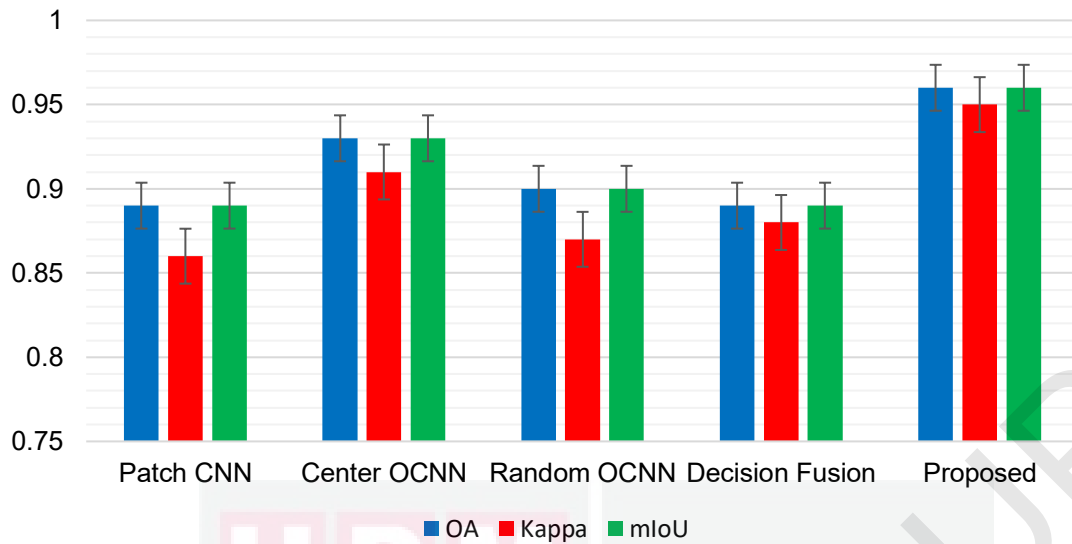


Figure 4.8 : The Overall Accuracy of the Proposed and Benchmark Classification Methods Based on Samples from the Training Area

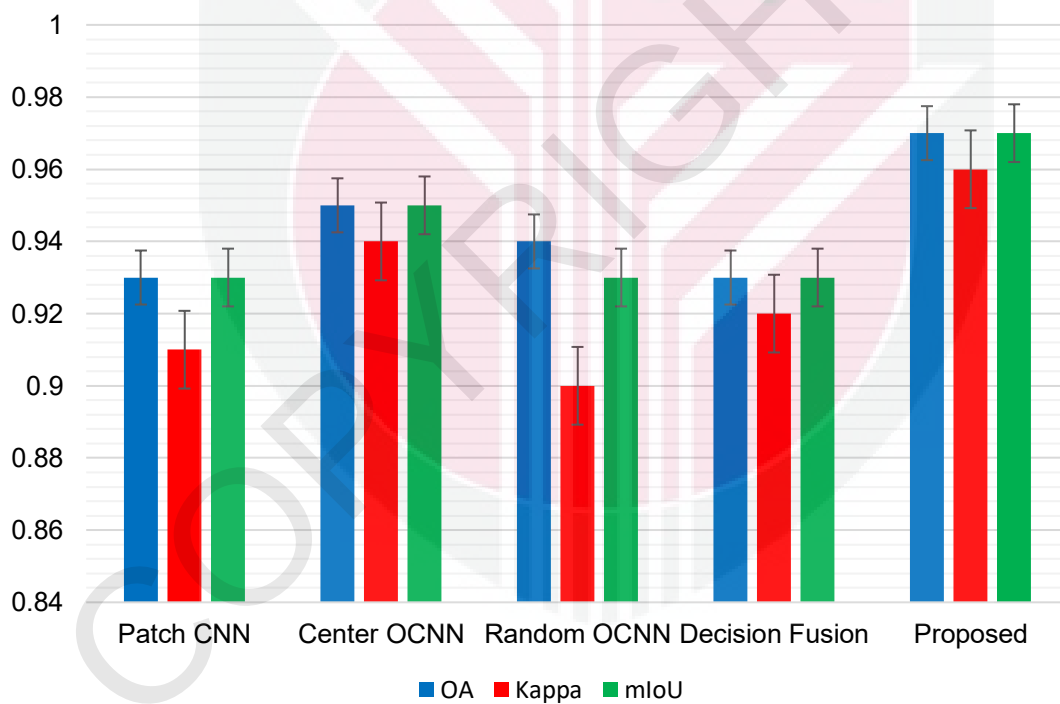


Figure 4.9 : The Overall Accuracy of the Proposed and Benchmark Classification Methods Based on Samples from the Testing Area

Table 4.7 : Per-Class Accuracies of the Proposed and Benchmark Classification Methods Based on Samples from the Training Area

Class	Patch CNN		Center OCNN		Random OCNN		Decision Fusion		Proposed	
	TR	TS	TR	TS	TR	TS	TR	TS	TR	TS
Buildings	0.85	0.9	0.96	0.96	0.85	0.96	0.84	0.92	0.97	0.97
Roads	0.85	0.91	0.87	0.93	0.83	0.94	0.83	0.89	0.94	0.98
Grass Land	0.91	0.89	0.95	0.94	0.94	0.89	0.8	0.9	0.95	0.94
Dense Vegetation/Trees	0.93	0.96	0.93	0.96	0.94	0.93	0.84	0.94	0.94	0.95
Water Body	0.96	0.99	0.98	0.97	0.98	0.98	0.93	0.98	0.98	0.99
Bare Land	0.93	0.97	0.98	0.98	0.98	0.98	0.89	0.95	0.98	0.98

TR : Training area

TS : Testing area

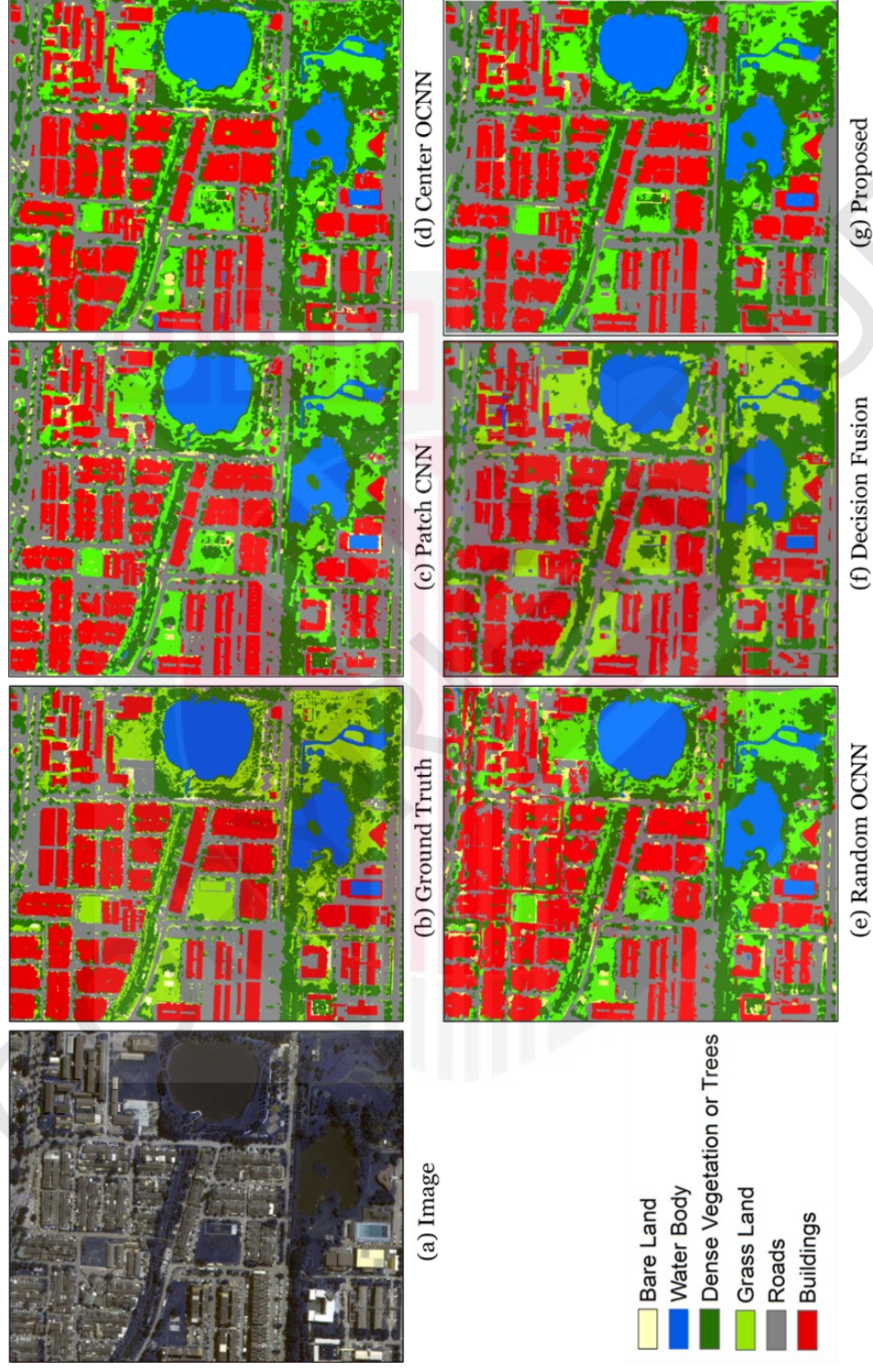


Figure 4.10 : Classification Maps of Different Methods Used In this Research for the Training Area

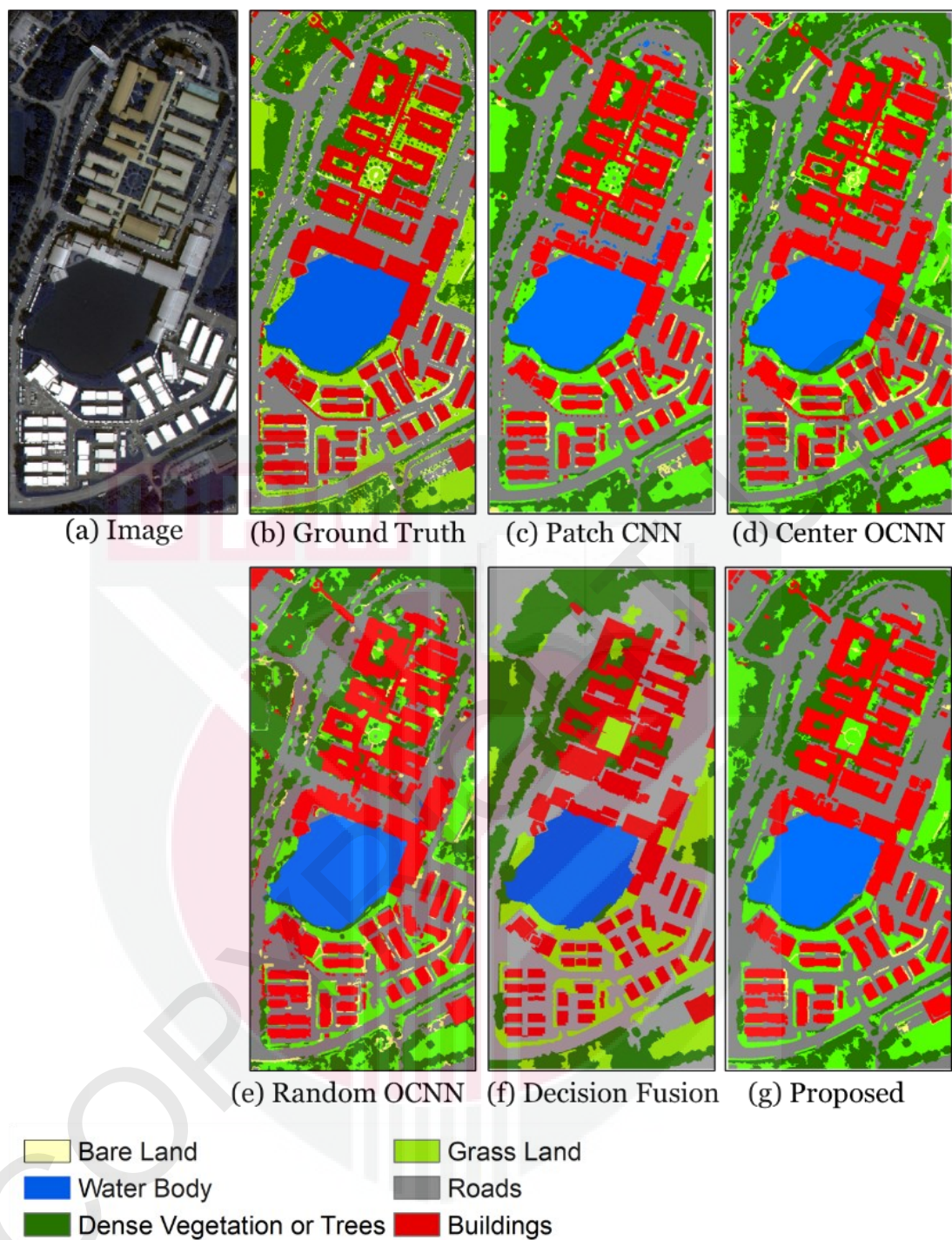


Figure 4.11 : Classification Maps of Different Methods Used in this Research for the Test Area

4.2.4 Sensitivity Analysis

4.2.4.1 Sensitivity Analysis of the Segmentation Parameters

The segmentation process played a vital role in the proposed classification model and its parameters (shape, scale, and compactness), which have a significant effect on the classification accuracy. This study optimized the segmentation parameters using the proposed joint Bayesian optimization approach. However, it is important to investigate how each parameter impacts the segmentation quality and ultimately the classification accuracy. Figure 4.12 presents the classification accuracy measured by mIoU based on different segmentation parameter values sampled by the Bayesian optimization method as expected improvement regions. As the optimization was performed jointly with the feature extraction, the results were presented for different base CNN models, namely, 2D CNN, 3D CNN, and HybridSN. With all three models, the results indicate that generally lower shape and compactness values yield better segmentation quality and classification accuracy. However, no systematic pattern was observed for the scale parameter. After n calls, the Bayesian optimization directed the search for regions of lower shape and compactness (<0.5). For the scale parameter, the optimal values ranged from 1000 to 5000, with no specific optimal region in between. The results highlight that as the area contains objects of different sizes, the optimal scale could be a small or large value. On the other hand, as the area is mostly covered by buildings of regular geometry (rectangular, square), lower values of shape and compactness achieved the best segmentation and classification accuracy in the area.

