Building Segmentation In Remote Sensing Images Using Region Merging Approach With Convolutional Neural Network-Based Model

ASIM SHOAIB

MASTER OF SCIENCE (COMPUTER SCIENCE)

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

UNIVERSITI TUNKU ABDUL RAHMAN

April 2025

This dissertation is submitted in partial fulfilment of the requirements for the degree of Master of Science (Computer Science) at Universiti Tunku Abdul Rahman (UTAR). This dissertation represents the work of the author, except where due acknowledgment has been made in the text. No part of this dissertation may be reproduced, stored, or transmitted in any form or by any means, whether electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the author or UTAR, in accordance with UTAR's Intellectual Property Policy.

DECLARATION

I Asim Shoaib hereby declare that the thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at UTAR or other institutions.

DEDICATION

I am dedicating this thesis to my family members and specially to my **Parents**.

ACKNOWLEDGMENTS

I express my deepest gratitude and respect for my main research advisor, Ts. Dr. Mogana A/P Vadiveloo. Her patience, encouragement, and continuous support have been pivotal in helping me find a clear path forward. Dr. Mogana A/P Vadiveloo has listened to my challenges patiently and pushed me to address them, making me a better researcher and learner.

I would also like to express my deepest thanks to my co-supervisor Ts. Dr. Lim Seng Poh for his unwavering support and guidance. He has been a friendly supervisor and his contributions has been invaluable to the success of my research work.

I am also thankful to Dr. Manoranjitham A/P Muniandy for recommending me for this opportunity. I also thankful to Engr. Salman Javed, Engr. Hasnain Sultan, and Engr. Muhammad Tarique Lakhiar for their technical guidance and support. Their contributions have been invaluable to the success of my research work.

I am also insistently and truly thankful to my parents “Muhammad Shoaib Khan and Yasmeen Bibi”, for their love, and unwavering support throughout my Master's journey. However, I must especially acknowledge my parents, whose sacrifices, guidance, encouragement, and endless prayers have been the cornerstone of my academic achievements. My parent’s selflessness and dedication to my well-being have been a constant source of inspiration, reminding me of the importance of resilience and determination in the face of challenges. My parents unwavering belief in me has instilled a sense of

confidence and determination that has been pivotal in my success. For my parents unwavering love, guidance, and sacrifices, I am forever grateful.

I would also like to thank my siblings, but especially my sister Ms. Shahana Shoaib, and my brother-in-law Ahmad Jamal, for their financial and emotional support and encouragement throughout this journey. Their trust in my capabilities and their push to achieve my goals has been paramount to my success.

Special thanks to the Universiti Tunku Abdul Rahman for provision of financial support based on research scholarship scheme (RSS) grant (Vote No.6550/1M01). It has been an honor to be a Master's student in the Department of Computer Science of Faculty of Information and Communication Technology, Universiti Tunku Abdul Rahman, Malaysia.

BUILDING SEGMENTATION IN REMOTE SENSING IMAGES USING REGION MERGING APPROACH WITH CONVOLUTIONAL NEURAL NETWORK-BASED MODEL

By

ASIM SHOAIB

A thesis submitted to the Department of Computer Science,
Faculty of Information and Communication Technology,
Universiti Tunku Abdul Rahman,
in partial fulfillment of the requirements for the degree of
Master of Science (Computer Science)
April 2025

ABSTRACT

BUILDING SEGMENTATION IN REMOTE SENSING IMAGES USING REGION MERGING APPROACH WITH CONVOLUTIONAL NEURAL NETWORK-BASED MODEL

ASIM SHOAIB

Image segmentation is a process used to delineate objects in an image as regions of interest (ROIs). Poor delineation can lead to over segmentation (OS), resulting in creation of small regions that do not represent meaningful segmented ROIs. Region merging is one of the common approaches used to prevent OS in images. This approach iteratively merges adjacent regions based on merging criterion (MC) that defines the similarity of features between them. In the existing research works, feature map of labelled images were generated either manually or using a specialised software to derive MC. This process is labour intensive and time consuming. Therefore, in this research MC is derived with the assistance of convolutional neural network (CNN)-based deep learning model to perform region merging without any human intervention. In this research, AttentionU-Net model is used to generate feature map that is used to derive MC for merging building regions in WHU remote sensing images dataset. From experiments conducted, prominent features of building regions which are colour, texture, shape, and edges were extracted from the feature map

to derive MC. This MC is used for merging the OS regions generated by simple linear iterative clustering (SLIC) algorithm. The proposed region merging approach has achieved an average F-measure of 0.91 in segmenting building regions in WHU remote sensing images. This is an improvement compared to previous research work on region merging, which achieved an average F-measure of 0.63 in delineating buildings regions in the same dataset. Moreover, the proposed region merging approach has achieved an average goodness of segmentation, G_s, of 0.92 compared to 0.83 average G_s achieved by the multiresolution segmentation (MRS) region merging approach in delineating building regions in the WHU dataset. This comparison demonstrates the efficacy of proposed approach in segmenting building regions in remote sensing images without any human intervention.

Keywords: Remote Sensing Images segmentation; Over segmentation; Region merging, Merging criterion; Convolutional Neural Network-based deep learning model.

QA75.5-76.95 Computer Science

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AGs	Attention Gates
ARE	Adapted Rand Error
ARI	Adapted Rand Index
BCE	Binary Cross-Entropy
BN	Batch Normalisation
CNN	Convolutional Neural Network
CNN-based	Convolutional Neural Network-based
CW	Compact Watershed
DN	Digitised Number
FP	False Positive
FH	Felzenszwalb and Huttenlocher
FN	False Negative
FCD u-net	Fully Convolutional Dense U-net
GTs	Ground Truths
GLCM	Grey Level Co-Occurrence Matrix
HDFV	High-Dimensional Feature-Vector
HRM	Hybrid Region Merging
HSV	Hue, Saturation, Value
ISRG	Improved Seeded Region Growing
IoU	Intersection Over Union
LBP	Local Binary Pattern
MC	Merging Criterion
MSE	Mean Squared Error

MO	Merging Order
MS	Mean Shift
MRS	Multiresolution Segmentation
OA	Overall Accuracy
OS	Over Segmentation
PReLU	Parametric Rectified Linear Unit
PSNR	Peak Signal to Noise Ratio
QS	Quick Shift
SA	Spectral Angle
SRG	Seeded Region Growing
SDG	Stochastic Gradient Descent
S2Former	Shift-Scale Transformer
RAM	Random Access Memory
RAG	Region Adjacency Graph
ReLU	Rectified Linear Unit
RGB	Red Green Blue
RF	Random Forest
ROI	Region of Interest
TE	Total Error
TN	True Negatives
TP	True Positive
SLIC	Simple Linear Iterative Clustering
SRG	Seeded Region Growing
URG	Unseeded Region Growing
VOI	Variation of Information

CHAPTER 1

INTRODUCTION

1.1 Background Study

In image processing, segmentation means the process of dividing an image into distinct, non-overlapping regions. The aim is to segment these regions into visual objects as perceived by the human perceptual system [1]. Generally, this segmentation is based on pixel features such as intensity, colour, texture, shape, edges, and other attributes [2]. Moreover, mostly image segmentation algorithms depends on two basic fundamental characteristics in delineating visual objects as regions of interest (ROIs) in terms of pixel features discontinuity and similarity [3]. Edge-based segmentation algorithms work based on pixels features discontinuity. While, the region-based segmentation algorithms are based on the pixel features similarity to generate homogeneous segments [4]. According to [5], the region-based segmentation algorithms generate segments that are aligned with object regions boundaries as compared to edge-based segmentation algorithms. Additionally, in contrast to the edge-based segmentation algorithms, the region-based segmentation algorithms are less affected by the noise [5]. This is because in these algorithms, pixels are evaluated within defined regions to form homogeneous region of interest (ROI). By considering a larger context of regions, the algorithms can average out noisy pixels. While edge-based algorithms rely on detecting boundaries between regions, which can be disrupted by noise leading to inaccurate segmentation [6].

Numerous region-based segmentation algorithms exist [7] such as region growing, watershed transform [8], the Felzenszwalb and Huttenlocher (FH) [9], Compact Watershed (CW) [10], random walk [11], Mean Shift (MS) [12], Quick Shift (QS) [13], and the simple linear iterative clustering (SLIC) [14]. Generally, these segmentation algorithms begin by identifying pixels having similarity of feature values then traverse until they reach the edge pixels of object regions for delineation [15]. Among the mentioned algorithms, SLIC [14] effectively delineates the regions in image compared to other region-based segmentation algorithms [16]. This is because in SLIC, a parameter known as the number of centroids, k has to be defined by the user. It controls the number of regions that the algorithm will segment the image into allowing for improved segmentation accuracy [14]. A higher value of parameter k 3000 [17] results in smaller segments of regions being generated, while a lower k value of 250 produces larger numbers of regions [18]. However, SLIC can still lead to over segmentation (OS), similar to other region-based segmentation algorithms [19]. Additionally, OS means the creation of small regions which do not provides meaningful segmented objects regions in the image [20]. Hence, to prevent the OS in the region-based segmentation algorithms, region merging approach is commonly used [21].

Region merging is utilised by iteratively merging the over segmented regions in order to form ROI [22]. Since this approach operates at the region level, it captures detailed information about object features for merging, compared to methods that work at the pixel level [23]. Furthermore, in region merging approach, the merging is conducted based on similarity or dissimilarity of features of regions selected for merging [24]. This measure of regions

features similarity or dissimilarity is known as merging criterion (MC) [25]. Region merging can be performed using a graph known as region adjacency graph (RAG) [26]. In RAG, the over segmented regions are the nodes while the MC depicts the edge connecting these nodes. Pairs of regions are merged iteratively until the MC is satisfied [27].

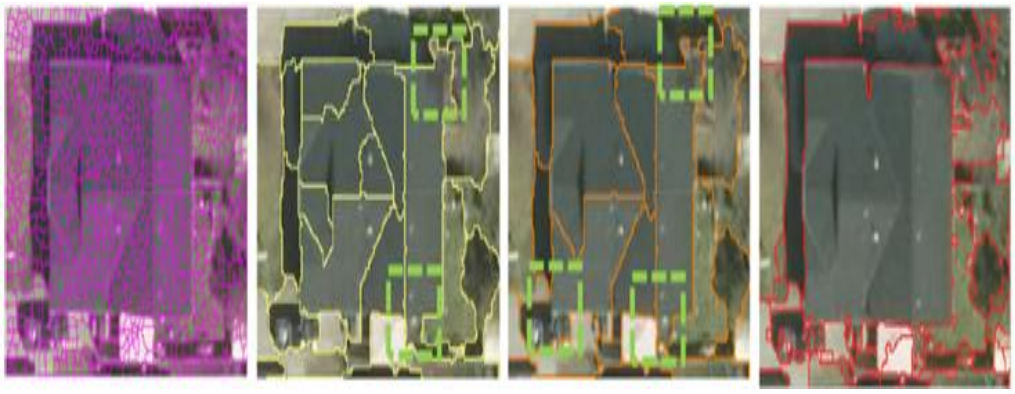


Figure 1.1: Over segmentation and regions merging referred from [28]

The Figure 1.1 demonstrates the over segmentation (OS) issue and the possible delineation for building ROI achieved through region merging. However, there are still some parts of building are left to merged appropriately. Resultantly, the regions merging has some limitations for delineating the building boundaries accurately.

1.2 Problem Statement

In region merging approach, an iterative merging process is commonly conducted between two adjacent regions using a merging criterion (MC) [26]. The MC is commonly derived from features of the two adjacent regions for merging [27, 29]. However, this approach still results in over segmentation (OS) [30]. This challenge has prompted researchers to propose various methods and feature utilisations for deriving the MC in this approach [31, 32].

In recent years, researchers have used deep learning and machine learning models to address the OS in region merging [25, 33]. The researchers in [25] have used a deep learning model in the preprocessing phase to generate over segmented region that effectively adheres to an object regions boundaries. By this, model enhances the precision of the initial segmentation results for accurate merging in the subsequent phase. In this work, a fully convolutional dense U-Net (FCD U-Net) deep learning model is used with the SLIC algorithm [14] to generate the over segmented regions in natural images for merging. This model processes images through multiple hierarchical levels to capture detailed features of regions which are then integrated with SLIC to generate the initial over segmented regions that adheres to object boundaries. These regions were represented in RAG then merged by statistical region merging (SRM) approach in [34] producing the final segmentation results. However, a limitation of this work is that deep learning is used only during the preprocessing phase for generating the over segmented regions. A conventional region merging was performed where the MC was derived only from colour, spatial and textures features to merge the regions [34].

In another work, a machine learning model is used to derive MC to perform merging [33]. In this work, the seeded region growing (SRG) algorithm [35] is used for generating initial over segmented regions. While the region merging is performed with random forest (RF) classifiers to segment building regions in remote sensing images. The RF is trained with a substantial number of labelled images which are collected based on human observation by using shape, compactness, smoothness, band-mean spectral standard deviation, band-mean square error, spectral heterogeneity, spectral angle, and edge features

values. It then classifies the test images into binary results of 0 and 1. The MC is derived from this classification results where pairs of regions labelled as 1 are merged while regions labelled as 0 are not merged. This approach has produced segmented buildings as ROI achieving a total error (TE) evaluation metric of only 0.64 which reflects the overall segmentation quality. Moreover, it is labour-intensive which limits its adaptability for generalisation.

While in [26], deep learning model is used to derive the MC for merging regions. This work has employed a Siamese Network and proposed a shift-scale transformer (S2Former) deep learning model to segment building as ROI in remote sensing images. The S2Former model is trained on manually selected positive and negative regions extracted from over segmented regions generated by a multiresolution algorithm (MRS) [36]. In this work, the positive regions correspond to buildings, while the negative regions refer to non-building object regions. These regions are labelled with the assistance of specialised software [26]. From these labelled regions of images, the S2Former transformer model shall produce feature maps that are used to calculate the MC for merging. The features used to derive MC consist of texture, shape, edge information, compactness, brightness, and standard deviation and mean of each Red, Green, Blue (RGB) channel. By the derived MC, the Siamese Network performs region merging to delineate the buildings as ROI. Despite its sophisticated method, the approach has a limitation in that software introduced in the work [26] can only be used a limited number of times per day by users for producing the labelled regions of images used for MC derivation. As a result, the method is both dependent on human involvement and labour-intensive.

In [28] the authors have proposed a region merging approach for segmenting buildings in remote sensing images using only labelled images to derive the MC. The labelled images contain feature map of buildings identified as ROI which are extracted using specialised software [28]. Despite that, image analysts have to manually cross-checked the generated labelled images against the original image to ensure accurate pixel matching. In this work, the MC is derived from feature map of the labelled images containing buildings of different colour, shapes, compactness and rectangularity feature values to perform merging [28]. As a result, this approach heavily relies on human intervention for the ROI delineation.

Hence, to address these issues, it is essential to explore the potential of using a deep learning model to derive the merging criterion (MC) and perform the merging process without human intervention, specifically for delineating buildings as regions of interest (ROIs) in remote sensing images.

1.3 Research Objectives

The objectives of this research are as follows:

1) To extract prominent buildings features from the feature map generated by a deep learning model to perform region merging.

2) To propose a merging criterion (MC) from the extracted features for merging buildings into ROI in remote sensing images without human intervention.

1.4 Research Scope

This research work focuses on delineating buildings in images as regions of interest (ROIs). Hence, the dataset used in this research is a publicly benchmarked high-resolution remote sensing images which is known as WHU aerial buildings dataset [37]. Hereafter, this dataset shall be referred as WHU buildings dataset. This dataset was chosen because high-resolution remote sensing images provide the ability to convey detailed spatial information in comparison to medium or low-resolution images. Moreover, these images capture the structure, shape, and other features of geographical objects effectively [38]. Furthermore, this dataset is a multisource collection of remote sensing images which consists of aerial and satellite buildings for delineation [37].

1.5 Research Contribution

The primary contribution of this research lies in deriving the merging criterion (MC) from the features extracted from the feature maps generated using deep learning model for merging. This research distinguishes itself from previous studies [25, 26, 28, 33] which utilised various methods in deriving the MC or to improve the final segmentation results. The earlier research work in [25] has employed a deep learning model only in the preprocessing phase for generating over segmented regions having aligned object region boundaries for merging to improve the segmentation results. While in [26] and [33] although the researchers have utilised deep learning and machine learning models to perform region merging, the labelled regions of images are obtained through specialised software for training these models in order to derive the MC.

Moreover, in [28], the labelled images containing building ROI feature map generated by a specialised software are further manually cross-checked by experienced professionals, known as image analysts. They compare these images against the building regions in the original images to ensure accurate pixel matching for the MC derivation. As a result, these approaches are labour-intensive since they require human intervention.

In contrast, the region-merging approach proposed in this research prevents the need for human intervention by leveraging a deep learning model to generate feature map for buildings regions. Based on this map, prominent features are extracted to derive the MC to merge the buildings regions as ROI in remote sensing images.

1.6 Dissertation Organisation

In Chapter 2, literature review is performed on region-based segmentation algorithms, CNN-based deep learning feature map generation for ROI and region merging approaches. Moreover, merging criterion (MC), features utilised for deriving MC and region adjacency graph (RAG) are covered. In Chapter 3 methodology, besides description on overall approach, the approach for comparison between different region-based segmentation algorithms, and CNN-based deep learning models have been demonstrated. Along with this, the experimental requirements have been demonstrated.

In Chapter 4, experiments of comparison are conducted on region-based segmentation algorithms, CNN-based deep learning models, and feature extraction approaches. In addition, preprocessing and hyperparameter settings for CNN-based deep learning models are covered. Moreover, analysis of results

is conducted to select best region-based segmentation algorithm for OS and CNN-based deep learning model for feature map generation to propose MC in Chapter 5. Moreover, in Chapter 5, MC is formulated and stated based on the experimental results explained in Chapter 4. In addition, implementation and discussion of proposed MC is provided. Furthermore, results and discussion are demonstrated in form of Tables and Figures with the utilisation of proposed MC during iterative regions merging process to propose region merging approach. Finally, in Chapter 6, conclusion is demonstrated which contains the limitation of proposed region merging approach in this research work, and possible future work.

CHAPTER 2

LITERATURE REVIEW

2.1 Region-based Segmentation Algorithm

In literature, there are many image segmentation approaches are proposed which are mainly divided into two main categories. These are the edge-based segmentation algorithms and region-based segmentation algorithms. Edge-based segmentation algorithms operate through determining the discontinuity of pixel features. Region-based segmentation algorithms used the similarity of pixel features to generate homogeneous regions [39]. Moreover, literature review on region-based segmentation algorithms has been discussed. Meanwhile, variations of region-based segmentation algorithms have been studied and discussed from subsections 2.1.1 to 2.1.4.

In literature, region growing is employed to achieve precise medical image segmentation while addressing the issue of over segmentation (OS) [40]. In [41] authors utilised a region growing approach that utilises gradient and variance as criterion to identify initial seed pixel and perform segmentation. The authors in [42], employed region growing by utilising edge and smoothness factors as criterion to identify initial seed pixels, subsequently applied seeded region growing (SRG) approach to perform final segmentation. In [43], author provides comparative review on region growing, seeded region growing (SRG), and improved seeded region growing (ISRG). The author mentioned that improved SRG shows promising results. As mentioned earlier, the selection of the initial seed in the SRG is critical, as it determines the overall segmentation

outcome. In previous studies of [4], determine the initial seed through edge-based segmentation, subsequently utilised the centroid as the initial seed to perform final segmentation. The authors in [44], employed the Harris corner detector to compute the initial seed pixels in SRG. In previous work [45], the authors utilised automatic seeded region growing (ASRG) algorithm in which the process of selecting initial seed pixel is selected automatically. While, genetic algorithm was employed to conduct a search for the selection of initial seed pixels. In [43], ASRG approach has been used for retinal image segmentation. In [46] proposed a method that utilises the Felzenszwalb and Huttenlocher (FH) algorithm [9] for the initial over segmentation (OS) generation. Subsequently, an autoencoder model is proposed to perform the final segmentation.

In previous work [47], the authors implemented Quick Shift (QS) algorithm for image segmentation, using a density probability function to connect each pixel to the next higher-density pixel. The function calculates distances based on colour and spatial differences. Distances exceeding a defined threshold are discarded, resulting in segments that correspond to the image region of interest (ROI). Meanwhile, in another previous work [48] the authors compared Region Growing, simple linear iterative clustering (SLIC) [14], Watershed [49]. Mean Shift (MS) [50], FH [9] and Voxel cloud connectivity segmentation (VCCS) [51] algorithms to generate over segmentation (OS) on medical images. The evaluation metrics such as homogeneity, moment of inertia, shape and size uniformity show that VCCS outperforms among others.

Recently, the researchers in [52] [53] compared the efficacy of four region-based segmentation algorithms in delineating images of medical [52] and natural [53], respectively. Moreover, Felzenszwalb and Huttenlocher (FH) [9], Compact Watershed (CW) [10], Quick Shift (QS) [13] and simple linear clustering (SLIC) algorithm [14] were included in this group of algorithms. The evaluation criteria were boundary recall (BR), under segmentation error (USE), and achievable segmentation accuracy (ASA). Based on this, findings reveal that boundary recall (BR) is much better among other three mentioned region-based segmentation algorithms. The authors of references [54, 55] compared the performance of FH, QS and the SLIC algorithms. The previous research work [54], utilised unmanned aerial vehicle (UAV) images to segment vehicles. The comparison was based on speed and accuracy as the evaluation and results indicates that FH shows better performance as compared to QS and SLIC in segmenting vehicles. Additionally, the authors in [56], compared FH, QS, and SLIC and mentioned that QS performs best in order to generate over segmentation (OS). The author in [55] utilised remote sensing images by employing evaluation criteria which is adapted rand error (ARE). Moreover, authors mentioned that the SLIC outperforms FH and QS algorithms in segmenting building regions. The segmentation results demonstrates that the boundaries of the image object regions were aligned correctly and obtains better average values of ARE metric [55].

Hence, based on the findings of previous works [52, 54, 55] this research work will utilised the FH [9], CW [10], QS [13] and SLIC [14] region-based segmentation algorithms for comparison purposes to identify most suitable one for generating initial over segmentation (OS) on WHU buildings dataset [37].

2.1.1 Region Growing Algorithm and its Variations

According to authors in [57], region growing algorithm starts from a pixel and grows iteratively in all directions to group the adjacent pixels on satisfying merging criterion (MC). The pixels are labelled as a region that grows based on the features characteristics similarity such as pixel intensity, colour, or texture. The edge pixels are associated with that region where the pixel intensity is the higher among its neighbours. In this right-left and top-down directional movement, the higher intensity pixels are grouped first. Once all the high intensity pixels grouped together, this generates a region. Then this process continues with the lower intensity pixels, and the pixels having lower intensity are compared with the higher intensity pixels and resultantly generates boundaries and objects as segmented image [57]. Region growing algorithm has the advantages of dividing regions with similar features and provides comprehensive edge information, this resultantly helps in generating good segmentation. While, the drawback is its excessive over segmentation (OS). Additionally, noise and uneven features causes severe OS [58].

In literature various variations to region growing algorithm have been proposed. Seeded region growing (SRG) proposed by [35], is a method that is based on the traditional region growing concept of grouping pixels having similarity within regions. It iteratively incorporates adjacent pixels that exhibit similar features including colour, texture, or pixel intensity. The SRG is characterised by its efficiency. However, its main drawback is the selection of initial seed pixel. This process can result in inconsistent regions [59]. The final

result of segmentation may differ with each execution, influenced by varying seed selections, which indicates the algorithm sensitivity [60].

Furthermore, to address the issue in SRG algorithm, the Improved Seeded Region Growing (ISRG) algorithm was proposed [61]. The ISRG is based on a pixel order independence and was improving SRG. It is because SRG has two inherent pixel order dependencies known as initial seed selection that result in different or irregular generated segments after each execution. The ISRG algorithm has the advantages of pixel order independence, and resultantly generate accurate segmentation than SRG [61]. However, the main drawback of ISRG being more complex than the SRG algorithm, it means that it is selecting the specific points to start the process. Besides SRG and ISRG, another improved variation of traditional region growing algorithm was proposed by [62] known as Unseeded region growing (URG) algorithm. The URG algorithm executes in 3×3 window and start from left to right and top to down. The main advantage of URG is that it does not rely on initial selection of pixel known as seeds as compared to SRG and ISRG, respectively. However, the drawback in URG is that it is also difficult to start the execution at initial stages [63]. Another improvement to region growing algorithm is known as automatic seeded region growing (ASRG) [64], having similarities to SRG, with the exception that no seed selection is required. The algorithm work as seed will be selected automatically. As a result of being a region-based segmentation algorithm, this algorithm generated better segmented regions. The issue in ASRG is that it will produce different segmentation results each time because of automatic selection of seed [59].

The region growing algorithms and its variation leads to proposed other approaches in which the image is represented as an undirected weighted graph. Based on this concept, FH [9] was proposed. The process depends on the selection of edges from graph, and similar pixels in graph represents the node and dissimilarity among pixels are interlinked by the undirected edges. The dissimilarity of neighbouring pixel referred as nodes is calculated by assigning weights to each edge based on their feature values, which is used for further merging process. Despite the fact that the FH algorithm produces over segmented regions, but still the regions are aligned with the image objects boundaries [9].

2.1.2 Watershed-based Algorithm and its Variations

Watershed algorithms segment images by a pixel-priority labelling in which pixels of different priority can have different priority to be labelled. There are mainly two such types of algorithms, immersion based and topographic distance watershed. The concept of an immersion approach, as proposed in [65], regards extensive simulation of water falling over a topographic surface, accumulating in catchment basins delimited by watershed lines. The basins fill and merge to bring in the boundaries of the final watershed until this process is repeated. The watershed by [66] topographic distance defined as an over the values or gradient magnitudes divides an image, so the image is seen as a terrain with ridges and valleys, and the watershed lines are the boundaries between, basins limited by watershed lines. As stated by [67], watershed algorithms have closure and region connectivity as advantages, and have object boundaries well aligned. However, these watershed-based algorithms suffer from the problem

of excessive over segmentation (OS) and issues due to noise effects in image regions [67].

As a variation of watershed-based, a marker-controlled watershed algorithm is proposed in [68]. The algorithm uses a prioritised queue set from pixel intensity values. Adding first the pixels to two queues which are Q1 of higher and Q2 for lower pixel intensity. It repeats until a queue is not filled with all pixels. Based on this, marker would be the local minimum or a pixel with the lowest intensity. Moreover, markerless pixels are assigned to the closest marker in order to traverse the process. This algorithm works well but suffers from over segmentation (OS) in which regions become smaller, more complicated shapes and less compact in size [69]. The Compact Watershed (CW) algorithm [10] is the improvement of marker-controlled watershed algorithm [68]. It was subsequently refined by the authors in previous research work referenced as [10]. The authors of the CW algorithm utilised the approach of data structure from marker-controlled watershed approach discussed in reference [68]. Based on that, the authors in [10] incorporated Euclidean distance function and the compactness to assign the non-marker pixels to the marker pixels. As a result, regions that are divided into smaller parts are both compact and have a regular shape [10].

2.1.3 Density-based Algorithm and its Variation

Mean shift (MS) is a density-based algorithm which is originally proposed by [12]. This algorithm estimates the density of pixels in an image by connecting each pixel with a probability density peak. This density peak is calculated by first creating a window around the pixel and then calculating the

average of the pixels that fall inside the window. Later, the window is moved to the mean point, and the process is repeated until convergence. The advantage of mean shift algorithm is that this algorithm uses a specific kernel where the number of clusters is determined automatically. According to authors in [70] the drawback of MS algorithm is its long execution of process.

Quick Shift (QS) algorithm [13] is another density-based algorithm which shifts each pixel in feature space to nearest neighbour for segmentation. Moreover, shifting procedure is based on a weight computed from two adjacent pixels in feature space. Furthermore, weight is a calculated measure for density estimation of either similarity or dissimilarity between adjacent regions. In addition, QS utilises a 5-dimensional (5D) feature space incorporating colour information and pixel position within the features space. In QS, one tree node corresponds to each pixel. Next, all these nodes are regenerated by pruning of tree branches whose weights are above a predefine threshold in order to create segmented region. It is because a weight between two adjacent pixels in QS is based on density estimation of the 5D feature space having pixels and over segmented regions obtained closely match a boundaries of image objects [13]. The advantages of QS include speed, and simplicity, which are used to generate uniformed image object boundaries and to reduce the number of under segmentation (US) and over segmented regions, as stated in [18].

2.1.4 Clustering-based Algorithm

One particular type of clustering-based is SLIC [14]. As stated by authors of [71], the SLIC algorithm is an adaption of the well-known k-means clustering approach. Also, SLIC runs on the concept that an image contains N

pixels already partitioned into R regions of equal size beforehand. The region of equal-size is approximately N/R pixels in each one. The SLIC algorithm also initialises centres of clusters through pixels that are approximately equal in size, and reside near an approximate region. It then determines those corresponding pixels in a 2 × 2 neighbourhood that are similar and associates them with their respective cluster centres. The process also calculates the values of the new cluster centres. For each iteration, the average of similar pixels in each approximately equal-sized region determines the cluster centre. Furthermore, average distance between two cluster centre will be calculated by $D = \sqrt{(N/R)}$. Thus, the whole process goes on in a repeat manner until all the pixels and regions of equal size are considered. After all the pixels have been traversed the process comes to an end. SLIC clusters pixels based on pixel intensity, which occurs in 5D feature space. The authors in [14] indicated that the SLIC algorithm requires only a single parameter which is k. This parameter represents the number of segments to be generated. As discussed in [14], the SLIC algorithm divides the image into segmented regions having well with object boundaries in an image [14].

2.2 Region Merging Approach

According to [26], the region merging approach progressively merging the over segmented regions to generate final segmentation having aligned region of closed boundaries [22]. Region merging process is divided into the graph-based, statistical-based, and marker-based region merging approach [72]. According to several previous research works [22, 25, 28, 33], graph-based region merging is easy to implement as compared to other two approaches. The

region merging approach is mostly influenced by the three key factors. First factor is the merging criterion (MC), which defines the similarity of neighbouring regions. Furthermore, the second factor is the order of merging, which decides in what order regions should be merged. Lastly, the region model, which represents over segmented regions [26]. Moreover, region merging model and MC are discussed in Sections 2.2.2 and 2.2.2.1, respectively.

According to author in [22], in region merging, the merging order can be categorised into three types depending on the merging strategies which are local, global, and hybrid region merging (HRM). In local-oriented region merging approach, similar neighbouring pairs of regions are merged based on defined criterion. According to [22], the global-oriented region merging approach works as the regions that are globally best fitted are merged during merging process at each iteration. The iterative searching and merging continue until finish the over segmented regions. The process of finding the global best fit regions searching takes time [22]. While, the integration of global- and local-oriented merging strategies are referred to as HRM. The global merging strategy of a growing region in HRM is defined by starting with some pair of neighbouring regions that are the most similar in the global sense. Next, merging most similar region neighbours of a growing region is performed following the local-oriented merging strategy [22].

As previous research works [22, 23, 73, 74] have utilised graph-based region merging method and employed four prominent features such as colour, texture, shape, and edge to derive MC and perform iterative merging process. Hence, the proposed approach in this research work will utilise graph-based

region merging method and four prominent features to derive MC to perform region merging.

