Fully Connected Neural Networks for Semantic Segmentation of High-Resolution Earth Observation Imagery for Building Rooftop Extraction

by

Zijian Jiang

A research paper
presented to the University of Waterloo
in fulfillment of the
research paper requirement for the degree of
Master of Environmental Studies
in
Geography

Waterloo, Ontario, Canada, 2021

Author's Declaration

I hereby declare that I am the sole author of this research paper. This is a true copy of the research paper, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

ABSTRACT

Building rooftop extraction from high-resolution earth observation imagery is the leading research direction for earth observation image comprehension and object detection. Automatic and accurate building extraction from earth observation images is of important application value and practical significance for the acquisition and update of basic geographical data, such as urban planning, demographic census, site localization, disaster management, etc. As the neural convolutional network has been introduced into earth observations, deep learning has shown great potential in image classification and target detection. Since 2015, a variant CNN architecture has been developed and widely used for semantic segmentation, in which category tags are assigned to all pixels in an image. These architectures can be collectively referred to as fully convolutional neural networks (FCN). In this research, the deep learning models U-Net, SegNet, RefineNet, DeepLabV3+, Pyramid Attention Network (PAN), and Bilateral Segmentation Network (BiSeNet) are trained for building extraction from high-resolution earth observation imagery. Then the state-of-art models trained on the recently proposed Waterloo Building Dataset (WBD) were evaluated by comparing the Overall Accuracy (OA), IoU, mIoU, Precision, Recall, *F*1-*score,* and testing rate (FPS). Experiments show that the accuracy of DeepLabV3+ with ResNet50 as the pre-training backbone was significantly enhanced compared to other state-of-the-art semantic segmentation models with the highest scores of 98.21%, 88.34%, 76.96%, 87.3%, and 90.01%, respectively by OA, Recall, IoU, mIoU, and F1−score. The subsequent conclusion discussed the future implementation of deep learning in the field of automatic building extraction, especially strategies of training the deep learning models. Meanwhile, also indicated the limitations and possible improvements of this experiment.

Keywords: Building Extraction, U-Net, SegNet, RefineNet, DeepLabV3+, Pyramid Attention Network (PAN), BiSeNet, earth observation imagery, building footprint

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

ASPP	Atrous Spatial Pyramid Pooling
BiSeNet	Bilateral Segmentation Network
CNN	Convolutional Neural Network
DCNN	Deep Convolutional Neural Network
DL	Deep Learning
FCN	Fully Convolutional Networks
FN	False Negative
FP	False Positive
FPA	Feature Pyramid Attention module
GAN	Generative Adversarial Network
GAU	Global Attention Upsample module
ILSVRC	ImageNet Large Scale Visual Recognition
OA	Overall Accuracy
PAN	Pyramid Attention Network
PASCAL VOC	PASCAL Visual Object Classes
ResNet	Deep Residual Net
RNN	Recursive Neural Networks
RS	Remote Sensing
TN	True Negative
TP	True Positive
VGG	Visual Geometry Group
WBD	Waterloo Building Dataset
WHU	Wuhan University

1. INTRODUCTION

1.1. Motivation

As an indispensable part of human production and life, buildings not only provide living space for human beings and are also the main component of cities, revealing the natural development of urban culture. With the continuous expansion of cities, scientists have paid great attention to studying the spatial-temporal distribution of urban buildings from images of earth observation. Building rooftop extraction from high-resolution earth observation imagery is the leading research direction for satellite image comprehension and object detection. Cartography has widely used high-resolution earth observation image because it can provide massive ground feature information with variant spectral characteristics. Accurate and timely gathering of building footprint information is one of the essential aspects of urban development, especially in the fields of urban planning, land use investigation, demographic census, disaster monitoring, real estate management, digital city and so on (Alshehhi et al., 2017; Gu et al., 2019; Huang et al., 2019). At the same time, as more countries begin to pay attention to the construction of smart cities in urban planning, the demand of buildings spatial information renewal is also enhancing, including its 3D model. As the main way to obtain such data, earth observation technology plays a key role in the construction of smart cities. With the rapid development of sensors applied to satellite or aerial imaging systems, the spatial resolution of optical earth observation satellites in commercial operation has reached the sub-meter level with tremendous output (Aasen et al., 2018; Mahabir et al., 2018; Zhang et al., 2020). Therefore, fast and accurate extraction of buildings has always been one of

the most essential and complex contents in earth observational image processing applications. A high-resolution earth observation image can highlight the detailed spectral characteristics of ground truth (Liu et al.,2019). However, the increased spatial resolution is accompanied by noise in the image, which limits the accuracy of building extraction. As a result, manual labeling remains the most widely used building extraction method at present. Human labor is time-consuming and inaccurate, seriously limiting the large-scale application of high-resolution earth observation images, resulting in waste of image data (Gu et al.,2019; Song et al.,2019; Transon et al.,2018).