4.2.4.2 The Effect of Patch Size

The process of sensitivity analysis was conducted to investigate the effect of patch size on the accuracy (mIoU) of the proposed model with different base CNN models. The patch sizes varied from 3×3 to 9×9 , with a step size of 2. Table 4.9 presents the accuracies obtained from this experiment. It can be seen that a larger patch size increases the accuracy of the models for all the base CNNs (Paoletti et al., 2018, KAVZOĞLU et al., 2022). The smallest patch size (3×3) yielded accuracies of 0.93, 0.94, and 0.95 for the models, i.e., 2D CNN, 3D CNN, and HybridSN, respectively. Higher accuracies were obtained with larger patch sizes and the largest patch size (9×9) achieved accuracies as high as 0.94, 0.95, and 0.97 for the mentioned CNN models, respectively. Figure 4.13.

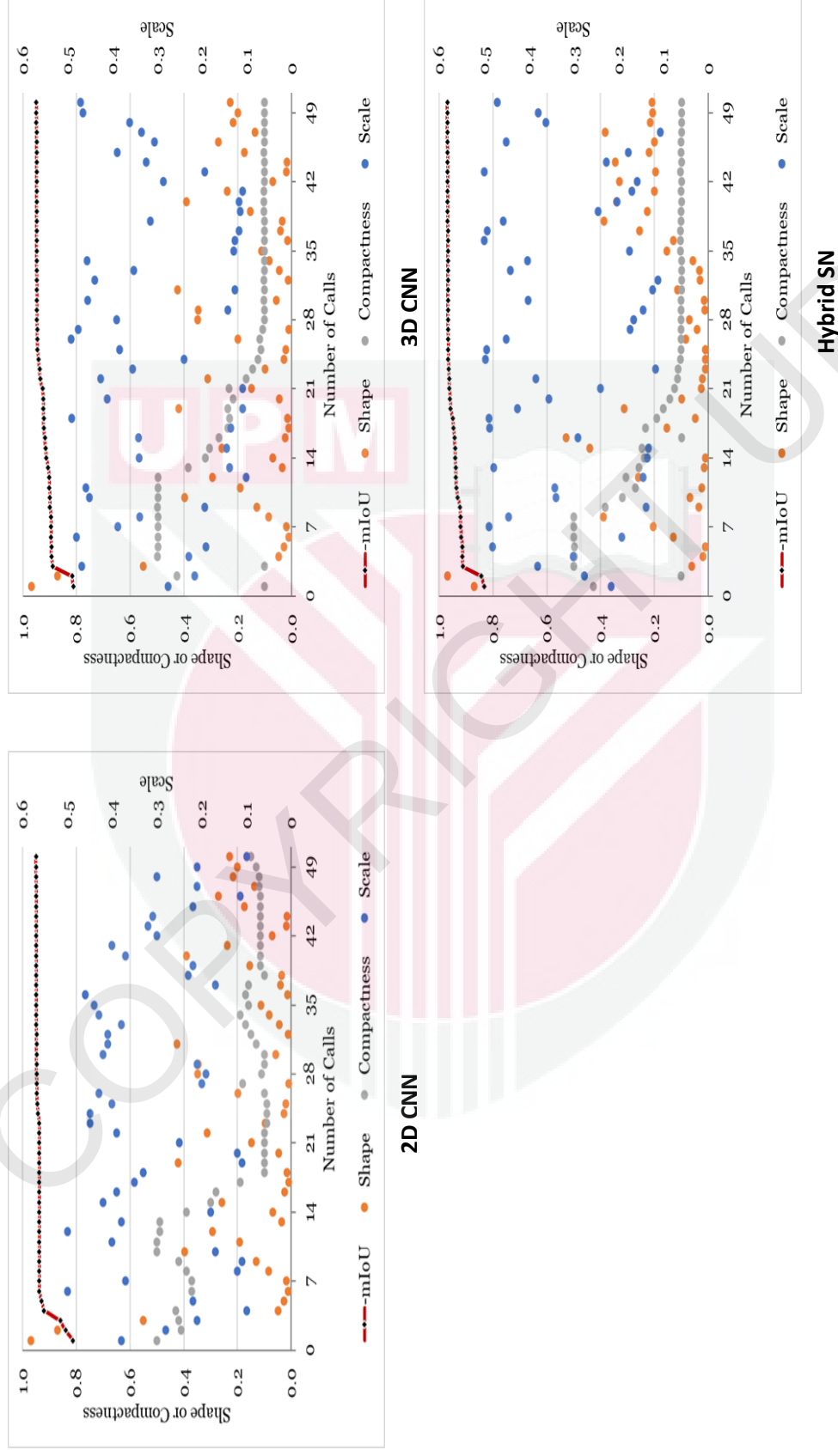


Figure 4.12 : Sensitivity Analysis on MRS Segmentation Parameters

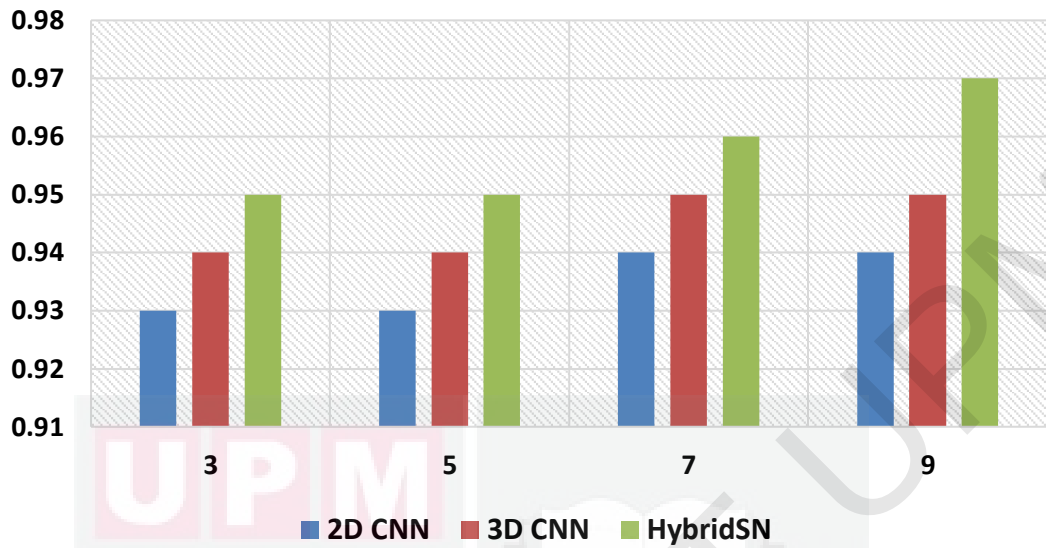


Figure 4.13 : Accuracy Assessment Based on mIoU of the Proposed Classification under Different CNN Models Using Patch Sizes from 3×3 to 9×9

4.2.5 Computational Efficiency

Table 4.8 shows the evaluation and comparison of the proposed framework with benchmark approaches in terms of computational efficiency. All analyses were conducted using the libraries (Keras and Tensorflow) within the Python environment (e.g., Python 3.7) and a computer laptop with Radeon Graphics (2.90 GHz), 16.0 GB of memory, an AMD Ryzen 7 4800H processor, and a 64-bit Windows 11 operating system. The networks' training was performed on the CPU. In each method, the processing times of different processing stages were calculated, including data preparation/preprocessing, image segmentation, model training, and postprocessing. As shown in Table 4.10, the image segmentation with its optimal parameters required only 2 s to obtain the image segments from the processed images. For the data preparation and pre-processing, all the methods except the Center OCNN and Random OCNN

methods required 5.3 s. This processing stage required a longer time for the Center OCNN and Random OCNN as they required additional processing of extracting segments' centers or random points within image segments, respectively. They took about 17.66 and 15.61 s, respectively. Regarding the training time for the base CNN models, the 2D CNN in Patch CNN, Decision Fusion, and the proposed framework, their training required 2.27 s. The Center OCNN and Random OCNN took much less time to train, 0.001 and 0.003 s, respectively, due to the lower number of training image patches. On the other hand, the proposed framework with the 3D CNN and HybridSN required the longest time to train, 11.13 and 12.14 s, respectively. Adding the time required for all the subprocesses, the total time required for each method shows that the patch CNN (7.57 s), followed by the proposed framework with 2D CNN (10.57 s), and the decision fusion (12.57 s) had the lowest time. The Center OCNN and Random OCNN had total processing times of 21.661 and 20.613 s, respectively.

The proposed framework with 3D CNN and HybridSN had the longest time of 23.63 and 24.64 s, respectively. Much more efficient implementation of the data pre-processing and post-processing can reduce the processing time of the methods that required a longer time.

Table 4.8 : The Computational Efficiency of the Proposed and Benchmark Classification Models Including Their Subprocesses

Model	Base CNN	Segmentation (s)	Data Preparation and Preprocessing (s)	Training Time (s/epoch)	Post-Processing Time (s)	Total (s)
Patch CNN	2D CNN	-	5.3	2.27	0	7.57
Center OCNN	2D CNN	2	17.66	0.001	2	21.661
Random OCNN	2D CNN	2	15.61	0.003	3	20.613
Decision Fusion	2D CNN	2	5.3	2.27	3	12.57
Proposed	2D CNN	2	5.3	2.27	5	10.57
Proposed	3D CNN	2	5.5	11.13	5	23.63
Proposed	HybridSN	2	5.5	12.14	5	24.64
Bayesian Optimization (per iteration)	-	-	-	13	-	13

4.2.6 Results of the Validation on Land Use/Land Cover Classification

In this study, we validated the classification results of the innovative framework based on orthorectified image. First, we performed image registration, ensuring accurate geometric and spatial matching between the base map and the classification maps. We performed the image geometric transformation using the first-order polynomial Affine. We achieved an RMSE of 1.25 and 0.32 meters on the training and testing areas, respectively. Tables 4.9 and 4.10 present the details, including the four GCPs and the residuals obtained from the process.

Table 4.9 : The Result of Image Registration of Training Area

Link	1	2	3	4
X source	556.8610	2334.4249	1496.9912	1315.1852
Y source	-1473.3085	-1446.9487	-878.4585	-2759.0943
X map	801270.9274	801800.5727	801549.3519	801494.5682
Y map	323578.7671	332584.9127	332755.6266	332187.9399
Residual x	0.8822	0.8725	-1.2119	-0.5428
Residual Y	0.8438	0.8346	-1.1592	-0.5192
Residual	1.2208	1.207	1.6771	0.7511

Table 4.10 : The Result of Image Registration of Testing Area

Link	1	2	3	4
X source	4517.5984	4816.7303	4592.0407	4249.5126
Y source	-279.7484	-746.8094	-2008.2600	-1857.7410
X map	802545.1123	802545.1123	802480.3378	802738.5259
Y map	332799.2582	332799.2582	332421.584	332467.110
Residual x	-0.1247	0.1555	-0.1599	0.1291
Residual Y	-0.2494	-0.3110	0.3198	-0.2582
Residual	0.2789	0.3477	0.3575	0.2887

Figures 4.14 and 4.15 show the base map (orthorectified image), the classification maps, the overlaid map, and the locations of the GCPs, as well as 10 randomly generated testing samples within the training and testing areas. As it can be seen, after image registration, the classification map accurately matches the base map. Based on the 10 testing samples, all the land cover types in the classification match the actual land use type in the base map. As a result, in addition to our previous accuracy assessment, this validation process shows the reliability of the proposed model for accurate classification from high-resolution satellite images.



Figure 4.14 : The Validation Results for the Training Area



Figure 4.15 : The Validation Results for the Testing Area

4.3 Results of Validation on Road Extraction

This study applied the proposed framework OCNN-JO, which has the same architecture and hyperparameters, for road extraction from satellite imagery. Figure 4.16 presents the road extraction map of the study area detected by the proposed model. The framework's accuracy assessment shows that it achieves 0.8637 OA and 0.8610 Kappa for road extraction. In addition, the framework also showed significant accuracy based on F1 score (0.8621) and mIoU (0.8546). Figure 4.17 shows the accuracy of the road extraction process. The results show that the framework is capable of extracting the roads accurately with few misdetections, especially the pixels around the road borders. As previously demonstrated, this study finds the proposed framework effective for both road extraction applications and LULC.