2.2.1 Review of Region Merging Approaches on Remote Sensing Images

In this subsection, a summary of literature has been presented in Table 2.1 having region merging approaches utilised for remote sensing images. The reference [26] utilised remote sensing images and proposed deep learning model to perform buildings segmentation. The MC is derived from different features such as texture, colour, edge, smoothness, mean value of pixels. The metrics utilised to evaluate the proposed deep learning-based region merging approach are F-measure, precision, recall, and total error. The results shows that the proposed approach of [26] achieved better results in delineating building regions. Afterwards, reference [28] as demonstrated in Table 2.1 utilised WHU buildings dataset and proposed region merging approach to delineate building regions. The approach utilised F-measure, precision, and recall metrics to evaluate the approach. The results indicate the better segmentation accuracy for building delineation. In reference [23], authors utilised synthetic aperture radar (SAR) images and proposed region merging approach. The texture and statistical values such as mean and standard deviation are utilised to derive MC. The results show superior performance in terms of evaluation by employing F-measure, precision, and recall. In [33], authors utilised remote sensing images and proposed machine learning-based region merging approach. The random forest classifier is used to decide merging. The classifier act as MC and perform region merging. The results indicate better performance of the proposed

approach in [33]. Furthermore, [75], utilised remote sensing images and proposed region merging approach. The work derives MC from edge feature. The approach was evaluated by employing F-measure, precision, and recall, where the results indicate the proposed approach perform well.

Table 2.1: Summary of region merging approaches for remote sensing images segmentation

Ref. / Year	Dataset / Image Type	Initial Segmentation	MC Features	Metrics	Pros	Cons
[26] / (2024)	RGB images of Arizona, U.S., (e.g., WorldView, QuickBird, IKONOS, etc.)	Multi-resolution Segmentation (MRS) algorithm [28]	Texture, statistical, shape features, standard deviation of each band, mean value of each band, shape indicator, compactness, brightness, border indicator, and resizing factors.	F-Measure, precision, recall. Total Error, global over segmentation error (GOSE), global under segmentation error (GUSE).	Proposed method shared the significance of accurately delineating building ROI having merging criterion (MC) defined through deep neural network.	Features of a newly merged superpixel can cause inefficiency in region merging if they are re-extracted by the pre-trained model.

[28] / (2022)	WHU Buildings dataset / Remote Sensing Images	Watershed transform algorithm	Homogeneity, heterogeneity, and illumination.	F-measure. precision, recall.	Used building features for threshold which shows reasonable segmentation accuracy.	An image analyst digitized the building features map. Manually delineated ground truths (GTs) of buildings.
[76] / (2021)	Three synthetic aperture radar (SAR) images with manually annotated	Initial Over segmentation based on multiscale Bhattacharyya approach.	Statistical similarity measure (SSM), Texture pattern similarity measure (TPSM) with the relative common	F-measure. precision, recall. Variation of information (VOI), rand	Utilised only texture features and perform segmentation on satellite images, resulting better segmentation results.	The final results have over segmentation in the village and urban regions in images which corresponds to

	boundaries of regions.		boundary length penalty (RCBLP).	index (RI), and segmentation covering criterion (Cov).		incorrect segmented regions.
[33] / (2020)	Chinese remote sensing satellite images, Gaofen-2.	Seeded Region Growing (SRG) algorithm.	8 merging criterions contains following features; region size, Compactness, Smoothness, Band-mean spectral standard deviation, Spectral heterogeneity,	Global over segmentation error (GOSE) and a global under segmentation error (GUSE).	Proposed method is the significantly different from traditional approaches such that the machine learning classifier is utilised to take the final decision about the merging of regions with the	The challenge in this work is training the Random Forest classifier, as obtaining test samples throughout the iterative region merging process requires numerous human-computer

			Band-mean square error, Spectral angle mapper, and edge strength.		assistance of different MCs.	interactions, even for a small image.
[75] / (2020)	Spaceborne PolSAR dataset / Remote Sensing Images	SLIC algorithm	Edge Information, and Homogeneity measurement (HoM)	F-measure, precision, recall.	No need of parameters during running time.	Used only edge map features for merging criterion derivation.

2.2.2 Graph-based region merging

The graph-based region merging method is outlined in previous research work reference as [77]. This method works by the consideration of initial over segmented regions as graph having edges demonstrating intensity differences between regions. The difference is computed by the minimum weight edge connecting any two adjacent regions. The region adjacency graph (RAG) is one of the graphs used for performing merging. It is further elaborated in below subsection 2.2.2.1.

2.2.2.1 Region Adjacency Graph (RAG)

The region merging process executed via RAG [26]. Using the RAG, an over segmented image can be represented as an undirected graph such as $G = (V, E)$ as shown in Figure 2.1 (a). While, V represents an over segmented regions as a set of nodes, and E is a set of edges connecting adjacent nodes. There will be an edge connecting the two nodes if they are next to each other or adjacent. Furthermore, as shown in Figure 2.1 (b), each edge has the corresponding edge weight to calculate the similarity or dissimilarity between two nodes [26]. Below image in Figure 2.1 (a) is over segmented via SLIC having parameter of k=100, and RAG Graph is generated from over segmented regions.

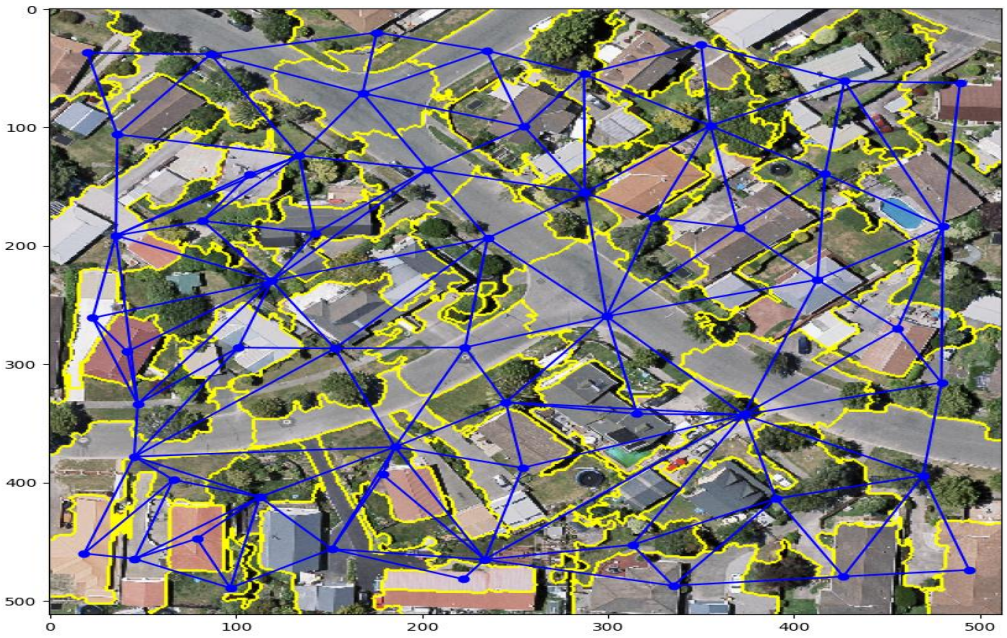


Figure 2.1 (a): RAG representation for Over segmented regions

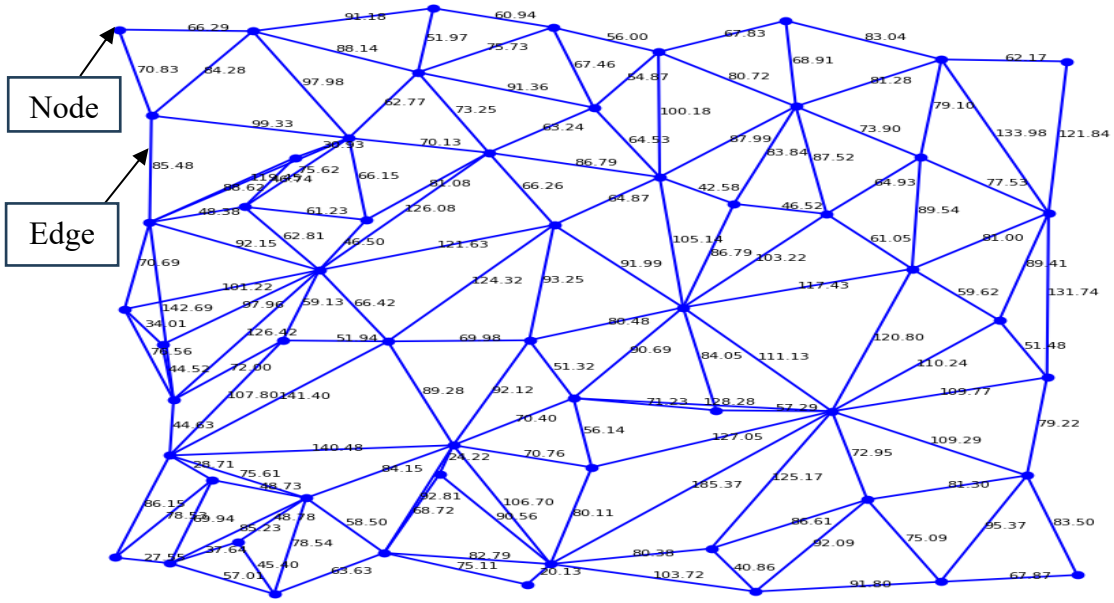


Figure 2.1 (b): RAG Nodes and Edges

2.2.2.1 Merging Criterion (MC)

In region merging approach, the formation of an appropriate merging criterion (MC) is essential for effective segmentation. Additionally, the one of the important aspects in region merging approach is to merge over segmented regions iteratively using the MC [22]. According to authors in [4, 29], MC

measures how similar the regions are and is used to determine whether a merge is desirable. In previous research works [33], a region merging approach based on machine learning is proposed, where a random forest (RF) classifier determines the merging by using various combinations of feature information as input to the binary classifier. This classifier act as MC to decide whether merging is feasible or not [33]. In [25], a deep learning based approach for deriving MC in unsupervised region merging segmentation is introduced, using the FCD U-Net model to extract feature maps. The model processes the input image at multiple levels in its hierarchical structure, generating feature map that capture various visual features. These feature maps are then used by the SLIC algorithm to improve region-based segmentation. The region merging approach takes these over segmented regions as input and merges them iteratively to create segmented regions. In another research work [26], the author presents a novel region merging approach that utilises deep learning. The proposed approach involves employing a Siamese network with a S2Former model as its backbone. The MC is derived based on deep learning model for iterative regions merging.

In literature four prominent features are mostly utilised to formulate MC which are colour, texture, shape, and edge. For instance, the research works [25, 33, 73] have utilised the combination of colour, texture, shape, and edge feature for deriving MC to perform iterative region merging. In previous work [25], authors proposed MC comprised of four features mentioned above and named as high dimensional feature vector (HDFV). While, [25] utilised different colour spaces along with red, green, blue (RGB), for texture feature extraction Gabor Filters has been employed, for shape feature an area attribute has been

considered, and for edge feature a spatial location has been identified of pixels. The authors in [73] used colour, texture, and edge information which were extracted through Canny edge detector and employed in MC. In another work [33], 8 MCs were derived and categorised into three groups such as geometry, spectral, and spatial context-based [33]. The method in [73] incorporates both within-segment and between-segment differences, while also considering segment area and common boundaries to derive MC and resultantly enhance the segmentation process [23]. The authors in [78] utilised watershed algorithm to generate initial over segmentation (OS) and subsequently employed region merging approach to reduce OS. The MC defined in work is based on the combination of the region homogeneity and edge integrity. Then RAG is created from OS and MC is applied to perform iterative merging process. The authors in [24] proposed region merging approach for remote sensing image segmentation. The MC named as reference merging criteria group (RMCG) was utilised and derived from spectral and spatial heterogeneities of region.

2.3 Feature Extraction

In literature, number of strategies are provided for feature extraction, with certain techniques relying on colour, texture, geometric, and statistical features as demonstrated in Figure 2.2 referred from previous work [79].

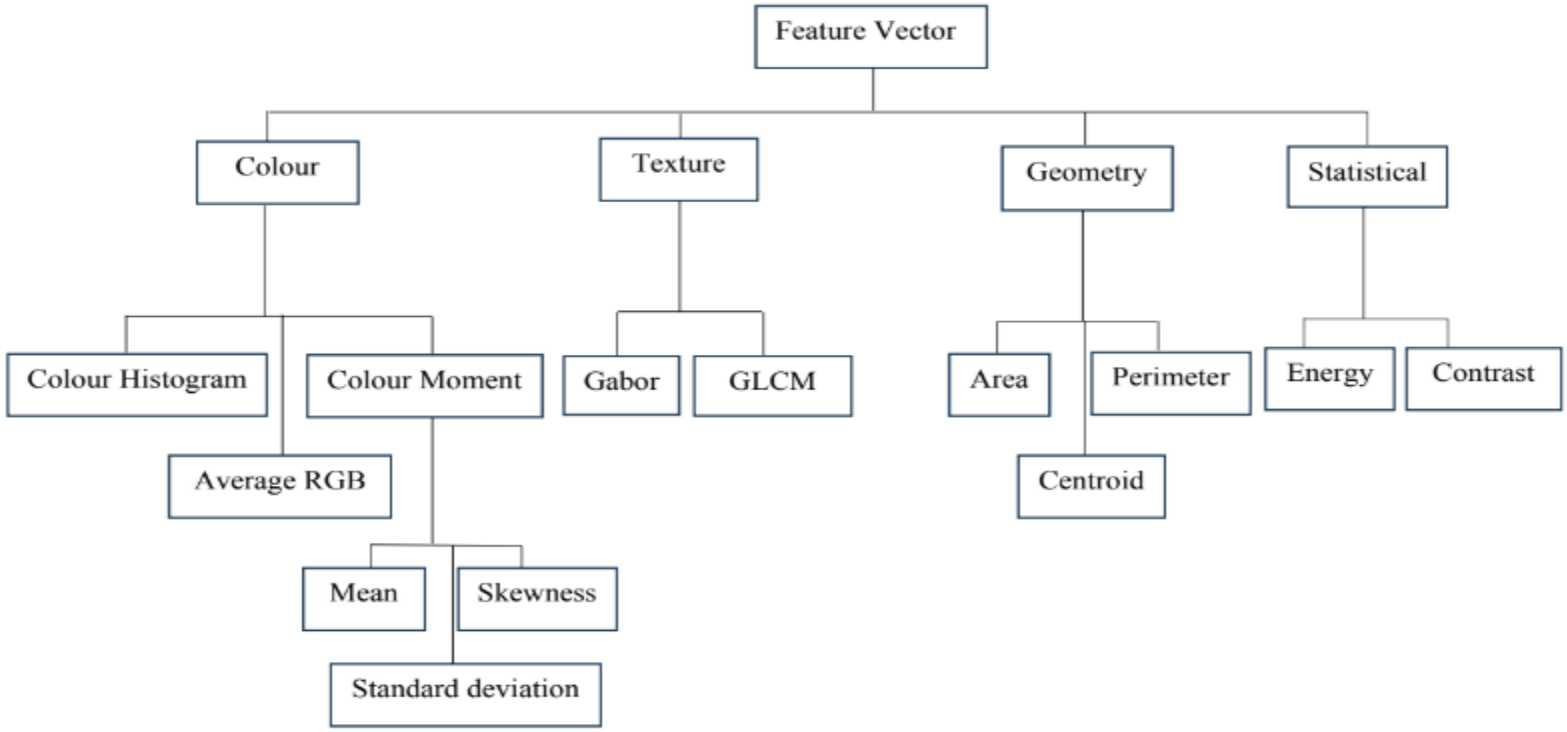


Figure 2.2: Feature Extraction Methods [79]

2.3.1 Colour Feature

The most important visual content about the image are colours, all images are collection of pixel values [80]. Secondly, colour spaces are used while describing or analysing image pixels and multidimensional space of colour components is called a colour space [81]. Many colour spaces exist in the literature to use for extracting colour features. According to [82]. RGB colour image is said to be of 3 channels which comprises of first red channel second green channel and the last blue channel [80]. In addition, RGB colour intensity is 0 to 255. Additionally, each of three main colours mixed creates different colour pixel [80].

In the RGB colour space, the combination of three channels (0,0,0) represents the black colour, while the combination of three channels (255,255,255) represents the colour white [83]. The (Hue, Saturation, Value) HSV colour space derived from the RGB colour space is calculated from RGB to HSV and vice versa. The HSV colour space is easier to define, invariant to illumination and capture orientation [82]. All values are 0 and 1 in HSV colour space and an image has 3 channels. Furthermore, any of the channels contain numerical values of the Hue (H), Saturation (S), and Value (V). Moreover, if the numerical value is 0 it shows dark, and shows white as increases with to 1. While the combination of three channels (255,255,255) represents the colour white [83]. The (Hue, Saturation, Value) HSV colour space is derived from RGB and easy to convert from RGB to HSV and vice versa. The HSV colour space is easier to define, invariant to illumination and capture orientation [82]. In HSV colour space, all values are between 0 and 1and an image has three

channels. All channels contain numerical values for Hue (H), Saturation (S), and Value (V). The V channel colour is dark when the numerical value is 0, but brightness as it approaches 1. Brightness is the term V. Colours change each one of them from unsaturated to fully saturated as saturation goes from 0 to 1 [84]. In an HSV colour space, however, each pixel of an image has one colour value, as given by the Hue channel. The S and V channels determine how much black and white colour has been added to the Hue to help differentiate objects from other colours. A colour channel extraction in RGB images for segmentation. In addition, different colour spaces have been utilised in terms of feature extraction not only for colour feature extraction. For example, the RGB, HSV, CIELAB and CIEXYZ colour spaces were used as in previous work [25]. The authors in [83] compared RGB and HSV colour spaces.

2.3.2 Texture Feature

Apart from colour, texture is visually perceptive and provides important image features information [85]. As humans view the image as a whole, it is observed that some of the objects do not appear regular in a small region. Hence, this combination of local irregularities and general regularities describe as Texture. It is challenging to find a single mathematical definition of texture, despite humans clear perceptual understand of the features in images [73]. Moreover, texture represent a surface and the structure of image and defined as a regular repeating of an element or pattern on a surface [7]. While, main properties of texture is the repeated pattern of the spatial pixels in an image, which is the set of small variations, mostly as the function of position [81]. The texture representation has two main approaches which are structural and

statistical [2]. A statistical approach calculates texture feature using the statistical distribution of observed intensities at given image pixels position. Moreover, number of pixels representing local feature, statistical approaches can be the first-order (one pixel), second-order (two pixels), or the higher-order (three or more pixels) [82]. This research will describe two texture feature extractors, which are Gabor filter and local binary pattern (LBP) for texture analysis to choose the best one.

LBP was introduced as an efficient texture feature descriptor and it represents the pixels local structure as texture [82]. In addition, LBP computes an eight-bit code information for each local neighbourhood pixel. For every pixel in the centre, it assumes that there is a neighbourhood of arbitrary size. After that, a binary code with eight bits is computed for the central pixel, and this code information can be represented by a decimal number [112].

The Gabor filters are bandpass and can be represented as the product of a complicated sinusoidal wave and a Gaussian envelope. The Gabor filters behaviour depends on both the frequency and the orientation of the sinusoidal wave. These filters have a major part in image processing and for feature extraction and texture analysis. As in past research works [25, 86], Gabor filters have found use in image segmentation. Moreover, a better result for feature extraction of textures on the Gabor filters is obtained than by other texture descriptors [87].

2.3.3 Shape Feature

In literature, shape is also considered as an important feature for deriving MC. A shape can be characterised by various attributes. The shape feature attribute include area [88]. Moreover, the previous works by [25], and [22] have employed shape feature with an area attribute for deriving MC.

2.3.4 Edge Feature

According to the authors in [73], edges in an image are defined as discontinuous variations in pixel intensity values, which typically occur at the boundaries between regions of distinct objects [73]. In [89], authors make comparison of two prominent edge detectors which are Canny and Sobel. In [89], the performance of these edge detector are evaluated by the use of metrics which are the peak signal to noise ratio (PSNR), the mean squared error (MSE), and the root mean squared error (RMSE). Their results illustrate that both the Canny and Sobel have distinct advantages that provide insights into their effectiveness in edge detection when dealing with image segmentation. [89]. Another previous research work as [90], depicts the importance of edge detection in object boundary extraction and recognition, resultantly highlighting the importance of understanding the differences between edge detection. In [90], authors compared the performance of edge detectors. Their results indicate that the Canny edge detector has successfully achieved superior accuracy in the detection of object edges with higher entropy, PSNR, MSE, and execution time compared to Sobel, Roberts, and Prewitt edge detectors [90]. In [91], authors compared Sobel, Prewitt and Canny edge detectors, which were used to extract edge information. Furthermore, performance of the mentioned edge detector are

analysed on the metrics which are MSE and PSNR. The experimental results proves that the Canny edge detector results better than Prewitt and Sobel edge detectors [91].

2.4 Convolutional Neural Network-Based (CNN-based) Deep Learning Models

Convolutional neural networks (CNNs) serve as a significant subtype of deep learning (DL) technologies, demonstrating rapid development and increasing attention in recent years [92]. Additionally, convolutional neural network (CNN) models consist of input, hidden, and output layers. The layer of the model that receives the data is referred to as the input layer, whereas the layer of the model that determines the appropriate action for that data is known as the output layer. Between the input and output layers, there are hidden layers that perform the calculations. Thus, a convolutional neural network (CNN) consists of multiple hidden layers situated between the input layer and the output layer [93].

Recently, the researchers have adopted CNNs for generating feature map of ROI [94]. CNN models have become the prominent choice for this task. It is because CNN models has the ability to learn automatically about relevant features from the images without requiring human intervention [95]. Thus, first model used in this research is U-Net, a widely recognised CNN-based deep learning model known for its most effectiveness in biomedical images segmentation [96].

The performance of standard V-Net, SegNet, SegU-Net, ResU-Net, MultiResU-Net, and the AttentionU-Net will be evaluated with the U-Net model in this research. The evaluation will be based on generating feature map of building regions ROI in WHU buildings dataset, and it consist of the various colours, shapes, sizes, and textures features.

Reference [95] states that U-Net, the CNN-based deep learning model was initially used in the medical image segmentation. It is also very popular in the area of remote sensing image segmentation. The authors in previous work [37] used the U-Net for generating buildings feature map from WHU buildings dataset of 512 × 512 dimension with the corresponding ground truths (GTs). The hyperparameters like binary cross-entropy (BCE) as loss function, rectified linear unit (ReLU) as an activation function and sigmoid classifier were used to train the model. Furthermore, U-Net benefits from the skip-connections, which allow to keep spatial feature information, and generating feature map of ROI accurately [37].

Previous works [97, 98] have experimented U-Net model and conclude that U-Net may have difficulty in inducing feature map for ROI with various characteristics, which motivates the exploration of U-Net variations. U-Net [96] has also demonstrated slightly irregular and blobby edge of objects in generating feature map of ROI or objects [97]. Based on this, various U-Net variants have been studied and some of the U-Net variants in this research were implemented. In addition, different variations or modified models based on the standard U-Net [96] are discussed in subsections 2.4.1 to 2.4.7.

Furthermore, the variants of U-Net [96] based models are V-Net [99], SegNet [100], SegU-Net [101], ResU-Net [102], MultiResU-Net [103], and AttentionU-Net [104]. The robustness of the mentioned models in producing the feature map of the buildings and the road regions as ROI in the remote sensing images is evaluated in the existing works [96, 102-104], [99]. These models are evaluated for their capability of generating feature map of irregular structure ROI in the medical images, both individually and in a pair. In precisely, complexity of ROI in chosen image determines the performance of CNN-based deep learning model in generating feature map to ROI [105]. However, ROI with the higher complexity is made by remote sensing image for the generation of feature map [106].

In several studies, U-Net has been compared to its variations, for instance, V-Net and SegNet for image segmentation tasks. In [107], authors also compared V-Net and U-Net, and V-Net achieved slightly better performance than U-Net with overall accuracy (OA) of 99.82 versus U-Net's 99.78 [107]. Additionally, on comparison of U-Net with SegNet for remote sensing image, results were found with slightly best OA of 88.1 for SegNet and 87.0 for U-Net [108]. In reference [109], authors compared U-Net with SegNet where U-Net, gets higher OA of 94.66 compared to SegNet 93.59. As presented in [110], authors compared SegNet with U-Net and stated that SegNet outperformed the U-Net especially achieving 0.4463 intersection over union (IoU) compared to U-Net 0.3752 in remote sensing images.

2.4.1 U-Net

The U-Net CNN-based deep learning model presents a symmetrical layout, consists of an encoder (left side) and a decoder (right side). It contains four blocks of convolutional layers, and along with these, the comprising of a bottleneck block (bridge between encoder and decoder). The encoder part of the model uses convolutional layers, ReLU activations, and max-pooling to reduce the dimension of the data. Moreover, in U-Net, each block contains two convolutional layers having kernel size of 3 × 3 and 64 feature channels [37]. In addition, the max-pooling phases in the encoder process decrease the size of the feature map, while simultaneously increasing the number of feature channels. This ultimately results in a bottleneck that stores the representation of dense features. The spatial dimension restoration is achieved by the use of up-sampling and the convolutional layers. The skip-connections are employed to maintain spatial feature information by connecting the encoder to decoder. This up-sampling step reduce the number of channels using the filter size of 2 × 2 (de-convolution), and skip-connections with the corresponding features map from an encoder. Moreover, each decoder block contains two convolutional layers of 3 × 3 filter, following one by one, a ReLU activation function, followed each after convolutional layer, and a 2 × 2 filter size max-pooling operation with stride 2. Then the final layer using a 1 × 1 convolution with having softmax classifier. The last layer of U-Net consists of the convolutional layer that utilises the softmax classifier to categorise the final feature map into ROI. In total, the U-Net model has the 23 convolutional layers, 18 ReLU activation functions, 4 max-pooling operations, and 4 skip-connections.

2.4.2 V-Net

The second model chosen in this research is V-Net, which is the extension of the U-Net [96] introduced for the 3-dimensional (3D) medical images segmentation [99]. The previous work [111] also applies this for road regions segmentation on the 2-dimensional (2D) remote sensing images. The V-Net model is similar to U-Net, using the encoder and decoder model, but additionally utilises element wise summation between encoder and decoder which helps in the propagation of fine detail feature map in skip-connections. V-Net also uses the de-convolutional layers for down-sampling, and changes up-sampling during training. Its encoder is made of 4 blocks with progressively increased feature channels, using 2 × 2 de- and up-convolutional layers for feature map up and down sampling. Model consists of the first block (single convolutional layer), the second block (two convolutional layers) and the third and fourth block with three convolutional layers, respectively. V-Net performs a change to the U-Net max-pooling replaced with convolutional processes and adds in the residual function via an element-wise summation and concatenation. V-Net also uses hyperparameters like binary cross entropy, PReLU as activation function, learning rate and batch size. Yet, V-Net suffers from inadequate generation of a feature map for features of various characteristics [111].

2.4.3 SegNet

The third model in this research is the SegNet, an encoder-decoder CNN-based deep learning model similar to U-Net but with key differences in its skip-connection process. SegNet does not employed skip-connection like the U-Net, rather to propagate feature map in the decoder, SegNet uses max-pooling

indices that are passed from encoder to decoder of the network. Convolutions, ReLU activation and 2 × 2 of max-pooling by stride 2 are applied to the encoder, capturing hierarchical features, and save indexes for later decoding by the decoder. Instead of U-Net, which uses full feature map, in the up-sampling process max-pooling indices are retrieved from the encoder. It is made of total 26 convolutional layers divide into five encoder and the decoder blocks, and concludes with the sigmoid classifier. In fact, reference [112] mentioned that SegNet is not suitable for generating feature map of small ROI. It is due to the ablation of spatial information which can occur during max-pooling, hence resulting inaccurate generation of feature map for building ROI in remote sensing images [112].

2.4.4 SegU-Net

The fourth model selected in this research is SegU-Net, which is originally proposed by the authors in reference [101] for traffic signs detection in videos. However, the same SegU-Net model proposed in [101] is employed by the authors in reference [113] for remote sensing images. Based on [113], SegU-Net is considered in this research for remote sensing images to generate feature map of building. Inspired by U-Net [96] and SegNet [100], SegU-Net [113] is proposed. The SegU Net [101] has an encoder and a decoder part. The encoder is composed of five blocks consisting of two 3 × 3 convolutional layers each followed by batch normalisation (BN) and ReLU activation, with max-pooling operations to increase receptive fields. Furthermore, receptive fields expansion is followed by some 3 × 3 convolutional layers with BN and ReLU in the third, fourth and fifth blocks. The number of feature channels double in

the fourth encoder block for down-sampling. At the bottleneck and the decoder stages, a series of convolutional layers are passed through the feature map and max-pooling indices from encoder is to be used for up-sampling. Final decoder layer [101] processes the up-sampled feature map and concatenated to corresponding encoder part.

2.4.5 ResU-Net

The ResU-Net is the fifth model utilised in this research, which is introduced in reference [102]. The authors in reference [102] proposed ResU-Net based on U-Net that integrates deep residual learning for enhanced road area segmentation in remote sensing images called the ResU-Net. First, compared with plain neural units, it employs residual units, which can achieve better efficient training and improve the feature map generation ability by using fewer convolutional layers. The decoder and encoder are composed of three separate blocks connected by a bottleneck. Unlike normal max-pooling, it does not employ stride 2 convolutional in residual units because that helps to retain more information of feature than just the upper portion. It combines a residual unit with up-sampled encoder's feature map to help better transfer information. The convolutional layer of 1×1 and sigmoid classifier is employed to make the final output. ResU-Net improves feature learning, but as experimented in previous research work [114], the authors stated that it has a longer training time because of its complex layout.

2.4.6 MultiResU-Net

The sixth model considered in this research is the MultiResU-Net, which is originally proposed by [103] for biomedical images segmentation. However,

the same MultiResU-Net model proposed in [103] is utilised by the authors in [115] for remote sensing images segmentation. As described by the authors in [103], MultiResU-Net uses the standard U-Net [96] encoder-decoder layout. In addition, the encoder part contains two 3 × 3 convolutional layers with a series that is finally followed by max-pooling operation of size 2 × 2 and stride of 2. Down-sampling is performed while iterated the mentioned series four times, subsequent to each one. The number of features channel in the encoder part is input twofold and it gets halved until bottleneck through the four times iterated series. On the contrary, the decoder first up samples the feature map via transposed convolution with 2 × 2 at which the feature channels are halved. Then, two convolutional with kernel size of 3 × 3 is done. The process is performed through the encoder, where up-sample and two convolutional layers iteratively executed. This iterative execution sequence would be repeated four times. The features channel is reduced by half in each afterwards block. Consequently, final segmented feature map is computed by performing 1 × 1 convolution operation. Thus, skip-connections utilised in U-Net [96] are used to concatenate the encoder feature map with corresponding decoder.