Although automatic semantic segmentation of earth observation images has been studied for over 30 years (Liow & Pavlidis, 1990), successful automatic building extraction methods are limited due to the following factors:

- Due to the complexity of the earth observation image, the contrast between the buildings and the background is not significant (Liu et al.,2018).
- The buildings have irregular and inconsistent shapes due to the shadowing of clouds, illumination and other ground objects, especially residential buildings in communities or suburban areas, parts of a building can be covered by trees or shadows, making it difficult for machine learning methods to distinguish buildings from their surroundings (Transon et al.,2018).
- In high spatial resolution earth observation imagery, pixels have similar spectral characteristics, because buildings have a large intra-class variance and a low inter-class variance, and it is challenging to extract texture and spatial geometry elements of the structures. (Alshehhi et al., 2017; Huang et al., 2019;Liu et al., 2018)

In recent years, researchers have attempted to achieve automatic extraction of buildings from high-resolution RS images using a variety of methods. The traditional machine learning for aeronautical/aerospace image building extraction focuses on expressing the building characteristics by empirically designing appropriate features and connection with corresponding feature sets on the automatic recognition and extraction of buildings (Huang & Zhang, 2011). Common index of machine learning building extraction includes pixel (Cui et al., 2014), spectral (Jin & Davis, 2005; Zhang et al., 2016), length, edge, shape, texture (Attarzadeh & Momeni, 2018; Huang & Zhang, 2011), shadow (Huang & Zhang, 2011), height, color, contours (Liasis & Stavrou, 2016), etc. Although these traditional machine learning algorithms enable automatic building extraction only focus on building geometrical characteristics; the performance of these indices also varies with datasets from different seasons, illumination, sensor types, scales and different environments (Ji, Wei & Lu, 2018). Therefore, the traditional machine learning methods are usually limited by specific data and cannot achieve fully automated building extraction.

Deep learning is a new stage in the development of machine learning. it effectively solves problems such as characterizing complex object features and analyzing intricate scenes (Dargan et al., 2019). The deep learning method of building extraction from high-resolution earth observation images can automatically extract the feature information of buildings, and achieve high building extraction accuracy and efficiency. The deep learning algorithm took first place in the ImageNet Large Scale Visual Recognition Competition (ILSVRC), and the result is much better than traditional methods in millions of imageNet datasets (Russakovsky et al., 2015). This deep learning algorithm proposes

an artificial neural network model, AlexNet (Krizhevskey et al., 2012). AlexNet compares samples with labels, defines the loss functions, obtains the residual between samples and labels, shifts upward by chain derivative rules, and hierarchically adjusts the weights and bias of each layer until the loss Function satisfies the globally optimal solution (Liu et al., 2018). From the perspective of computer vision, building extraction from high-resolution earth observation imagery is not only a semantic segmentation problem but also object detection and instance segmentation, which indicates that every pixel in the image has its own class and pixel-level segmentation can produce the most accurate results (Liu et al., 2018; Song et al., 2019). Since the successful application of deep Convolutional Neural Network (CNN) in 2012 in image recognition and semantic segmentation, several improved convolutional neural network architectures emerged, including Visual Geometry Group (VGG) Net proposed by the Visual Geometry Group of Oxford University in 2014 (Simonyan & Zisserman, 2014), GoogleNet by (Szegedy et al., 2015) and Deep Residual Net (ResNet) launched by Kaiming He (He et al., 2016). These classical CNNs have been widely improved and applied as semantic segmentation models. The proposed architectures have more network layers to further improve the accuracy through the development of a convolutional neural network.

1.2. Objectives of the study

The existing researches on deep learning based building extraction are usually conducted by proposing a novel network structure based on current semantic segmentation models. The increasing model prediction accuracy was carried out.