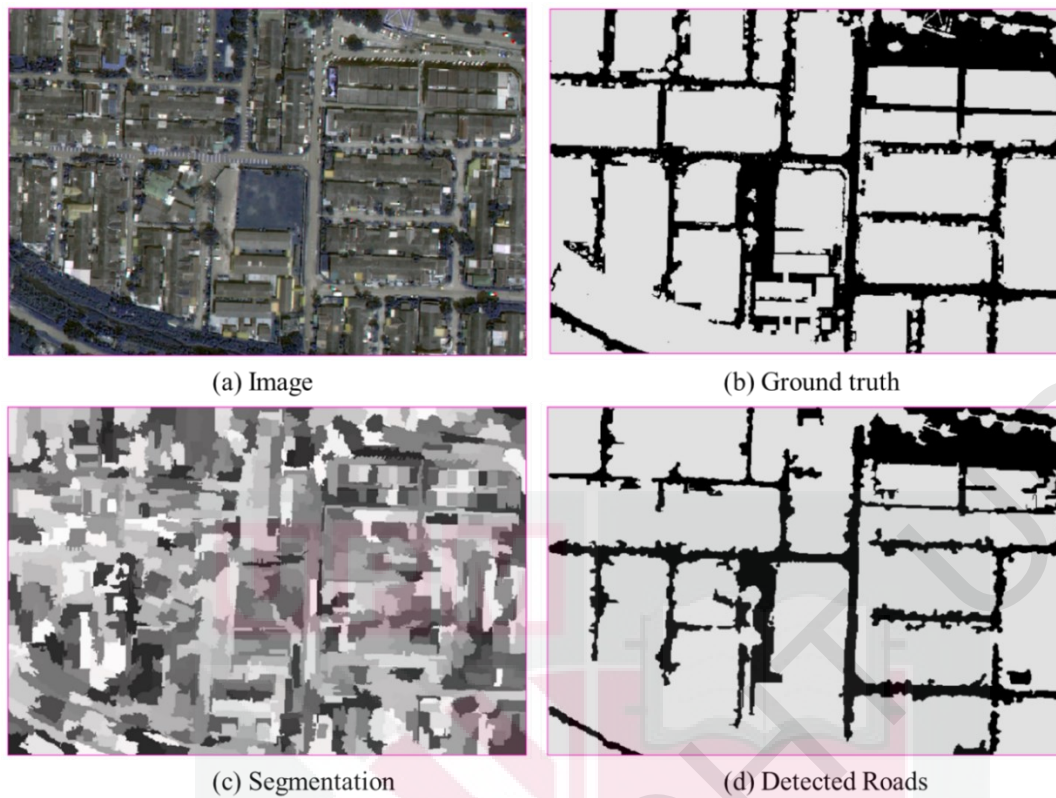


Figure 4.16 : Results of Road Extraction Based on the Proposed Model, (a) the Satellite Image (RGB True Color Composite), (b) Ground Truth Map, (c) Segmentation Map, and (d) Road Extraction Map of the Study Area

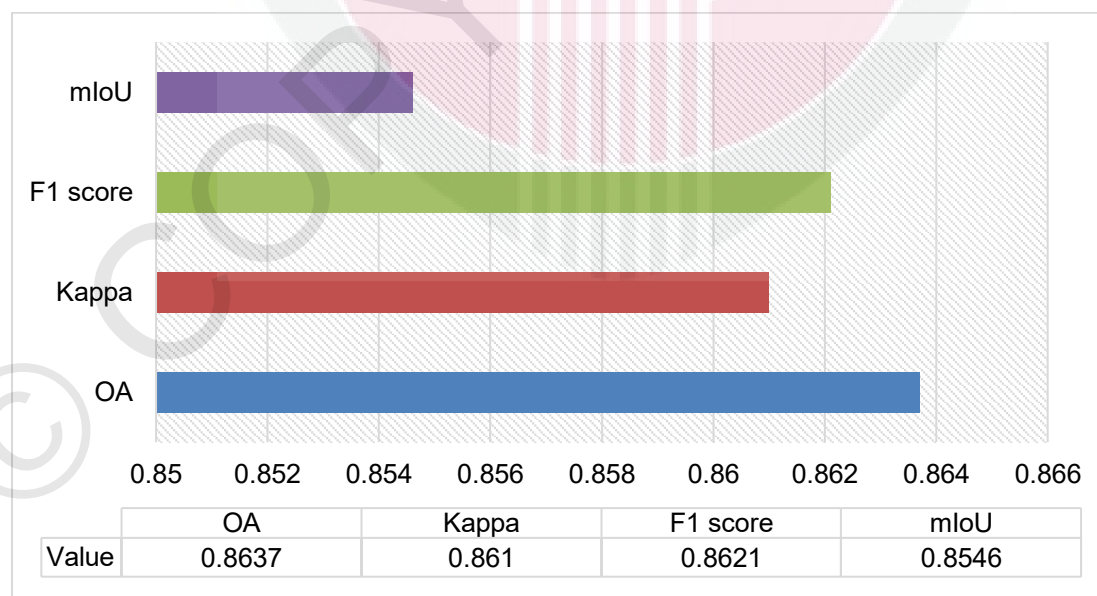


Figure 4.17 : Accuracy Assessment of the Proposed Framework for Road Extraction

4.4 Discussions

In deep learning models, pixels that belong to the same object are not enforced to be classified as the same semantic category. This reduces the artifacts in the class boundaries and the salt and pepper effect. However, deep learning has outstanding capability in learning abstract features that are generalizable to new areas. Therefore, deep learning-based classification models often achieve high overall accuracy. In contrast, OBIA through segmentation offers classification outputs that have few artifacts in the class boundaries and less salt and pepper effect. The combination of two methods can complement the strengths of each method, resulting in classification models with high accuracy and quality.

The integration frameworks evaluated in this research aim to blend the strengths of each method in a unique way. The deep learning based on object-level features, for example, combines the strengths of OBIA, i.e., semantic segmentation and providing detailed spatial and textural features, and the ability of the neural network to learn contextual features such that the final models have an understanding of the relationship between the features. The main limitation of this approach is that CNN and similar models are not suitable for tabular data. CNN models are incapable of capturing the spatial relationship among the table's cells (Hossain et al., 2024a). The feature fusion method aims to extract features by each method and use the combined features for classification. This method can lead to redundant information (Ubben et al., 2024), as OBIA and deep learning may learn the same spatial and textural features from the image data. In order to avoid the negative effects

of redundant information in these models, dimensionality reduction techniques may be necessary (Abdollahi et al., 2021). Or adding more layers with strong regularization after the concatenation layer, which combines the features from the OBIA and deep learning methods.

Decision fusion uses only segmentation from OBIA and features from the deep learning method. Thus, there are no impacts of redundant information. However, some features of OBIA that may be useful for classification tasks are neglected. Decision fusion methods are easier to compute and optimize than feature fusion method as there is no calculation of OBIA features (Chen et al., 2024). Finally, patch filtering methods are complex due to the requirement of matching between image objects and image patches. This framework involves several assumptions and abstractions stemming from the matching process between image objects and image patches. In addition, the patch filtering framework does not utilize OBIA features as the decision fusion framework. This method uses local semantics based on image patches, which may not take full advantage of the OBIA segmentation (Pan et al., 2019).

This study presented the advantages of combining OBIA and deep learning for very high-resolution satellite classification. Results showed that integrated frameworks, especially patch filtering, had a better result than the CNN-only methods and pixel-based ANN. Integrated models contributed to the reduction of artifacts in the class boundaries and the salt and pepper effect, which resulted in higher quality classification maps. In general, very-high-resolution (VHR) satellite images contain complex and heterogeneous objects which

require efficient feature extraction and classification models to convert them into meaningful thematic maps. Precise characterization of image objects is critical for robust representation of spatial contexts. In addition, appropriate feature extraction methods and the way image patches are extracted from image objects also play a significant role in producing accurate thematic maps from VHR images.

This study presents a joint optimization of image segmentation and deep feature learning for VHR satellite image classification. Previous methods applied segmentation optimization independently from deep learning feature extraction, which lacks knowledge sharing between the two tasks (Li et al., 2021). Using image segmentation and deep learning sequentially or combining them in post-processing can overcome several issues with pixel-level classification, such as "salt-and-pepper" noise, smoothness of object boundary, and separation of adjacent buildings (Vinod et al., 2023). However, joint optimization of segmentation and feature extraction can even further improve upon the traditional methods (Nababan et al., 2021). The proposed framework is different from those presented in previous works in multiple aspects: (1) jointly optimizing the extraction of image objects and feature learning; (2) utilizing hybrid CNN models to extract hierarchical features that combine 2D and 3D convolutional operations; and (3) using transfer learning to achieve efficient integration of segmentation and deep feature extraction.

In sequential methods, the errors are propagated from one task to another. For example, poor segmentation will result in less-accurate classification

because the latter task is based on the output of the former task. In this case, optimizing the segmentation is highly needed (KAVZOĞLU et al., 2022).

The problem with optimizing segmentation independently from feature extraction is that most deep learning methods utilize image segments as a spatial unit for extracting image patches and ultimately learning the image features. Considering segmentation, a separate step may lead to an optimized solution for specific integration workflows and not a general optimal solution (Pathak et al., 2017). With joint optimization, this issue can be overcome, and the optimal segmentation will always be obtained regardless of the integration workflow. In the methods that use post-processing to combine segmentation and classification predictions, poor segmentation continues to have a significant impact on the final classification. Not sharing knowledge between the two tasks results in learning fewer effective features (Sun et al., 2020). However, joint optimization showed feasibility in the modeling process. It was considered a complicated and time-consuming process due to the number of iterations during the parameter/hyperparameter updating. As a result, such processes require advanced methods such as Bayesian optimization that provide a smart technique to find optimal parameters/hyperparameters with the least number of iterations (Wu et al., 2019).

Figure 4.18 shows the improved land-cover extraction depending on the proposed OCNN framework compared to some traditional methods.

The pixel-based CNN algorithm that depends on image patches often weakens the boundary information of the land use, somewhat like the Gabor filter or morphological methods. In these methods, blurred boundaries occur between classified objects, resulting in a loss of useful land features. Object-based CNNs can overcome this problem. Object-based deep learning classification demonstrates a strong ability for classifying complex land uses through deep feature representation while maintaining the fine spatial details of image objects (Wang et al., 2020, Zaabar et al., 2022).

Accurate segmentation is very important for learning effective spatial features that can easily distinguish different land-use classes, especially for methods that are based on context patches, which are extracted based on the object center or random points within objects. The way these objects are created decides the image patches that are used for feature extraction by the deep learning models. Optimal segmentation requires not only the segmentation of small objects but also the prevention of over-segmentation of large objects. Furthermore, the parameters of the MRS algorithm can be used to control the boundary of image objects through shape and compactness. However, object-based CNNs require innovative use of appropriate functional units and convolutional processes based on image segmentation. The methods Center OCNN and Random OCNN share the same problem. The center point or the generated random points may exist near the boundary of the object due to the complexity of the object's outline shape. This will lead to some undesirable deep features that might be involved and thus severely affect classification. In Random OCNN, the effect of this problem can be decreased by using several

random points and taking the majority voting among the class labels predicted at those random points. However, this can significantly increase the model's computation efficiency. Some studies have attempted to overcome this and related issues by using the appropriate distribution of voting points within image objects with geometry conditions or advanced analysis of convolutional processes.

In this study, some parameters have been selected experimentally such as patch size. These parameters are related to image segmentation in object-based deep learning classification. Therefore, it is significant to be optimized jointly with the segmentation parameters. This can be investigated in future research either by our proposed Bayesian optimization approach or any other suitable optimization methods.

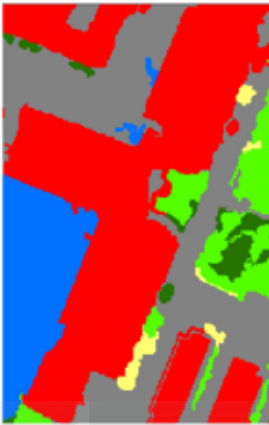
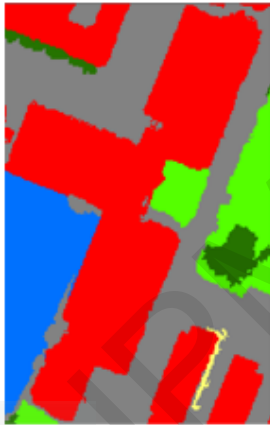
Patch CNN	Random OCNN	Proposed OCNN
		
		
		

Figure 4.18 : The Improved Land-Cover Extraction Depending on the Proposed OCNN Method Compared to Traditional Methods

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORKS

This chapter presents the summary of this thesis, the main findings, and the specific conclusions for the research objectives. Furthermore, it discusses some of the potential for future work based on the current study's limitations and weaknesses.