The authors in [103] build on top of the U-Net described within by adding an idea previously studied in reference [116], that two convolutional operations of size 3 × 3 is similar to a single convolutional operation of the size 5 × 5, enabling a multi-resolution approach. Moreover, in order to achieve more comprehensive feature map generation, the authors introduce the MultiRes Block, comprised of a sequence of three 3 × 3 convolutional layers with 5 × 5 and 7 × 7 convolutions following. The final layer is fed with the input features via a 1 × 1 convolutional so that dimensions are preserved, and a residual

connection is applied. In order to bridge the semantic gap between the encoder and decoder feature map, the encoder map is fed through additional convolutional layers rather than concatenating the encoder and decoder feature map directly. Residual connections are kept in focus as they facilitate feature learning and enhancing the final results of deep model [103].

2.4.7 AttentionU-Net

Finally, the seventh model utilised in this research is AttentionU-Net. The AttentionU-Net is the extension of U-Net, and it is utilised in generating feature map of ROI for remote sensing images [104]. Moreover, at the end of each skip-connection, AttentionU-Net integrates attention gates (AGs). While designing AGs, the element-wise multiplication, activation function of ReLU, and the sigmoid classifier are being utilised. These AGs assist to improve the resolution or representation of feature map traversed from the encoder to the decoder part by carrying out the element-wise multiplication between the input feature map and the attention coefficients [104]. The previous research referenced as [117], authors introduced the grid-attention mechanism into AGs. Based on this, the AttentionU-Net was utilised for the purpose of distinguishing or segmenting building regions in remote sensing images. The incorporation of AGs makes it possible for gating signals to extract the information about spatial objects in the feature map. Resultantly, this creates an improved generated the feature map at decoder part, which concludes with the sigmoid classifier [117].

2.5 Summary

In this chapter, a comprehensive literature review has been conducted, focusing on region-based segmentation algorithms, CNN-based deep learning models, region merging approaches. The aim was to identify optimal region-based segmentation algorithms, CNN-based deep learning models, and region merging approaches most effectively applied for image segmentation. In the region merging approach, the formation of an appropriate MC is crucial for determining segmentation efficacy. Hence, four prominent features from the literature have been explored for designing effective MC. Moreover, approaches to extract feature have been reviewed and discussed in detail. Furthermore, Chapter 3 builds upon this literature by outlining the overall experimental approach for implementing various region-based segmentation algorithms, CNN-based deep learning models, and a proposed region merging approach.

CHAPTER 3

RESEARCH METHODOLOGY

In remote sensing image segmentation, challenges such as over segmentation (OS) often arise, prompting researchers to propose various region merging approaches. Moreover, to solve the issue of OS, researchers have explored different techniques to derive the merging criterion (MC) and perform the merging process, particularly for delineating buildings as regions of interest (ROIs) in remote sensing images. The proposed approach in this research is useful to address challenges of buildings segmentation having varying shapes, and complex background from other structures in remote sensing imagery. By improving the accuracy and robustness of building segmentation, it enhances applications like urban planning and disaster management.

3.1 Research Framework

Figure 3.1 illustrates the framework for this research as describe below.

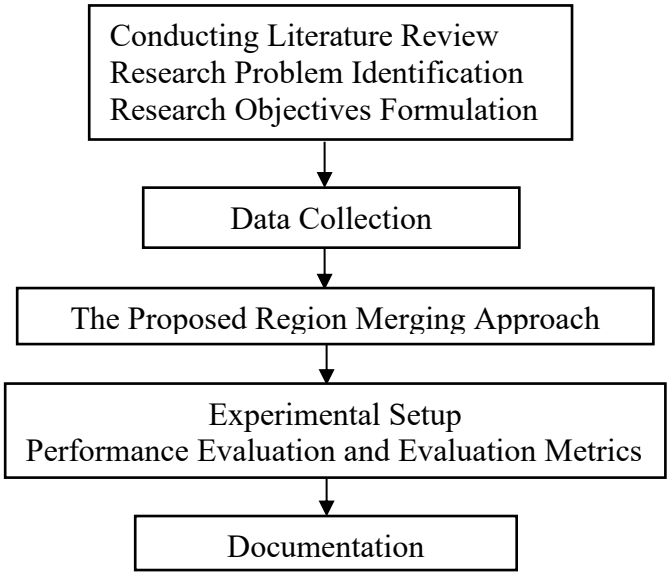


Figure 3.1: Research Framework

3.1.1 Conducting Literature Review, Research Problem Identification, and Research Objectives Formulation

As shown in Figure 3.1, the initial phase involves conducting a comprehensive literature review on the traditional region-based segmentation algorithms, region merging approaches, and convolutional neural network (CNN)-based deep learning models. Moreover, this review phase explores the existing methods on region merging approaches mainly on the variations of merging criterion (MC) derivation to perform merging for building regions in remote sensing images. After reviewing the existing approaches, a research gap was identified, leading to the identification of the research problem. In this research, the identified problem is that in similar existing works, the derivation of the MC required human intervention to perform region merging for buildings segmentation in remote sensing images. Subsequently, the research objectives were formulated based on the identified research problem. The objective of this research is to generate a building feature map by utilising a CNN-based deep learning model. Based on the generated feature map, prominent features were extracted to derive the MC without human intervention to perform region merging to delineate buildings in remote sensing images.

3.1.2 Data Collection

In this research, a publicly benchmarked images dataset known as WHU buildings dataset [118] were chosen. The dataset was released in 2019 and consists of 0.075-m resolution aerial images of Christchurch, New Zealand, covering 450 km^2 area. It contains a total of 8189 images, each with dimensions

of 512 × 512 pixels. Figure 3.2 illustrates the original images with the corresponding ground truth (GT) images.

This dataset was selected as it provides detailed spatial and geographical information of visual objects regions in the images [38]. Moreover, this dataset contains a collection of remote sensing images from multiple sources including satellite and aerial buildings [37] since the focus of this research is to delineate buildings region as ROI from the dataset. The images and their corresponding ground truths (GTs) that either display limited building regions or contain no building regions have been excluded from this research work.

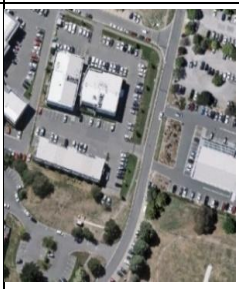
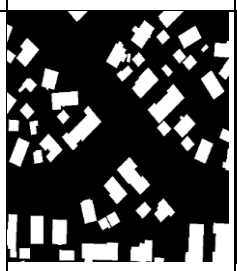
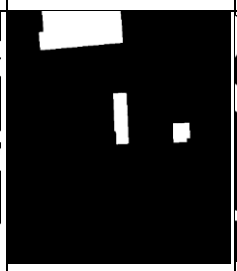
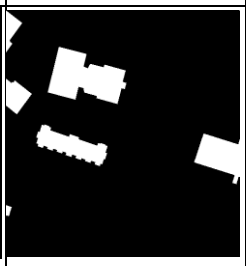
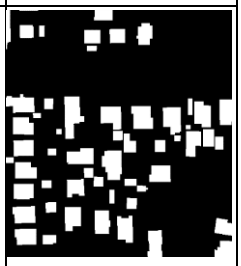
Image No.	**101**	**829**	**1027**	**1781**
Image				
GTs				

Figure 3.2: Original images and ground truths (GTs) of remote sensing images in WHU buildings dataset [37]

3.1.3 The Proposed Region Merging Approach

Figure 3.3 presents the flowchart of the proposed region merging approach which is discussed in detail in the following subsections.

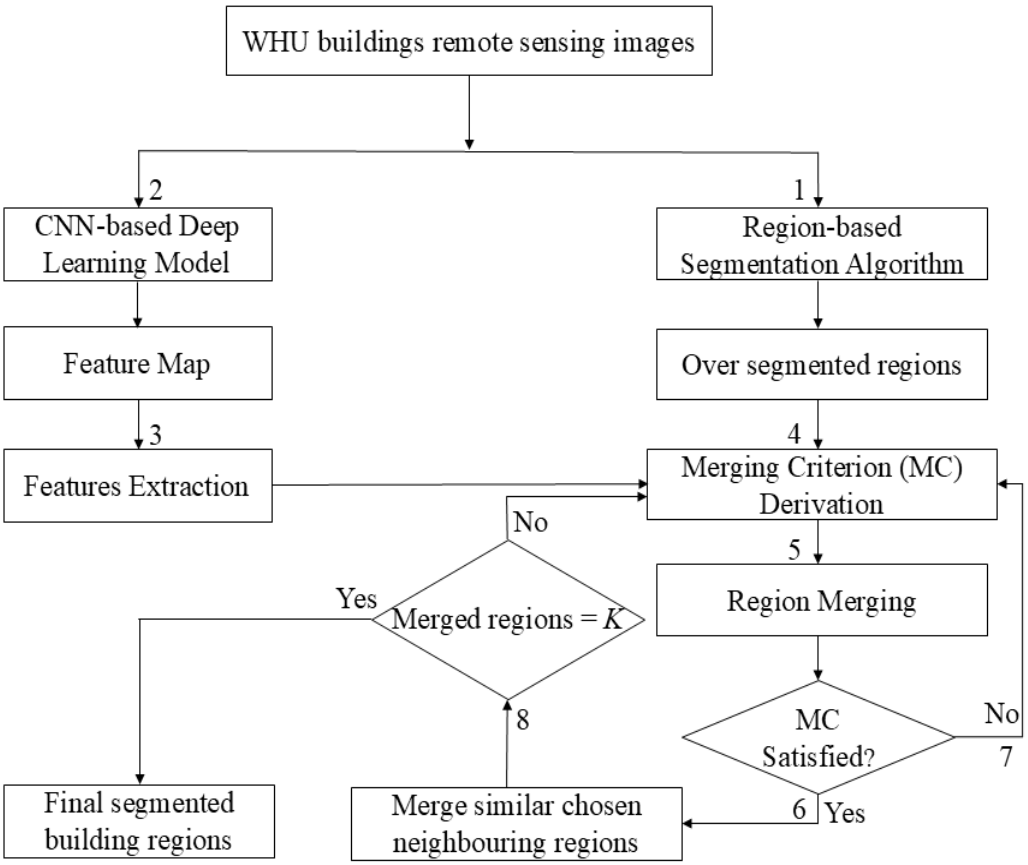


Figure 3.3: The Proposed Region Merging Approach

3.1.3.1 Region-based Segmentation Algorithm

As presented in Figure 3.3, first step of the proposed region merging approach involves the implementation and the performance comparison of four region-based segmentation algorithms on WHU buildings dataset [37]. The experiments and performance comparison are described in Chapter 4 Section 4.2.3. The purpose of the comparison is to select the best performing region-based segmentation algorithm in generating the over segmented regions which

are well-aligned with the image objects regions boundaries for delineating building regions in the WHU buildings dataset.

3.1.3.2 CNN-based Deep Learning Model

The second step is to select a suitable CNN-based deep learning model from the seven models being reviewed. The seven models were trained, validated, and tested on WHU buildings dataset to generate feature map of buildings regions. The performance comparison among these models have been included in Chapter 4 Section 4.2.4. Moreover, the original CNN-based deep learning models utilised varying input image sizes, as referenced in their respective research works [96, 99-104]. Therefore, for a standardised performance comparison among these models, this research has used the same input image size, batch number, learning rate, and epoch value for all the CNN-based deep learning models during training. During the training of all CNN-based deep learning models, only the hyperparameters were adjusted based on the results of the hyperparameter tuning experiments. As outlined in the second step in Figure 3.3, the best feature map generated for buildings as ROI from the CNN-based deep learning model will be used for features extraction to derive the MC for merging.

3.1.3.3 Features Extraction

As shown in Figure 3.3, the third step involves features extraction. The most prominent features of buildings regions in the WHU buildings dataset shall be identified and extracted from the feature map generated by the CNN-based deep learning model during the second step. The prominent features are extracted by identifying the best performing feature extractors. This is because

extracting the prominent features is crucial for merging criterion (MC) derivation to perform merging. [119]. Further discussions on these are provided in Chapter 4 Section 4.2.5.

3.1.3.4 Merging Criterion (MC) Derivation

As demonstrated in the fourth step in Figure 3.3, the extracted features were used to derive the merging criterion (MC) from the over segmented regions to perform the merging. The over segmented regions generated in a first step of Figure 3.3 shall be represented in a graph as commonly addressed in the existing works [26, 28, 120]. The graph is known as region adjacency graph (RAG) that represents regions as nodes while the MC is represented by the edge connecting pairs of nodes. Further details are discussed in Chapter 5 Section 5.2.4.

3.1.3.5 Region Merging

In the fifth step of Figure 3.3, the region merging is performed by iteratively merging pair of selected neighboring regions that were represented in the RAG using MC.

3.1.3.6 Condition of Merging Criterion (MC)

In the sixth step of Figure 3.3, the pair of regions in RAG will be merged if they fulfilled the condition of merging criterion (MC). Moreover, the MC will be recalculated after each merging is performed.

3.1.3.7 If Condition of Merging Criterion (MC) Not Satisfied

As shown in seventh step of Figure 3.3, if the adjacent pair regions do not meet the MC condition, they will remain unmerged in RAG.

3.1.3.8 Final Segmented Building Regions

In the final eighth step as illustrated in Figure 3.3, the merging process will continue until all the pairs of regions of similar features are successfully merged if the number of regions is equivalent to *K*, where *K* is the number of buildings in the feature map. The updated region adjacency graph (RAG) is checked against the condition "Merged Regions = *K*." This step ensures that merging stops once the number of merged regions matches *K*, the estimated number of buildings in the feature map. Resultantly, all the over segmented regions are iteratively merged to delineate building region as ROI in WHU buildings dataset. However, if the condition is not fulfilled, the process loops back to the merging criterion (MC) derivation step, where the merging conditions are re-evaluated for further refinement.

3.1.4 Experimental Setup

All the seven CNN-based deep learning models were implemented using TensorFlow [121], Keras [122] and scikit-image [123] open-sourced libraries. Python V3.10 was utilised as a programming language in which the execution of all seven CNN-based deep learning models was conducted on the Google Colab Pro+ integrated development environment (IDE) installed with the graphics processing unit (GPU) A100 along with 40GB of the random access memory (RAM). As presented in Table 3.1, desktop computer Core i7 having 11th Gen Intel (R), operating system (OS) of Windows 10, and 64GB RAM was utilised. Furthermore, Spyder IDE was used for the visualisation of the results while Google Colab Pro+ IDE was used to evaluate the performance of the proposed approach, respectively, using Python V3.10 having open-source

library of scikit-image [123]. Additionally, laptop with Intel Core i5 having Windows 10 OS was utilised to perform the experiments.

Table 3.1: Detail of system specification

System	OS	RAM	IDE
Desktop Computer	Windows 10	64GB	Google Colab
Laptop	Windows 10	32 GB	Spyder & Google Colab

Furthermore, Table 3.2, shows the open-source packages or libraries that were utilised for the implementation of the proposed region merging approach.

Table 3.2: Open-Source Packages or Libraries

Libraries	Functions
Python Imaging Library (PIL)	Loading and reading images
Numerical Python (Numpy)	Numerical operations on image data
Open-source computer vision (OpenCV)	Calculation of quantitative results
Pandas	Data saving in excel format
MatPlotLib	Visualisation of results as an image

3.1.5 Performance Evaluation and Evaluation Metrics

3.1.5.1 Performance Evaluation of Region-based Segmentation Algorithm

The region-based segmentation algorithms performance has been evaluated using two evaluation metrics which are the adapted rand error (ARE) [55] and the variation of information (VOI) [25]. The results are demonstrated in Chapter 4 subsection 4.1.2.2 of Section 4.1.2.

ARE measures the difference between incorrectly segmented boundary pixels against the ground truth (GT) as shown in equation 3.1 [55].

$$ARE = 1 - \frac{2\,(precision\ x\ recall)}{precision + recall} \tag{3.1}$$

where precision means positive predictive value while recall is true positive value or sensitivity of a segmented objects regions.

The range of the ARE is between 0 and 1 with lower values indicating better delineation results in relation to the GT [124].

Precision and recall are defined as in equations 3.2 and 3.3, respectively.

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

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

where true positive (*TP)* means the number of correctly segmented buildings region pixels in comparison to the GT. While false positive (*FP)* is the number of incorrectly segmented buildings pixels as compared to the GT and false negative (*FN)* is the number of incorrectly segmented non-buildings pixels in relation to the GT [125].

The VOI measures the distance between the pixels of the segmented results and the ground truth (GT) by their average conditional entropy [126] as represented in equation 3.4.

$$VOI = FalseSplit + FalseMerge \tag{3.4}$$

where *FalseSplit* are the missed pixels in the delineation result in relation to GT, while *FalseMerge* means that the pixels are merged incorrectly during delineation in comparison to the GT [126]. *FalseSplit* computed from the pixels that are incorrectly assigned to different segments or regions in the segmented result compared to the ground truth (GT). The calculation of *FalseMerge* is based on the pixels that are incorrectly merged during segmentation, when they should actually be merged to different segments according to the GT. The VOI often yields values greater than 1, as it is computed over the entire image and includes the sum of all segmented pixels. Consequently, this results in higher values. Conversely, a lower VOI value indicates better segmentation results in relation to the GT [125].

3.1.5.2 Performance Evaluation of CNN-based Deep Learning Model

The performance evaluation of all the seven CNN-based deep learning models involves visualising the generated feature map for buildings as region of interest (ROI) and using evaluation metrics. These metrics are intersection over union (IoU) [95], F-measure [127], precision [128], recall [128], and pixel accuracy (PA) [95]. These results are presented and discussed in Chapter 4 Section 4.1.3.3.

IoU or also known as Jaccard score known as the cardinality of intersecting pixels between the delineated results, *P* and the GT images, *G* divided by total number of pixels in a union of segmented results and GT as defined in equation 3.5 [112].

$$IoU = J(P,G) = \frac{|P \cap G|}{|P \cup G|} \tag{3.5}$$

The IoU has a range of 0 to 1, where lower values correspond to better segmentation results in comparison to the GT [22].

F-measure is known as harmonic mean of the precision and the recall of segmented buildings regions as presented in equation 3.6 [129, 130].

$$F - measure = \frac{2(precision\ x\ recall)}{precision + recall} \tag{3.6}$$

F-measure value of a 1 indicates good delineation that accurately matches GT, while a value of 0 indicates poor segmentation that does not align accurately in relation to GT [28].

Moreover, pixel accuracy (PA) represents the proportion of correctly segmented pixels relative to the total number of pixels in the ground truth (GT). It encompasses true positives *(TP)*, true negatives *(TN)*, false positives *(FP)*, and the false negatives *(FN)* during the segmentation process as shown in equation 3.7 [95].

$$PA = \frac{TP+TN}{TP+TN+FP+FN} \tag{3.7}$$

The PA values of 1 refers to segmentation results that are well aligned with the GT while values of 0 refers to poor delineation results in comparison to GT.

3.1.5.3 Performance Evaluation of the Proposed Region Merging Approach

The evaluation metrics used to evaluate the final segmentation results of the proposed region merging approach are the adapted rand error (ARE), and

the variation of information (VOI) as stated in Section 3.1.5.1 as equations 3.1 and 3.4, respectively. The other evaluation metrics include F-measure, precision, and recall as referred in Section 3.1.5.1 as equations 3.2 and 3.3, and equation 3.6 in Section 3.1.5.2, respectively. Additionally, the adapted rand index (ARI) [131] was also used to evaluate the performance of the final segmentation results. ARI is the measure of correspondence between the pixels of the segmented results, $Seg(x)$ and the GT, $GT(x)$ [22].

$$ARI = \frac{\sum GT(x).Seg(x)}{\sum GT(x)} \tag{3.8}$$

The range of the metric ARI is between 0 and 1 with lower values indicating better segmentation output in comparison to GT [124]. Moreover, these results are presented and discussed in Chapter 5 Section 5.4.

3.1.5.4 Performance Comparison of the Proposed Region Merging Approach with Existing Works

The final segmentation results of the proposed region merging approach were compared with two similar existing works [28, 36]. The evaluation metrics used for this comparison are F-measure, precision, and recall as stated in Section 3.1.5.B as from equations 3.4 to 3.6, respectively.

The reference in [132] introduced an evaluation metric known as goodness of segmentation, G_s as presented in equation 3.9. G_s is defined as the ratio of the correctly segmented pixels, $A_{overlap}$ to the GT, A_{refer} with A_{diff} representing the difference between correctly segmented pixels and misclassified pixels. The G_s is only measuring for the specific building region of interest (ROI) as compared to considering other objects or regions [132].

$$G_s = \frac{A_{overlap}}{A_{refer} . e^{\frac{A_{diff}}{A_{refer}}}} \quad (3.9)$$

The $A_{overlap}$ and A_{diff} are presented in equations 3.10 and 3.11, respectively.

$$A_{overlap} = \sum A_i^{overlap} \quad (3.10)$$

$$A_{diff} = \sum A_i^{diff} \quad (3.11)$$

The value 1 of G_s indicates good segmentation results in comparison to GT. In contrast, a G_s value of 0 indicates that segmentation results are poorly aligned with the GT [22, 28]. These experimental results and discussions are presented in Chapter 5 Section 5.5.

3.1.6 Documentation

The experimental results obtained from region-based segmentation algorithms and CNN-based deep learning models will be documented in detail for the thesis write-up. Additionally, an in-depth analysis of the final segmentation evaluation results will also be included in the thesis.

3.2 Summary

This chapter presented the methodology of the proposed region merging approach for delineating buildings as ROI in remote sensing images. Additionally, this chapter outlined the experimental setup and performance evaluation of this approach including the evaluation metrics used. Furthermore, Chapter 4 covers the region-based segmentation algorithms, CNN-based deep learning models, and feature extraction processes. Chapter 5 then focuses on the derivation of the merging criterion (MC) to perform region merging for segmenting the ROI in remote sensing images.

CHAPTER 4

GENERATION OF OVER SEGMENTED REGIONS AND FEATURES EXTRACTION FROM THE FEATURE MAP GENERATED BY CONVOLUTIONAL NEURAL NETWORK (CNN)-BASED DEEP LEARNING MODEL

This chapter focuses on experiments related to selecting the appropriate region-based segmentation algorithm for generating over segmented regions, followed by generating the feature map from a suitable convolutional neural network (CNN)-based deep learning model. These generated feature map by the identified CNN-based deep learning model consists of building regions of interest (ROIs) are then utilised to extract prominent features which are colour, texture, shape, and edge information in order to derive merging criterion (MC) to perform region merging.

4.1 Experimental Flow

Figure 4.1 shows the flowchart of experimental flow for the generation of the over segmented regions, feature map from the convolutional neural network (CNN)-based deep learning model, and features extraction from generated feature map, respectively. Further details on these are discussed in the following subsections:

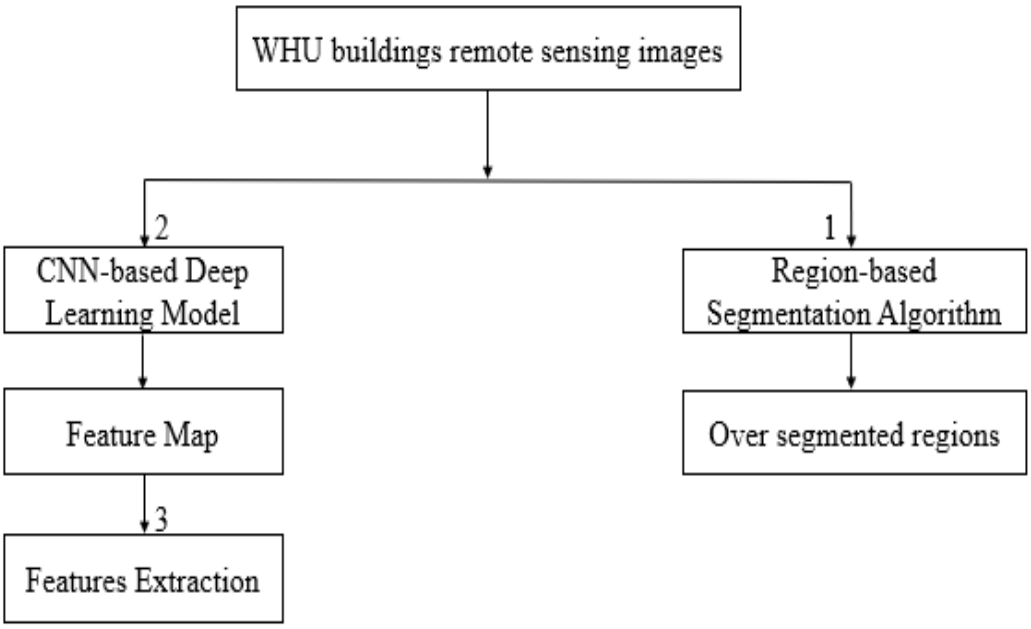


Figure 4.1: Flowchart of experimental flow

4.1.1 WHU Buildings Remote Sensing Images Dataset

The experiments were performed by utilising the publicly benchmarked remote sensing images named as WHU buildings dataset [37]. The dataset [37] has a total of 8189 images and the corresponding ground truths (GTs) with the 512 × 512 dimensions. However, there are images that only displayed a small number of buildings or contains no building. In this research, images and GTs that contains limited building regions have been excluded. Hence, a total of 1550 images and their corresponding GTs of 512 × 512 dimensions are selected and utilised for experiments in this research.

4.1.2 Over Segmented Regions Generation using Region-based Segmentation Algorithm

Recently, the researchers in [52] [53] have compared and evaluated the efficacy of four region-based segmentation algorithms in a process of segmenting medical [52] and natural [53] images simultaneously. Moreover,

four region-based segmentation algorithms includes Felzenszwalb and Huttenlocher (FH) [9], the Compact Watershed (CW) [10], the Quick Shift (QS) [13], and the simple linear iterative clustering (SLIC) [14]. Additionally, the accuracy of an algorithm is based on a complexity of ROI in selected image [15]. The complexity of an ROI refers to factors like the shape, size, texture, and boundaries of the region, as well as how well it contrasts with its surrounding areas. These characteristics directly affect how challenging it is for an algorithm to accurately segment the region. For instance, in medical images, ROIs may be highly irregular in shape which may perform segmentation difficult. In contrast, in natural images, the ROIs could involve large, distinct regions with clearer boundaries, which might be easier to segment for some region-based segmentation algorithm. It means that each region-based segmentation algorithm has its own effectiveness on either medical, natural or remote sensing images. Based on this, the reason for choosing the mentioned four region-based segmentation algorithms is to select a suitable one for this research on remote sensing images of WHU buildings dataset. Moreover, experiments are conducted among these algorithms for finding out which algorithm generate over segmented regions which adheres well to the object boundaries. Further details on the parameter utilised for region-based segmentation algorithms and their experimental results are explained in Sections 4.1.2.1 and 4.1.2.2, respectively.

4.1.2.1 Parameters of Region-based Segmentation Algorithm

In this research work, the parameters defined for region-based segmentation algorithms. In addition, parameters need to be defined for the FH

algorithm are the scale, sigma, and minimum size. Moreover, the parameters for CW are markers and compactness. The Sobel filter [133] is utilised to execute CW, it is because watershed algorithm always requires the gradient image as an input image to conduct segmentation [134]. Meanwhile, kernel size needs to be defined as the parameter for QS. As for the SLIC, k parameter has to be defined [123].

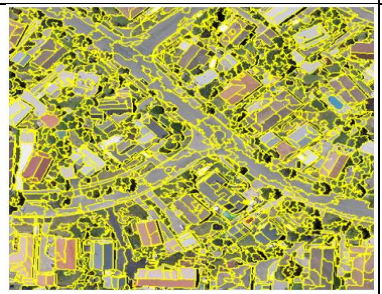
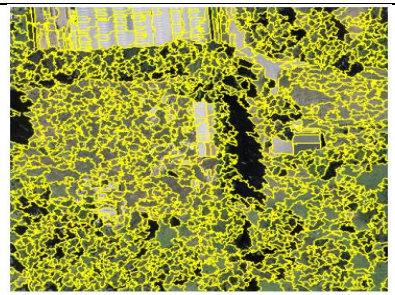
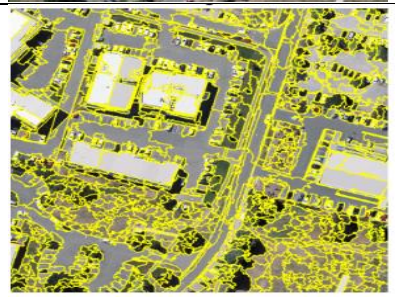
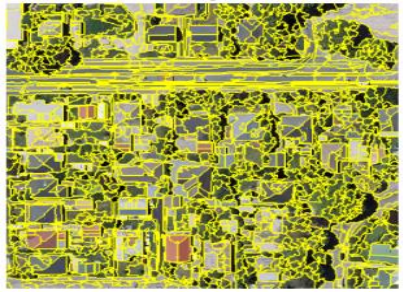
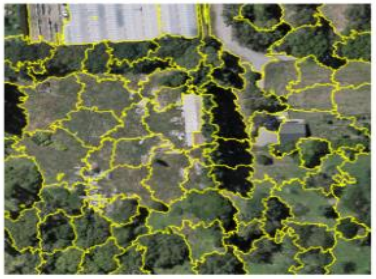
Moreover, for FH, the scale parameter is considered as 50, sigma as 0.5, and the minimum size as 50 in previous research work [55]. Meanwhile, in CW, marker and compactness set as 100 [135] and 0.001 [123], respectively. The value 3 is used for the parameter kernel size in QS [55]. In addition, in SLIC algorithm, parameter value of k is taken as 100 [123].

4.1.2.2 Experimental Results and Discussion for region-based segmentation algorithm

Figure 4.2 demonstrates the over segmentation (OS) results and Table 4.1 presents the performance evaluation results of the four chosen region-based segmentation algorithms which are FH, CW, QS and SLIC on WHU buildings dataset [37]. Furthermore, two evaluation metrics are utilised for performance evaluation for selecting a suitable region-based segmentation algorithm. These evaluation metrics are the adapted rand error (ARE) and the variation of information (VOI) formulas are stated in Chapter 3 Section 3.1.5 and subsection 3.1.5.1 as equations 3.1 and 3.4, respectively.

From Figure 4.2 (left to right), in third row the FH has delineated images into 1262, 1237, 970 and 1302 regions. Moreover, CW (fourth row) generated

76 segments of the regions, respectively. Additionally, remote sensing images were over segmented into total of 3054, 2733, 1982 and 2595 regions by QS (fifth row). Meanwhile, SLIC (last row) generated 57, 40, 64 and 56 over segmented regions, respectively.