Researchers usually design their architecture for higher building recognition accuracy by integrating state-of-art computer vision techniques and/or combining the advantages of two structures together. Then their proposed models were compared with the baseline models by training on building datasets. Although there have been increasing trends in deep learning-based automatic building extraction in recent years, there is still a clear gap compared with the other earth observation imagery object extraction tasks, such as road segmentation and land classification applications. In particular, the state-of-art semantic segmentation models: PAN and BiSeNet are rarely discussed in this field of research.

In order to fulfill the research gap above, this paper is conducted on the recent development of CNN especially FCN-based automatic building extraction of high-resolution earth observation images. Then, several state-of-art semantic segmentation models were trained including U-Net, SegNet, RefineNet, DeepLabV3+, BiSeNet and PAN on the Waterloo Building Dataset (WBD) using Tensorflow as a deep learning framework. Meanwhile, the results were evaluated by the comparison of both qualitative and quantitative analysis. For quantitative analysis, seven commonly used evaluation matrices are applied to each model. The results show that DeepLabV3+ outperforms other models with higher accuracy on the dataset, achieving the highest scores of 98.21%, 91.75%, 88.34%, 76.96%, 87.3%, and 90.01%, respectively by OA, precision, Recall, IoU, mIoU, and F_1*−score.* The index FPS indicates image frames per second that each model can deal with, which is also an intuitive index representing the running speed of the deep learning model.

1.3. Structure of the research paper

The rest of this research paper is organized as follow: Chapter 2 summaries the development of machine learning to deep learning and the implementation of Deep Learning (DL) in the field of building extraction from high-resolution earth observation imagery, and emphasizes introducing the development and structures of the state-of-art semantic segmentation models used in this experiment: U-Net, SegNet, RefineNet, DeepLabV3+, BiSeNet, and PAN. Chapter 3 presents the description of existing building footprint datasets: Massachusetts Building Dataset, Inria Aerial Image Labeling Dataset and WHU Aerial imagery Dataset. Chapter 4 describes the methodology of this experiment including implementation details, dataset WBD, briefly compared the FCN baseline models, and introduction of evaluation matrices are used for assessing the efficiency of models. Chapter 5 concludes the quantitative and qualitative analysis of results and possible improvement and the difficulties encountered during the experiment. Chapter 6 summarizes the important findings of this research paper and addresses the most promising directions for improving the system.

2. RELATED WORK

With the development of earth observation technology, the human ability to observe the earth comprehensively has reached an unprecedented level. Earth observation data with different imaging modes, wavebands, resolutions, observation scales, and dimensions have become key carriers for the acquisition, processing, and application of geoscience information (Gu et al., 2019; Transon et al., 2018). Currently, the spatial resolution of optical satellites in commercial operation has reached the submeter level. For example, the U.S. World View-4 satellite launched in 2016 can provide high-definition ground images with a resolution of 0.3 m (Sozzi et al., 2018). However, the efficiency of processing earth observation information is not compatible with the speed of data acquisition. The main challenge is how to extract and interpret information from these big data since the increasing amount of data did not lead to improved system ability to predict (Reichstein et al., 2019). The rapid development of artificial intelligence technologies, the emergence of large-scale labeled data, and significantly improved computational performance lay the foundation for intelligent analysis of earth observation data, and also provides conditions for the classification and prediction of earth observation imagery using deep learning (Sun et al., 2017). From the perspective of a computer vision, building extraction from high-resolution earth observation imagery is not only a semantic segmentation problem, but also involves object detection and instance segmentation (Gu et al., 2019; Liu et al., 2018; Song et al., 2019). Building extraction belongs to the object/image segmentation, which aims to segment the corresponding portion of the target from the image. For general optical images, the common goal for object extraction is to extract the ground truth (Liu & Zhou, 2018; Song

et al., 2019). It can also be considered as a classification task, in which each pixel is classified to its belonging class. In contrast with object extraction, building recognition is a classification-based task. Based on the given data, it can be explained which samples are classified as targets and which are not. In this case, it does not classify pixels but gives segments or the image itself and defines what class it belongs to. Unlike object extraction and object recognition, object detection was developed from the origin of the detection system and has received extensive research by radar professionals. The most straightforward task of object detection is to extract information-bearing patterns from the signal that is random and full of interference and noise. Therefore, the main difference between object detection and object extraction/recognition is that it requires the prerequisite of being noisy and interfering with the data (Lin et al., 2017; Liu & Huang, 2018; Papageoriou & Poggio, 2000).