5.1 Conclusion

Deep learning-based image classification has become widespread, especially in remote sensing applications, due to its automatic and strong feature extraction capability. However, as deep learning methods operate on rectangular-shaped image patches, they cannot accurately extract objects' boundaries, especially in complex urban areas. As a result, combining deep learning and object-based image analysis (OBIA) has become a new avenue in remote sensing studies. Several integration frameworks have been proposed to combine OBIA with deep learning, such as deep learning based on object-level features, feature-level fusion, decision-level fusion, and deep learning with filtered patches. There is a significant lack of understanding of how these integration frameworks can be compared, and there have been no assessment studies that compare the effectiveness of each of the methods listed above. Nevertheless, researchers are pursuing different streams of study to enhance the performance of each method. Furthermore, the majority of these methods implement the optimization process separately. This often results in a knowledge gap between the segmentation and feature learning

processes; consequently, there is a lack of joint optimization between OBIA and CNN, which is necessary to achieve the best performance in the classification of remotely sensed data. On the other hand, despite the robust performance of combining OBIA and DL methods, there is a noticeable lack of implementation in transportation studies (e.g., road extraction from remotely sensed data). To that end, this study provides a methodology that includes three parts.

The first part involves assessing four OBIA-deep learning integration frameworks and comparing them to a patch-based convolutional neural network (CNN) and a pixel-based artificial neural network (ANN) for high-resolution image classification. The second part includes the innovation of a specific framework that depends on the decision-level fusion to resolve the fundamental issue with methods in this category, namely, that the "segmentation step is independent of feature extraction and classification." For precise image classification, The innovative framework includes a combination of CNN and OBIA based on joint segmentation parameter optimization and deep feature extraction. The proposed framework learns the best segmentation parameters and high-level features jointly. The final part includes the validation of the proposed framework for road extraction from high-resolution satellite images. The comparative assessment results indicated that patch filtering outperformed other frameworks such as decision-level, feature-level fusion, and OBIA CNN.

The patch filtering achieved the highest overall accuracy (OA) in both training and testing areas, 0.919 and 0.917, respectively. On the other hand, for the training area, decision-level fusion and feature fusion achieved 0.864 and 0.869, respectively. In the testing area, decision-level fusion and feature fusion achieved 0.862 and 0.860, respectively. In contrast, the OBIA CNN framework scored the lowest OA in both training and testing areas, which achieved 0.838 and 0.842, respectively. The classification process resulted in six distinct classes: buildings, roads, grassland, dense vegetation or trees, waterbodies, and bare land. The OBIA CNN framework best classified the building class (0.93), while the patch filtering framework best classified the roads class (0.84), according to the per-class accuracy. Furthermore, the patch filtering framework outperformed other frameworks in grasslands classification (0.82). On the other hand, patch filtering accurately classified the class of dense vegetation or trees (0.89). Conversely, the patch filtering method best classified the waterbody class (0.97) and the bare land (0.93).

Three MRS segmentation parameters were optimized using the Bayesian technique (scale, shape, and compactness). The best scales were 4705 and 4617 for the training and test areas, respectively. The best shape and compactness values were 0.2 and 0.1 for the training area and 0.14 and 0.1 for the test area, respectively. The best parameter values for the training area are very close to those for the test area. In both areas, higher scale value and lower shape and compactness are preferred.

The innovative classification framework achieved the best accuracy, with 0.96 OA, 0.95 Kappa, and 0.96 mIoU in the training area and 0.97 OA, 0.96 Kappa, and 0.97 mIoU in the test area, outperforming several benchmark methods including Patch CNN, Center OCNN, Random OCNN, and Decision Fusion. The analysis of CNN variants within the proposed classification framework showed that the HybridSN model achieved the best results compared to 2D and 3D CNNs. The 3D CNN and the combined 3D-2D CNN layers (HybridSN) yielded slightly better accuracies than the 2D CNN layers regarding geometric fidelity, object boundary extraction, and separation of adjacent objects. The Bayesian optimization could find comparable optimal MRS parameters for the training and test areas, with excellent quality measured by AFI (0.046, 0.037) and QR (0.945, 0.932). In the proposed framework, higher accuracies could be obtained with larger patch sizes (e.g., 9×9 compared to 3×3). Moreover, the proposed framework is computationally efficient, with the longest training being fewer than 25 s considering all the sub-processes and a single training epoch. The validation process shows the reliability of the proposed model for accurate classification from high-resolution satellite images. The framework's accuracy assessment shows that it achieves 0.8637 OA and 0.8610 Kappa for road extraction. In addition, the framework also showed significant accuracy based on F1 score (0.8621) and mIoU (0.8546). The results show that the framework is capable of extracting the roads accurately with few misdetections, especially the pixels around the road borders. Therefore, urban and environmental applications that depend on VHR satellite images and necessitate land use information can utilize the proposed framework.

5.2 Recommendations for Future Work

Despite the excellent results achieved by the proposed framework, this thesis suggests some areas for improvement in future work:

1. Additional research should be carried out to improve the patch filtering framework by finding better methods to merge image objects and image patches. Moreover, ensemble frameworks may also be applied. For example, OBIA features can be used in decision fusion methods by refining with segmentation a classification map produced with an integrated model (e.g., feature fusion).
2. According to the results listed above, the proposed classification framework can serve as an efficient tool for extracting land-cover information from VHR satellite imagery. In addition, the proposed framework can be used for a wide range of urban and environmental applications that are based on remote sensing data or use land-cover products as a data layer. However, further improvements can be made to jointly optimize patch size and other network-related parameters with the MRS parameters and feature extraction models.
3. Further research is required to optimize the proposed framework for applications such as road geometry characterization, building footprint extraction, and urban tree segmentation.

REFERENCES

- Abbasi, M., Hosseiny, B., Stewart, R. A., Kalantari, M., Patorniti, N., Mostafa, S., & Awrangjeb, M. (2024). Multi-temporal change detection of asbestos roofing: A hybrid object-based deep learning framework with post-classification structure. *Remote Sensing Applications: Society and Environment*, 34, 101167.
- Abdi, G., Samadzadegan, F., & Reinartz, P. (2018). Deep learning decision fusion for the classification of urban remote sensing data. *Journal of Applied Remote Sensing*, 12(1), 016038-016038.
- 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, 569-583.
- Agatonovic-Kustrin, S., & Beresford, R. (2000). Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research. *Journal of pharmaceutical and biomedical analysis*, 22(5), 717-727.
- Aguilar, M. A., Aguilar, F. J., García Lorca, A., Guirado, E., Betlej, M., Cichón, P., ... & Parente, C. (2016). Assessment of multiresolution segmentation for extracting greenhouses from WorldView-2 imagery. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 41, 145-152.
- Alaieri, F. (2024). Precision Agriculture based on Machine Learning and Remote Sensing Techniques. *Engineering, Technology & Applied Science Research*, 14(3), 14206-14211.
- Ameslek, O., Zahir, H., Mitro, S., & Bachaoui, E. M. (2023). Identification and mapping of Individual Trees from Unmanned Aerial Vehicle Imagery Using an object oriented Convolutional Neural Network. <https://doi.org/10.21203/rs.3.rs-3486635/v1>
- Anderson, J. R. (1976). A land use and land cover classification system for use with remote sensor data (Vol. 964). US Government Printing Office. <https://doi.org/10.3133/pp964>
- Arrechea-Castillo, D. A., Solano-Correa, Y. T., Muñoz-Ordóñez, J. F., Pencue-Fierro, E. L., & Figueroa-Casas, A. (2023). Multiclass Land Use and Land Cover Classification of Andean Sub-Basins in Colombia with Sentinel-2 and Deep Learning. *Remote Sensing*, 15(10), 2521.
- Atef, I., Ahmed, W., & Abdel-Maguid, R. H. (2023). Modelling of land use land cover changes using machine learning and GIS techniques: a case study in El-Fayoum Governorate, Egypt. *Environmental Monitoring and Assessment*, 195(6), 637.

- Atik, S. O., & Ipbuker, C. (2021). Integrating convolutional neural network and multiresolution segmentation for land cover and land use mapping using satellite imagery. *Applied Sciences*, 11(12), 5551.
- Attaf, D., Djerriri, K., Cheriguene, S. R., & Karoui, M. S. (2018, October). One-dimensional convolution neural networks for object-based feature selection. In *Image and Signal Processing for Remote Sensing XXIV* (Vol. 10789, pp. 505-512). SPIE.
- Baatz, M. (2000). Multiresolution segmentation: an optimization approach for high quality multi-scale image segmentation. *Angewandte geographische informationsverarbeitung*, 12-23.
<https://cir.nii.ac.jp/crid/1572261550679971840>
- Bai, T., An, Q., Deng, S., Li, P., Chen, Y., Sun, K., ... & Song, Z. (2024). A Novel UNet 3+ Change Detection Method Considering Scale Uncertainty in High-Resolution Imagery. *Remote Sensing*, 16(11), 1846
- Bebis, G., & Georgiopoulos, M. (1994). Feed-forward neural networks. *Ieee Potentials*, 13(4), 27-31.
- Bengoufa, S., Niculescu, S., Mihoubi, M. K., Belkessa, R., & Abbad, K. (2021). Rocky shoreline extraction using a deep learning model and object-based image analysis. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 43, 23-29.
- Berezina, P., & Liu, D. (2022). Hurricane damage assessment using coupled convolutional neural networks: a case study of hurricane Michael. *Geomatics, Natural Hazards and Risk*, 13(1), 414-431.
- Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2), 281-350.
- Bhandari, A. K., Kumar, A., & Singh, G. K. (2012). Feature extraction using Normalized Difference Vegetation Index (NDVI): A case study of Jabalpur city. *Procedia technology*, 6, 612-621.
- Blaschke, T. (2010). Object based image analysis for remote sensing. *ISPRS journal of photogrammetry and remote sensing*, 65(1), 2-16.
- Boykov, Y., & Funka-Lea, G. (2006). Graph cuts and efficient ND image segmentation. *International journal of computer vision*, 70(2), 109-131.
- Brochu, E., Cora, V. M., & De Freitas, N. (2010). A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*. <https://doi.org/10.48550/arXiv.1012.2599>
- Bycroft, R., Leon, J. X., & Schoeman, D. (2019). Comparing random forests and convoluted neural networks for mapping ghost crab burrows using imagery from an unmanned aerial vehicle. *Estuarine, Coastal and Shelf*

Science, 224, 84-93.

- Cai, G., Ren, H., Yang, L., Zhang, N., Du, M., & Wu, C. (2019). Detailed urban land use land cover classification at the metropolitan scale using a three-layer classification scheme. *Sensors*, 19(14), 3120.
- Chen, D., & Stow, D. (2002). The effect of training strategies on supervised classification at different spatial resolutions. *Photogrammetric Engineering and Remote Sensing*, 68(11), 1155-1162.
- Chen, T., Trinder, J. C., & Niu, R. (2017). Object-oriented landslide mapping using ZY-3 satellite imagery, random forest and mathematical morphology, for the Three-Gorges Reservoir, China. *Remote sensing*, 9(4), 333.
- Chen, Y., Tang, L., Yang, X., Bilal, M., & Li, Q. (2020). Object-based multi-modal convolution neural networks for building extraction using panchromatic and multispectral imagery. *Neurocomputing*, 386, 136-146.
- Chen, Z., Chen, J., Yue, Y., Lan, Y., Ling, M., Li, X., ... & Zhou, G. (2024). Tradeoffs among multi-source remote sensing images, spatial resolution, and accuracy for the classification of wetland plant species and surface objects based on the MRS_DeepLabV3+ model. *Ecological Informatics*, 81, 102594.
- Cheng, Y. (1995). Mean shift, mode seeking, and clustering. *IEEE transactions on pattern analysis and machine intelligence*, 17(8), 790-799.
- Coll, C., Galve, J. M., Sanchez, J. M., & Caselles, V. (2009). Validation of Landsat-7/ETM+ thermal-band calibration and atmospheric correction with ground-based measurements. *IEEE Transactions on Geoscience and Remote Sensing*, 48(1), 547-555.
- Cui, W., Zheng, Z., Zhou, Q., Huang, J., & Yuan, Y. (2018). Application of a parallel spectral-spatial convolution neural network in object-oriented remote sensing land use classification. *Remote Sensing Letters*, 9(4), 334-342.
- Dawei, L. I. U., & Shujing, G. A. O. (2021, August). Object Oriented Road Extraction from Remote Sensing Images Using Improved Watershed Segmentation. In *Journal of Physics: Conference Series* (Vol. 2005, No. 1, p. 012077). IOP Publishing. <https://dx.doi.org/10.1088/1742-6596/2005/1/012077>
- Definiens, A. G. (2006). User Guide for Definiens Professional 5. München. Germany. <https://www.yumpu.com/en/document/read/5092858/definiens-professional-5-installationguidepdf>
- Derivaux, S., Lefevre, S., Wemmert, C., & Korczak, J. (2006, July). Watershed segmentation of remotely sensed images based on a supervised fuzzy