Input Image No.	101	829	1027	1781
Input Image				
FH				
CW				

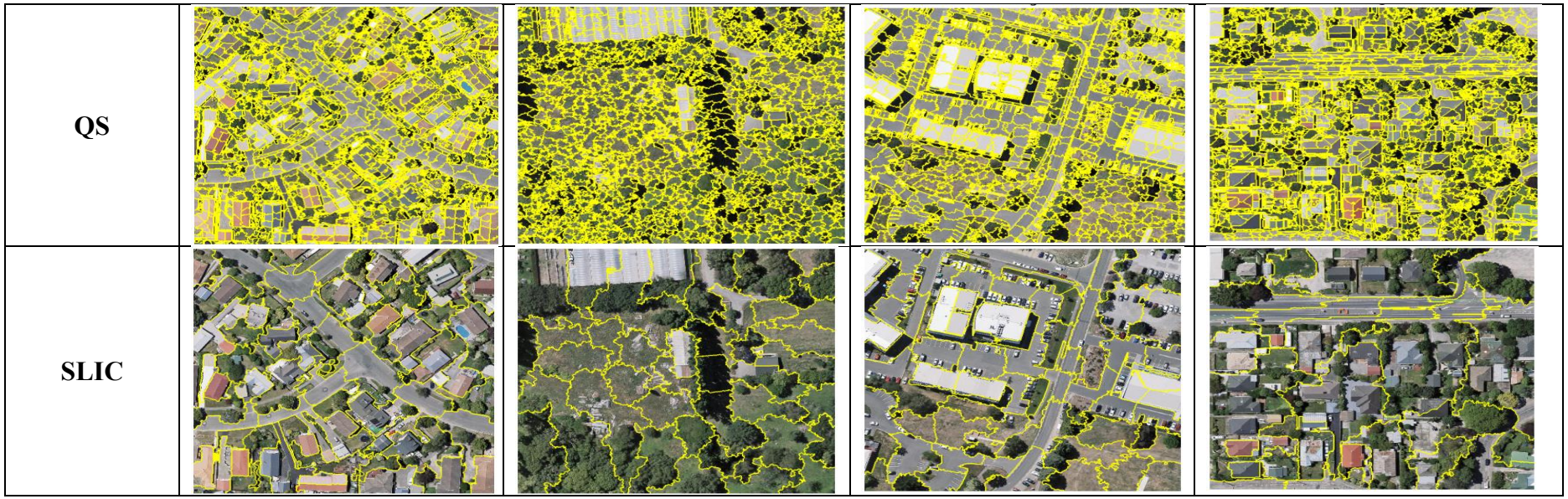


Figure 4.2: Over segmentation results of FH [9], CW [10], QS [13], and SLIC [14] on WHU buildings dataset [37]

Table 4.1 presents the performance evaluations of implemented four region-based segmentation algorithms in delineating images from WHU buildings dataset [37]. Moreover, SLIC has achieved average ARE of 0.2426. While, the CW has obtained an average ARE of 0.1811. Furthermore, average VOI of CW is 5.2066. Additionally, SLIC achieved better results for average VOI than the CW. In comparison to SLIC and the CW, FH obtained better results only than the QS. Hence, the QS algorithm ranked last as it has highest results of average ARE and VOI in delineating the remote sensing images demonstrating poor results.

Table 4.1: Performance comparison of FH [9], CW [10], QS [13], and SLIC [14] algorithms on 1550 images from WHU buildings dataset [37]

Algorithms	Average Adapted Rand Error (ARE)	Average Variation of Information (VOI)
FH	0.6714	7.9363
CW	**0.1811**	5.2066
QS	0.7597	8.3387
SLIC	0.2426	**4.9297**

Though severe over segmentation (OS) was observed in the images, the results still showed acceptable segmentation performance across all four region-based segmentation algorithms. Moreover, the performance of these experimented region-based segmentation algorithms are majorly relies on the selection of parameter(s) values which are scale for the FH algorithm, compactness for the CW, kernel size for the QS and k for the SLIC. In previous work [135], the authors tested various parameter(s) values for the above-mentioned algorithms to determine the

optimal ones. Based on these results, this research has selected the parameter values that showed the best performance in [135]. The authors in [135] stated that, parameter values has great impact on the segmentation output of algorithm. Therefore, this can lead to severe under or over segmentation (OS) in the final results if the parameter values are not correctly chosen [135]. Meanwhile, to be more specific, Figure 4.2 demonstrate that SLIC adheres to image object boundaries well as compared to other three region-based segmentation algorithms.

Table 4.1 shows that SLIC performed better as compared to other three region-based segmentation algorithms by achieving average VOI of 4.9297. This algorithm is capable of making almost optimal use of memory when segmenting high resolution images as opposed to generating over segmented regions that obey image object boundaries. The SLIC over segmented regions are regular and uniform in shape [64]. On the other hand, CW also produced images over segmented area, which only came up to the images objects boundaries. The CW achieves average VOI of 5.2066 than SLIC indicating more variation of pixels loss. It is because the CW considers the high-contrast pixels and ignored some of the true but relatively low-contrast pixels [134]. While for the ARE, CW outperform SLIC with lower average ARE 0.1811 and also lower average VOI 5.2066 than QS and FH. This is mainly due to the compactness and marker parameters in the CW algorithm [10], because the compactness influences how non-marker pixels are assigned to object regions and the marker parameters changes other smaller regions to zero. The FH and QS algorithms do not adhere to image boundaries too well and also have higher average ARE and VOI values. From [53] and [136], the FH and

QS algorithms has no compactness parameter being defined to control the generation of the over segmented regions. Based on above experimental results, SLIC was found to be the effective due to its ability to produce regular and uniform over segmented regions. Therefore, SLIC will be utilised for further segmentation tasks.

4.1.3 Generation of Feature Map Using CNN-based Deep Learning Model

In this research, the CNN-based deep learning models are utilised to generate the feature map. The chosen CNN-based deep learning models are used to generate feature map in remote sensing images because they have been effectively utilised to delineate ROIs in various remote sensing image datasets in the existing research works [37, 100, 102, 111, 113, 115, 117]. In this research the experiments are conducted among the chosen seven models to select the best model generating accurate buildings regions feature map. Further details on the construction of CNN-based deep learning model along with hyperparameter tuning and experimental results are explained in Sections 4.1.3.2 and 4.1.3.3, respectively. The implemented CNN-based deep learning models are compared by employing IoU, F-measure, precision, recall, and pixels accuracy (PA) metrics formulas as stated in Chapter 3 Section 3.1.5.2 from equations 3.3 to 3.7, respectively. Hence, the comparison among CNN-based deep learning models will assists in finding the suitable model in order to generate the feature map of buildings as ROI.

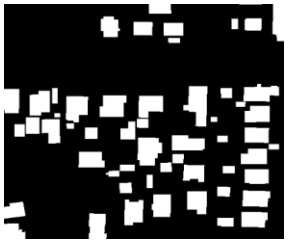
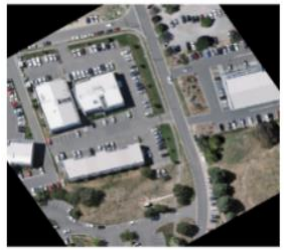
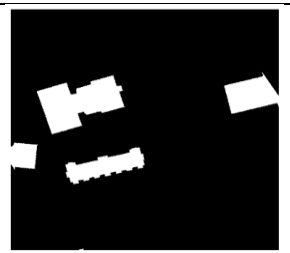
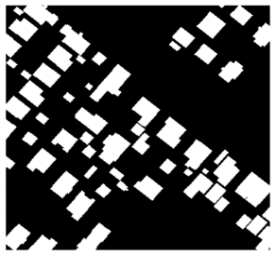
4.1.3.1 Preprocessing and Data Augmentation of WHU buildings dataset for CNN-based Deep Learning Model

The WHU buildings dataset of 1550 images and corresponding ground truths (GTs), a total of 1395 images and GTs are utilised for training the CNN-based deep learning models. On the other hand, the testing dataset consist of 155 images with their corresponding GTs. Based on this, the initial splits of the WHU buildings dataset of 1550 images and GTs into training and testing datasets becomes a ratio of 90:10. According to the authors in [137], the utilisation of data augmentation process for training CNN-based deep learning model is of utmost importance and plays a substantial role in generating an efficient result. The data augmentation demonstrates images to different changes and positions. It also increases diversity and size of training dataset. This leads to improve the model generalisation ability and performance [95]. Thus, the training images 1395 are augmented by applying random rotation of 45 degree, and operations of horizontal and vertical flips as recommended in previous work [138].

During the data augmentation process, random rotation of 45 degrees results in images that exhibit empty spaces, as illustrated in Figure 4.3, third row. These empty spaces are subsequently filled by replacing them with the nearest pixels to ensure a complete image representation. After augmentation, the total images and corresponding GTs of training dataset 1395 resulting in generating 5580 images and corresponding GTs, which are then split into final training dataset 4464, and 1116 validation dataset of images along with the GTs. Moreover, the validation

dataset refers to validating the model's performance during training simultaneously. The testing dataset of 155 was not augmented, it is because to keep it as images unseen for CNN-based deep learning models [95]. Based on this, the seven CNN-based deep learning models are trained and validated on 5580 generated images and GTs and tested on 155 images. Hence, WHU buildings dataset splits into the training, the validation, and the testing datasets in a ratio of the 70:20:10 [127], respectively.

As demonstrated in Figure 4.3, the first and second row shows the original images and their corresponding GTs, respectively. During the data augmentation process, random rotation of images and GTs generated with black regions are presented in Figure 4.3 third and fourth row, respectively. Resultantly, third row in Figure 4.3 consists of Random Rotated Image (empty spaces after rotation) and fourth row containing Random Rotated GTs (empty spaces after rotation) have been excluded for training the CNN-based deep learning models. However, they are utilised after filling the black or empty regions with nearest pixels to generate completely new image and GT as demonstrated in Figure 4.3 fifth and sixth row, respectively. Hence, images and GTs from the fifth and sixth row (Random Rotated Images and filled empty spaces) are utilised for training CNN-based deep learning models, respectively. Furthermore, Figure 4.4 shows the horizontal and vertical flips demonstrated from seventh till tenth row images and GTs. These are used for training CNN-based deep learning models.

Input Image	101	827	1027	1781
Original Image				
GT				
Random Rotated Image (empty spaces after rotation)				
Random Rotated GT (empty spaces after rotation)				

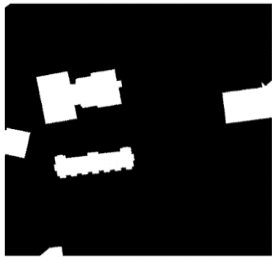
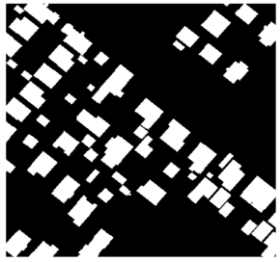
Random Rotated Image (filled empty spaces after rotation)				
Random Rotated GT (filled empty spaces after rotation)				

Figure 4.3: Random rotation of WHU buildings dataset images and GTs

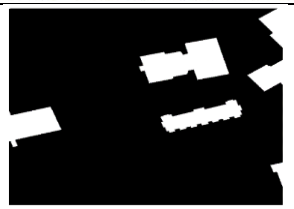
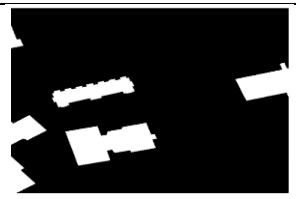
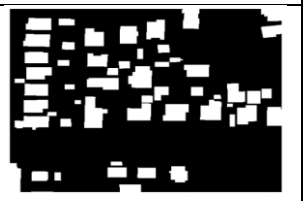
Horizontal Flip of Original Image				
Horizontal Flip of Original GT				
Vertical Flip of Original Image				
Vertical Flip of Original GT				

Figure 4.4: Horizontal and Vertical Flips of WHU buildings dataset images and GTs

4.1.3.2 Construction of CNN-based Deep Learning Model and Hyperparameter Tuning

This research chose CNN-based deep learning models from the literature, specifically focusing on U-Net and modified or enhancements versions based on standard U-Net. All these standard CNN-based deep learning models consist of encoder-decoder parts. The encoder part is responsible for extracting the features of the input image and progressively reducing the spatial dimensions of the image. While the decoder part is responsible for reconstructing the features of the image obtained from the encoder part [95]. The implementation of seven chosen CNN-based deep learning models includes the U-Net [37], V-Net [111], SegNet [100], SegU-Net [113], ResU-Net [102], MultiResU-Net [115], and AttentionU-Net [117].

The hyperparameters for CNN-based deep learning model are chosen from previous research work of [37] and hyperparameters are further validated through performing experiments. The experiments for hyperparameter tuning starts by training U-Net model with two optimisers, and different batch sizes on WHU buildings dataset referred in [37] for 50 epochs. This research utilises the specified hyperparameters from previous research work [37], which were implemented in the U-Net. Consequently, this serves as baseline model for the performance comparison with other models in this research work.

The chosen seven CNN-based deep learning models have been trained from scratch. The training process of the seven chosen models starts by taking the

original size of input images from WHU buildings dataset having 512 × 512 dimensions. As stated in Section 4.1.3.1, the dataset splits into ratio of 70:20:10 as training, validation, and testing, respectively. The training dataset consists of 4464 images and corresponding GTs for training the CNN-based deep learning models and 1116 images along with the GTs as validation dataset for validating the models performance during training. Moreover, the test dataset consists of 155 images and corresponding GTs for final performance evaluation of all models. Finally, the results and discussion are explained in Section 4.1.3.3 which contains Tables 4.2 to 4.5. These tables present the results of seven CNN-based deep learning models by employing evaluation metrics of the intersection over union (IoU), F-measure, precision, recall, and pixel accuracy (PA).

4.1.3.3 Experimental Results and Discussion on Construction of CNN-based Deep Learning Model and Hyperparameter Tuning

Table 4.2 presents the hyperparameters utilisation and their results for the U-Net model. Based on these results, the suitable hyperparameters have been selected for training all the chosen seven CNN-based deep learning models in this research. Furthermore, same set of the hyperparameters values are considered among all seven CNN-based deep learning models for the fair comparison of performance evaluation.

Table 4.2: Summary of hyperparameter tuning and training results for U-Net model using Adam [139] and stochastic gradient descent (SGD) [140] Optimisers on WHU buildings dataset for 50 epochs with different batch sizes

Optimiser	Batch size	Average IoU	Average F-measure	Average precision	Average recall	Average Pixel Accuracy (PA)
Adam [139]	6	**0.9353**	**0.9113**	0.9196	**0.9121**	**0.9650**
	8	0.9268	0.8807	0.8752	0.9283	0.9577
	16	0.9296	0.9029	0.9131	0.9045	0.9621
SGD [140]	6	0.9035	0.7860	0.9055	0.8214	0.9439
	8	0.8884	0.7677	**0.9206**	0.7642	0.9356
	16	0.9103	0.8016	0.8784	0.8714	0.9481

The hyperparameter results as presented in the Table 4.2, the U-Net model training results achieved higher average intersection over union (IoU) of 0.9353 by employing Adam optimiser as compared to stochastic gradient descent (SGD) optimiser of 0.9035 for batch size 6. The other two batch sizes of 8 and 16 have also comparable average IoU results, however lower than Adam with batch size 6. Hence, the Adam optimiser and batch size 6, along with ReLU as activation function, binary cross-entropy (BCE) as loss function, learning rate 0.0001, final layer classifier as sigmoid has shown reasonable results. Additionally, an early stopping and regularisation technique has been implemented to prevent the overfitting [141].

Table 4.3 and 4.4 presents performance evaluation results of the U-Net [37], V-Net [111], SegNet [100], SegU-Net [113], ResU-Net [102], MultiResU-Net [115], and AttentionU-Net from [117] in generating feature map of buildings as ROI from training and validation dataset [24], respectively. Table 4.5 shows the test dataset from the WHU buildings dataset that the models had not seen during training and validation, along with an analysis of the best performing model.

Table 4.3: Average IoU, average F-measure, average precision, average recall, average pixel accuracy (PA) and total epochs of seven CNN-based deep learning models for WHU buildings remote sensing training images dataset

Models	Total epochs	IoU	F-measure	precision	recall	PA
U-Net [37]	100	**0.9939**	**0.9873**	**0.9923**	**0.9904**	**0.9963**
V-Net [111]	24	0.9609	0.9185	0.9482	0.9415	0.9768
SegNet [100]	33	0.9347	0.8582	0.9309	0.8755	0.9614
SegU-Net [113]	85	0.8568	0.6975	0.8827	0.7009	0.9170
ResU-Net [102]	38	0.9819	0.9627	0.9749	0.9736	0.9891
MultiResU-Net [115]	54	0.9733	0.9458	0.9623	0.9621	0.9840
AttentionU-Net [117]	62	0.9780	0.9475	0.9690	0.9690	0.9869

Table 4.4: Average IoU, average F-measure, average precision, average recall, average pixel accuracy (PA) and total epochs of seven CNN-based deep learning models for WHU buildings validation images dataset

Models	Total epochs	IoU	F-measure	precision	recall	PA
U-Net	100	0.9298	**0.9107**	0.9208	0.9043	0.9631
V-Net	24	0.9360	0.8869	0.9144	0.9079	0.9624
SegNet	33	0.9216	0.8428	0.9205	0.8537	0.9547
SegU-Net	85	0.8612	0.7048	0.8906	0.7092	0.9197
ResU-Net	38	0.9351	0.9036	0.9219	0.9032	0.9632
MultiResU-Net	54	0.9338	0.8964	0.9092	0.9114	0.9618
AttentionU-Net	62	**0.9436**	0.9046	**0.9277**	**0.9185**	**0.9675**

Tables 4.3 and 4.4 present the training and validation results. In the training dataset, the U-Net model achieved the highest average IoU of 0.9939 over 100 epochs, outperforming other models. This is attributed to its efficient feature map processing from the encoder to decoder, which enables accurate generation of building region feature maps as regions of interest (ROI) in the training dataset. [142]. In the validation dataset presented in Table 4.4, U-Net achieved an average IoU of 0.9298, higher than SegNet and SegU-Net while lower than other models which are V-Net, ResU-Net, MultiResU-Net, and AttentionU-Net. It is due to its simpler architecture having fewer convolutional layers which may have led to poorer feature map generation during validation.

The SegU-Net model achieved the lowest average IoU of 0.8568 in training and 0.8612 in validation. Its deep structure, combining SegNet and U-Net, affects feature learning, particularly at deeper layers, leading to poor results [112]. SegNet achieved the lowest IoU of 0.9347 in training and 0.9216 in validation, outperforming only SegU-Net. It is because of the use of pooling indices for upsampling may miss fine details, unlike U-Net's skip connections [112]. AttentionU-Net achieved an average IoU of 0.9780 in training and 0.9436 in validation. It is due to the attention gates (AGs) enhanced feature map generation by focusing on prominent features, outperforming other models. [104].

V-Net, ResU-Net, and MultiResU-Net achieved satisfactory results with IoU of 0.9609 for V-Net and 0.9819 for ResU-Net in training, and 0.9360 and

0.9351 in validation, respectively. The V-Net, similar to U-Net, includes additional convolutional layers [111]. The ResU-Net combines U-Net with residual blocks, enhancing feature extraction for efficient ROI feature map generation [143]. MultiResU-Net achieved IoU of 0.9733 in training and 0.9338 in validation time, which are lower than ResU-Net and it is due to its multiple residual learning blocks [143]. The structure of ResU-Net is much deeper than those of deep CNN-based models, which may also affect the performance of ResU-Net. Table 4.4 shows AttentionU-Net achieved the highest average IoU of 0.9469, while SegU-Net had the lowest of 0.8726.

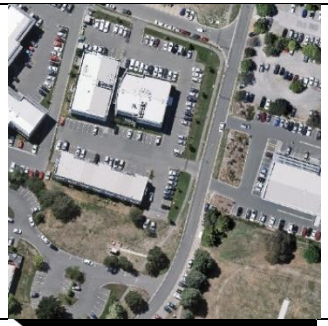
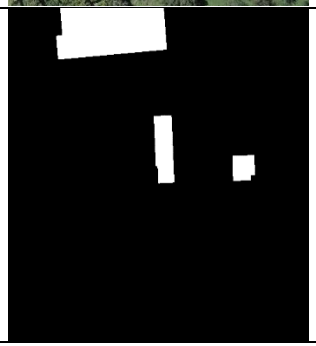
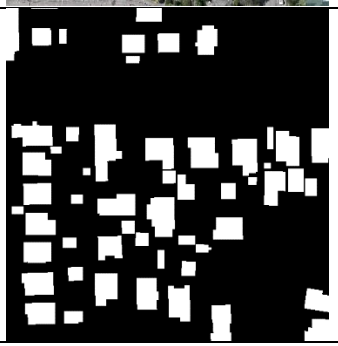
Table 4.5: Average IoU, average F-measure, average precision, average recall, average pixel accuracy (PA) and total epochs of seven CNN-based deep learning models for WHU buildings testing images dataset

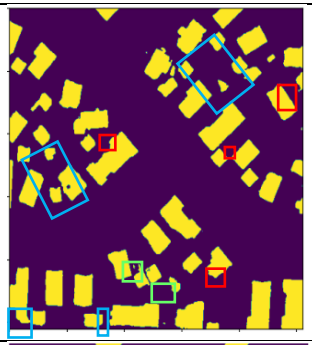
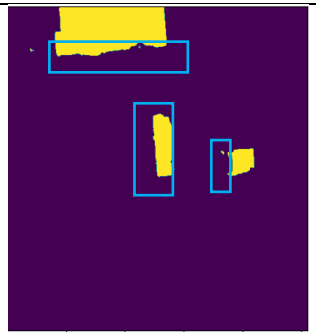
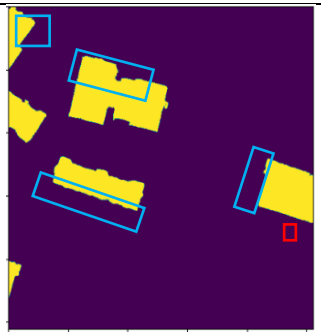
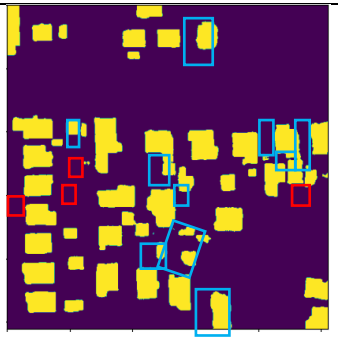
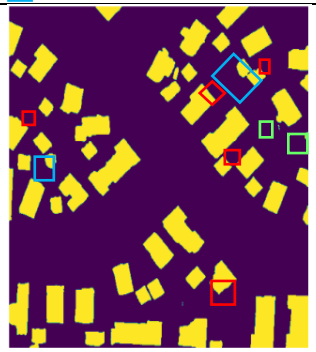
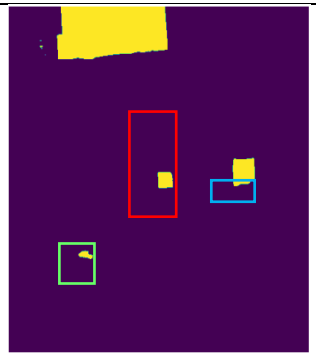
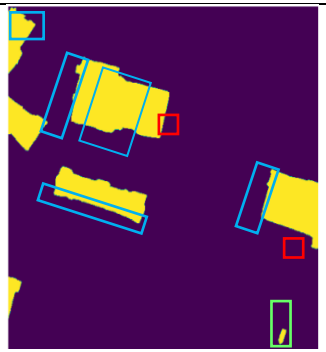
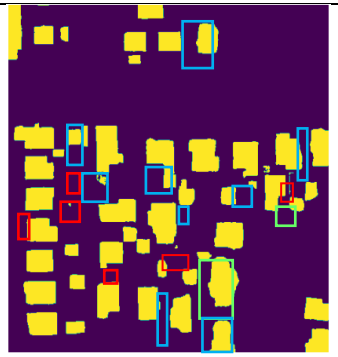
Models	Total epochs	IoU	F-measure	precision	recall	PA
U-Net	100	0.9339	**0.9150**	0.9218	0.9119	0.9654
V-Net	24	0.9376	0.8922	0.9169	0.9146	0.9638
SegNet	33	0.9291	0.8558	0.9314	0.8713	0.9592
SegU-Net	85	0.8726	0.7239	0.9061	0.7383	0.9274
ResU-Net	38	0.9396	0.9124	0.9268	0.9155	0.9662
MultiResU-Net	54	0.9388	0.9052	0.9141	0.9240	0.9650
AttentionU-Net	62	**0.9469**	0.9119	**0.9309**	**0.9293**	**0.9698**

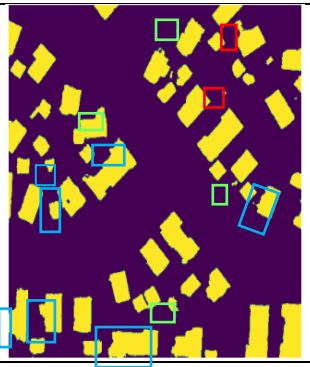
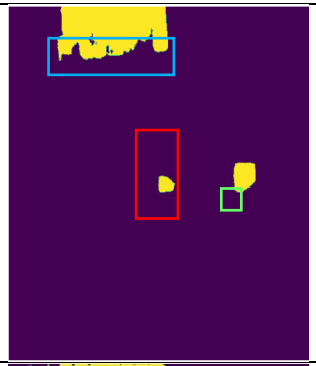
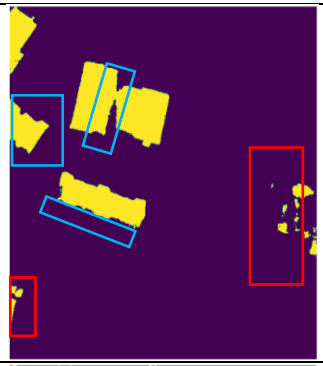
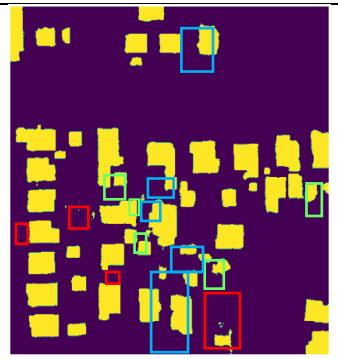
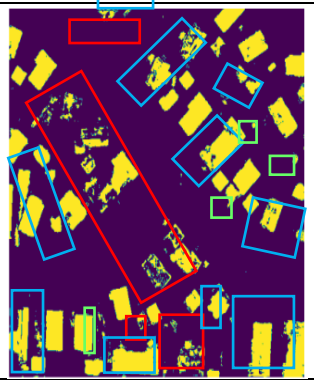
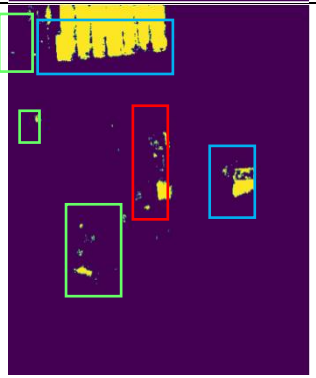
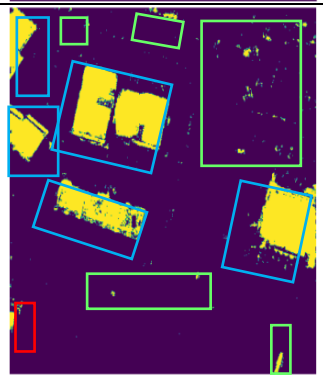
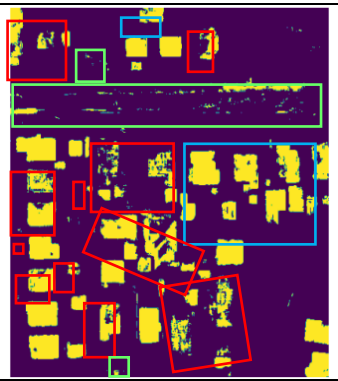
The CNN-based models achieved higher accuracy in generating building ROI feature maps in training presented in Table 4.3 compared to validation time results shown in Table 4.4, indicating effective learning and good generalisation to new images, with minimal overfitting. [37]. Additionally, the difference between Table 4.3 and Table 4.5 indicates that the models are less likely to be affected by overfitting on the test images dataset. This demonstrates that the models are able to generalise accurately when it comes to delineating ROI in new images.

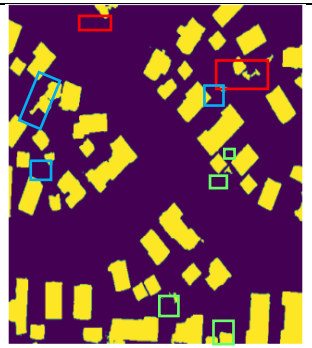
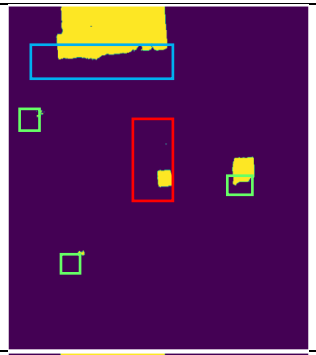
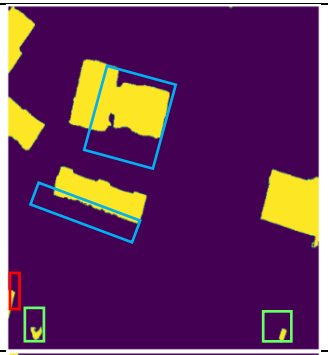
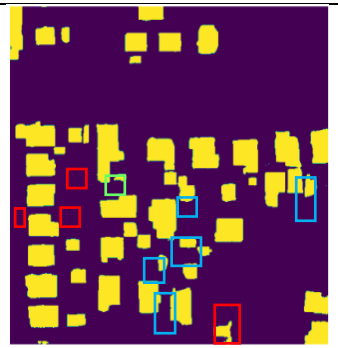
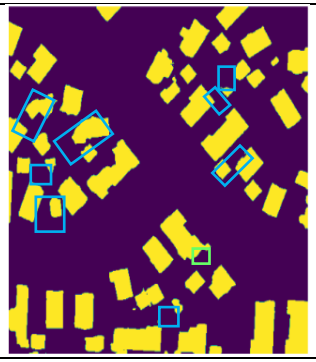
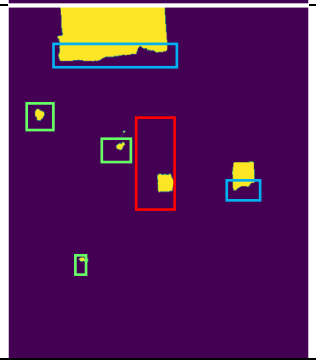
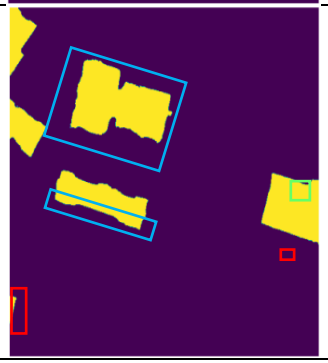
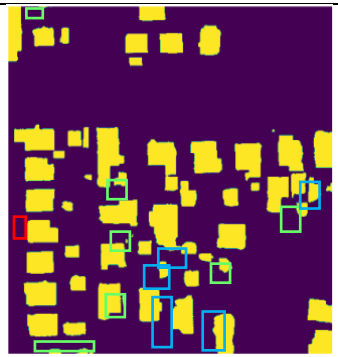
Figure 4.5 shows the U-Net, V-Net, SegNet, SegU-Net, ResU-Net, MultiResU-Net, and AttentionU-Net results on the WHU buildings test dataset, with blue, red, and green boxes highlighting building ROIs, including inaccurate boundaries or misclassified pixels. The blue colour boxes representing segmented regions with inaccurate boundary, the green colour boxes mean incorrect regions segmented as buildings, and red colour boxes means under segmented regions. Figure 4.5 shows AttentionU-Net accurately generates building ROI feature maps with diverse colours, sizes, shapes, and textures compared to other models. Figure 4.5 shows AttentionU-Net accurately generates building ROI feature maps, despite minor boundary discrepancies. It is because of the use of attention gates (AGs) effectively captures fine feature details in the WHU buildings dataset [37]. In comparison, SegNet and SegU-Net fail to generate accurate building ROI maps compared to AttentionU-Net, due to their structure and the inclusion of attention gates in AttentionU-Net. Figure 4.5 shows that for image ID 829, none of the models generated a building ROI feature map (blue box), and for image ID 1027,

SegNet failed to generate a feature map for the building regions indicated within the red box. SegNet's poor performance is due to its reliance on pooling indices for upsampling, which cannot capture fine details like shape, size, and texture of building ROI regions [108]. SegU-Net shows poor visual results for many images, mainly due to its complex structure [112]. The U-Net, V-Net, ResU-Net, and MultiResU-Net generate satisfactory building ROI maps, but their results are poorer than AttentionU-Net as shown in Figure 4.5. Since the U-Net model contains lower number of convolutional layers, resulting in lower performance as compared to the AttentionU-Net model. The unavailability of an attention mechanism in V-Net, ResU-Net, and MultiResU-Net makes it harder for these models to generate diverse building features. Therefore, AttentionU-Net is considered the best model based on experimental results in this research work.