This section first summarizes the development of CNN including the introduction of two classic classification networks, ResNet and VGG16, which also serve as backbone networks in this study. We then introduce some state-of-the-art FCN-based semantic segmentation models and their application in the field of earth observation imagery building extraction, followed by the comparison of the three most commonly used open-source building footprint datasets in deep learning building extraction tasks. Recent work on deep learning-based building extraction from high-resolution earth obser-vation imagery is then presented.

2.1. Emerge of CNN

Deep learning as a subfield of machine learning research, is motivated to establish and simulate neural networks in the human brain for analytical learning. Deep learning algori-

thms can be mainly divided into three types: CNN is usually used for image data analysis and processing; Recursive Neural Networks (RNN) for text analysis or natural language processing; Generative Adversarial Network (GAN), usually for data generation or unsupervised learning applications (Pröve, 2017). After several years of development, many scholars have proposed deep neural networks-based change detection methods, such as Deep Belief Networks (DBN) (Mohamed et al., 2009), Stacked Auto-Encoders (SAE) (Wang et al., 2014), Convolutional Auto-Encoders (CAE) (Masci et al., 2011), PCA Network (PCANet) (Masci et al., 2011) and so on. The development of CNN and FCN has injected new vitality into pixel-level change detection methods. In particular, the FCN architecture, developed from CNN, has become an essential aspect of semantic segmentation. Through the development of deep learning-based building extraction in recent years, it is proved that deep learning approaches can well extract spatial-temporal characteristics of earth observation data and generate new mechanisms and understandings for computer vision.

Convolutional Neural Network, also known as CNNs, is a neural network that is widely used in image processing tasks. CNNs inspired by the human visual neural system, can extract hierarchical characteristics as feature vectors (Albawi et al., 2017). CNNs have two main features: they can effectively reduce the dimensionality of large data images to small data volumes, and retain image features, in accordance with the principles of image processing. Before developing a convolutional neural network, image processing was challenging for AI for two reasons: the amount of data needed to be processed, leading to high cost and low efficiency. Meanwhile, it is difficult to retain the original features in the digitization process, resulting in low image processing accuracy. CNNs addresses the obstacles ima-

ge classification, which is due to large variations in entire images via modeling smaller pieces and using deep networks to connect them (Gollapudi,2019). The typical CNN is composed of three parts: the convolutional layer, pooling layer, and the connection layer. Each layer can be described briefly and the convolution layer can extract local features in the image. The pooling layer is used to significantly reduce the parameter magnitude, in other words, to reduce the dimensionality; the fully connected layer resembles the section required for the output of a typical neural network. (Lenc & Vedaldi,2015). Figure 2.1 shows the structure of a typical five-layer CNNs. The convolutional layer is composed of filters (convolution kernel) which used to filter

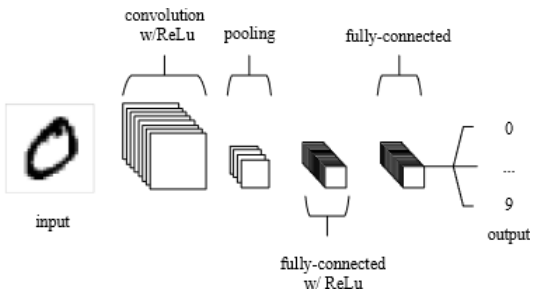


Figure 2.1 *A simple five-layer CNN architecture. (O'Shea and Nash,2015)*

various small regions of the image and obtain the eigenvalues of these small regions. Then a pooling layer is added after the convolutional layer, designed to reduce the dimensionality of images by down sampling. The pooling layer can compress the data dimension more efficiently than the convolutional layer, reducing the amount of computation and effectively avoiding overfitting. Enter convolutional and pooling layers processing data into the fully connected layer to obtain the desired output. The full connection layer is responsible for classifying the data with its size reduction. Hence, the images are converted to a 1D array, which can be processed and compared to the template. A typical CNN is not just a three-tier structure as mentioned above, but a

multi-tier structure. For example, the structure of LetNet-5 is shown in Figure 2.2 below: the structure is the input layer → convulational layer → pooling layer → activation function → convulational layer → pooling layer → activation function → convulational layer → fully connect layer → fully connect layer → output layer (LeCun et al., 1998).