- pixel classification. In 2006 IEEE International Symposium on Geoscience and Remote Sensing (pp. 3712-3715). IEEE. <https://doi.org/10.1109/IGARSS.2006.951>
- Detka, J., Coyle, H., Gomez, M., & Gilbert, G. S. (2023). A Drone-Powered Deep Learning Methodology for High Precision Remote Sensing in California's Coastal Shrubs. *Drones*, 7(7), 421.
- Dewali, S. K., Jain, K., Varshney, D., Dhamija, S., & Pundir, E. (2023). Combining OBIA, CNN, and UAV photogrammetry for automated avalanche deposit detection and characterization. *Advances in Space Research*, 72(8), 3109-3132.
- Dharmarajan, R., & Kannan, K. (2010). A hypergraph-based algorithm for image restoration from salt and pepper noise. *AEU-International Journal of Electronics and Communications*, 64(12), 1114-1122.
- Difference in data mining vs Machine Learning vs Artificial Intelligence. *Software Testing Help*. (2023, June 20). <https://www.softwaretestinghelp.com/data-mining-vs-machine-learning-vs-ai/>
- Du, X., Cai, Y., Wang, S., & Zhang, L. (2016, November). Overview of deep learning. In 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC) (pp. 159-164). IEEE.
- El-naggar, A. M. (2018). Determination of optimum segmentation parameter values for extracting building from remote sensing images. *Alexandria engineering journal*, 57(4), 3089-3097.
- Elshaw, R., Maher, M., & Sakr, S. (2019). Automated machine learning: State-of-the-art and open challenges. *arXiv preprint arXiv:1906.02287*. <https://doi.org/10.48550/arXiv.1906.02287>
- Eugenio González, F., & Marcello Ruiz, F. J. (2019). Very High Resolution (VHR) Satellite Imagery. Processing and Applications. MDPI. <https://doi.org/10.3390/books978-3-03921-757-1>
- Firat, H., Asker, M. E., Bayindir, M. İ., & Hanbay, D. (2023a). 3D residual spatial-spectral convolution network for hyperspectral remote sensing image classification. *Neural Computing and Applications*, 35(6), 4479-4497.
- Firat, H., Asker, M. E., Bayındır, M. İ., & Hanbay, D. (2023b). Hybrid 3D/2D complete inception module and convolutional neural network for hyperspectral remote sensing image classification. *Neural Processing Letters*, 55(2), 1087-1130.
- Franklin, S. E. (2001). Remote sensing for sustainable forest management. CRC press. <https://doi.org/10.1201/9781420032857>

- Frazier, P. I. (2018). A tutorial on Bayesian optimization. arXiv preprint arXiv:1807.02811. <https://doi.org/10.48550/arXiv.1807.02811>
- Fu, T., Ma, L., Li, M., & Johnson, B. A. (2018). Using convolutional neural network to identify irregular segmentation objects from very high-resolution remote sensing imagery. *Journal of Applied Remote Sensing*, 12(2), 025010-025010.
- Fu, Y., Liu, K., Shen, Z., Deng, J., Gan, M., Liu, X., Lu, D., & Wang, K. (2019). Mapping impervious surfaces in town–rural transition belts using China’s GF-2 imagery and object-based DEEP CNNs. *Remote Sensing*, 11(3), 280.
- Gandhimathi Alias Usha, S., & Vasuki, S. (2018). Improved segmentation and change detection of multi-spectral satellite imagery using graph cut based clustering and multiclass SVM. *Multimedia Tools and Applications*, 77, 15353-15383.
- Garajeh, M. K., Feizizadeh, B., Weng, Q., Moghaddam, M. H. R., & Garajeh, A. K. (2022). Desert landform detection and mapping using a semi-automated object-based image analysis approach. *Journal of Arid Environments*, 199, 104721.
- Ge, S., Lu, J., Gu, H., Yuan, Z., & Su, W. (2017, September). Polarimetric SAR image classification based on deep belief network and superpixel segmentation. In *2017 3rd International Conference on Frontiers of Signal Processing (ICFSP)* (pp. 114-119). IEEE.
- Geiß, C., Zhu, Y., Qiu, C., Mou, L., Zhu, X. X., & Taubenböck, H. (2020). Deep Relearning in the Geospatial Domain for Semantic Remote Sensing Image Segmentation. *IEEE Geoscience and Remote Sensing Letters*, 19, 1-5.
- Ghassabeh, Y. A. (2015). A sufficient condition for the convergence of the mean shift algorithm with Gaussian kernel. *Journal of Multivariate Analysis*, 135, 1-10.
- Ghorbanzadeh, O., Crivellari, A., Tiede, D., Ghamisi, P., & Lang, S. (2022 a). Mapping Dwellings in IDP/Refugee Settlements Using Deep Learning. *Remote Sensing*, 14(24), 6382.
- Ghorbanzadeh, O., Shahabi, H., Crivellari, A., Homayouni, S., Blaschke, T., & Ghamisi, P. (2022 b). Landslide detection using deep learning and object-based image analysis. *Landslides*, 19(4), 929-939.
- Ghorbanzadeh, O., Tiede, D., Wendt, L., Sudmanns, M., & Lang, S. (2021). Transferable instance segmentation of dwellings in a refugee camp-integrating CNN and OBIA. *European Journal of Remote Sensing*, 54(sup1), 127-140.
- Ghorbanzadeh, O., Gholamnia, K., & Ghamisi, P. (2023). The application of ResU-net and OBIA for landslide detection from multi-temporal

- sentinel-2 images. *Big Earth Data*, 7(4), 961-985.
- Giles, A. B., Ren, K., Davies, J. E., Abrego, D., & Kelaher, B. (2023). Combining drones and deep learning to automate coral reef assessment with RGB imagery. *Remote Sensing*, 15(9), 2238.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, 27.
- Guirado, E., Blanco-Sacristán, J., Rodríguez-Caballero, E., Tabik, S., Alcaraz-Segura, D., Martínez-Valderrama, J., & Cabello, J. (2021). Mask R-CNN and OBIA Fusion Improves the Segmentation of Scattered Vegetation in Very High-Resolution Optical Sensors. *Sensors*, 21(1), 320.
- Hamza, A., Khan, M. A., Ur Rehman, S., Albarakati, H. M., Alroobaea, R., Baqasah, A. M., ... & Masood, A. (2023). An integrated parallel inner deep learning models information fusion with Bayesian optimization for land scene classification in satellite images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*.
- Hardini, I. R. (2020, November). Comparative analysis of mean-shift based object tracking using simulated annealing and locust search algorithm approaches. In *2020 International Conference on ICT for Smart Society (ICISS)* (pp. 1-6). IEEE.
- He, S., Du, H., Zhou, G., Li, X., Mao, F., Zhu, D., Xu, Y., Zhang, M., Huang, Z., Liu, H., & Luo, X. (2020). Intelligent Mapping of Urban Forests from High-Resolution Remotely Sensed Imagery Using Object-Based U-Net-DenseNet-Coupled Network. *Remote Sensing*, 12(23), 3928.
- Herold, M., Couclelis, H., & Clarke, K. C. (2005). The role of spatial metrics in the analysis and modeling of urban land use change. *Computers, environment and urban systems*, 29(4), 369-399.
- Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, 18(7), 1527-1554.
- Hong, L., & Zhang, M. (2020). Object-oriented multiscale deep features for hyperspectral image classification. *International Journal of Remote Sensing*, 41(14), 5549-5572.
- Hongtao, L., & Qinchuan, Z. (2016). Applications of deep convolutional neural network in computer vision. *Journal of Data Acquisition and Processing*, 31(1), 1-17.
- Hossain, M. D., & Chen, D. (2019). Segmentation for Object-Based Image Analysis (OBIA): A review of algorithms and challenges from remote sensing perspective. *ISPRS Journal of Photogrammetry and Remote Sensing*, 150, 115-134.

- Hossain, M. D., & Chen, D. (2024a). Performance Comparison of Deep Learning (DL)-Based Tabular Models for Building Mapping Using High-Resolution Red, Green, and Blue Imagery and the Geographic Object-Based Image Analysis Framework. *Remote Sensing*, 16(5), 878.
- Hossain, M. D., & Chen, D. (2024 b). Target-based Building Extraction from High-Resolution RGB Imagery using GEOBIA Framework and Tabular Deep Learning Model. *Geomatica*, 76(1), 100007.
- Huang, B., Zhao, B., & Song, Y. (2018). Urban land-use mapping using a deep convolutional neural network with high spatial resolution multispectral remote sensing imagery. *Remote Sensing of Environment*, 214, 73-86.
- Sussi, N., Husni, E., Yusuf, R., Harto, A. B., Suwardhi, D., & Siburian, A. (2024). Utilization of Improved Annotations from Object-Based Image Analysis as Training Data for DeepLab V3+ Model: A Focus on Road Extraction in Very High-Resolution Orthophotos. *IEEE Access*, 12(1), 67910 – 67923.
- Ienco, D., Gaetano, R., Dupaquier, C., & Maurel, P. (2017). Land cover classification via multitemporal spatial data by deep recurrent neural networks. *IEEE Geoscience and Remote Sensing Letters*, 14(10), 1685-1689.
- Ienco, D., Gbodjo, Y. J. E., Gaetano, R., & Interdonato, R. (2020). Weakly supervised learning for land cover mapping of satellite image time series via attention-based CNN. *IEEE Access*, 8, 179547-179560.
- Ismayilova, I., & Timpf, S. (2023). Semantic Identification of Urban Green Spaces: Forest. *AGILE: GIScience Series*, 4, 28.
- Jain, A. K., Mao, J., & Mohiuddin, K. M. (1996). Artificial neural networks: A tutorial. *Computer*, 29(3), 31-44.
- Jena, R., & Pradhan, B. (2020). Integrated ANN-cross-validation and AHP-TOPSIS model to improve earthquake risk assessment. *International Journal of Disaster Risk Reduction*, 50, 101723.
- Jensen, J.R. (1996) *Introductory Digital Image Processing: A Remote Sensing Perspective*. 2nd Edition, Prentice Hall, Inc., Upper Saddle River, NJ., 197-256.
- Ji, S., Shen, Y., Lu, M., & Zhang, Y. (2019). Building instance change detection from large-scale aerial images using convolutional neural networks and simulated samples. *Remote Sensing*, 11(11), 1343.
- Jin, B., Ye, P., Zhang, X., Song, W., & Li, S. (2019). Object-oriented method combined with deep convolutional neural networks for land-use-type classification of remote sensing images. *Journal of the Indian Society of Remote Sensing*, 47, 951-965.

- Jones, D. R., Schonlau, M., & Welch, W. J. (1998). Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13, 455-492.
- Joshaghani, M., Davari, A., Hatamian, F. N., Maier, A., & Riess, C. (2023). Bayesian convolutional neural networks for limited data hyperspectral remote sensing image classification. *IEEE Geoscience and Remote Sensing Letters*, 20, 1-5.
- Jozdani, S. E., Johnson, B. A., & Chen, D. (2019). Comparing deep neural networks, ensemble classifiers, and support vector machine algorithms for object-based urban land use/land cover classification. *Remote Sensing*, 11(14), 1713.
- KAVZOĞLU, T., & YILMAZ, E. Ö. (2022). Analysis of patch and sample size effects for 2D-3D CNN models using multiplatform dataset: hyperspectral image classification of ROSIS and Jilin-1 GP01 imagery. *Turkish Journal of Electrical Engineering and Computer Sciences*, 30(6), 2124-2144.
- Karimi, F., & Sultana, S. (2024). Urban Expansion Prediction and Land Use/Land Cover Change Modeling for Sustainable Urban Development. *Sustainability*, 16(6), 2285.
- Khatami, R., Mountrakis, G., & Stehman, S. V. (2016). A meta-analysis of remote sensing research on supervised pixel-based land-cover image classification processes: General guidelines for practitioners and future research. *Remote sensing of environment*, 177, 89-100.
- KRAYEM, D., FALLAHA, A., ASHOUR, M. T. E., & ALQARALEH, S. (2022). CNN HYPERPARAMETERS OPTIMIZATION USING RANDOM SEARCH FOR IMAGE CLASSIFICATION. Özseven, T (Eds.), 6th International Symposium on Innovative Approaches in Smart Technologies (pp.34-37).SETSCI. <https://doi.org/10.36287/setsoci.5.2.007>
- Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*. <https://doi.org/10.48550/arXiv.1412.6980>
- Kong, J., Ryu, Y., Jeong, S., Zhong, Z., Choi, W., Kim, J., Lee, K., Lim, J., Jang, K., Chun, J., Kim, K. M., & Houborg, R. (2023). Super resolution of historic Landsat imagery using a dual generative adversarial network (GAN) model with CubeSat constellation imagery for spatially enhanced long-term vegetation monitoring. *ISPRS Journal of Photogrammetry and Remote Sensing*, 200, 1–23.
- Lam, O. H. Y., Dogotari, M., Prüm, M., Vithlani, H. N., Roers, C., Melville, B., Zimmer, F., & Becker, R. (2020). An open source workflow for weed mapping in native grassland using unmanned aerial vehicle: using *Rumex obtusifolius* as a case study. *European Journal of Remote*

Sensing, 54(sup1), 71–88.