Input Image No.	101	829	1027	1781
Input Image				
GT				

U-Net				
V-Net				

SegNet				
SegU-Net				

ResU-Net				
Multi ResU-Net				

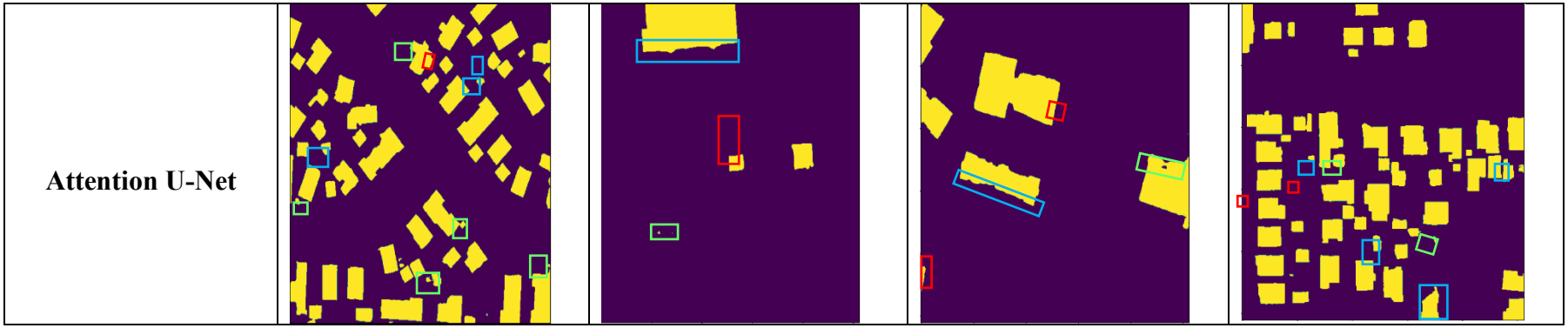


Figure 4.5: Visualisation results of seven CNN-based deep learning models in generation of buildings ROI feature map for test image IDs 101, 829, 1027, and 1781 from WHU buildings dataset in comparison to ground truths (GTs), respectively

In Figure 4.6, AttentionU-Net's classified building ROI (second row) matches the actual building ROI indices (first row), with pixel values ranging from 0 to 1. The threshold value of 0.5 is used to retrieve the actual building ROI from the image based on the AttentionU-Net generated feature map of building ROI [97]. AttentionU-Net uses a sigmoid classifier in the last layer with a 0.5 threshold to output the building ROI feature map [28]. The process of generating feature map of buildings ROI from actual image building follows the previous work [28]. This process first verifies the indexes on x and y coordinates of feature map generated by AttentionU-Net with the pixels of actual building ROI in image by using the built-in capabilities of python libraries [144, 145]. As a result, the matched pixels are considered as buildings regions. The rest of the pixels values which are unmatched on indexes of x and y are taken as 0, resultantly buildings feature map generated having black background as shown in Figure 4.6 third/last row. Finally, the building feature map generated by AttentionU-Net (third row of Figure 4.6) is used to extract colour, texture, shape, and edge features as discussed in Section 4.1.4.

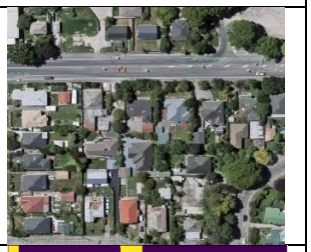
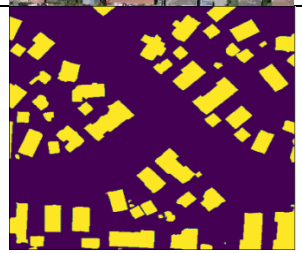
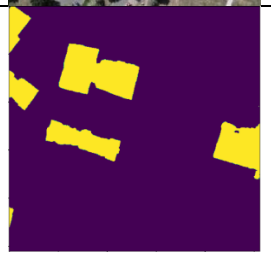
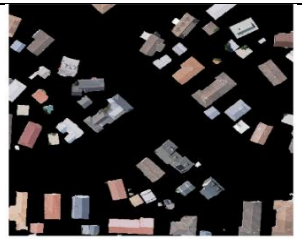
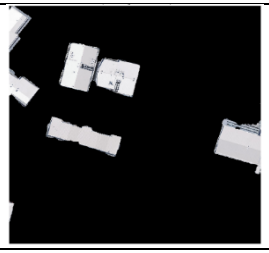
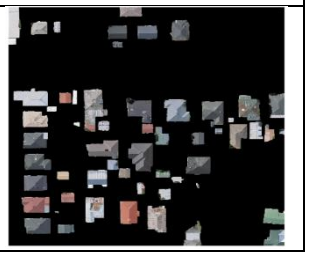
Input Image No.	101	829	1027	1781
Input Image				
AttentionU-Net				
Buildings Feature Map Generated				

Figure 4.6: Building ROI feature map generated through AttentionU-Net

4.1.4 Features Extraction from the Feature Map

The features extraction experiments (represented as step 3 in Figure 4.1) have been conducted on the feature map obtained from the best performing AttentionU-Net CNN-based deep learning model (represented as step 2 in Figure 4.1). Based on this feature map, colour, texture, shape, and edge features will be extracted. These four features are prominent features that are commonly used in most of previous works [25, 28] for region merging approaches. The four features will be computed after the generation of the feature map by the AttentionU-Net model. The colour features will be extracted from the colour channels (such as RGB and HSV) of the image. By analysing the colour distribution within each region of interest (ROI), statistical measures like mean, of the colour channels will be derived. Texture features are computed by analysing the spatial patterns within the feature map, using techniques such as Local Binary Patterns (LBP), or Gabor filters, which quantify the surface characteristics of each region. Shape features will be derived from building ROIs. Lastly, edge features will be computed through edge detection algorithms like Sobel, or Canny filters applied to the feature map. Further details on these features extraction are described in the subsections from 4.1.4.1 to 4.1.4.4.

4.1.4.1 Colour Feature Extraction

In this research, two different colour spaces which are Red, Green, Blue (RGB) and Hue, Saturation, Value (HSV) are used for colour features extraction to achieve the best colour feature space, especially for WHU buildings dataset [37]. The colour information can be extracted through many

ways such as colour histogram, colour moment, and average RGB [79]. However, one efficient way is colour moment which refers to express the distribution of pixel intensity values in an image [82]. The concept of colour moment was first developed by Stricker and Orengo [146]. The colour moment is used as a descriptor and it actually defines the relationship of pixel with its surrounding neighbours.

The colour moments cannot be computed for a single pixel where it requires a number of pixels to compute it [82]. Colour moments can be employed for extracting colour feature from colour space be it RGB or HSV when considering image segmentation approaches [147]. In addition, colour can be measured by following attributes and there are colour moments, specifically the mean, standard deviation, skewness, and kurtosis as shown in Figure 2.2.

As shown in Figure 4.6 third row as building feature map generated, experiments are conducted in order to extract colour feature. If an image is considered as discrete function *f (x, y)* with *x*=0, 1...M and y=0, 1....N, then the moment of order *(p + q)* is defined as equation 4.1.

$$Mpq = \sum_{x=0}^{M} \sum_{y=0}^{N} \varphi pq(x,y) f(x,y), p,q = 0,1,2,\ldots \qquad (4.1)$$

where Mpq is the momentum having moment weighting kernel demonstrated as φpq. This kernel is a function or a set of coefficients f that is applied to each pixel intensity value or a combination of values represented as (x, y) in the case of colour images to compute the colour moments of each channel having dimension represented as M and N in the colour space [148].

Colour moments are utilised to characterised the colour distribution within an image. The mean, or first moment, quantifies the average colour intensity across each channel [82]. It is calculated by summing the pixel intensity values for the entire channel and normalising by the total number of pixels. The mean for each individual channel red, green, and blue (RGB) has been calculated as follows in equation 4.2.

$$E_c = \frac{1}{MxN}\sum_{i=1}^{M}\sum_{j=1}^{N} P_{ij} \tag{4.2}$$

where E_c represents the mean intensity value for a specific colour channel *c*, *c* can be red (R), green (G), or blue (B). The total number of pixels in the image is determined by multiplying the number of rows *M* and columns *N*, ensuring that the mean is calculated over all pixels in the image. The term P_{ij} denotes the intensity of a pixel located at coordinates *(i, j)* in the respective colour channel. By summing the pixel values across all positions in the image and then dividing by the total number of pixels, this equation provides the average intensity for each colour channel, which helps in analyzing the overall colour features in the image. Thus, E_c represents the mean pixel intensity over all pixels in an image, and subscript l is representing the AttentionU-Net CNN-based deep learning model generated feature map.

The conversion from RGB to HSV colour space is based on the open-source image processing library *skimage.color* [149]. These two colour spaces are utilised in order to select the suitable colour space for feature extraction which are colour, texture, shape, and edge feature. Along with this, colour moments and the attributes such as mean, standard deviation, skewness, and

kurtosis are utilised from open-sourced image processing library [150-152]. The mean value is used of channels for comparing pixel distributions based on a threshold (having higher mean intensity of single channel) [153].

4.1.4.2 Texture Feature Extraction

Experiments are conducted on building feature map generated as shown in Figure 4.6 third row in order to extract texture feature. Furthermore, to extract texture features from the CNN-based deep learning model generated feature map, two texture extractor consisting of Gabor and Local Binary Pattern (LBP) filters have been utilised. These filters are used for finding the suitable descriptor for textures feature extraction in both colour spaces RGB and HSV, respectively. According to [82] both descriptors generate matrices, and afterwards, mean values from those matrices of texture feature can be utilised for further computation. Gabor and LBP have been utilised from an open-source image processing library which are *skimage.filters.gabor* [154] [155].

4.1.4.3 Shape Feature Extraction

In this research, for shape feature the area attribute has been considered and it refers to the total number of pixels in each building ROI. Since, the number of pixels would be the same for each building ROI in both the colour spaces RGB and HSV. Thus, in this case area attribute of shape feature is not further compared or evaluated.

4.1.4.4 Edge Feature Extraction

As presented in Figure 4.6 third row as building feature map generated, the Canny and Sobel edge detectors are implemented on the feature map. Moreover, two colour spaces RGB and HSV have been utilised for the experimentation of Canny and Sobel in order to identify the best performing edge detector for edge information extraction. Both the edge detectors are utilised from an open-sourced image processing library having Canny as [156] and for Sobel edge detector as [157].

4.2 Evaluation of Extracted Features from Generated Feature Map

In this research work, chi-squared (x^2) statistical test is utilised in terms of selecting the suitable feature in order to incorporate for merging criterion (MC) derivation. The chi-squared (x^2) statistical test is often used to find the most important and appropriate features by evaluating the relationship of independence between each feature in the feature map and the target variable [158]. In this research, target variable is the RGB or HSV channel value from feature map generated by AttentionU-Net CNN-based deep learning model. The chi-squared is a statistical test in order to ascertain if two categorical variables have a significant relationship [159]. The relationship is more significant and the feature is more relevant if the chi-squared value is higher. Lower p-values (probability values) denote more significant relationships, while the chi-squared test p-value represents the likelihood of finding the chi-squared statistic as defined in equation 4.3 [160].

$$(x_c)^2 = \sum \frac{(O_i - E_i)^2}{E_i} \quad (4.3)$$

where $(x_c)^2$ refers to chi-squared for RGB channels, E_i means the expected count of pixels and O_i is representing the observed count of pixels of the *i-th* channel [158].

In chi-squared statistical test context, the aim is to evaluate the significance of the relationship between the colour moment features such as mean, standard deviation, skewness and kurtosis of RGB and HSV colour spaces as well as the texture features extracted via Gabor and LBP filters. The threshold for mean calculation is the Mean value itself derived from channels having highest Mean value. It means that the highest value of any channel from R, G, B, the value will be considered as threshold value. Similarly, the threshold for standard deviation, skewness, and kurtosis is their respective values. During the chi-squared (χ^2) statistical test, mean value is used for comparing pixel distributions based on a threshold having highest Mean intensity of each channel [153].

The test typically involves two categories: one for pixels below or equal to the threshold and another one for pixels above the threshold. The threshold below or equal to is considered the best. Moreover, there is no particular range for a maximum value, the lowest possible value for a chi-squared x^2 variable is 0 as referred in [161] equation 4.4.

$$(x_c)^2 = \frac{(O_1 - E_1)^2}{E_1} + \frac{(O_2 - E_2)^2}{E_2} \quad (4.4)$$

Categories:

Category 1: Pixels with intensities below or equal to the threshold.

Category 2: Pixels with intensities above the threshold.

Observed Counts:

O_1: The observed count of pixels in Category 1.

O_2: The observed count of pixels in Category 2.

Expected Counts:

E_1: The expected count of pixels in Category 1 based on a segmented pixels value.

E_2: The expected count of pixels in Category 2 based on outcome pixels value.

The chi-squared statistical test evaluates whether there is a significant difference between the observed and expected distributions of pixel intensities relative to the threshold (mean intensity value from r, g, b channels) [25]. The chi-squared statistical test is utilised from an open-source image processing library [162].

4.2.1 Experimental Results and Discussion for Colour Feature Extraction

Table 4.6 presents the results of features extracted from AttentionU-Net model generated feature map. In Table 4.7, the upward direction arrows represent that higher the value the better is the result, and for downward arrows it is vice versa. The RGB colour space achieved 0.58 chi-squared value as

compared to HSV colour space which achieves 0.45. The standard deviation (STDV) achieved 0.17 chi-squared value for RGB colour space in contrast to HSV as 0.11. The skewness and kurtosis have been computed in both colour spaces RGB and HSV, respectively. The skewness and kurtosis achieved 0.16 and 0.21 chi-squared value in RGB colour space, respectively. While, HSV colour space achieves 0.06 and 0.19 chi-squared value for skewness and kurtosis, respectively.

Table 4.6: Colour feature extraction in RGB and HSV colour spaces

Feature	**Colour Spaces**	**Feature Extractor / Attributes**	**Eval. Metric**	**chi-squared Results ↑**		
				Attributes	**RGB**	**HSV**
Colour [153]	RGB & HSV	colour moments (mean, standard deviation, skewness, kurtosis)	chi-squared	Mean	**0.58**	0.45
				STDV	**0.17**	0.11
				Skewness	**0.16**	0.06
				kurtosis	**0.21**	0.19

Table 4.6 demonstrated experimental results of colour feature extraction from feature map generated through AttentionU-Net model. During the experiments for colour feature extraction, results demonstrate the highest values of chi-squared test for Red, Green, Blue (RGB) colour space as compared to hue, saturation, value (HSV) colour space. Moreover, in HSV colour space, hue channel determines the colour, the saturation channel determines the different shades of that colour, and the value channel describes the intensity of lightness or darkness [83]. The higher value of chi-squared values indicates that the RGB

colour space provides better colour information in WHU buildings dataset [37] for building feature map generated through AttentionU-Net. Hence, the RGB has higher chi-squared value which means that the RGB colour space is more efficient than HSV in this research.

4.2.2 Experimental Results and Discussion for Texture Feature Extraction

Table 4.7 presents results of texture feature extraction. Gabor filters achieved 0.15 chi-squared value in RGB colour space and HSV colour space depicts 0.12 chi-squared value. The Local Binar Pattern (LBP) descriptor shows 0.05 chi-squared value for the RGB colour space in contrast to HSV colour space which depicts 0.09 chi-squared value.

Table 4.7: Comparison of texture feature extraction in RGB and HSV colour spaces

Feature	**Colour Spaces**	**Feature Extractor / Attributes**	**Eval. Metric**	**chi-squared Results ↑**				
Texture	RGB & HSV	Gabor & LBP (Mean)	chi-squared	**Attributes**	**Gabor**		**LBP**	
					RGB	HSV	RGB	HSV
				Mean	**0.15**	0.12	**0.05**	0.09

The results in Table 4.7 for two texture extractors Gabor and LBP in RGB and HSV colour spaces indicates that Gabor filter demonstrates highest chi-squared value in RGB colour space. This shows the efficacy of RGB colour space for texture feature extraction. The main reason of RGB colour space

showing best chi-squared test results for colour and texture feature extraction is due to the significant difference between building ROI pixels and background pixels. Hence, RGB colour space performs better than HSV.

4.2.3 Experimental Results and Discussion for Shape Feature Extraction

During shape feature extraction, area attribute has been considered as the total number of pixels. Hence, it has been extracted during experiments that the area has same number of pixels in both colour spaces RGB and HSV, respectively.

4.2.4 Experimental Results and Discussion for Edge Feature Extraction

Table 4.8 presents the results for Canny and Sobel edge detectors for extracting edge feature. The Canny achieved average mean squared error (MSE) 17446.93 in RGB colour space as compared to Sobel which shows 10910.20, respectively. Furthermore, Canny achieves average MSE 18313.86 in contrast to Sobel 12092.30 in HSV colour space, respectively. In addition, Canny achieved average peak signal to noise ratio (PSNR) 5.75 as compared to Sobel edge detector 7.81 in RGB colour space, respectively. Moreover, Canny achieved average MSE 18313.86 and Sobel shows 12092.30 in HSV colour space, respectively. Additionally, the PSNR value achieved by Canny is 5.53 and Sobel demonstrated 7.35, respectively. The MSE and PSNR has been calculated from each building ROI feature map which is generated through the AttentionU-Net and each building is represented as K. The highest values for

both metrics is due to the *K* of building ROI. As *K* grows, the values of two metrics increase as much as *K*. Resultantly, the values are calculated as for overall *K* in order to find the optimal results for each metric.

Table 4.8: Comparison of edge feature extraction in RGB and HSV colour spaces

<table>
<tr><th>Feature</th><th>Colour Spaces</th><th>Feature Extractor / Attributes</th><th>Eval. Metric</th><th>chi-squared Results ⬆</th></tr>
<tr><td>Edge</td><td>RGB & HSV</td><td>Canny & Sobel Detectors</td><td>MSE, PSNR</td><td>
<table>
<tr><th rowspan="2">Attributes</th><th colspan="2">Canny</th><th colspan="2">Sobel</th></tr>
<tr><th>RGB</th><th>HSV</th><th>RGB</th><th>HSV</th></tr>
<tr><td>Average MSE ⬆</td><td>17446.93</td><td>18313.86</td><td>10910.20</td><td>12092.30</td></tr>
<tr><td>Average PSNR ⬇</td><td>5.75</td><td>5.53</td><td>7.81</td><td>7.35</td></tr>
</table>
</td></tr>
</table>

As presented in Table 4.8, during experimentation in two colour spaces, in general the HSV depicts higher values for MSE and lowest values for PSNR as compared to RGB colour space. This shows that HSV colour space is providing more edge information as compared to RGB colour space. However, in terms of two edge detectors comparison, Canny edge detector shows good results by achieving higher values of MSE and PSNR in both colour spaces as compared to Sobel edge detector. The main reason for good results for the edge detectors in HSV colour space is because of the edge pixels identification of buildings ROI. As the edge detectors only focus on the edge pixels of the buildings ROI then both the edge detectors performed slightly better in HSV colour space as compared to RGB colour space. The Canny edge detector depicts good results as compared to Sobel in HSV colour space as presented in Table 4.8. The HSV colour space provides good results for edge detectors. Based on the results of colour and texture feature extraction in RGB colour space, the Canny edge detector has been utilised in RGB to maintain pixel correlation.

4.3 Summary

This chapter outlines the implementation of region-based segmentation algorithms, along with CNN-based deep learning models. It also details the process of feature extraction from the generated feature map utilising the selected AttentionU-Net model. Total four region-based segmentation algorithms were evaluated for generating over segmented region, and results demonstrating SLIC with superior performance. Afterwards, seven CNN-based deep learning models are compared, and AttentionU-Net has been selected based on the best

performance. The experiments are performed between RGB and HSV colour spaces in order to extract colour feature. The comparison between Gabor and LBP for texture feature, and Canny and Sobel for edge detection are performed in RGB and HSV colour spaces. Following the results of these experiments of Chapter 4, prominent features which are colour, texture, shape, and edge information will be utilised to derive merging criterion (MC) for the proposed region merging approach in Chapter 5 for delineating buildings as ROI in remote sensing images.

CHAPTER 5

MERGING CRITERION DERIVATION FOR MERGING BUILDINGS REGION IN REMOTE SENSING IMAGES

This chapter focuses on experimental flow which involves generating the over segmented regions, followed by extracting the feature map from the AttentionU-Net convolutional neural network (CNN)-based deep learning model. These extracted features are then used to propose a merging criterion (MC) for merging buildings into regions of interest (ROIs) in remote sensing images. This process aims to address over segmentation (OS) and improve building delineation, while eliminating the requirement of human intervention in the merging process.

5.1 Derivation and Implementation of Merging Criterion

Figure 5.1 presents the flowchart for derivation of merging criterion (MC) and implementation of region merging, which is further discussed in detail in the following subsections. While, based on the results of experiments performed in Chapter 4, simple linear iterative clustering (SLIC) among four region-based segmentation algorithms, AttentionU-Net among seven CNN-based deep learning models, and colour, texture, shape, and edge features extraction were selected to derive MC. Finally, all the pairs of regions of similar features will be successfully merged if the number of regions is equivalent to K, where K is the number of buildings in the feature map.

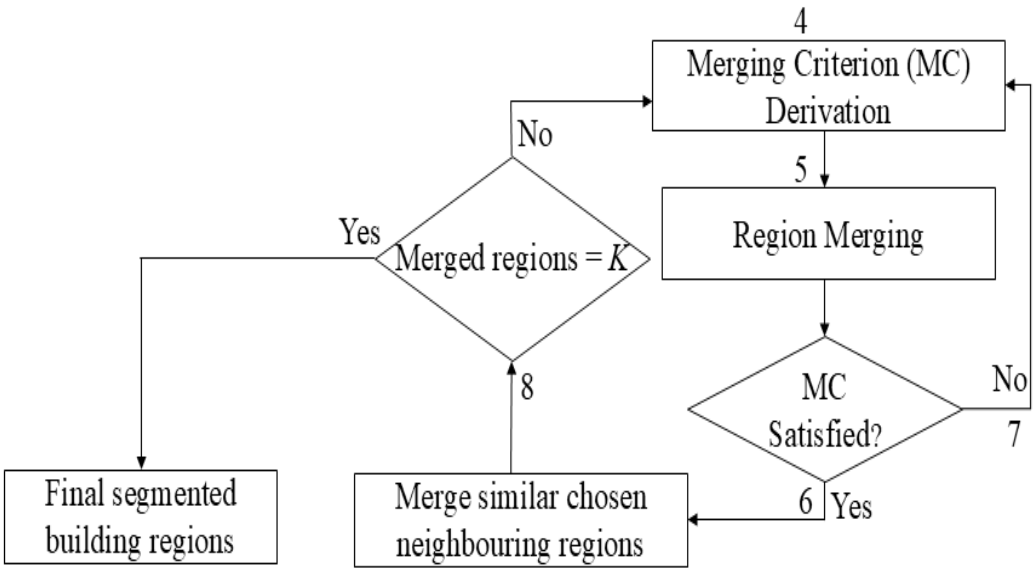


Figure 5.1: Flowchart for derivation and implementation of MC

5.1.1 Over segmented regions by SLIC represented in Region Adjacency Graph (RAG)

In Chapter 4 Section 4.1.2, experiments were performed for finding suitable region-based segmentation algorithm for the proposed region merging approach. Although, in Chapter 4 Section 4.1.2.2 results demonstrate that SLIC had outperformed to generate over segmented regions which are aligned with object boundaries. However, as mentioned earlier that SLIC generates different number of over segmented regions based on k parameter. According to authors in [138], although the process of over segmenting an image is faster with a lower value of k parameter in SLIC which produces less and larger over segmented regions boundaries, which completely missed the object regions. However, with higher k values in SLIC algorithm, generating severe over segmentation (OS) [138]. Hence, in this Chapter, experiments have been conducted on SLIC with different values of parameter k and incorporated the proposed MC. The goal is to find the suitable

value of parameter k in the SLIC to merge the over segmented regions. Moreover, the authors in [17] experimented on SLIC algorithm for finding appropriate value for parameter k and their experimental results demonstrated that best results were achieved using k=3000. In addition, the segmentation results are demonstrated from Figure 5.3 to 5.7 by utilising different values for k parameter in SLIC and MC derived from three and four features, respectively. The over segmented regions generated by SLIC are represented on RAG graph which is presented in Figures 2.1(a) and (b) Chapter 2 Section 2.2.2.1.

5.1.2 Features Extraction from Over segmented region

In below equations from 5.1 to 5.10, the following formulas are utilised to extract features from over segmented regions, and region of over segmented image is represented by I_r.

5.1.2.1 Colour Feature

I represent the input image having three channels red, green, blue (R, G, B). The image has over segmented regions where I_r represents these over segmented regions. The I_r contains pixels represented as N_r demonstrated in equation 5.1. Therefore, C_r representing mean values of colour feature for an over segmented region I_r can be calculated via equation 5.2.

$$I_r = \{(x_0, y_0), (x_1, y_1), \dots, (x_{N_r}, y_{N_r})\} \tag{5.1}$$

$$C_r = \frac{1}{N_r} \sum_{z \,\epsilon\, I_r}^{O} z(x, y) \tag{5.2}$$

where I_r showing the over segmented regions such as r = 1, . . ., *O,* N_r is representing the number of pixels in each over segmented region, and z is the pixels location. Moreover, the summation is taken over all pixels in RGB colour space representing through $z\ (x, y)$ in each over segmented region I_r.

5.1.2.2 Texture Feature

The texture feature for each over segmented region are identified through Gabor filters. The Gabor filters have been utilised by employing the equations 5.3 to 5.7 as defined in [163]. The author in [163] defined 2D Gabor filter kernel function as presented in equation 5.4.

$$G^c{}_{f,\theta}\ (x, y) = g_\theta(x', y')\ .exp[i(2\pi\ f\ x'\ +\ \psi)] \qquad (5.3)$$

where superscript $c \in \{R, G, B\}$ represents each channel, θ is the orientation, f is the specific frequency, the phase offset is represented with ψ, and $g_\theta(x', y')$ is the Gaussian envelope.

The Gaussian Envelope formulation controls the spatial extent of the Gabor function $G^c{}_{f,\theta}$ and phase offset ψ, and can be written as mentioned in equation 5.4. Furthermore, equations 5.5 and 5.6 describes the rotation of filters bank during implementation on original pixel *(x, y)* coordinates based on orientation θ.

$$g\sigma(x', y') = \exp\left[-\frac{1}{2}\left(\frac{x'^2+y'^2}{\sigma^2}\right)\right] \qquad (5.4)$$

$$x' = xcos(\theta) + ysin(\theta) \qquad (5.5)$$

$$y' = -xsin(\theta) + ycos(\theta) \tag{5.6}$$

where Gaussian Envelope $g\sigma$, has the standard deviation represented as σ. Moreover, *(x, y)* represents the original pixels values and cosine and sin θ indicates the different angles of the filters.

Based on this, the result of Gabor filters for all three channels *(R, G, B)* will be extract at pixel location *z (x, y)* by utilising equation 5.7.

$$G_{I_r} = \psi_{(m,n)}(z) = [\psi^R_{(m,n)}, \psi^G_{(m,n)}, \psi^B_{(m,n)}] \tag{5.7}$$

where G_{I_r} refers to the result of Gabor filters after appending the phase offset represented with ψ for three channels *(R, G, B)* with *m* and *n* are the direction and scale factors. It holds the Gabor response for all three channels at pixel location *z (x, y)*. I_r is representing the over segmented regions.

5.1.2.3 Shape Feature

For shape feature of each over segmented region I_r, the area attribute A_r can be computed via equation 5.8. The area contains the total number of pixels that belong to the each over segmented region.

$$A_r = N_r \tag{5.8}$$

where N_r is the total number of pixels that belong to the over segmented region I_r.

5.1.2.4 Edge Feature

Following the previous research work [73] for integrating edge features into merging criterion (MC), the edge information is extracted by applying Canny edge detector. According to [73], the Canny edge detector mathematically formulates the three criteria which low error rate, exact position, and single edge point. Moreover, authors calculate the edge intensity representing by *D* and the normal direction θ of the image at the pixel location *(x, y)* presented in equations 5.9 and 5.10, respectively.

$$D_{(x,y)} = \sqrt{\left(\frac{\partial f_s}{\partial x}\right)^2 + \left(\frac{\partial f_s}{\partial y}\right)^2} \tag{5.9}$$

$$\theta_{(x,y)} = \arctan\left(\frac{\frac{\partial f_s}{\partial y}}{\frac{\partial f_s}{\partial x}}\right) \tag{5.10}$$

where equations 5.9 and 5.10 are referring to the 3×3 kernel implemented in *x*-axis and *y*-axis direction θ on the image. ∂f_s is referring to frequently occurrence of pixels at location ∂x and ∂y. Detailed mathematical representation of Canny edge detector provided in [73].

Thus, the over segmented regions have been utilised to extract edge information by using the equations 5.9 and 5.10 of Canny edge detector. Thus, the equation of extracting edge information from the over segmented regions would be as in equation 5.11.

$$ED_{I_r} = \sum_{z \in I_r} D_z \tag{5.11}$$

where ED_{I_r} Canny Edge Detector has D_z which is referring to pixels in different over segmented regions, region is represented with $r = 1, \ldots, O$ consist of the pixels $z \in I_r$ having the pixels location of $z = (x, y)$ in over segmented region I_r.

5.1.2.5 Over segmented region Features

The four features that are colour, texture, shape, and edge are used to derive the MC from the over segmented regions as referred in the formulations of Section 5.1.2. The over segmented region features containing the colour, texture, shape, and edge are extracted by following the formulations from Section 5.1.2.1 to 5.1.2.4 and stacked through horizontal stacking [138]. According to the authors in [138], horizontal stacking is the process where different features having different dimensions are stacked as one after another. In this way, colour feature $C_{I_r \in I}$, texture feature $G_{I_r \in I}$, shape feature $A_{I_r \in I}$ and edge feature $ED_{I_r \in I}$ are horizontally stacked each after one another and can be represented as in equation 5.12.

$$F_{I_r \in I} = hstack(C_{I_r \in I}, G_{I_r \in I}, A_{I_r \in I}, ED_{I_r \in I}) \quad (5.12)$$

where $F_{I_r \in I}$ are the features from over segmented regions, and $hstack$ is referring to the horizontal stacking of four features as stated above. The dimension of $F_{I_r \in I}$ will be 1 x 6, which is having horizontal stacking where 6 rows and 1 column. The 3 dimensions are for colour having mean of R, G, B channel, 1 for texture feature resultant from Gabor matrix as mean, 1 for shape as area attribute having number of pixels, and 1 for edge information as indices. The merging process will use these stacked features when the difference in these feature values (colour, texture, shape,

and edge information) between neighbouring regions is below a threshold as defined in below Section 5.1.4.

5.1.3 Feature Extraction from AttentionU-Net generated Feature Map

The above process mentioned in Section 5.1.2 is demonstrated for features extraction from over segmented regions. The following formulas from Section 5.1.2.1 to 5.1.2.4 are utilised for the feature extraction from the AttentionU-Net CNN-based deep learning model generated feature map. Additionally, the AttentionU-Net model includes different connected components which are the building ROI and black background. These connected components were derived through the AttentionU-Net CNN-based deep learning model. Thus, it is necessary to utilised the only pixels having non-background and consists of buildings ROI to derive MC.

The feature map generated by AttentionU-Net is represented by I_l which contains connected components. These connected components correspond to individual building regions of interest (ROIs) and consist of K ROI, such that the $I_l = \{1, \ldots, K\}$ with each ROI containing N_l number of pixels. Afterwards, an indicator function referred from previous work of [164] as demonstrated in equation 5.13 will assist in the feature calculation for each building ROI in feature map generated through the AttentionU-Net.

$$\psi(z) = \begin{cases} 1 & z > 0, \quad z \in I_l \\ 0 & Otherwise \end{cases} \tag{5.13}$$

where ψ(z) an indicator function [164], it calculates the feature values when the pixel value *z* is non-zero and otherwise refers that the calculation for zero value pixels will not be computed. It is used for AttentionU-Net generated feature map building ROI in order to specifically used building ROI pixels for calculation and ignore the background pixels which are in black colour and would have zero value. In indicator function ψ(z), $z = (x, y)$ is the pixels position, and $z \in I_l$. Thus, the indicator function will identify and calculate only the building regions. It will extract the exact information of features without considering the background pixels.

As defined in equation 5.13, the indicator function utilised to extract feature from each building ROI generated feature map through AttentionU-Net and representing a labelled region $I_l \in I$. Moreover, labelled region $l = 1, \ldots K$, K representing each building from feature map generated through AttentionU-Net. Then, the formulas for colour, texture, shape, and edge feature, presented in Sections 5.1.2.1 to 5.1.2.4 for the over segmented region to be applied.

Thus, formulas for extracting feature from buildings ROI feature map generated through AttentionU-Net which are colour, texture, shape, and edge feature. Hence, buildings ROI feature map generated through AttentionU-Net can be represented as in equation 5.14.

$$F_{I_l \in I} = hstack(C_{I_l \in I}, G_{I_l \in I}, A_{I_l \in I}, ED_{I_l \in I}) \qquad (5.14)$$

where the $F_{I_l \in I}$ is referring to the horizontal stacking [138]. Moreover, colour represented as $C_{I_l \in I}$, texture as $G_{I_l \in I}$, shape as $A_{I_l \in I}$, and edge feature represented as $ED_{I_l \in I}$.

5.1.4 Threshold Calculation

The threshold is defined in order to put the limit in merging edges and employed in merging criterion (MC)**.** According to previous work [28], the authors stated that due to the high variability of the remote sensing image features, utilisation of a static value or hard threshold affects the overall segmentation efficiency [25]. Hence, the threshold represented with th is computed for all the edge weights in the region adjacency graph (RAG) by utilising equation 5.15.

$$th = \mu\left(F_{I_l \in I}\right)K \tag{5.15}$$

where μ is the mean of the building ROI generated feature map through AttentionU-Net, and $F_{I_l \in I}$ is calculated through equation 5.14. As demonstrated in equation 5.15, the *K* denotes the total number of buildings in the feature map. The different buildings ROI will have different mean values and for each building mean value will act as the threshold for iterative regions merging.

5.2 Merging Criterion (MC)

This step involves deriving the MC from the AttentionU-Net building ROI. Moreover, the MC based on the three (colour, texture, and shape) and four features (colour, texture, shape, and edge information) is incorporated into the merging

process over the RAG generated from the SLIC over segmentation (OS). Based on the over segmented image having adjacent regions such as I_r , $I_j \in I$, a RAG graph is presented in Figure 2.2, Section 2.3.4 Chapter 2. The RAG graph consists of node in each over segmented region and edges between adjacent over segmented regions. Each node represents the calculated over segmented region features vector for each over segmented region. The edges in RAG represent the dissimilarity between adjacent regions I_r , $I_j \in I$. Hence, I_r , I_j is representing the two adjacent over segmented regions in an over segmented image.

As presented in equation 5.16, $w_{(I_r\,,\ I_j)\,\in\, I}$ will be the weight calculation formula of the edge connecting nodes on RAG representing regions I_r and I_j, which corresponds to the dissimilarity between the over segmented regions having features vector of the two adjacent regions, I_r and I_j in the over segmented image. Moreover, I_j represents the adjacent regions to I_r in the over segmented image.

$$E = w_{(I_r, I_j)\,\in\, I} = \sum_{I_r=1}^{O} \sum_{I_j=1}^{b} \left\| F_{I_r} - F_{I_j} \right\| \quad \forall \ \ I_r \neq I_j \qquad (5.16)$$

where $w_{(I_r\,,\, j_r)\,\in\, I}$ represents the edge weights between I_r and I_j, which are the neighbouring over segmented regions. Moreover, O is the total number of over segmented regions, and b is the total number of neighbouring over segmented regions for any given node I_r. While, $\sum$ is used to calculate the total features values of over segmented region, which are then subtracted from the adjacent over segmented region using $\left\| F_{I_r} - F_{I_j} \right\|$. This results in a feature value that will be compared with the feature values of the building ROI generated feature map.

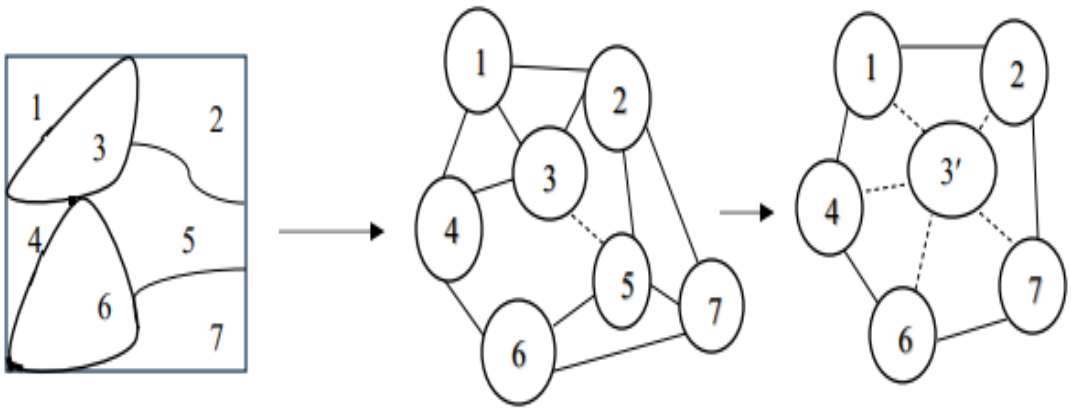
Finally, a merging criterion $M(I_r\ ,\ I_j)$ will determine when to merge two nodes I_r and I_j based on the weight of the edge connecting them and the computed threshold through equation 5.15. Based on the above formulas, the following merging criterion (MC) performs the merging process, and hence can be represented as in equation 5.17.

$$M_{(I_r\ ,\ I_j)\in I} = \begin{cases} 1 & if\ w_{(I_r,I_j)\in I} = min(w_{(I_r,I_j)\in I})\ and\ w_{(I_r,I_j)\in I} < th \\ 0 & Otherwise \end{cases} \quad (5.17)$$

where $min(w_{(I_r\ ,\ I_j)\in I})$ is the minimum edge weight in the RAG and *th* represents the threshold.

5.2.1 Iterative Region Merging

When the MC $M_{(I_r\ ,I_j)\in I}$ is equals or less than the threshold, then regions I_r and I_j can be merged into a single a region. If the difference of edge weights is below the threshold computed in equation 5.15 and satisfied the merging criterion (MC) presented in equation 5.17, the two nodes merge to generate a new region. After merging nodes, the new region features are computed and assigned to new node in the RAG. Following each merge, the edges of the newly formed region are recalculated based on the four features of the merged regions. The region merging process continues until all the nodes in the RAG are merged based on the merging criterion (MC).



(a) **(b)** **(c)**

Figure 5.2: Construction of RAG and iterative region merging process in RAG

In graph-based region merging that involves constructing a region adjacency graph (RAG) to represent the over segmented regions. The feature information of the corresponding over segmented region is represented by each node, with each edge representing the feature distance (or dissimilarity) between the connected nodes. Figure 5.2 shows the concept of a RAG model. Beginning with seven over segmented regions as shown in Figure 5.2 (a), Figure 5.2 (b) shows the nodes representing the over segmented regions, together with the edges connecting them representing the relations between the regions. The merging criterion (MC) formula of equation 5.17 assigns the weight of each edge as a dissimilarity of its two adjacent nodes. For instance, the weight of the edge connecting nodes n3 and n5 in Figure 5.2 (b) is found to be smallest, this means the nodes are highly similar. As a result, n3 and n5 will be merged into a new node, a new n3′, as shown in Figure 5.2 (c). The edge weights of node n3′ are subsequently changed. The process iteratively proceeds with the RAG model until no merging can be performed.

5.3 Analysis and Discussion

The performance evaluation results have been demonstrated in Tables 5.1 and 5.2 (see Appendix A) in terms of comparison between the integration of three (colour, texture, and shape) and four features (colour, texture, shape, and edge information) into MC for different SLIC algorithm k parameter. In addition, the upward direction arrows represent that the higher the value the better is the result, and for downward arrows it is vice versa in both the tables. The adapted rand error (ARE), variation of information (VOI), F-measure, precision, recall, and adapted rand index (ARI) metrics have been employed as stated in Chapter 3 from Sections 3.1.5.1 to 3.1.5.3, respectively. Furthermore, in order to find the better resultant feature integrated into MC, the comparison have been performed between three features as MC and Four features as MC. Finally, the proposed region merging approach and previous work are compared by utilising metrics which are the G_s, the F-measure, precision, and recall stated in Chapter 3 Section 3.1.5.4 and 3.1.5.3, respectively.

5.3.1 Results for Three Features with specified k parameter value in SLIC

The performance evaluation results of incorporating three features into MC for different k parameter values in SLIC are presented in Table 5.1. These values for parameter k are 500, 1000, 1500, 2000, and 3000, respectively. As presented in Table 5.1 that the three features incorporating in MC, image ID 101 has achieved F-Measure of 0.8465 through SLIC algorithm with parameter k=3000 by

generating 2165 number of over segmented regions. The same image with k=3000 parameter of SLIC achieved values of 0.8632 for ARI, 0.1535 for ARE, and VOI value is reached to 0.7594, respectively.

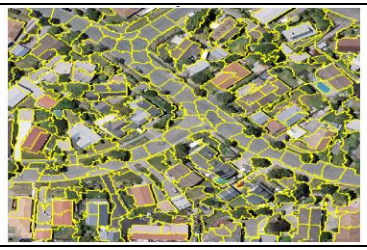
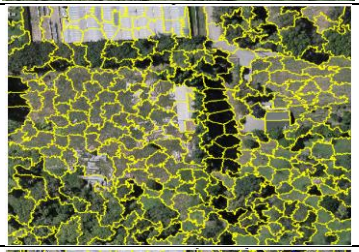
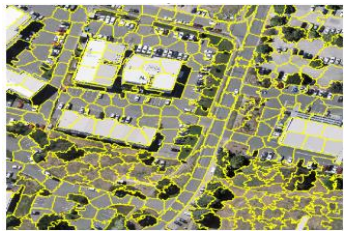
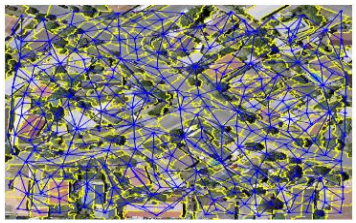
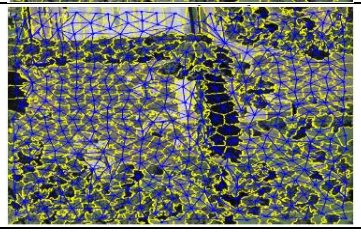
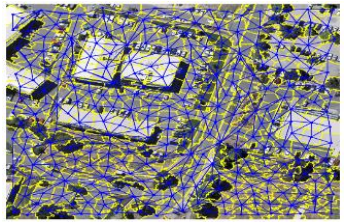
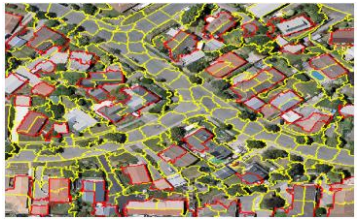
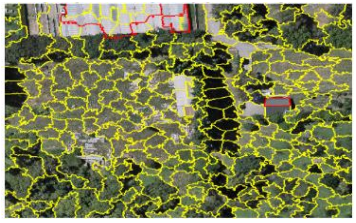
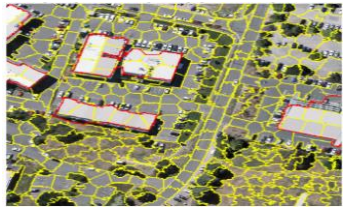
The image ID 829 has demonstrated the F-measure value of 0.9865 via SLIC algorithm k parameter of value 3000 by generating 2068 over segmented regions. While, the same image with achieved values of 0.9553 for ARI, ARE is 0.0153, and VOI is 0.0781, with the SLIC k=3000, respectively. The image ID 1027 has shown the value 0.9548 for F-measure metric by setting SLIC k=1000 which generated over segmented regions of 661. Along with this, the same image with achieved values of ARE is 0.0452. However, for ARI, and VOI, the values are 0.9307 and 0.2083, with the SLIC k=3000, respectively.

5.3.2 Results for Four Features with specified k parameter value in SLIC

As demonstrated in Table 5.2 for utilising the four features which are colour, texture, shape, and edge information incorporated into MC from AttentionU-Net generated feature map. The results of different initial over segmentation (OS) have been demonstrated by employing MC during the regions merging. The image ID 101 has achieved F-Measure of 0.9040 through SLIC with parameter k=3000 by generating 2165 number of over segmented regions. For the same image, having parameter k=3000 of SLIC achieved values of 0.9102 for ARI, 0.0873 for ARE, and VOI value is reached to 0.3592, respectively.

The image ID 829 has demonstrated the F-measure value of 0.9990 for SLIC k parameter of value 3000 by generating 2068 over segmented regions. Additionally, the same image has achieved values of 0.9781 for ARI, ARE is 0.0127, and VOI is 0.0301, with the SLIC k=3000, respectively. The image ID 1027 has shown the value 0.9825 for F-measure metric by setting SLIC k=3000 by generating number of 2324 of over segmented regions. Additionally, the same image achieved ARI values of 0.9893. The metric ARE demonstrated 0.0397 and VOI shows 0.0912 with the SLIC k=1500, respectively.

The difference of F-measure between Tables 5.1 and 5.2 are also notable. As from the results in above Table 5.2, the average F-measure values for all the images have achieved higher value as compared to Table 5.1. One of the reasons of achieving higher values of Table 5.2 is because of the features incorporation into MC. It means that the four features integration as MC has significantly impacts on the final segmentation outcomes. It is because that, the more features are utilised the better will be segmentation results. Furthermore, from Figure 5.3 to 5.7, the segmentation results of parameter k different values in SLIC algorithm have been presented visually.

Input Image No.	101	829	1027
Input Image			
Over Segmentation via SLIC algorithm			
RAG Graph			
Merging Regions with Three Features in MC			

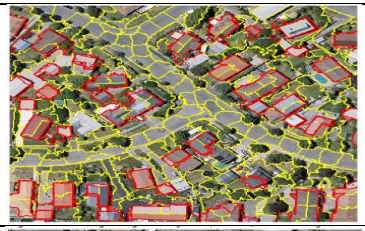
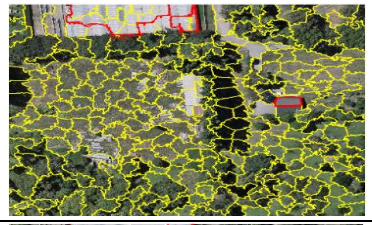
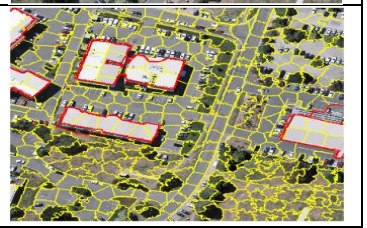
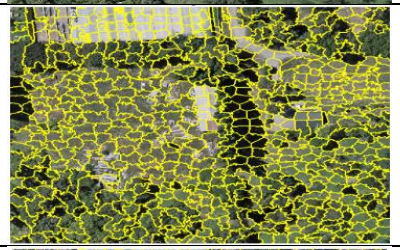
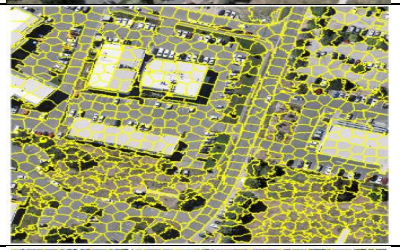
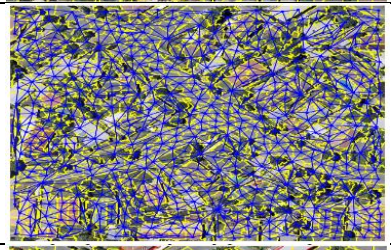
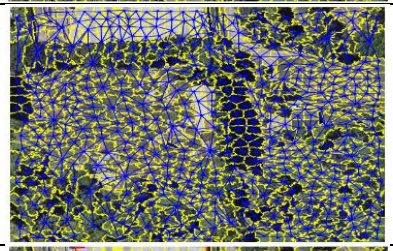
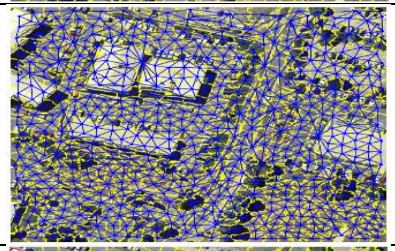
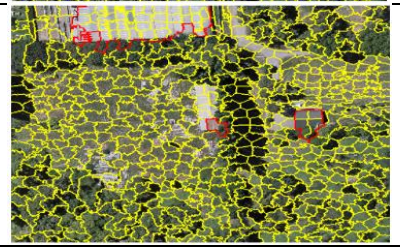
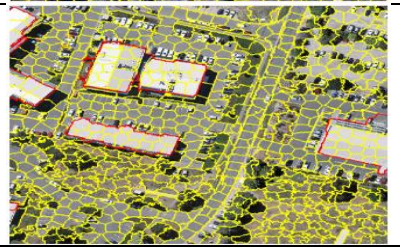
Segmented Regions for Three Features in MC			
Merging Regions with Four Features in MC			
Final Segmented Regions having Four Features in MC			

Figure 5.3: SLIC Parameter value of k = 500

Input Image No.	101	829	1027
Input Image			
Over Segmentation via SLIC algorithm			
RAG Graph			
Merging Regions with Three Features in MC			

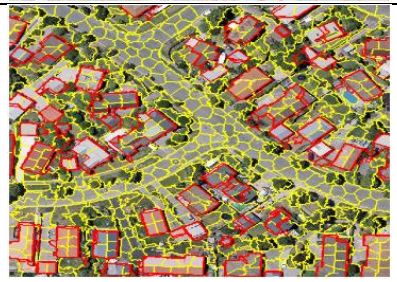
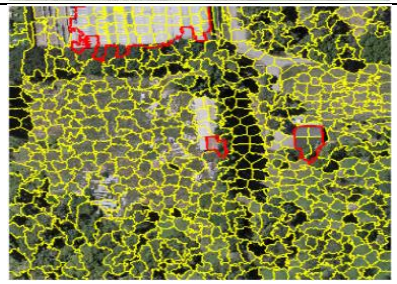
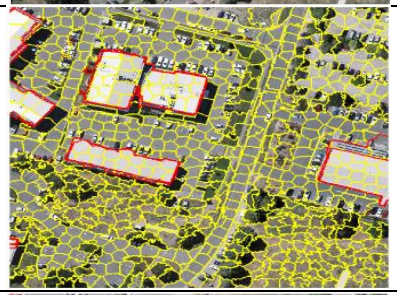
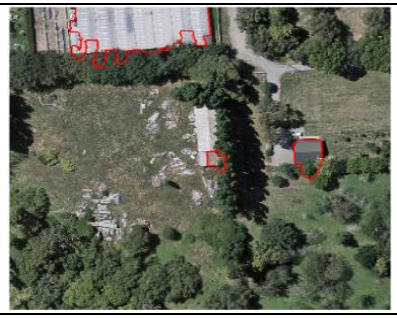
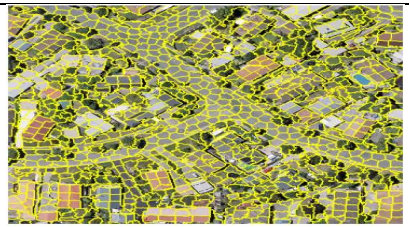
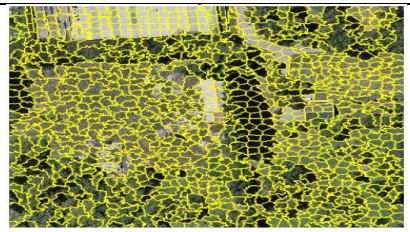
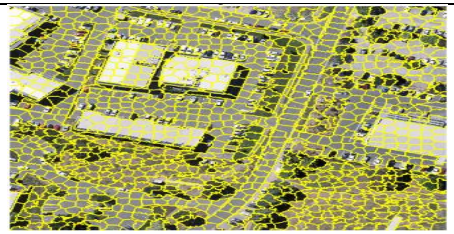
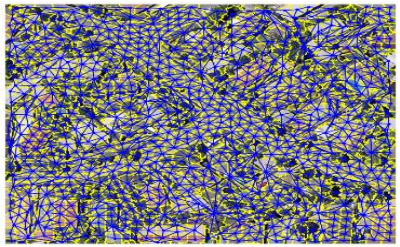
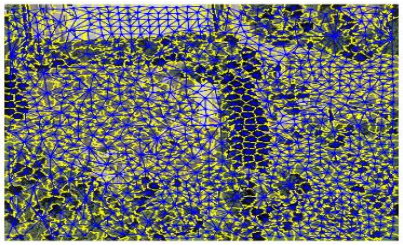
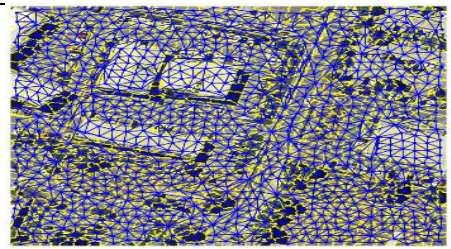
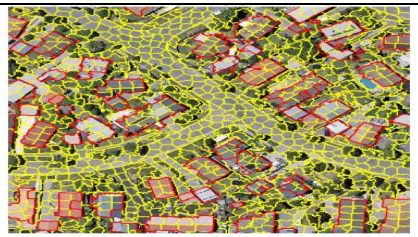
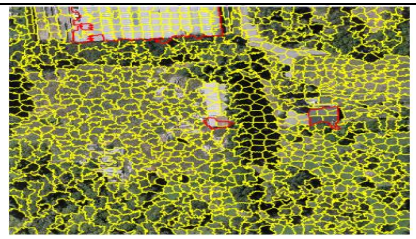
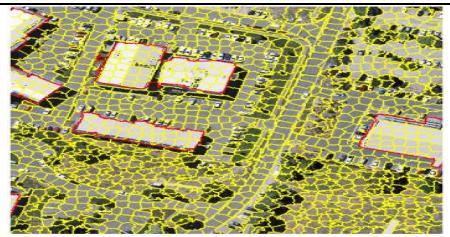
Segmented Regions for Three Features in MC			
Merging Regions with Four Features in MC			
Final Segmented Regions having Four Features in MC			

Figure 5.4: SLIC Parameter value of k = 1000

Input Image No.	101	829	1027
Input Image			
Over Segmentation via SLIC algorithm			
RAG Graph			
Merging Regions with Thress Features in MC			

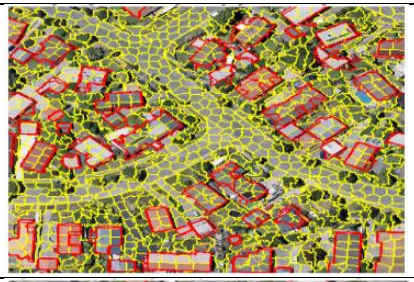
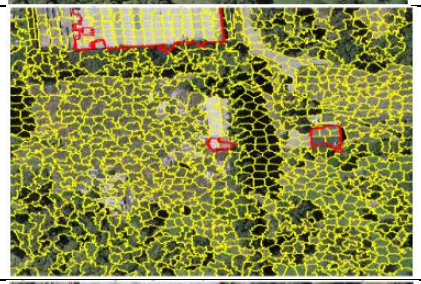
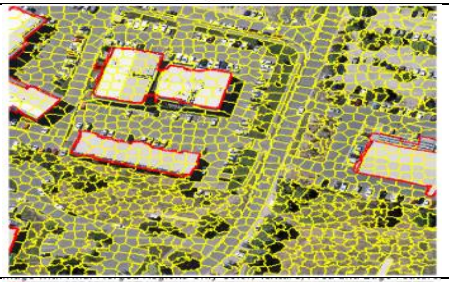
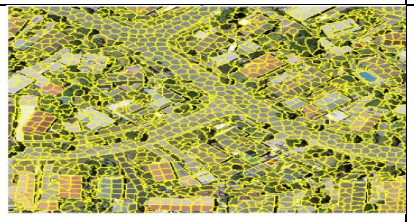
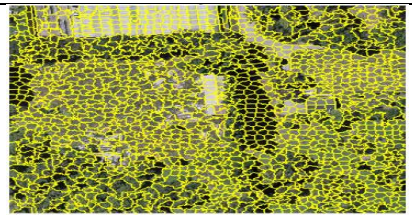
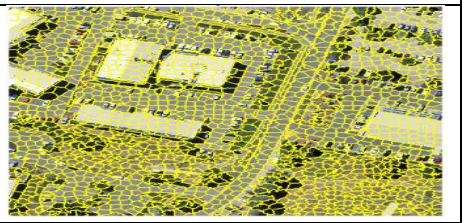
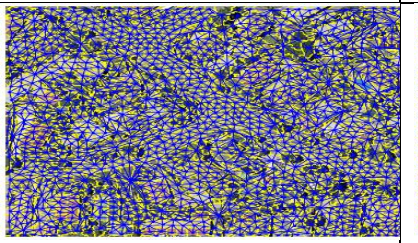
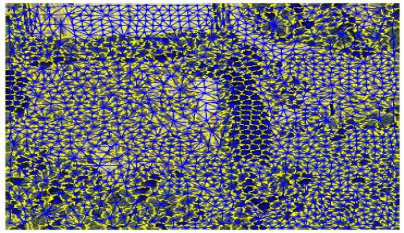
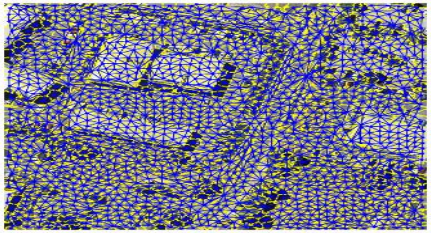
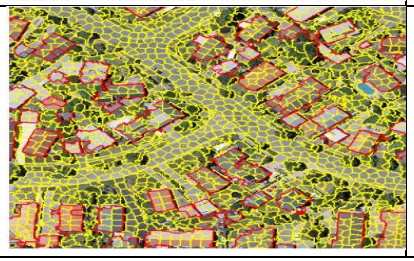
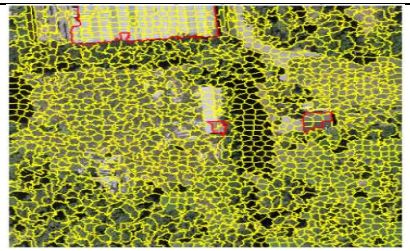
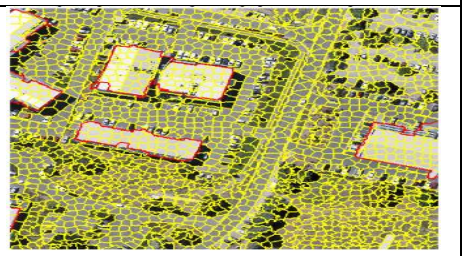
Segmented Regions for Three Features in MC			
Merging Regions with Four Features in MC			
Final Segmented Regions having Four Features in MC			

Figure 5.5: SLIC Parameter value of k = 1500

Input Image No.	101	829	1027
Input Image			
Over Segmentation via SLIC algorithm			
RAG Graph			
Merging Regions with Three Features in MC			

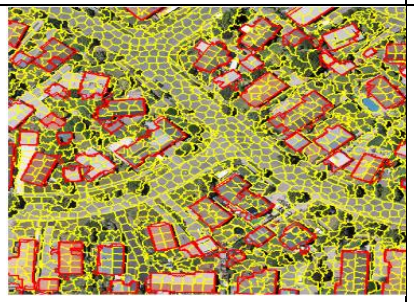
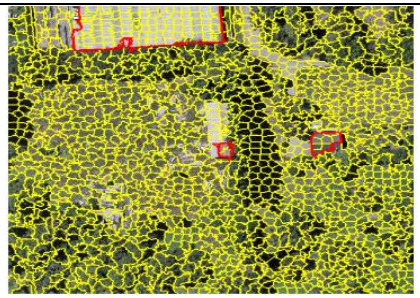
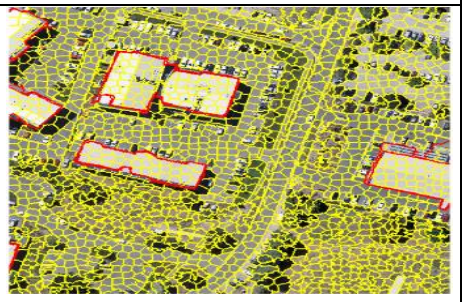
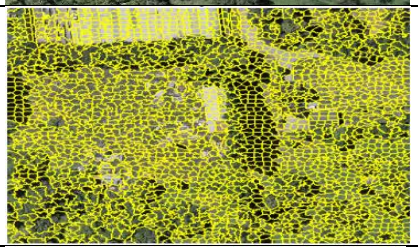
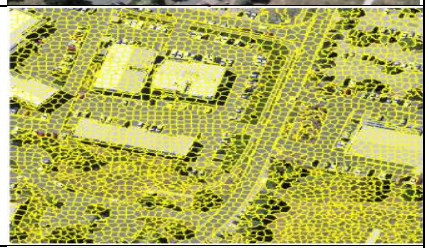
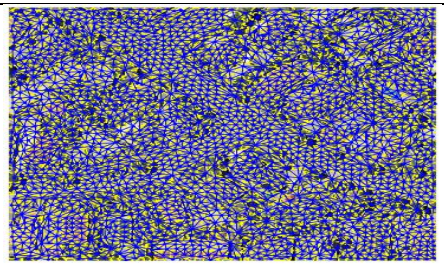
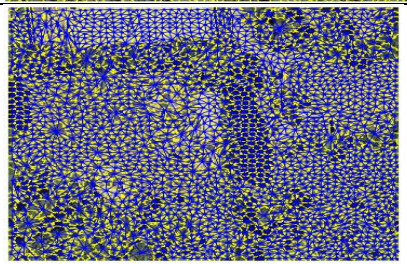
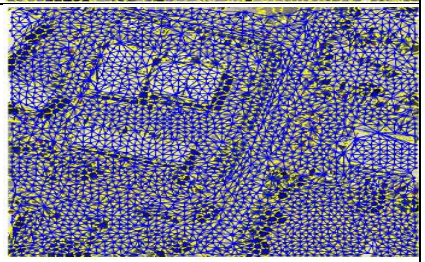
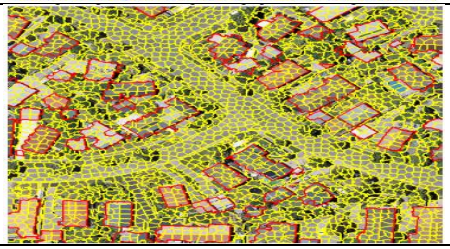
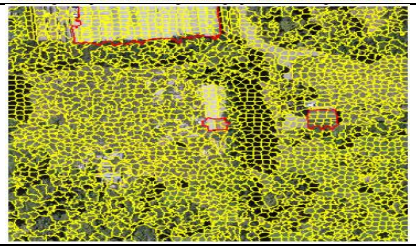
Segmented Regions for Three Features in MC			
Merging Regions with Four Features in MC			
Final Segmented Regions having Four Features in MC			

Figure 5.6: SLIC Parameter value of k = 2000

Input Image No.	101	829	1027
Input Image			
Over Segmentation via SLIC algorithm			
RAG Graph			
Merging Regions with Three Features in MC			

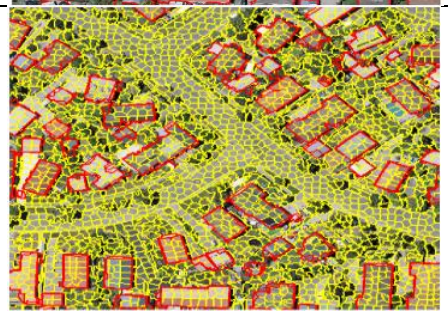
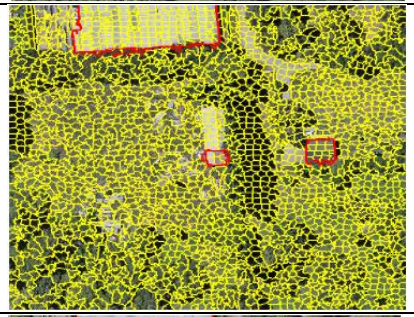
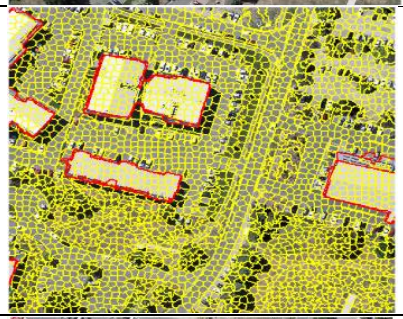
Segmented Regions for Three Features in MC			
Merging Regions with Four Features in MC			
Final Segmented Regions having Four Features in MC			

Figure 5.7: SLIC Parameter value of k = 3000

5.3.3 Analysis on specified k parameter value in SLIC by incorporation of Three and Four Features as MC

From results presented in Table 5.1 and 5.2 as well as from Figure 5.3 to 5.7, it has been observed that the initial over segmentation (OS) has influence on regions merging process. The results indicate that, the higher the OS the better is the final segmentation. The tables 5.1 present three features results and 5.2 demonstrates the results of four features as MC (see Appendix A). The metric used for comparison between three and four features as MC are F-measure, precision, recall, ARI, ARE, and VOI. The results of both the tables 5.1 and 5.2 proves that SLIC with parameter k=3000 outperform among k values of 500, 1000, 1500, and 2000. Moreover, the results also indicate that, four features which includes colour, texture, shape, and edge feature are providing better segmentation results as compared to three features as MC.