- Landgrebe, D. A. (2003). *Signal theory methods in multispectral remote sensing* (Vol. 24). John Wiley & Sons.
- Långkvist, M., Kiselev, A., Alirezaie, M., & Loutfi, A. (2016). Classification and segmentation of satellite orthoimagery using convolutional neural networks. *Remote Sensing*, 8(4), 329.
- Lakhal, M. I., Çevikalp, H., Escalera, S., & Ofli, F. (2018). Recurrent neural networks for remote sensing image classification. *IET Computer Vision*, 12(7), 1040-1045.
- Lassalle, G., Ferreira, M. P., La Rosa, L. E. C., & de Souza Filho, C. R. (2022). Deep learning-based individual tree crown delineation in mangrove forests using very-high-resolution satellite imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 189, 220-235.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436-444.
- Lee, T., Lee, K. B., & Kim, C. O. (2016). Performance of machine learning algorithms for class-imbalanced process fault detection problems. *IEEE Transactions on Semiconductor Manufacturing*, 29(4), 436-445.
- Lefsky, M. A., & Cohen, W. B. (2003). Selection of remotely sensed data. In *Remote sensing of forest environments: concepts and case studies* (pp. 13-46). Boston, MA: Springer US.
- Lekfuangfu, A., Kasetkasem, T., Rakwatin, P., Tanarat, S., Kumazawa, I., & Chanwimaluang, T. (2016, June). Land cover mapping classification based on multi Restricted Boltzmann machines and Support Vector Machines. In *2016 13th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)* (pp. 1-5). IEEE.
- Lemenkova, P. (2024). Random Forest Classifier Algorithm of Geographic Resources Analysis Support System Geographic Information System for Satellite Image Processing: Case Study of Bight of Sofala, Mozambique. *Coasts*, 4(1), 127-149.
- Li, M., Dai, W., Fan, M., Qian, W., Yang, X., Tao, Y., & Zhao, C. (2023). Combining Deep Learning and Hydrological Analysis for Identifying Check Dam Systems from Remote Sensing Images and DEMs in the Yellow River Basin. *International Journal of Environmental Research and Public Health*, 20(5), 4636.
- Li, H., Tian, Y., Zhang, C., Zhang, S., & Atkinson, P. M. (2022 a). Temporal sequence Object-based CNN (TS-OCNN) for crop classification from fine resolution remote sensing image time-series. *The Crop Journal*, 10(5), 1507-1516.

- Li, H., Zhang, C., Zhang, S., & Atkinson, P. M. (2019). A hybrid OSVM-OCNN method for crop classification from fine spatial resolution remotely sensed imagery. *Remote Sensing*, 11(20), 2370.
- Li, H., Zhang, C., Zhang, Y., Zhang, S., Ding, X., & Atkinson, P. M. (2021). A Scale Sequence Object-based Convolutional Neural Network (SS-OCNN) for crop classification from fine spatial resolution remotely sensed imagery. *International Journal of Digital Earth*, 14(11), 1528-1546.
- Li, L., Han, L., Hu, H., Liu, Z., & Cao, H. (2020). Standardized object-based dual CNNs for very high-resolution remote sensing image classification and standardization combination effect analysis. *International Journal of Remote Sensing*, 41(17), 6635-6663.
- Li, M., Zang, S., Zhang, B., Li, S., & Wu, C. (2014). A review of remote sensing image classification techniques: The role of spatio-contextual information. *European Journal of Remote Sensing*, 47(1), 389-411.
- Li, W., Fu, H., Yu, L., Gong, P., Feng, D., Li, C., & Clinton, N. (2016). Stacked Autoencoder-based deep learning for remote-sensing image classification: a case study of African land-cover mapping. *International journal of remote sensing*, 37(23), 5632-5646.
- Li, Y., Ouyang, S., & Zhang, Y. (2022 b). Combining deep learning and ontology reasoning for remote sensing image semantic segmentation. *Knowledge-Based Systems*, 243, 108469.
- Lilay, M. Y., & Taye, G. D. (2023). Semantic segmentation model for land cover classification from satellite images in Gambella National Park, Ethiopia. *SN Applied Sciences*, 5(3), 76.
- Lin, Y., Lv, F., Zhu, S., Yang, M., Cour, T., Yu, K., Cao, L., & Huang, T. (2011, June). Large-scale image classification: Fast feature extraction and SVM training. In *CVPR 2011* (pp. 1689-1696). IEEE.
- Liu, S., Qi, Z., Li, X., & Yeh, A. G. O. (2019). Integration of convolutional neural networks and object-based post-classification refinement for land use and land cover mapping with optical and SAR data. *Remote Sensing*, 11(6), 690.
- Liu, T., & Abd-Elrahman, A. (2018a). An object-based image analysis method for enhancing classification of land covers using fully convolutional networks and multi-view images of small unmanned aerial system. *Remote Sensing*, 10(3), 457.
- Liu, T., & Abd-Elrahman, A. (2018b). Deep convolutional neural network training enrichment using multi-view object-based analysis of Unmanned Aerial systems imagery for wetlands classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 139, 154-170.

- López-Fandiño, J., B. Heras, D., Argüello, F., & Dalla Mura, M. (2019). GPU framework for change detection in multitemporal hyperspectral images. *International Journal of Parallel Programming*, 47(2), 272-292.
- Lourenço, P., Teodoro, A. C., Gonçalves, J. A., Honrado, J. P., Cunha, M., & Sillero, N. (2021). Assessing the performance of different OBIA software approaches for mapping invasive alien plants along roads with remote sensing data. *International Journal of Applied Earth Observation and Geoinformation*, 95, 102263.
- Lu, D., & Weng, Q. (2007). A survey of image classification methods and techniques for improving classification performance. *International journal of Remote sensing*, 28(5), 823-870.
- Lu, Y., Li, H., Zhang, C., & Zhang, S. (2024). Object-Based Semi-Supervised Spatial Attention Residual UNet for Urban High-Resolution Remote Sensing Image Classification. *Remote Sensing*, 16(8), 1444.
- Lv, X., Ming, D., Lu, T., Zhou, K., Wang, M., & Bao, H. (2018). A new method for region-based majority voting CNNs for very high resolution image classification. *Remote Sensing*, 10(12), 1946.
- Lv, X., Shao, Z., Ming, D., Diao, C., Zhou, K., & Tong, C. (2021). Improved object-based convolutional neural network (IOCNN) to classify very high-resolution remote sensing images. *International Journal of Remote Sensing*, 42(21), 8318-8344.
- Lyu, H., Lu, H., & Mou, L. (2016). Learning a transferable change rule from a recurrent neural network for land cover change detection. *Remote Sensing*, 8(6), 506.
- Ma, L., Liu, Y., Zhang, X., Ye, Y., Yin, G., & Johnson, B. A. (2019). Deep learning in remote sensing applications: A meta-analysis and review. *ISPRS journal of photogrammetry and remote sensing*, 152, 166-177.
- Ma, X., Man, Q., Yang, X., Dong, P., Yang, Z., Wu, J., & Liu, C. (2023). Urban Feature Extraction within a Complex Urban Area with an Improved 3D-CNN Using Airborne Hyperspectral Data. *Remote Sensing*, 15(4), 992.
- Maggiori, E., Tarabalka, Y., Charpiat, G., & Alliez, P. (2016, July). Fully convolutional neural networks for remote sensing image classification. In 2016 IEEE international geoscience and remote sensing symposium (IGARSS) (pp. 5071-5074). IEEE.
- Majd, R. D., Momeni, M., & Moallem, P. (2019). Transferable object-based framework based on deep convolutional neural networks for building extraction. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(8), 2627-2635.
- Martins, V. S., Kaleita, A. L., Gelder, B. K., da Silveira, H. L., & Abe, C. A. (2020). Exploring multiscale object-based convolutional neural network (multi-OCNN) for remote sensing image classification at high spatial

resolution. *ISPRS Journal of Photogrammetry and Remote Sensing*, 168, 56-73.

Masci, J., Meier, U., Cireşan, D., & Schmidhuber, J. (2011). Stacked convolutional auto-encoders for hierarchical feature extraction. In *Artificial Neural Networks and Machine Learning–ICANN 2011: 21st International Conference on Artificial Neural Networks*, Espoo, Finland, June 14-17, 2011, Proceedings, Part I 21 (pp. 52-59). Springer Berlin Heidelberg.

Mayr, A., Klambauer, G., Unterthiner, T., Steijaert, M., Wegner, J. K., Ceulemans, H., Clevert, D. A., & Hochreiter, S. (2018). Large-scale comparison of machine learning methods for drug target prediction on ChEMBL. *Chemical science*, 9(24), 5441-5451.

Mboga, N., Georganos, S., Grippa, T., Lennert, M., Vanhuysse, S., & Wolff, E. (2019a). Fully convolutional networks and geographic object-based image analysis for the classification of VHR imagery. *Remote Sensing*, 11(5), 597.

Mboga, N., Georganos, S., Grippa, T., Lennert, M., Vanhuysse, S., & Wolff, E. (2019 b). Weakly supervised fully convolutional networks using OBIA classification output. In *2019 Joint Urban Remote Sensing Event (JURSE)* (pp. 1-4). IEEE.

Merchant, M. A. (2020). Classifying open water features using optical satellite imagery and an object-oriented convolutional neural network. *Remote Sensing Letters*, 11(12), 1127-1136.

Ming, D., Li, J., Wang, J., & Zhang, M. (2015). Scale parameter selection by spatial statistics for GeOBIA: Using mean-shift based multi-scale segmentation as an example. *ISPRS Journal of Photogrammetry and Remote Sensing*, 106, 28-41.

Mohammadi, M., & Sharifi, A. (2021). Evaluation of convolutional neural networks for urban mapping using satellite images. *Journal of the Indian Society of Remote Sensing*, 49, 2125-2131.

Munyati, C. (2018). Optimizing multiresolution segmentation: Delineating savannah vegetation boundaries in the Kruger National Park, South Africa, using Sentinel 2 MSI imagery. *International Journal of Remote Sensing*, 39(18), 5997-6019.

Nababan, B., Mastu, L. O. K., Idris, N. H., & Panjaitan, J. P. (2021). Shallow-water benthic habitat mapping using drone with object based image analyses. *Remote Sensing*, 13(21), 4452.

Niphadkar, M., Nagendra, H., Tarantino, C., Adamo, M., & Blonda, P. (2017). Comparing pixel and object-based approaches to map an understory invasive shrub in tropical mixed forests. *Frontiers in plant science*, 8, 892.

- Özdemir, A. O. B., Gedik, B. E., & Çetin, C. Y. Y. (2014, June). Hyperspectral classification using stacked autoencoders with deep learning. In 2014 6th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS) (pp. 1-4). IEEE.
- Pal, R., Mukhopadhyay, S., Chakraborty, D., & Suganthan, P. N. (2023). A hybrid algorithm for urban LULC change detection for building smart-city by using worldview images. *IETE Journal of Research*, 69(9), 5748-5754.
- Pan, X., Zhao, J., & Xu, J. (2019). An object-based and heterogeneous segment filter convolutional neural network for high-resolution remote sensing image classification. *International Journal of Remote Sensing*, 40(15), 5892-5916.
- Paoletti, M. E., Haut, J. M., Plaza, J., & Plaza, A. (2018). A new deep convolutional neural network for fast hyperspectral image classification. *ISPRS journal of photogrammetry and remote sensing*, 145, 120-147.
- Papadomanolaki, M., Vakalopoulou, M., & Karantzas, K. (2019). A novel object-based deep learning framework for semantic segmentation of very high-resolution remote sensing data: Comparison with convolutional and fully convolutional networks. *Remote Sensing*, 11(6), 684.
- Pathak, D., Girshick, R., Dollár, P., Darrell, T., & Hariharan, B. (2017). Learning features by watching objects move. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2701-2710).
- Qing, Y., Ming, D., Wen, Q., Weng, Q., Xu, L., Chen, Y., ... & Zeng, B. (2022). Operational earthquake-induced building damage assessment using CNN-based direct remote sensing change detection on superpixel level. *International Journal of Applied Earth Observation and Geoinformation*, 112, 102899.
- Quattrochi, D. A., & Goodchild, M. F. (Eds.). (1997). *Scale in remote sensing and GIS*. CRC press. <http://doi:10.1201/9780203740170>
- Rafael, C. G., & Richard, E. W. (2002). *Digital image processing second edition*. Digital Image Processing Sencond Edition" in, Beijing: Publishing House of Electronics Industry, 626-630.
- Rajesh, S., Nisia, T. G., Arivazhagan, S., & Abisekaraj, R. (2020). Land cover/land use mapping of LISS IV imagery using object-based convolutional neural network with deep features. *Journal of the Indian Society of Remote Sensing*, 48, 145-154.
- Rathbun, S. L. (2000). Interpolation of spatial data: some theory for kriging. *Journal of the American Statistical Association*, 95(451), 1010.

- Reza, M. N., Na, I. S., Baek, S. W., & Lee, K. H. (2019). Rice yield estimation based on K-means clustering with graph-cut segmentation using low-altitude UAV images. *Biosystems engineering*, 177, 109-121.
- Rezaee, M., Zhang, Y., Mishra, R., Tong, F., & Tong, H. (2018, August). Using a vgg-16 network for individual tree species detection with an object-based approach. In 2018 10th IAPR Workshop on Pattern Recognition in Remote Sensing (PRRS) (pp. 1-7). IEEE.
- Robson, B. A., Bolch, T., MacDonell, S., Hölbling, D., Rastner, P., & Schaffer, N. (2020). Automated detection of rock glaciers using deep learning and object-based image analysis. *Remote sensing of environment*, 250, 112033.
- Rom, A. R. M., Jamil, N., & Ibrahim, S. (2024). Multi objective hyperparameter tuning via random search on deep learning models. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 22(4), 956-968.
- Russ, J. C. (2006). *The image processing handbook*. CRC press. <https://doi.org/10.1201/9780203881095>
- Sanam, H., Mathai, A. K., & Lakshmanan, G. (2023, January). Multi-resolution remote sensing for the specieslevel classification of mangroves. In 2023 International Conference on Machine Intelligence for GeoAnalytics and Remote Sensing (MIGARS) (Vol. 1, pp. 1-4). IEEE.
- Sameen, M. I., Pradhan, B., & Lee, S. (2020). Application of convolutional neural networks featuring Bayesian optimization for landslide susceptibility assessment. *Catena*, 186, 104249.
- Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, 61, 85-117.
- Shahabi, H., Jarihani, B., Tavakkoli Piralilou, S., Chittleborough, D., Avand, M., & Ghorbanzadeh, O. (2019). A semi-automated object-based gully networks detection using different machine learning models: a case study of Bowen catchment, Queensland, Australia. *Sensors*, 19(22), 4893.
- Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., & De Freitas, N. (2015). Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104(1), 148-175.
- Shakya, A., Biswas, M., & Pal, M. (2021). Parametric study of convolutional neural network based remote sensing image classification. *International Journal of Remote Sensing*, 42(7), 2663-2685.
- Sharma, A., Liu, X., Yang, X., & Shi, D. (2017). A patch-based convolutional neural network for remote sensing image classification. *Neural Networks*, 95, 19-28.