It has been observed in Tables 5.1 and 5.2 (see appendix A), respectively that the SLIC algorithm parameter k=3000 shows better results for three images in terms of performance evaluation. While, in Table 5.3, SLIC parameter k=3000 also achieved best results on WHU buildings Dataset [37]. It means that when the initial OS is excessive, then the features employed into MC are most correlate to regions in the over segmented region. In this way, it becomes easy for the MC to be satisfied and merge regions, hence provides better segmentation results.

Furthermore, in terms of evaluation results for different initial OS, it has also been observed from Figure 5.3 to 5.7 for image ID 101, 829, and 1027 that the

SLIC algorithm having parameter k value of 500, 1000, 1500, 2000 have missed some buildings region either completely or partially for three features as MC as well as for four features as MC, respectively. The parameter k value 1500, and 2000 has segmented the buildings ROI but due to some inaccurate boundaries identified for regions as compared to k parameter value 3000. Although, k parameter value 3000 of SLIC algorithm also contains some missed or irregular boundaries even though the edge information is present in AttentionU-Net generated feature map to provide accurate boundary. The edge features implementation is based on the satisfying of three features merging criterion (MC), and once the three features incorporated provides initial merging range, then the edge information assists in uniforming those boundaries which have high similarity with over segmented region. It means that even though the edge features from the Attention U-Net generated feature map and the over segmented region are included in the feature vector, they will not merge outside the three features range. This is due to the strength of the information between the edges. As it can be seen in Figure 5.3 of SLIC algorithm having parameter k=500 from row fourth to row seventh row for image ID 101, 829, and 1027 has missed two buildings in top-centre of image, centre building, and bottom-left, respectively. Although, the buildings in top-centre of image, centre building, and bottom-left, are present in the AttentionU-Net generated feature map as well as the edge features intensity map, respectively.

The process of incorporating edge features along with three features will work as initially the colour, texture, and shape features criterion will be satisfied and based on which the edge features would assist in generating the boundaries of

within the generated or satisfied colour, texture, and shape region in terms of generating uniformed edges. It means that once the three features satisfied the process of merging region, the edge features map would fulfil the region boundaries within the merged region and will produce the uniform boundaries. This way, the resultant image would have straight boundaries with incorporating four features into MC as compared to three features. Meanwhile, to be more specific, the colour, texture, and shape features will select the relevant features of object and later will be uniformed with edge information. During the region merging process, colour, texture, shape, and edge features collectively enhance segmentation. Initially, colour features are grouped based on mean values similarity, followed by texture, which refines the grouping by considering surface patterns via Gabor filter matrices mean values. Afterwards, shape features further refined region having number of pixels. Finally, edge features are used to precisely define boundaries, ensuring clear segmentation of ROI. In a brief it means that, the entire four features have been used to merge but the three features (colour, texture, and shape) help with identifying and grouping the regions. Once complete this, the fourth feature which is edge information will refine the segmentation by clearly marking the building region boundaries.

As mentioned in literature that, region growing algorithm starts from within the objects and finish until identifying the boundaries. Hence, this process has similarity to region growing algorithm which is discussed in detail in Chapter 2 Section 2.1.1. In this proposed region merging approach, if initially the edge features are integrated and then the other three features mentioned above are

utilised to perform merging. Then the chances are there that the buildings ROI feature map generated by AttentionU-Net having edges that are consists of extra pixels, such edges would influence the region merging and resultantly the building segmentation would be inaccurate. Based on the comparisons in tables 5.1 and 5.2 (see Appendix A) and F-measure values, as well as the visual results from Figures 5.3 to 5.7, this research explores the generation of initial over segmentation (OS). While, different k parameter values of the SLIC algorithm are applied, and the results are compared using both three and four features based MC. The findings demonstrated that SLIC algorithm provides better results by using parameter k=3000.

5.3.4 Comparison of Three and Four features as MC

The results in below Table 5.3 are demonstrated having the comparison between three features and four features employed into MC for different parameter of k in SLIC. While, the SLIC algorithm with parameter k=500, 1000, 1500, 2000, and 3000 is utilised to generate initial OS. The best results achieved with k=3000 as average results for F-measure on WHU buildings dataset [37] for three features achieved 0.8826, ARI as 0.8976, ARE as 0.1174, and VOI as 0.5057, respectively. The four features acting as MC produces the average results on WHU Buildings dataset for F-measure of 0.9101, ARI of 0.7603, ARE as 0.0793, and VOI as 0.3951, respectively.

Table 5.3 Average Results for Three and Four features incorporated as MC for parameter k value 500, 1000, 1500, 2000, 3000, respectively on 1550 images of WHU buildings Dataset [37]

Features into MC	OS Value of parameter k for SLIC Algorithm	Evaluation Metrics					
		F-Measure ↑	precision ↑	recall ↑	Adjusted Rand Index (ARI) ↑	Adapted Rand Error (ARE) ↓	Variation of Information (VOI) ↓
Three Features	500	0.6354	0.6954	0.5857	0.5751	0.2061	0.6437
	1000	0.6606	0.7132	0.6145	0.6903	0.1899	0.5711
	1500	0.7182	0.7691	0.6783	0.7999	0.1701	0.5552
	2000	0.7571	0.7999	0.7201	0.8019	0.1699	0.5476
	3000	0.8826	0.8840	0.8825	**0.8976**	0.1174	0.5057
Four Features	500	0.8910	0.9010	0.8850	0.6452	0.1560	0.4931
	1000	0.9012	0.9104	0.8910	0.6903	0.1395	0.3075
	1500	0.9064	0.9150	0.8958	0.7370	0.1291	0.1590
	2000	0.9092	0.9180	0.8999	0.7601	0.1099	0.0631
	3000	**0.9101**	**0.9204**	**0.9001**	0.7603	**0.0793**	**0.3951**

Image No.	Three Features as MC having SLIC k=3000	Four Features as MC having SLIC k=3000
101		

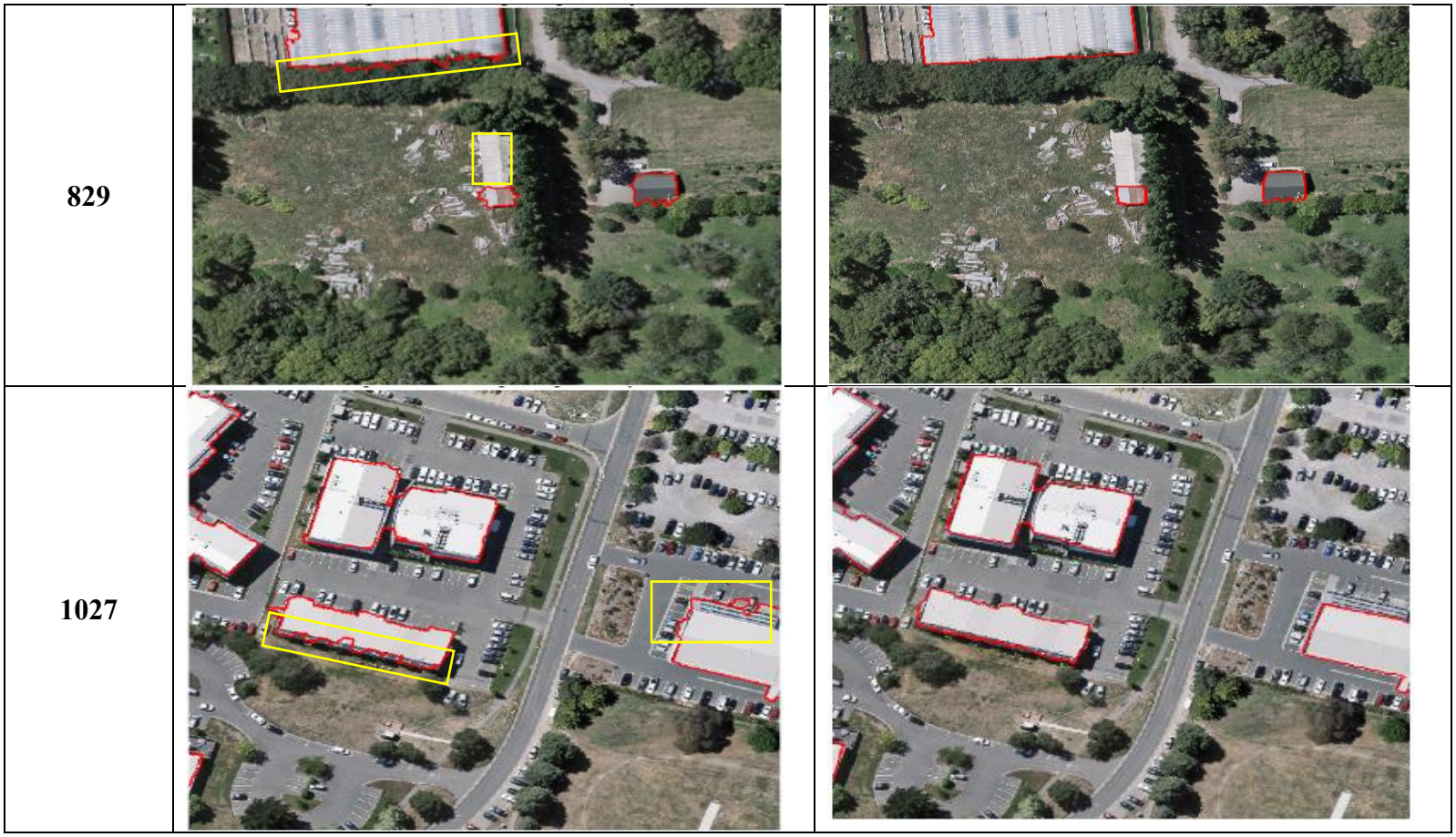


Figure 5.8: Comparison of segmentation results by incorporating Three and Four Features into MC, respectively

5.3.4.1 Analysis on Three and Four Features incorporation into MC

In Table 5.3, the comparison between integrating three features in MC and four features in MC has been performed. The results proves that four features MC containing colour, texture, shape and edge features achieved higher F-measure of 0.9101 as compare to three features as MC achieved 0.8826. Furthermore, it can be visualised in Figure 5.8 that the image ID 101, 829, and 1027 has buildings extracted through three features integration in MC, however the boundaries of buildings are not uniform or regular as shown in yellow colour boxes. Hence, the visual results of four features incorporated into MC demonstrated uniform buildings boundaries as compared to three features integration into MC. This is because the edge information from building feature map based on AttentionU-Net assists in the generation of uniformed boundaries of building ROI. The edge feature is used in order to identify the location of boundaries that are very similar or nearby to the over segmented region boundary pixels. This process executes when the three features colour, texture, and shape achieved the initial merging range.

Thus, it shows that the edge features will not consider the buildings boundaries outside the three features range as mentioned earlier. This comparison between three features and four features integration into MC indicates the trade-off about the utilisation of three features into MC for implementing region merging process in this research. Hence, the SLIC parameter k=3000 and four features integrated into MC is utilised for comparison with previous research works proposed region merging approaches.

5.4 Comparison with Previous Research Works

The results comparison between the proposed region merging approach results from four features incorporated into MC and with previous research work [36], and [28] on WHU buildings dataset [37] demonstrated in Table 5.4 for 1550 images. The authors in [36] proposed the MRS algorithm, which is widely regarded as a highly effective segmentation algorithm to segment remote sensing images. The MRS algorithm has been successfully implemented in commercial software named as eCognition, which includes a module for MRS algorithm [36].

According to the authors in [36], MRS algorithm is a bottom-up region merging algorithm that starts by treating each pixel as an object and then merges the pixels according to the MC. Furthermore, the MC of MRS algorithm is regulated by two parameters which are (1) shape, which also controls colour, and (2) compactness, which oversee the smoothness. The shape criteria range from 0 to 0.9 and measures the extent to which the shape of an image object affects the segmentation process relative to the object spectral features or colour [36]. Moreover, the compactness parameter is determined by the standard deviations of the spectral values linked to the initial pixel [36]. Besides this, MRS algorithm also consists of colour, and shape features as MC and scale parameter act as threshold for optimisation process of regions merging. Based on the properties mentioned above, the MRS algorithm is used in this research to compare the final segmentation results on the WHU buildings dataset. Furthermore, as mentioned above that MRS [36] requires three parameters to implement and perform segmentation. The three parameters consist of scale parameter, shape and compactness. According to the

comparison of previous work [28], the author compared different scale parameter (SP) and resultantly the 130 provides better buildings segmentation on WHU buildings dataset in since shape and compactness were kept as 0.5 for the experiments. Hence, in the experimentation of MRS algorithm [36] in this research, scale parameter (SP) is taken as 130 and other two parameters having shape and compactness are kept as 0.5.

Although, the results produced by [28] are promising such as the average F-measure achieved in [28] is 0.63 on WHU buildings dataset. However, the achieved value of [28] is lower than the proposed region merging approach in this research work which is average F-measure 0.91 on the same dataset. Furthermore, the precision value of the proposed region merging approach in this research is 0.92, whereas [28] achieved 0.82. Moreover, the recall is 0.90 in this research work, whereas the [28] achieved 0.52 average recall. Furthermore, the results of MRS algorithm in Table 5.5 demonstrates the G_s metric values in comparison to the proposed region merging approach. The MRS achieved 0.83 average G_s as compared to the proposed region merging approach in this research work which achieves 0.92 average G_s for WHU buildings dataset.

Table 5.4: Comparison of final segmentation results of proposed approach with previous research work [28]

Ref / Year	Initial Segmentation Algorithm	Dataset	Metrics Used	Average F-Measure	Average precision	Average recall
[28] / 2022	Watershed transformation	WHU buildings	F-Measure, Precision, and Recall.	0.63	0.82	0.52
Proposed Approach / Four Features as MC	SLIC algorithm parameter k =3000 value			**0.91**	**0.92**	**0.90**

Table 5.5: Comparison of final segmentation results of proposed approach with previous research work [36]

Ref	Initial Segmentation Algorithm	Dataset	Metric Used	Average G_s
[36]	Region Growing	WHU buildings	G_s (Goodness for segmentation)	0.83
Proposed Approach / Four Features as MC	SLIC algorithm parameter k =3000 value			**0.92**

Figure 5.9 demonstrated the comparison between the proposed region merging approach in this research and the MRS algorithm proposed by [36]. The proposed region merging approach and MRS algorithm [36] are evaluated on the same WHU buildings dataset. In Figure 5.9, the first row shows the image ID 829 buildings segmentation through the proposed region merging approach and in comparison, to the MRS algorithm for the same image. In this first row, the yellow boxes represents that the building is either has missed the ROI or other regions are been segment which can be further analysed in third row via the zoomed building of image ID 829 for both approaches. In the third row having image ID 1027, it can be analysed that some buildings have slightly under-segment in the proposed region merging approach shown in yellow boxes at centre-right side and the MRS algorithm generate slightly over segmented regions demonstrated in yellow boxes at centre in third row. The reason of two comparison having different metrics is that the previous work [28] used the segmentation specifically for buildings ROI, while the [36] algorithm perform segmentation for other objects as well. Hence, it is difficult to analyse the performance of segmentation with other metric, resultantly the G_s metric has been utilised for comparing the buildings delineation of proposed approach with MRS [36].

Image No.	Four Features as MC having SLIC k=3000	Buildings Segmented by MRS Algorithm
829		
829 Zoomed-In Building ROI		

1027		
1027 Zoomed-In Building ROI		

Figure 5.9: Comparison of final segmentation results with utilisation of four features into MC, with MRS algorithm [36] on WHU buildings dataset [37]

5.4.1 Analysis on the Proposed Region Merging Approach in comparison to previous research works

As demonstrated the results comparison in Table 5.4, average F-measure, precision, and recall of the proposed region merging approach in this research are higher than previous research work region merging approach [28]. The main reason of higher result of the proposed region merging approach in this research as compared to previous work [28] is due to the WHU buildings dataset preprocessing phase. It means that the WHU buildings dataset used in previous work [28] is not preprocessed by the authors. In [28] authors stated that the preprocessing is not performed in order to analyse the performance of the algorithm proposed in [28]. While, the dataset which is utilised in this research is actually preprocessed by the authors [37] and made publicly available. In this research, the utilised features as MC can also be the factor of higher results. In this research colour, texture, shape, and edge features are employed as MC. The researchers in [28] utilised only spectral angle (SA) as average homogeneity and heterogeneity for MC derivation. Hence, these could be one of the reasons of high difference of results between this research and previous work of [28]. Moreover, in the methodology of [28], the approach to segment buildings in remote sensing images is similar to this research in such a way that building features from labelled image are used as feature map to derive MC. Although, in the previous work of [28], the feature map employed to derive MC is containing building features identified as ROI which are generated using specialised software [28]. Furthermore, in Table 5.5 MRS achieved lower G_s (goodness of segmentation) as compared to the proposed region merging

approach in this research work. It is mainly due to the features integrated into MC. Moreover, the MRS algorithm is also depending on the colour, texture, and smoothness features as MC. While in this research work, MC is derived from the colour, texture, shape, and edge features. Hence, this could be one of the factors of higher results of the proposed region merging approach.

In Figure 5.9, visual results comparison of the proposed region merging approach in this research and the MRS algorithm [36] are demonstrated. It can be observed that both the region merging approaches are segmenting buildings accurately and efficiently. However, there are some small boundaries of building regions which still shows under segmentation (US) in the proposed approach and over segmentation (OS) in the MRS algorithm results. These US or OS can be seen in yellow boxes highlighted in Figure 5.9 in image ID 829 and 1027, respectively. The limitation of MRS algorithm is that it is difficult in terms of finding the specific threshold. It has also been observed in MRS approach that during extracting the mean value for colours in order to merge the over segmented regions, the difficulty lies to find the difference between different features as demonstrated in Figure 5.9 last row image ID 1027 Zoomed-In Building ROI having buildings colour similarity. Moreover, to be more specific, the mean value of buildings roof has the correlation to roads mean values due to similarity in colour. Hence it becomes challenging to merge two different objects based on this criterion in MRS approach. However, in the proposed region merging approach, this limitation is minimal because of the MC derivation based on the feature map generated through AttentionU-Net CNN-based deep learning model to perform region merging without any human

intervention. The MC derivation, utilising the feature map generated through the AttentionU-Net CNN-based deep learning model, enables region merging without the need for human intervention. This approach enhances efficiency, accuracy, and, making it valuable for tasks such as buildings delineation in remote sensing images. Finally, the results proves that the proposed region merging approach provides better segmentation results.

5.5 Summary

In this chapter, a region merging approach is proposed by deriving merging criterion (MC) from the experimental results of Chapter 4. This chapter describes the formulation of the MC including colour, texture, shape, and edge. The results of the proposed MC are presented in tables and figures and compare segmentation performance with previous region merging approaches. Finally, the chapter outlined a region merging approach implementation through the utilisation of derived MC.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

In this research work, a region merging approach is proposed to delineate the buildings regions as region of interest (ROI) in the WHU buildings remote sensing images dataset. In this proposed approach, Attention U-Net, a CNN-based deep learning model is used to generate the feature map of the ROI in the images. From this feature map, prominent features of the ROI which are colour, texture, shape, and edge were extracted. These features are then used to derive the merging criterion (MC) using the over segmented regions generated from the SLIC algorithm for merging without any human intervention.

The proposed region merging approach achieved a higher average F-measure of 0.91 in comparison to a similar existing work that required human intervention for merging buildings regions in the WHU images dataset [28] which achieved an average F-measure of 0.63. Furthermore, in comparison to another similar work known as the multiresolution segmentation (MRS) algorithm [36], the proposed approach achieved an average G_s of 0.92 while MRS scored an average G_s of 0.83 for delineating building regions in the same dataset. The higher results of the proposed region merging approach is mainly because of the feature map generated by the AttentionU-Net model. This feature map accurately defines buildings as ROI, allowing the subsequent feature extraction process to select the prominent features for deriving the merging

criterion (MC) to perform the merging without any human intervention. The proposed region merging approach proves best results for building ROI delineation as compared to previous research works [35, 36].

6.2 Limitation and Future Work

In the proposed region merging approach, there are instances where AttentionU-Net CNN-based deep learning model used fails to generate feature map for small buildings regions. As a result, these regions will not be delineated by the proposed region merging approach. This indicates that the feature map produced by the AttentionU-Net model will have a significant impact on the feature extraction process and the merging criterion (MC) derivation for merging the buildings regions in the proposed approach. The proposed region merging approach may not be ideal for real time buildings segmentation in remote sensing images. The limitation is because of the dependence on the AttentionU-Net model which requires extensive training that results in delays in the iterative merging process.

In future, to address the challenges for training the AttentionU-Net CNN-based deep learning model, a transfer learning technique can be developed to generate feature map with a reduced number of training images to minimise delays in the merging process. Additionally, the AttentionU-Net model can be integrated with additional attention mechanisms such as you only look once (YOLO) or ResNet-based networks to enhance its capabilities in generating feature map for small objects regions as ROI in remote sensing

images. By this, the proposed region merging approach shall accurately delineate the ROI.

REFERENCES

[1] S. K. Abdulateef and M. D. Salman, "A Comprehensive Review of Image Segmentation Techniques," *Iraqi Journal for Electrical & Electronic Engineering,* vol. 17, no. 2, pp-166-175 , 2021.

[2] J. Freixenet, X. Munoz, D. Raba, J. Martí, and X. Cufí, "Yet another survey on image segmentation: Region and boundary information integration," *7th European Conference on Computer Vision Copenhagen, Denmark*, pp. 408-422, 2002.

[3] T. Pavlidis and Y.-T. Liow, "Integrating region growing and edge detection," *IEEE transactions on pattern analysis and machine intelligence,* vol. 12, no. 3, pp. 225-233, 1990.

[4] J. Fan, D. K. Yau, A. K. Elmagarmid, and W. G. Aref, "Automatic image segmentation by integrating color-edge extraction and seeded region growing," *IEEE transactions on image processing,* vol. 10, no. 10, pp. 1454-1466, 2001.

[5] M. D. Hossain and D. Chen, "Segmentation for Object-Based Image Analysis (OBIA): A review of algorithms and challenges from remote sensing perspective," *ISPRS Journal of Photogrammetry and Remote Sensing,* vol. 150, pp. 115-134, 2019.

[6] N. Bhargava, P. Trivedi, A. Toshniwal, and H. Swarnkar, "Iterative region merging and object retrieval method using mean shift segmentation and flood fill algorithm," *IEEE Third International Conference on Advances in Computing and Communications*, pp. 157-160, 2013.

[7] I. Kotaridis and M. Lazaridou, "Remote sensing image segmentation advances: A meta-analysis," *ISPRS Journal of Photogrammetry and Remote Sensing,* vol. 173, pp. 309-322, 2021.

[8] J. B. Roerdink and A. Meijster, "The watershed transform: Definitions, algorithms and parallelization strategies," *Fundamenta informaticae,* vol. 41, no. 1-2, pp. 187-228, 2000.

[9] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *International journal of computer vision,* vol. 59, pp. 167-181, 2004.

[10] P. Neubert and P. Protzel, "Compact watershed and preemptive slic: On improving trade-offs of superpixel segmentation algorithms," in *2014 22nd international conference on pattern recognition*, 2014: IEEE, pp. 996-1001.

[11] L. Grady, "Random walks for image segmentation," *IEEE transactions on pattern analysis and machine intelligence,* vol. 28, no. 11, pp. 1768-1783, 2006.

[12] K. Fukunaga and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *IEEE Transactions on information theory,* vol. 21, no. 1, pp. 32-40, 1975.

[13] A. Vedaldi and S. Soatto, "Quick shift and kernel methods for mode seeking," *10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008,* pp. 705-718, 2008.

[14] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE*

transactions on pattern analysis and machine intelligence, vol. 34, no. 11, pp. 2274-2282, 2012.

[15] A. P. Moore, S. J. Prince, J. Warrell, U. Mohammed, and G. Jones, "Superpixel lattices," *IEEE conference on computer vision and pattern recognition*, pp. 1-8, 2008.

[16] K. L. Goh, G. W. Ng, M. Hamzah, and S. S. Chai, "Effects of Different Superpixel Algorithms on Interactive Segmentations," *International Journal of Circuits, Systems and Signal Processing,* vol. 16, pp. 1084-1092, 2022.

[17] J. Martins, D. A. Sant'Ana, J. M. Junior, H. Pistori, and W. N. Gonçalves, "Aerial image segmentation in urban environment for vegetation monitoring," *IEEE Latin American GRSS & ISPRS Remote Sensing Conference (LAGIRS)*, pp. 375-379, 2020.

[18] S. Patel and B. Kadhiwala, "Comparative analysis of cluster based superpixel segmentation techniques," *IEEE 2nd International Conference on Trends in Electronics and Informatics (ICOEI)*, pp. 1454-1459, 2018.

[19] A. Z. Zhang, G. Y. Sun, S. H. Liu, Z. J. Wang, P. Wang, and J. S. Ma, "Multi-scale segmentation of very high resolution remote sensing image based on gravitational field and optimized region merging," *Multimedia Tools and Applications,* vol. 76, pp. 15105-15122, 2017.

[20] T. He, J. Chen, L. Kang, and Q. Zhu, "Evaluation of Global-Scale and Local-Scale Optimized Segmentation Algorithms in GEOBIA with SAM on Land Use and Land Cover," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing,* vol 17, pp. 6721-6738, 2024.

[21] D. Liu and F. Xia, "Assessing object-based classification: advantages and limitations. Remote Sens Lett 1 (4): 187–194," ed, 2010.

[22] X. Zhang, P. Xiao, X. Feng, J. Wang, and Z. Wang, "Hybrid region merging method for segmentation of high-resolution remote sensing images," *ISPRS Journal of Photogrammetry and Remote Sensing,* vol. 98, pp. 19-28, 2014.

[23] Y. Wang, Q. Meng, Q. Qi, J. Yang, and Y. Liu, "Region merging considering within-and between-segment heterogeneity: An improved hybrid remote-sensing image segmentation method," *Remote Sensing,* vol. 10, no. 5, p. 781, 2018.

[24] H. Wang *et al.*, "Improvement of region-merging image segmentation accuracy using multiple merging criteria," *Remote Sensing,* vol. 13, no. 14, p. 2782, 2021.

[25] Z. Khan and J. Yang, "Bottom-up unsupervised image segmentation using FC-Dense u-net based deep representation clustering and multidimensional feature fusion based region merging," *Image and Vision Computing,* vol. 94, p. 103871, 2020.

[26] X. Lv, C. Persello, W. Li, X. Huang, D. Ming, and A. Stein, "DeepMerge: Deep-Learning-Based Region-Merging for Image Segmentation," *arXiv preprint arXiv:2305.19787,* 2023.

[27] X. Zhang, P. Xiao, X. Song, and J. She, "Boundary-constrained multi-scale segmentation method for remote sensing images," *ISPRS Journal of Photogrammetry and Remote Sensing,* vol. 78, pp. 15-25, 2013.

[28] M. D. Hossain and D. Chen, "A hybrid image segmentation method for building extraction from high-resolution RGB images," *ISPRS Journal of Photogrammetry and Remote Sensing,* vol. 192, pp. 299-314, 2022.

[29] J. R. Beveridge, J. Griffith, R. R. Kohler, A. R. Hanson, and E. M. Riseman, "Segmenting images using localized histograms and region merging," *International Journal of Computer Vision,* vol. 2, pp. 311-347, 1989.

[30] P. J. Salembier Clairon, L. Garrido Ostermann, and D. Garcia, "Image sequence analysis and merging," *International Workshop on Coding Techniques for Very Low Bit-rate Video: Linköping, Sweden,* LINKÖPING UNIVERSITY, pp. 1-8, 1997.

[31] T. Su and S. Zhang, "Local and global evaluation for remote sensing image segmentation," *ISPRS Journal of Photogrammetry and Remote Sensing,* vol. 130, pp. 256-276, 2017.

[32] Y. Liu *et al.*, "Discrepancy measures for selecting optimal combination of parameter values in object-based image analysis," *ISPRS journal of photogrammetry and remote sensing,* vol. 68, pp. 144-156, 2012.

[33] T. Su, T. Liu, S. Zhang, Z. Qu, and R. Li, "Machine learning-assisted region merging for remote sensing image segmentation," *ISPRS Journal of Photogrammetry and Remote Sensing,* vol. 168, pp. 89-123, 2020.

[34] R. Nock and F. Nielsen, "Statistical region merging," *IEEE Transactions on pattern analysis and machine intelligence,* vol. 26, no. 11, pp. 1452-1458, 2004.

[35] R. Adams and L. Bischof, "Seeded region growing," *IEEE Transactions on pattern analysis and machine intelligence,* vol. 16, no. 6, pp. 641-647, 1994.

[36] M. Baatz, "Multiresolution segmentation: an optimization approach for high quality multi-scale image segmentation," *Angewandte geographische informationsverarbeitung,* pp. 12-23, 2000.

[37] S. Ji, S. Wei, and M. Lu, "Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set," *IEEE Transactions on geoscience and remote sensing,* vol. 57, no. 1, pp. 574-586, 2018.

[38] C. Wang, W. Xu, X.-f. Pei, and X.-y. Zhou, "An unsupervised multi-scale segmentation method based on automated parameterization," *Arabian Journal of Geosciences,* vol. 9, pp. 1-10, 2016.

[39] J. Schiewe, "Segmentation of high-resolution remotely sensed data-concepts, applications and problems," *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences,* vol. 34, no. 4, pp. 380-385, 2002.

[40] E. A. Zanaty and A. S. Ghiduk, "A novel approach based on genetic algorithms and region growing for magnetic resonance image (MRI) segmentation," *Computer Science and Information Systems,* vol. 10, no. 3, pp. 1319-1342, 2013.

[41] W. Deng, W. Xiao, H. Deng, and J. Liu, "MRI brain tumor segmentation with region growing method based on the gradients and variances along and inside of the boundary curve," *IEEE 3rd International Conference on Biomedical Engineering and Informatics*, vol. 1, pp. 393-396, 2010.

[42] C. Huang, Q. Liu, and X. Li, "Color image segmentation by seeded region growing and region merging," *IEEE Seventh International Conference on Fuzzy Systems and Knowledge Discovery*, vol. 2, pp. 533-536, 2010.

[43] A. K. Dubey, R. Arora, and A. Ahmed, "A comparative review of various segmentation methods and its application," *IEEE 6th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)*, pp. 645-650, *2017*.

[44] W. Cui, Z. Guan, and Z. Zhang, "An improved region growing algorithm for image segmentation," *IEEE International Conference on Computer Science and Software Engineering*, vol. 6, pp. 93-96, 2008.
[45] R. Dehdasht-Heydari and S. Gholami, "Automatic seeded region growing (ASRG) using genetic algorithm for brain MRI segmentation," *Wireless Personal Communications,* vol. 109, no. 2, pp. 897-908, 2019.
[46] Q. Sun, R. Zheng, and B. Xu, "Bayesian-optimized unsupervised semantic segmentation model for structural crack detection," *Advances in Structural Engineering,* vol. 27, no. 3, pp. 432-449, 2024.
[47] B. Fulkerson and S. Soatto, "Really quick shift: Image segmentation on a GPU," *ECCV 2010 Workshops, Heraklion, Crete, Greece,* pp. 350-358, 2012.
[48] S. Mohammadi and M. Allali, "Advancing Brain Tumor Segmentation with Spectral–Spatial Graph Neural Networks," *Applied Sciences,* vol. 14, no. 8, p. 3424, 2024.
[49] S. Beucher and F. Meyer, "The morphological approach to segmentation: the watershed transformation," *Mathematical morphology in image processing*: CRC Press, pp. 433-481, 2018.
[50] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on pattern analysis and machine intelligence,* vol. 24, no. 5, pp. 603-619, 2002.
[51] J. Papon, A. Abramov, M. Schoeler, and F. Worgotter, "Voxel cloud connectivity segmentation-supervoxels for point clouds," *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2027-2034, 2013.
[52] N. Agrawal and S. Aurelia, "A Review on Segmentation of Vitiligo image," *Materials Science and Engineering*, vol. 1131, no. 1, p. 012003, 2021.
[53] S. Kaur and R. Bansal, "Comparative analysis of superpixel segmentation methods," *Int. J. Eng. Technol. Manag. Res,* vol. 5, pp. 1-9, 2018.
[54] K. V. Abramov, P. V. Skribtsov, and P. A. Kazantsev, "Image segmentation method selection for vehicle detection using unmanned aerial vehicle," *Modern Applied Science,* vol. 9, no. 5, p. 295, 2015.
[55] A. S. Sahadevan, "Extraction of spatial-spectral homogeneous patches and fractional abundances for field-scale agriculture monitoring using airborne hyperspectral images," *Computers and Electronics in Agriculture,* vol. 188, p. 106325, 2021.
[56] U. Nazir, H. Wang, and M. Taj, "Survey of image based graph neural networks," *arXiv preprint arXiv:2106.06307,* 2021.
[57] S. Hojjatoleslami and J. Kittler, "Region growing: a new approach," *IEEE Transactions on Image processing,* vol. 7, no. 7, pp. 1079-1084, 1998.
[58] S. Yuheng and Y. Hao, "Image segmentation algorithms overview," *arXiv preprint arXiv:1707.02051,* 2017.
[59] N. Shrivastava and J. Bharti, "Automatic seeded region growing image segmentation for medical image segmentation: a brief review," *International Journal of Image and Graphics,* vol. 20, no. 03, p. 2050018, 2020.
[60] P.-L. Shui and Z.-J. Zhang, "Fast SAR image segmentation via merging cost with relative common boundary length penalty," *IEEE Transactions on Geoscience and Remote Sensing,* vol. 52, no. 10, pp. 6434-6448, 2014.
[61] A. Mehnert and P. Jackway, "An improved seeded region growing algorithm," *Pattern Recognition Letters,* vol. 18, no. 10, pp. 1065-1071, 1997.

[62] Z. Lin, J. Jin, and H. Talbot, "Unseeded Region Growing for 3D Image Segmentation," in *ACM International Conference Proceeding Series*, 2000, vol. 9: Citeseer, pp. 31-37.

[63] J. Fan, G. Zeng, M. Body, and M.-S. Hacid, "Seeded region growing: an extensive and comparative study," *Pattern recognition letters,* vol. 26, no. 8, pp. 1139-1156, 2005.

[64] F. Y. Shih and S. Cheng, "Automatic seeded region growing for color image segmentation," *Image and vision computing,* vol. 23, no. 10, pp. 877-886, 2005.

[65] L. Vincent and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," *IEEE Transactions on Pattern Analysis & Machine Intelligence,* vol. 13, no. 06, pp. 583-598, 1991.

[66] S. Beucher, "Use of watersheds in contour detection," in *Proc. Int. Workshop on Image Processing, Sept. 1979*, 1979, pp. 17-21.

[67] N. Amoda and R. K. Kulkarni, "Image segmentation and detection using watershed transform and region based image retrieval," *Int. J. Emerg. Trends & Techno. Comp. Sci,* vol. 2, pp. 89-94, 2013.

[68] F. Meyer, "Color image segmentation," in *1992 international conference on image processing and its applications*, 1992: IET, pp. 303-306.

[69] H. Palus, "Color image segmentation: selected techniques," in *Color Image Processing*: CRC Press, 2018, pp. 123-148.

[70] M. A. Carreira-Perpinán, "A review of mean-shift algorithms for clustering," *arXiv preprint arXiv:1503.00687,* 2015.

[71] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "Slic superpixels," *EPFL Technical Report* 149300, 2010.

[72] A. Sarwar, A. A. Sheikh, J. Manhas, and V. Sharma, "Segmentation of cervical cells for automated screening of cervical cancer: a review," *Artificial Intelligence Review,* vol. 53, pp. 2341-2379, 2020.

[73] X. Shen, Y. Guo, and J. Cao, "Object-based multiscale segmentation incorporating texture and edge features of high-resolution remote sensing images," *PeerJ Computer Science,* vol. 9, p. e1290, 2023.

[74] K. Haris, S. N. Efstratiadis, N. Maglaveras, and A. K. Katsaggelos, "Hybrid image segmentation using watersheds and fast region merging," *IEEE Transactions on image processing,* vol. 7, no. 12, pp. 1684-1699, 1998.

[75] D. Xiang *et al.*, "Adaptive statistical superpixel merging with edge penalty for PolSAR image segmentation," *IEEE Transactions on Geoscience and Remote Sensing,* vol. 58, no. 4, pp. 2412-2429, 2019.

[76] S. Fan, Y. Sun, and P. Shui, "Region-merging method with texture pattern attention for SAR image segmentation," *IEEE Geoscience and Remote Sensing Letters,* vol. 18, no. 1, pp. 112-116, 2020.

[77] S. Ahmed, Z. Khan, and A. Jain, "Image Segmentation by using Mean-Shift with Dynamic Region Merging-A Survey," *Image,* vol. 1, no. 7, 2012.

[78] S. E. Hernandez, K. E. Barner, and Y. Yuan, "Region merging using homogeneity and edge integrity for watershed-based image segmentation," *Optical engineering,* vol. 44, no. 1, pp. 017004-017004-14, 2005.

[79] W. K. Mutlag, S. K. Ali, Z. M. Aydam, and B. H. Taher, "Feature extraction methods: a review," in *Journal of Physics: Conference Series*, vol. 1591, no. 1: IOP Publishing, p. 012028, 2020.

[80] Z. Alqadi, M. Khrisat, A. Hindi, and M. Dwairi, "Features Analysis of RGB Color Image based on Wavelet Packet Information," *International Journal of Computer Science and Mobile Computing,* vol. 9, no. 3, pp. 149-156, 2020.

[81] Y. Wicaksono, R. Wahono, and V. Suhartono, "Color and texture feature extraction using gabor filter-local binary patterns for image segmentation with fuzzy C-means," *Journal of Intelligent Systems,* vol. 1, no. 1, pp. 15-21, 2015.

[82] R. Kusumaningrum and A. M. Arymurthy, "Color and texture feature for remote sensing-image retrieval system: a comparative study," *International Journal of Computer Science Issues (IJCSI),* vol. 8, no. 5, p. 125, 2011.

[83] P. M. B. Ong and E. R. Punzalan, "Comparative analysis of RGB and HSV color models in extracting color features of green dye solutions," *DLSU Research Congress*, pp. 1500-20, 2014.

[84] K. Erdogan and N. Yilmaz, "Shifting colors to overcome not realizing objects problem due to color vision deficiency," *2nd Int. Conf. on Advances in Computing, Electronics and Electrical Technology-CEET*, pp. 11-14, 2014.

[85] S. G. A. Usha and S. Vasuki, "Significance of texture features in the segmentation of remotely sensed images," *Optik,* vol. 249, p. 168241, 2022.

[86] A. K. Jain and F. Farrokhnia, "Unsupervised texture segmentation using Gabor filters," *Pattern recognition,* vol. 24, no. 12, pp. 1167-1186, 1991.

[87] A. Gorai, K. Cetina, L. Baumela, and A. Ghosh, "A comparative study of local binary pattern descriptors and Gabor Filter for electron microscopy image segmentation," *IEEE 2014 International Conference on Parallel, Distributed and Grid Computing*, pp. 76-81, 2014.

[88] Y. Mingqiang, K. Kidiyo, and R. Joseph, "A survey of shape feature extraction techniques," *Pattern recognition,* vol. 15, no. 7, pp. 43-90, 2008.

[89] R. Rana and A. Verma, "Comparison and enhancement of digital image by using canny filter and sobel filter," *IOSR Journal of Computer Engineering,* vol. 16, no. 1, pp. 06-10, 2014.

[90] P. P. Acharjya, R. Das, and D. Ghoshal, "Study and comparison of different edge detectors for image segmentation," *Global journal of computer science and technology,* vol. 12, no. 13, pp. 28-32, 2012.

[91] A. S. Ahmed, "Comparative study among Sobel, Prewitt and Canny edge detection operators used in image processing," *J. Theor. Appl. Inf. Technol,* vol. 96, no. 19, pp. 6517-6525, 2018.

[92] Z.-H. Zhan, J.-Y. Li, and J. Zhang, "Evolutionary deep learning: A survey," *Neurocomputing,* vol. 483, pp. 42-58, 2022.

[93] S. Dargan, M. Kumar, M. R. Ayyagari, and G. Kumar, "A survey of deep learning and its applications: a new paradigm to machine learning," *Archives of Computational Methods in Engineering,* vol. 27, pp. 1071-1092, 2020.

[94] Y. Zhang, S. Mehta, and A. Caspi, "Rethinking semantic segmentation evaluation for explainability and model selection," *arXiv preprint arXiv:2101.08418,* 2021.

[95] L. Luo, P. Li, and X. Yan, "Deep learning-based building extraction from remote sensing images: A comprehensive review," *Energies,* vol. 14, no. 23, p. 7982, 2021.

[96] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *Proceedings of 18th international*

conference in Medical image computing and computer-assisted intervention–MICCAI, Munich, Germany, pp. 234-241, 2015.

[97] P. Ps and B. H. Aithal, "Building footprint extraction from very high-resolution satellite images using deep learning," *Journal of Spatial Science,* vol. 68, no. 3, pp. 487-503, 2023.

[98] P. Guo, X. Su, H. Zhang, M. Wang, and F. Bao, "A multi-scaled receptive field learning approach for medical image segmentation," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1414-1418, *2020*.

[99] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," *fourth international conference on 3D vision (3DV)*, pp. 565-571, 2016.

[100] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence,* vol. 39, no. 12, pp. 2481-2495, 2017.

[101] U. Kamal, T. I. Tonmoy, S. Das, and M. K. Hasan, "Automatic traffic sign detection and recognition using SegU-Net and a modified Tversky loss function with L1-constraint," *IEEE Transactions on Intelligent Transportation Systems,* vol. 21, no. 4, pp. 1467-1479, 2019.

[102] Z. Zhang, Q. Liu, and Y. Wang, "Road extraction by deep residual u-net," *IEEE Geoscience and Remote Sensing Letters,* vol. 15, no. 5, pp. 749-753, 2018.

[103] N. Ibtehaz and M. S. Rahman, "MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation," *Neural networks,* vol. 121, pp. 74-87, 2020.

[104] O. Oktay, "Attention u-net: Learning where to look for the pancreas," *arXiv preprint arXiv:1804.03999,* 2018.

[105] P. Dias, Y. Tian, S. Newsam, A. Tsaris, J. Hinkle, and D. Lunga, "Model assumptions and data characteristics: Impacts on domain adaptation in building segmentation," *IEEE Transactions on Geoscience and Remote Sensing,* vol. 60, pp. 1-18, 2022.

[106] G. Zhang, W. Li, H. Dong, and G. Gui, "High Spatial Resolution Remote Sensing Classification with Lightweight CNN Using Dilated Convolution," *14th EAI International Conference* in *Mobile Multimedia Communications: Mobimedia, Virtual Event,* pp. 757-767, 2021.

[107] A. Uslucuk and H. Öcal, "Comparative Analysis of Baseline Vnet and Unet Architectures on Pancreas Segmentation," *Orclever Proceedings of Research and Development,* vol. 3, no. 1, pp. 146-157, 2023.

[108] M. Alam, J.-F. Wang, C. Guangpei, L. Yunrong, and Y. Chen, "Convolutional neural network for the semantic segmentation of remote sensing images," *Mobile Networks and Applications,* vol. 26, pp. 200-215, 2021.

[109] V. Khryashchev, L. Ivanovsky, V. Pavlov, A. Ostrovskaya, and A. Rubtsov, "Comparison of different convolutional neural network architectures for satellite image segmentation," *IEEE 2018 23rd conference of open innovations association (FRUCT)*, pp. 172-179, 2018.

[110] A. K. Srivastava, M. Shayan, S. Gupta, and M. Dixit, "Multiple Features Extraction from High Resolution Multi-Spectral Satellite Images using Deep Learning Technique," *4th International Conference on Advances in*

Computing, Communication Control and Networking (ICAC3N), pp. 1015-1019, 2022.

[111] A. Abdollahi, B. Pradhan, and A. Alamri, "VNet: An end-to-end fully convolutional neural network for road extraction from high-resolution remote sensing data," *Ieee Access,* vol. 8, pp. 179424-179436, 2020.

[112] B. Sariturk, D. Kumbasar, and D. Z. Seker, "Comparative analysis of different CNN models for building segmentation from satellite and UAV images," *Photogrammetric Engineering & Remote Sensing,* vol. 89, no. 2, pp. 97-105, 2023.

[113] A. Abdollahi, B. Pradhan, and A. M. Alamri, "An ensemble architecture of deep convolutional Segnet and Unet networks for building semantic segmentation from high-resolution aerial images," *Geocarto International,* vol. 37, no. 12, pp. 3355-3370, 2022.

[114] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778, 2016.

[115] A. Abdollahi and B. Pradhan, "Integrating semantic edges and segmentation information for building extraction from aerial images using UNet," *Machine Learning with Applications,* vol. 6, p. 100194, 2021.

[116] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818-2826, 2016.

[117] A. Temenos, N. Temenos, A. Doulamis, and N. Doulamis, "On the exploration of automatic building extraction from RGB satellite images using deep learning architectures based on U-Net," *Technologies,* vol. 10, no. 1, p. 19, 2022.

[118] http://gpcv.whu.edu.cn."building_dataset." http://gpcv.whu.edu.cn/data/building_dataset.html

[119] J. Yang, Y. He, and J. Caspersen, "Region merging using local spectral angle thresholds: A more accurate method for hybrid segmentation of remote sensing images," *Remote sensing of environment,* vol. 190, pp. 137-148, 2017.

[120] B. Peng, L. Zhang, and D. Zhang, "Automatic image segmentation by dynamic region merging," *IEEE Transactions on image processing,* vol. 20, no. 12, pp. 3592-3605, 2011.

[121] M. Abadi *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467,* 2016.

[122] "“Keras: Deep Learning for humans.” " https://keras.io/.

[123] S. Van der Walt *et al.*, "scikit-image: image processing in Python," *PeerJ,* vol. 2, p. e453, 2014.

[124] R. H. Van Leuken, L. Garcia, X. Olivares, and R. Van Zwol, "Visual diversification of image search results," *Proceedings of the 18th international conference on World wide web*, pp. 341-350, 2009.

[125] J. Pont-Tuset and F. Marques, "Supervised evaluation of image segmentation and object proposal techniques," *IEEE transactions on pattern analysis and machine intelligence,* vol. 38, no. 7, pp. 1465-1478, 2015.

[126] D. Xiang *et al.*, "Fast pixel-superpixel region merging for SAR image segmentation," *IEEE Transactions on Geoscience and Remote Sensing,* vol. 59, no. 11, pp. 9319-9335, 2020.

[127] T. Bakirman, I. Komurcu, and E. Sertel, "Comparative analysis of deep learning based building extraction methods with the new VHR Istanbul dataset," *Expert Systems with Applications,* vol. 202, p. 117346, 2022.
[128] X. Zhang, X. Feng, P. Xiao, G. He, and L. Zhu, "Segmentation quality evaluation using region-based precision and recall measures for remote sensing images," *ISPRS Journal of Photogrammetry and Remote Sensing,* vol. 102, pp. 73-84, 2015.
[129] J. Zhang *et al.*, "Segmenting purple rapeseed leaves in the field from UAV RGB imagery using deep learning as an auxiliary means for nitrogen stress detection," *Remote Sensing,* vol. 12, no. 9, p. 1403, 2020.
[130] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey," *IEEE transactions on pattern analysis and machine intelligence,* vol. 44, no. 7, pp. 3523-3542, 2021.
[131] F. Pedregosa, "Scikit-learn: Machine learning in python Fabian," *Journal of machine learning research,* vol. 12, p. 2825, 2011.
[132] J. Tian and D. M. Chen, "Optimization in multi-scale segmentation of high-resolution satellite images for artificial feature recognition," *International Journal of Remote Sensing,* vol. 28, no. 20, pp. 4625-4644, 2007.
[133] N. Kanopoulos, N. Vasanthavada, and R. L. Baker, "Design of an image edge detection filter using the Sobel operator," *IEEE Journal of solid-state circuits,* vol. 23, no. 2, pp. 358-367, 1988.
[134] Y. Yuan, Z. Zhu, H. Yu, and W. Zhang, "Watershed-based superpixels with global and local boundary marching," *IEEE Transactions on Image Processing,* vol. 29, pp. 7375-7388, 2020.
[135] N. Liao, B. Guo, C. Li, H. Liu, and C. Zhang, "BACA: Superpixel segmentation with boundary awareness and content adaptation," *Remote Sensing,* vol. 14, no. 18, p. 4572, 2022.
[136] D. Stutz, A. Hermans, and B. Leibe, "Superpixels: An evaluation of the state-of-the-art," *Computer Vision and Image Understanding,* vol. 166, pp. 1-27, 2018.
[137] A. Khalel and M. El-Saban, "Automatic pixelwise object labeling for aerial imagery using stacked u-nets," *arXiv preprint arXiv:1803.04953,* 2018.
[138] A. M. Ayalew, A. O. Salau, B. T. Abeje, and B. Enyew, "Detection and classification of COVID-19 disease from X-ray images using convolutional neural networks and histogram of oriented gradients," *Biomedical Signal Processing and Control,* vol. 74, p. 103530, 2022.
[139] D. P. Kingma, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980,* 2014.
[140] L. Bottou, "Large-scale machine learning with stochastic gradient descent," *19th International Conference on Computational StatisticsParis France,* pp. 177-186, 2010.
[141] D. Abriha and S. Szabó, "Strategies in training deep learning models to extract building from multisource images with small training sample sizes," *International Journal of Digital Earth,* vol. 16, no. 1, pp. 1707-1724, 2023.
[142] Y. Cai *et al.*, "A comparative study of deep learning approaches to rooftop detection in aerial images," *Canadian Journal of Remote Sensing,* vol. 47, no. 3, pp. 413-431, 2021.

[143] Y. Yi, Z. Zhang, W. Zhang, C. Zhang, W. Li, and T. Zhao, "Semantic segmentation of urban buildings from VHR remote sensing imagery using a deep convolutional neural network," *Remote sensing,* vol. 11, no. 15, p. 1774, 2019.
[144] "NumPy documentation—NumPy v1.26 Manual" https://numpy.org/doc/stable/.
[145] T. E. Oliphant, *Guide to numpy*. Trelgol Publishing USA, 2006.
[146] M. A. Stricker and M. Orengo, "Similarity of color images," *Storage and retrieval for image and video databases*, vol. 2420: SPiE, pp. 381-392, 1995.
[147] Y. Kurmi, V. Chaurasia, and N. Kapoor, "Design of a histopathology image segmentation algorithm for CAD of cancer," *Optik,* vol. 218, p. 164636, 2020.
[148] S. Ashtputre, M. Sharma, and M. Ramaiya, "Image segmentation based on moments and color map," *IEEE 2013 International Conference on Communication Systems and Network Technologies*, pp. 133-136, 2013.
[149] https://scikit-image.org."skimage.color."https://scikit-image.org/docs/stable/api/skimage.color.html.
[150] https://scikit-image.org."skimage.measure."https://scikit-image.org/docs/stable/api/skimage.measure.html.
[151] https://docs.scipy.org."scipy.stats.skew." https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.skew.html
[152] https://docs.scipy.org."scipy.stats.kurtosis." https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.kurtosis.html.
[153] M. D. Hossain and D. Chen, "Target-based Building Extraction from High-Resolution RGB Imagery using GEOBIA Framework and Tabular Deep Learning Model," *Geomatica,* p. 100007, 2024.
[154] https://scikit-image.org."skimage.filters.gabor."https://scikit-image.org/docs/stable/api/skimage.filters.html#skimage.filters.gabor.
[155] https://scikit-image.org."skimage.feature.local_binary_pattern." https://scikit-image.org/docs/stable/api/skimage.feature.html#skimage.feature.local_binary_pattern.
[156] [https://scikit-image.org."edges/plot_canny."[https://scikit-image.org/docs/stable/auto_examples/edges/plot_canny.html.
[157] https://scikit-image.org,"edges/plot_edge_filter."https://scikit-image.org/docs/stable/auto_examples/edges/plot_edge_filter.html.
[158] S. K. Dey, K. M. M. Uddin, H. M. H. Babu, M. M. Rahman, A. Howlader, and K. A. Uddin, "Chi2-MI: A hybrid feature selection based machine learning approach in diagnosis of chronic kidney disease," *Intelligent Systems with Applications,* vol. 16, p. 200144, 2022.
[159] R. Adipranata and G. S. Budhi, "Java characters recognition using evolutionary neural network and combination of Chi2 and backpropagation neural network," *International Journal of Applied Engineering Research,* vol. 9, no. 22, pp. 15395-15406, 2014.
[160] H. Xu and L. Xu, "Multi-label feature selection algorithm based on label pairwise ranking comparison transformation," *IEEE 2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 1210-1217, 2017.
[161] Y. Wang, Q. Qi, Y. Liu, L. Jiang, and J. Wang, "Unsupervised segmentation parameter selection using the local spatial statistics for remote sensing image

segmentation," *International Journal of Applied Earth Observation and Geoinformation,* vol. 81, pp. 98-109, 2019.

[162] https://scikit-learn.org,"sklearn.feature_selection.chi2."https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.chi2.html.

[163] D. Feng, Z. Zhang, and K. Yan, "A semantic segmentation method for remote sensing images based on the Swin transformer fusion Gabor filter," *IEEE Access,* vol. 10, pp. 77432-77451, 2022.

[164] K. Li, W. Tao, X. Liu, and L. Liu, "Iterative image segmentation with feature driven heuristic four-color labeling," *Pattern Recognition,* vol. 76, pp. 69-79, 2018.

LIST OF PUBLICATION

1) Shoaib, A., Vadiveloo, M. and Lim, S.P., 2024. Comparative Studies of Region-Based Segmentation Algorithms on Natural and Remote Sensing Images. In *ITM Web of Conferences* (Vol. 67, p. 01048). EDP Sciences. https://doi.org/10.1051/itmconf/20246701048

2) Shoaib, A., Vadiveloo, M., Lim, S.P. (2024). Performance Comparison of Convolutional Neural Network Deep Learning Architectures for Remote Sensing Image Segmentation. In: Thiruchelvam, V., Alfred, R., Ismail, Z.I.B.A., Haviluddin, H., Baharum, A. (eds) Proceedings of the 4th International Conference on Advances in Computational Science and Engineering. ICACSE 2023. Lecture Notes in Electrical Engineering, vol 1199. Springer, Singapore. https://doi.org/10.1007/978-981-97-2977-7_12

APPENDIX A

Table 5.1 Results of Three Features as MC for parameter k in SLIC value 500, 1000, 1500, 2000, 3000, respectively

Input Image No.	OS Value of parameter k for SLIC Algorithm	Total Number of Regions Generated	Evaluation Metrics					
			F-Measure ↑	precision ↑	recall ↑	Adjusted Rand Index (ARI) ↑	Adapted Rand Error (ARE) ↓	Variation of Information (VOI) ↓
101	500	250	0.3944	0.8282	0.2588	0.6501	0.6056	1.4231
	1000	612	0.6235	0.8076	0.5077	0.7607	0.3765	1.1186
	1500	940	0.6864	0.8333	0.5835	0.7961	0.3136	0.9818
	2000	1421	0.7772	0.8582	0.7103	0.8312	0.2228	0.8851
	3000	**2165**	**0.8465**	**0.8930**	**0.8046**	**0.8632**	**0.1535**	**0.7594**
829	500	234	0.5748	0.4387	0.8333	0.7054	0.4252	0.2928
	1000	554	0.8826	0.7963	0.9900	0.8571	0.1174	0.2015
	1500	902	0.9407	0.8936	0.9931	0.9166	0.0593	0.1294
	2000	1324	0.9575	0.9238	0.9938	0.9398	0.0425	0.0942
	3000	**2068**	**0.9865**	**0.9742**	**0.9991**	**0.9553**	**0.0135**	**0.0781**
1027	500	303	0.8981	0.8403	0.9646	0.8852	0.1019	0.3044
	1000	**661**	**0.9548**	**0.9228**	0.9890	0.9154	**0.0452**	0.2443
	1500	1027	0.9536	0.9167	**0.9937**	0.9284	0.0464	0.2162
	2000	1538	0.9330	0.8831	0.9888	0.9245	0.0670	0.2236
	3000	2324	0.9423	0.8990	0.9900	**0.9307**	0.0577	**0.2083**

Table 5.2 Results for Four features as MC for parameter k value 500, 1000, 1500, 2000, 3000, respectively

Input Image No.	OS Value of parameter k for SLIC Algorithm	Total Number of Regions Generated	Evaluation Metrics					
			F-Measure ↑	precision ↑	recall ↑	Adjusted Rand Index (ARI) ↑	Adapted Rand Error (ARE) ↓	Variation of Information (VOI) ↓
101	500	250	0.8781	0.8996	0.8587	0.7295	0.1442	0.6210
	1000	612	0.8799	0.8998	0.8610	0.7791	0.1255	0.5729
	1500	940	0.8850	0.9015	0.8693	0.8049	0.1098	0.4601
	2000	1421	0.8950	0.9097	0.8810	0.8610	0.0990	0.3919
	3000	**2165**	**0.9040**	**0.9184**	**0.8902**	**0.9102**	**0.0873**	**0.3592**
829	500	234	0.7978	0.7510	0.8509	0.8195	0.0971	0.0886
	1000	554	0.9359	0.9017	0.9731	0.8850	0.0496	0.0690
	1500	902	0.9548	0.9296	0.9815	0.9279	0.0310	0.0513
	2000	1324	0.9781	0.9611	0.9958	0.9547	0.0199	0.0309
	3000	**2068**	**0.9990**	**0.9985**	**0.9996**	**0.9781**	**0.0127**	**0.0301**
1027	500	303	0.9296	0.8901	0.9730	0.9501	0.0891	0.1792
	1000	661	0.9569	0.9351	0.9798	0.9599	0.0919	0.1401
	1500	1027	0.9804	0.9697	0.9919	0.9700	**0.0397**	**0.0912**
	2000	1538	0.9603	0.9410	0.9799	0.9781	0.0401	0.1001
	3000	**2324**	**0.9825**	**0.9704**	**0.9951**	**0.9893**	0.0399	0.1197