- Shams, M. Y., Elshewey, A. M., El-Kenawy, E. S. M., Ibrahim, A., Talaat, F. M., & Tarek, Z. (2024). Water quality prediction using machine learning models based on grid search method. *Multimedia Tools and Applications*, 83(12), 35307-35334.
- Shendryk, Y., Rist, Y., Ticehurst, C., & Thorburn, P. (2019). Deep learning for multi-modal classification of cloud, shadow and land cover scenes in PlanetScope and Sentinel-2 imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 157, 124-136.
- Singh, M. K., Mohan, S., & Kumar, B. (2022). Fusion of hyperspectral and LiDAR data using sparse stacked autoencoder for land cover classification with 3D-2D convolutional neural network. *Journal of Applied Remote Sensing*, 16(3), 034523-034523.
- Snoek, J., Larochelle, H., & Adams, R. P., (2012). Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, 25.
- Srinivas, S., Sarvadevabhatla, R. K., Mopuri, K. R., Prabhu, N., Kruthiventi, S. S., & Babu, R. V. (2016). A taxonomy of deep convolutional neural nets for computer vision. *Frontiers in Robotics and AI*, 2, 36.
- Sun, T., Shao, Y., Li, X., Liu, P., Yan, H., Qiu, X., & Huang, X. (2020, April). Learning sparse sharing architectures for multiple tasks. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 34, No. 05, pp. 8936-8943).
- Superczynski, S. D., & Christopher, S. A. (2011). Exploring land use and land cover effects on air quality in Central Alabama using GIS and remote sensing. *Remote sensing*, 3(12), 2552-2567.
- Sutha, J. (2020). Object based classification of high resolution remote sensing image using HRSVM-CNN classifier. *European Journal of Remote Sensing*, 53(sup1), 16-30.
- Tan, K., Wu, F., Du, Q., Du, P., & Chen, Y. (2019). A parallel Gaussian–Bernoulli restricted Boltzmann machine for mining area classification with hyperspectral imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(2), 627-636.
- Tang, Q., & Wang, L. (2012). Urban land-use, land-cover classification through watershed segmentation in the v–i–s feature space. *Remote sensing letters*, 3(8), 677-685.
- Tang, Z., Li, M., & Wang, X. (2020). Mapping tea plantations from VHR images using OBIA and convolutional neural networks. *Remote Sensing*, 12(18), 2935.
- Thomas, D. J., Robson, B. A., & Racoviteanu, A. (2023). An integrated deep learning and object-based image analysis approach for mapping debris-covered glaciers. *Frontiers in Remote Sensing*, 4, 1161530.

- Timilsina, S., Aryal, J., & Kirkpatrick, J. B. (2020). Mapping urban tree cover changes using object-based convolution neural network (OB-CNN). *Remote Sensing*, 12(18), 3017.
- Ubben, N., Pukrop, M., & Jarmer, T. (2024). Spatial Resolution as a Factor for Efficient UAV-Based Weed Mapping—A Soybean Field Case Study. *Remote Sensing*, 16(10), 1778.
- Vignesh, T., & Thyagarajan, K. K. (2019). Land Use and Land Cover Classification Using Deep Belief Network for LISS-III Multispectral Satellite Images. *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*. 9(1), 94-98.
- Vinod, P. V., Trivedi, S., Hebbar, R., & Jha, C. S. (2023). Assessment of Trees Outside Forest (TOF) in Urban Landscape Using High-Resolution Satellite Images and Deep Learning Techniques. *Journal of the Indian Society of Remote Sensing*, 51(3), 549-564.
- Waghwal, R. K., & Agnihotri, P. G. (2019). Flood risk assessment and resilience strategies for flood risk management: A case study of Surat City. *International Journal of Disaster Risk Reduction*, 40, 101155.
- Wan, H., Tang, P., Tian, B., Yu, H., Jin, C., Zhao, B., & Wang, H. (2023). Water Extraction in PolSAR Image Based on Superpixel and Graph Convolutional Network. *Applied Sciences*, 13(4), 2610.
- Wang, C., Shu, Q., Wang, X., Guo, B., Liu, P., & Li, Q. (2019). A random forest classifier based on pixel comparison features for urban LiDAR data. *ISPRS journal of photogrammetry and remote sensing*, 148, 75-86.
- Wang, F. (1990). Fuzzy supervised classification of remote sensing images. *IEEE Transactions on geoscience and remote sensing*, 28(2), 194-201.
- Wang, J., Huang, B., & Wang, F. (2023). Extraction and Classification of Flood-Affected Areas Based on MRF and Deep Learning. *Water*, 15(7), 1288.
- Wang, J., Thiesson, B., Xu, Y., & Cohen, M. (2004). Image and video segmentation by anisotropic kernel mean shift. In *Computer Vision-ECCV 2004: 8th European Conference on Computer Vision*, Prague, Czech Republic, May 11-14, 2004. Proceedings, Part II 8 (pp. 238-249). Springer Berlin Heidelberg.
- Wang, J., Zheng, Y., Wang, M., Shen, Q., & Huang, J. (2020). Object-scale adaptive convolutional neural networks for high-spatial resolution remote sensing image classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14, 283-299.
- Wilkinson, G. G. (2005). Results and implications of a study of fifteen years of satellite image classification experiments. *IEEE Transactions on Geoscience and remote sensing*, 43(3), 433-440.

- Wiskes, W., Blesius, L., & Hines, E. (2023). Identification of Abandoned Logging Roads in Point Reyes National Seashore. *Remote Sensing*, 15(13), 3369.
- Wu, J., Chen, X. Y., Zhang, H., Xiong, L. D., Lei, H., & Deng, S. H. (2019). Hyperparameter optimization for machine learning models based on Bayesian optimization. *Journal of Electronic Science and Technology*, 17(1), 26-40.
- Wu, T., Luo, J., Zhou, Y. N., Wang, C., Xi, J., & Fang, J. (2020). Geo-object-based land cover map update for high-spatial-resolution remote sensing images via change detection and label transfer. *Remote Sensing*, 12(1), 174.
- Xie, F., Shi, M., Shi, Z., Yin, J., & Zhao, D. (2017). Multilevel cloud detection in remote sensing images based on deep learning. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(8), 3631-3640.
- Xie, P., Wang, S., Wang, M., Ma, R., Tian, Z., Liang, Y., & Shi, X. (2024). Waterlogged Area Identification Models Based on Object-Oriented Image Analysis and Deep Learning Methods in Sloping Croplands of Northeast China. *Sustainability*, 16(10), 3917.
- Xu, Y., Du, B., Zhang, L., Cerra, D., Pato, M., Carmona, E., ... & Le Saux, B. (2019). Advanced multi-sensor optical remote sensing for urban land use and land cover classification: Outcome of the 2018 IEEE GRSS data fusion contest. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(6), 1709-1724.
- Xue, Y., Zhao, J., & Zhang, M. (2021). A watershed-segmentation-based improved algorithm for extracting cultivated land boundaries. *Remote Sensing*, 13(5), 939.
- Yalcin, I., Karakas, G., Kocaman, S., Saunier, S., & Albinet, C. (2022). Investigations on the Effect of HD Processing in Land Cover Classification. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences-ISPRS Archives*, 43(B1-2022), 293-300.
- Zaabar, N., Niculescu, S., & Kamel, M. M. (2022). Application of convolutional neural networks with object-based image analysis for land cover and land use mapping in coastal areas: A case study in Ain Témouchent, Algeria. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15, 5177-5189.
- Zhang, C., Pan, X., Li, H., Zhang, S., & Atkinson, P. (2020a). Joint Deep Learning: A novel framework for urban land cover and land use classification. *GISRUk 2020*, 1-5.
- Zhang, C., Sargent, I., Pan, X., Gardiner, A., Hare, J., & Atkinson, P. M. (2018 b). VPRS-based regional decision fusion of CNN and MRF

- classifications for very fine resolution remotely sensed images. *IEEE Transactions on Geoscience and Remote Sensing*, 56(8), 4507-4521.
- Zhang, C., Sargent, I., Pan, X., Li, H., Gardiner, A., Hare, J., & Atkinson, P. M. (2018c). An object-based convolutional neural network (OCNN) for urban land use classification. *Remote sensing of environment*, 216, 57-70.
- Zhang, C., Sargent, I., Pan, X., Li, H., Gardiner, A., Hare, J., & Atkinson, P. M. (2019). Joint Deep Learning for land cover and land use classification. *Remote sensing of environment*, 221, 173-187.
- Zhang, H., Altham, C., Trovati, M., Zhang, C., Rolland, I., Lawal, L., Wegbu, D., & Ajienka, N. (2022). Adaptive Multiscale Superpixel Embedding Convolutional Neural Network for Land Use Classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15, 7631-7642.
- Zhang, T., Su, J., Xu, Z., Luo, Y., & Li, J. (2021). Sentinel-2 satellite imagery for urban land cover classification by optimized random forest classifier. *Applied Sciences*, 11(2), 543.
- Zhang, X., Chen, G., Wang, W., Wang, Q., & Dai, F. (2017). Object-based land-cover supervised classification for very-high-resolution UAV images using stacked denoising autoencoders. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(7), 3373-3385.
- Zhang, X., Han, L., Han, L., & Zhu, L. (2020). How well do deep learning-based methods for land cover classification and object detection perform on high resolution remote sensing imagery?. *Remote Sensing*, 12(3), 417.
- Zhang, C., Pan, X., Li, H., Gardiner, A., Sargent, I., Hare, J., & Atkinson, P. M. (2018 a). A hybrid MLP-CNN classifier for very fine resolution remotely sensed image classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 140, 133-144.
- Zhang, K., Ming, D., Du, S., Xu, L., Ling, X., Zeng, B., & Lv, X. (2022). Distance Weight-Graph Attention Model-Based High-Resolution Remote Sensing Urban Functional Zone Identification. *IEEE Trans. Geosci. Remote. Sens.*, 60, 1-18.
- Zhao, C., Qin, B., Feng, S., & Zhu, W. (2022). Multiple superpixel graphs learning based on adaptive multiscale segmentation for hyperspectral image classification. *Remote Sensing*, 14(3), 681.
- Zhao, J., Hu, L., Dong, Y., Huang, L., Weng, S., & Zhang, D. (2021). A combination method of stacked autoencoder and 3D deep residual network for hyperspectral image classification. *International Journal of Applied Earth Observation and Geoinformation*, 102, 102459.

- Zhao, W., Du, S., & Emery, W. J. (2017). Object-based convolutional neural network for high-resolution imagery classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(7), 3386-3396.
- Zhou, T., Fu, H., Sun, C., & Wang, S. (2021). Shadow detection and compensation from remote sensing images under complex urban conditions. *Remote Sensing*, 13(4), 699.
- Zhu, L., Chen, Y., Ghamisi, P., & Benediktsson, J. A. (2018). Generative adversarial networks for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 56(9), 5046-5063.
- Zhu, M., Wang, Z., Cui, H., & Yao, S. (2022). Rural road network planning based on 5g and traffic big data. *Journal of Advanced Transportation*, 2022(1), 1991757.
- Zhu, Q., Lei, Y., Sun, X., Guan, Q., Zhong, Y., Zhang, L., & Li, D. (2022). Knowledge-guided land pattern depiction for urban land use mapping: A case study of Chinese cities. *Remote Sensing of Environment*, 272, 112916.
- Zi, Y., Xie, F., & Jiang, Z. (2018). A cloud detection method for Landsat 8 images based on PCANet. *Remote Sensing*, 10(6), 877.
- Zitova, B., & Flusser, J. (2003). Image registration methods: a survey. *Image and vision computing*, 21(11), 977-1000.
- Zöller, M. A., & Huber, M. F. (2021). Benchmark and survey of automated machine learning frameworks. *Journal of artificial intelligence research*, 70, 409-472.

APPENDICES

Appendix A

Python Codes for Calculating OBIA Features

```
import numpy as np
from skimage.measure import perimeter,
EllipseModel

from gis import minimum_bounding_rectangle

class compute_spatial_features:
    def __init__(self, binary_arr, pixels):
        self.binary_arr = binary_arr
        self.pixels = pixels

    def compute(self):
        farea = area(self.binary_arr)
        fop =
object_perimeter(self.binary_arr)
        fei =
elongation_index(self.binary_arr)
        fsi = shape_index(self.binary_arr)
        fd = density(self.binary_arr,
self.pixels)
        fo = orientation(self.pixels)
        faw = average_width(self.pixels)
        fah = average_height(self.pixels)
        frf =
rectangular_fit(self.binary_arr,
self.pixels)
        features = {farea.name: farea(),
                    fop.name: fop(),
                    fei.name: fei(),
                    fsi.name: fsi(),
                    fd.name: fd(),
                    fo.name: fo(),
                    faw.name: faw(),
                    fah.name: fah(),
                    frf.name: frf()}

        return features

class area:
    def __init__(self, binary_arr):
        self._binary_arr = binary_arr
        self.name = 'area'
```

```

    def __call__(self):
        return
np.sum(self._binary_arr[self._binary_arr ==
1])

class object_perimeter:
    def __init__(self, binary_arr):
        self._binary_arr = binary_arr
        self.name = 'perimeter'

    def __call__(self):
        return perimeter(self._binary_arr)

class elongation_index:
    def __init__(self, binary_arr):
        self._binary_arr = binary_arr
        self.name = 'elongation_index'

    def __call__(self):
        a = area(self._binary_arr)()
        p =
object_perimeter(self._binary_arr)()
        return (2 * np.pi * a) / (p ** 2)

class shape_index:
    def __init__(self, binary_arr):
        self._binary_arr = binary_arr
        self.name = 'shape_index'

    def __call__(self):
        a = area(self._binary_arr)()
        p =
object_perimeter(self._binary_arr)()
        return p / (4 * np.sqrt(a))

class density:
    def __init__(self, binary_arr, pixels):
        self._binary_arr = binary_arr
        self._pixels = pixels
        self.name = 'density'

    def __call__(self):
        ar = area(self._binary_arr)()
        ell = EllipseModel()
        if ell.estimate(self._pixels):
            _, _, a, b, _ = ell.params

```

```

        else:
            a, b = 1.5, 1
            return ar / ((a + b) / 2)

class orientation:
    def __init__(self, pixels):
        self._pixels = pixels
        self.name = 'orientation'

    def __call__(self):
        ell = EllipseModel()
        if ell.estimate(self._pixels):
            _, _, _, _ = theta = ell.params
        else:
            theta = 0
        return theta

class average_width:
    def __init__(self, pixels):
        self._pixels = pixels
        self.name = 'average_width'

    def __call__(self):
        ell = EllipseModel()
        if ell.estimate(self._pixels):
            _, _, _, b, _ = ell.params
        else:
            b = 0
        return b

class average_height:
    def __init__(self, pixels):
        self._pixels = pixels
        self.name = 'average_height'

    def __call__(self):
        ell = EllipseModel()
        if ell.estimate(self._pixels):
            _, _, a, _, _ = ell.params
        else:
            a = 0
        return a

```

```

class rectangular_fit:
    def __init__(self, binary_arr, pixels):
        self._binary_arr = binary_arr
        self._pixels = pixels
        self.name = 'rectangular_fit'

    def __call__(self):
        try:
            a = area(self._binary_arr)()
            rect =
minimum_bounding_rectangle(self._pixels)
            xs = list(set([int(row[0]) for
row in rect]))
            ys = list(set([int(row[1]) for
row in rect]))
            rect_area = xs[xs != 0] * ys[ys
!= 0]
            return abs(a - rect_area) /
abs(a + rect_area)
        except:
            return 0

import numpy as np
from scipy.stats import kurtosis as kurt
from scipy.stats import skew

class compute_spectral_features:
    def __init__(self, arr, pixels, rgb):
        self.arr = arr
        self.pixels = pixels
        self.rgb = rgb

    def compute(self):
        fnop = number_of_pixels(self.pixels)
        fmin = minimum(self.rgb)
        fmax = maximum(self.rgb)
        fmean = mean(self.rgb)
        fstd = standard_deviation(self.rgb)
        fkurt = kurtosis(self.rgb)
        fskew = skewness(self.rgb)
        fvrange = val_range(self.rgb)
        features = {fnop.name: fnop(),
                    fmin.name: fmin(),
                    fmax.name: fmax(),
                    fmean.name: fmean(),
                    fstd.name: fstd(),
                    fkurt.name: fkurt(),
                    fskew.name: fskew(),

```

```

        fvrange.name: fvrange() }
    return features

class number_of_pixels:
    def __init__(self, pixels):
        self._pixels = pixels
        self.name = 'number_of_pixels'

    def __call__(self):
        return len(self._pixels)

class minimum:
    def __init__(self, rgb_segments):
        self._rgb_segments = rgb_segments
        self.name = 'minimum'

    def __call__(self):
        r, g, b = self._rgb_segments
        min_r = np.nanmin(r) if
np.isfinite(np.nanmin(r)) else 0
        min_g = np.nanmin(g) if
np.isfinite(np.nanmin(g)) else 0
        min_b = np.nanmin(b) if
np.isfinite(np.nanmin(b)) else 0
        return min_r, min_g, min_b

class maximum:
    def __init__(self, rgb_segments):
        self._rgb_segments = rgb_segments
        self.name = 'maximum'

    def __call__(self):
        r, g, b = self._rgb_segments
        max_r = np.nanmax(r) if
np.isfinite(np.nanmax(r)) else 0
        max_g = np.nanmax(g) if
np.isfinite(np.nanmax(g)) else 0
        max_b = np.nanmax(b) if
np.isfinite(np.nanmax(b)) else 0
        return max_r, max_g, max_b

class mean:
    def __init__(self, rgb_segments):
        self._rgb_segments = rgb_segments
        self.name = 'mean'

```

```

def __call__(self):
    r, g, b = self._rgb_segments
    mean_r = np.nanmean(r) if
np.isfinite(np.nanmean(r)) else 0
    mean_g = np.nanmean(g) if
np.isfinite(np.nanmean(g)) else 0
    mean_b = np.nanmean(b) if
np.isfinite(np.nanmean(b)) else 0
    return mean_r, mean_g, mean_b

class standard_deviation:
    def __init__(self, rgb_segments):
        self._rgb_segments = rgb_segments
        self.name = 'standard_deviation'

    def __call__(self):
        r, g, b = self._rgb_segments
        std_r = np.nanstd(r) if
np.isfinite(np.nanstd(r)) else 0
        std_g = np.nanstd(g) if
np.isfinite(np.nanstd(g)) else 0
        std_b = np.nanstd(b) if
np.isfinite(np.nanstd(b)) else 0
        return std_r, std_g, std_b

class kurtosis:
    def __init__(self, rgb_segments):
        self._rgb_segments = rgb_segments
        self.name = 'kurtosis'

    def __call__(self):
        r, g, b = self._rgb_segments
        return kurt(r), kurt(g), kurt(b)

class skewness:
    def __init__(self, rgb_segments):
        self._rgb_segments = rgb_segments
        self.name = 'skewness'

    def __call__(self):
        r, g, b = self._rgb_segments
        return skew(r), skew(g), skew(b)

class val_range:
    def __init__(self, rgb_segments):
        self._rgb_segments = rgb_segments

```

```

        self.name = 'val_range'

    def __call__(self):
        r, g, b = self._rgb_segments
        range_r = max(r) - min(r)
        range_g = max(g) - min(g)
        range_b = max(b) - min(b)
        return range_r, range_g, range_b

from skimage.feature import greycoprops

class compute_textural_features:
    def __init__(self, glcm):
        self.glcm = glcm

    def compute(self):
        fcon = contrast(self.glcm)
        fdis = dissimilarity(self.glcm)
        fhom = homogeneity(self.glcm)
        fene = energy(self.glcm)
        fcor = correlation(self.glcm)
        fang =
angular_second_moment(self.glcm)
        features = {fcon.name: fcon(),
                    fdis.name: fdis(),
                    fhom.name: fhom(),
                    fene.name: fene(),
                    fcor.name: fcor(),
                    fang.name: fang()}

        return features

class contrast:
    def __init__(self, glcm):
        self._glcm = glcm
        self.name = 'contrast'

    def __call__(self):
        glcm_r, glcm_g, glcm_b = self._glcm
        contrast_r =
float(greycoprops(glcm_r, 'contrast'))
        contrast_g =
float(greycoprops(glcm_g, 'contrast'))
        contrast_b =
float(greycoprops(glcm_b, 'contrast'))
        return contrast_r, contrast_g,
contrast_b

```

```

class dissimilarity:
    def __init__(self, glcm):
        self._glcm = glcm
        self.name = 'dissimilarity'

    def __call__(self):
        glcm_r, glcm_g, glcm_b = self._glcm
        dissimilarity_r =
float(greycoprops(glcm_r, 'dissimilarity'))
        dissimilarity_g =
float(greycoprops(glcm_g, 'dissimilarity'))
        dissimilarity_b =
float(greycoprops(glcm_b, 'dissimilarity'))
        return dissimilarity_r,
dissimilarity_g, dissimilarity_b

class homogeneity:
    def __init__(self, glcm):
        self._glcm = glcm
        self.name = 'homogeneity'

    def __call__(self):
        glcm_r, glcm_g, glcm_b = self._glcm
        homogeneity_r =
float(greycoprops(glcm_r, 'homogeneity'))
        homogeneity_g =
float(greycoprops(glcm_g, 'homogeneity'))
        homogeneity_b =
float(greycoprops(glcm_b, 'homogeneity'))
        return homogeneity_r, homogeneity_g,
homogeneity_b

class energy:
    def __init__(self, glcm):
        self._glcm = glcm
        self.name = 'energy'

    def __call__(self):
        glcm_r, glcm_g, glcm_b = self._glcm
        energy_r = float(greycoprops(glcm_r,
'energy'))
        energy_g = float(greycoprops(glcm_g,
'energy'))
        energy_b = float(greycoprops(glcm_b,
'energy'))
        return energy_r, energy_g, energy_b

```

```

class correlation:
    def __init__(self, glcm):
        self._glcm = glcm
        self.name = 'correlation'

    def __call__(self):
        glcm_r, glcm_g, glcm_b = self._glcm
        correlation_r =
float(greycoprops(glcm_r, 'correlation'))
        correlation_g =
float(greycoprops(glcm_g, 'correlation'))
        correlation_b =
float(greycoprops(glcm_b, 'correlation'))
        return correlation_r, correlation_g,
correlation_b
class angular_second_moment:
    def __init__(self, glcm):
        self._glcm = glcm
        self.name = 'angular_second_moment'

    def __call__(self):
        glcm_r, glcm_g, glcm_b = self._glcm
        ASM_r = float(greycoprops(glcm_r,
'ASM'))
        ASM_g = float(greycoprops(glcm_g,
'ASM'))
        ASM_b = float(greycoprops(glcm_b,
'ASM'))
        return ASM_r, ASM_g, ASM_b

```

Appendix B

WorldView-3 Satellite Sensor Specifications

Launch Date	August 13, 2014
Orbit	Altitude: 617 km Type: SunSync, 1:30 pm descending Node Period: 97 min.
Life	Spec Mission Life: 7.25 years Estimated Service Life: 10 to 12 years
Spacecraft Size, Mass and Power	Size: 5.7 m (18.7 feet) tall x 2.5 m (8 feet) across, 7.1 m (23 feet) across the deployed solar arrays Mass: 2800 kilograms (6200 pounds) Power: 3.1 kW solar array, 100 Ahr battery
Sensor Bands	Panchromatic: 450-800 nm 8 Multispectral: (red, red edge, coastal, blue, green, yellow, near-IR1, and near-IR2) - 400 nm - 1040 nm 8 SWIR: 1195 nm - 2365 nm 12 CAVIS Bands: (desert clouds, aerosol-1, aerosol-2, aerosol-3, green, water-1, water- 2, water-3, NDVI-SWIR, cirrus, snow) - 405 nm - 2245 nm
Sensor Resolution (GSD)	Panchromatic Nadir: 30cm GSD at Nadir 0.34 m at 20° Off-Nadir Multispectral Nadir: 1.24 m at Nadir, 1.38 m at 20° Off-Nadir SWIR Nadir: 3.70 m at Nadir, 4.10 m at 20° Off-Nadir CAVIS Nadir: 30.00 m
Dynamic Range	11-bits per pixel Pan and MS; 14-bits per pixel SWIR
Swath Width	At nadir: 13.1 km
Attitude Determination and Control	Type: 3-axis stabilized Actuators: Control Moment Gyros (CMGs) Sensors: Star trackers, precision, IRU, GPS
Pointing Accuracy and Knowledge	Accuracy: <500 m at image start and stop Knowledge: Supports geolocation accuracy below
Retargeting Agility	Time to slew 200 km: 12 seconds
Onboard Storage	2199 Gb solid state with EDAC
Communications	Image & Ancillary: 800 & 1200 Mbps X-band Housekeeping: 4, 16, 32 or 64 kbps real-time, 524 kbps stored, X-band Command: 2 or 64 kbps S-band
Max Contiguous Area Collected in a Single Pass (30° off-nadir angle)	Mono: 66.5 km x 112 km (5 strips) Stereo: 26.6 km x 112 km (2 pairs)
Revisit Frequency(at 40°N Latitude)	1 m GSD: <1.0 day 4.5 days at 20° off-nadir or less
Geolocation Accuracy(CE90)	Predicted Performance: <3.5 m CE90 without ground control
Capacity	680,000 km ² per day

LIST OF PUBLICATIONS

- Azeez, O. S., Shafri, H. Z., Alias, A. H., & Haron, N. A. (2022). Integration of Object-Based Image Analysis and Convolutional Neural Network for the Classification of High-Resolution Satellite Image: A Comparative Assessment. *Applied Sciences*, 12(21), 10890.
- Azeez, O. S., Shafri, H. Z., Alias, A. H., & Haron, N. A. (2022). A Joint Bayesian Optimization for the Classification of Fine Spatial Resolution Remotely Sensed Imagery Using Object-Based Convolutional Neural Networks. *Land*, 11(11), 1905.