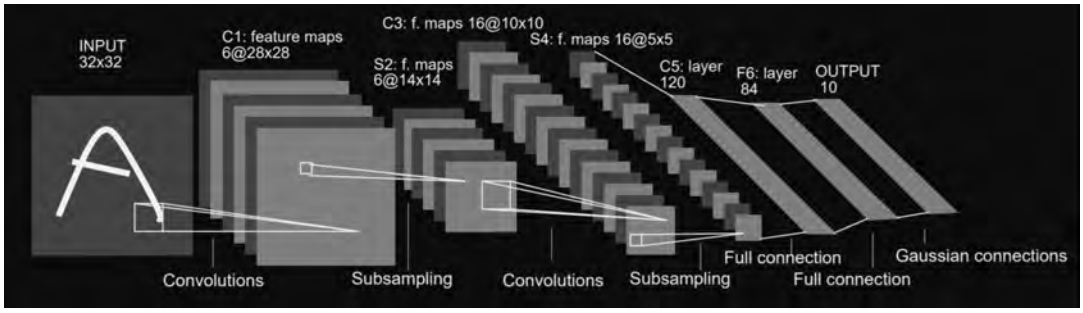


Figure 2.2 Structure of *LeNet network, which defines the basic components of CNN: convolution layer, pooling layer, full connection layer, etc. (LeCun et al.,1998).*

LetNet is one of the classic CNN classification networks. Some other early CNN models are classified chronologically as Letnet →Alexnet →ZFnet →VGG →GoogLeNet→ResNet →DenseNet. In this section, the focus is on VGG and ResNet, because they can serve as backbone models for this experiment. Understanding the structure and function of the pre-trained backbone model can facilitate comprehending the semantic segmentation architecture. VGG is the related work for ILSVRC in 2014. The main goal of this research is to show that expanding the network's depth has a limited impact on the network's overall performance. The main capacity of VGG can be summarized such as:

- The stack of two 3x3 convolutional layers is equivalent to the receptive field size of a 5x5 convolutional layer, but its parameters are less than that of 5x5. There is also an additional convolutional layer followed by a non-linear structure.

- Network weight pretraining techniques are used to train shallow networks and fine-

tune the deep network with the shallow network weight (Simonyan & Zisserman, 2014).

The ResNet network was modified based on the VGG16 (He et al., 2016). ResNets addresses the "degradation" problem of deep neural networks. When layers are gradually added to a simple network, the model's performance on the training set and test set will improve because the model's complexity is higher, and the potential mapping relationship can be better fitted. On the other hand, degradation occurs when more layers are added to the network, and performance reduces rapidly. The degradation problem is attributed to the optimization problem. When the model becomes complicated, the SGD optimization becomes more difficult, resulting in the model's failure to achieve a good learning effect. To solve this problem, ordinary networks of all stacked layers 18 and 34 were first constructed. The remaining networks of layers 18 and layer 34 were constructed, as shown in Figure 2.3. Only a shortcut was inserted into the plain. Moreover, the number of parameters and computations are identical for the two networks. And compared with the previously well-effective VGG-19, the calculations were much smaller (3.6 billion FLOPs VS 19.6 billion FLOPs, or floating-point operations per second). This is repeatedly emphasized by the author, and also the most significant advantage of the model.

2.2. CNN models for building extraction

Researchers constructed semantic segmentation classifiers using machine learning techniques such as textural Forest or Random Forest before deep learning for computer vision (Schroff et al., 2008; Zhang et al., 2016). not only does CNN make image classification but also made great progress in segmentation. First, the categorization

layer name	output size	18-layer	34-layer
cov1	112×112	7×7,64,stride 2	
conv2_x	56×56	3×3 max pool, stride 2	
		$\begin{cases} 3\times3, 64 \\ 3\times3, 64 \end{cases}$ ×2	$\begin{cases} 3\times3, 64 \\ 3\times3, 64 \end{cases}$ ×3
conv3_x	28×28	$\begin{cases} 3\times3, 128 \\ 3\times3, 128 \end{cases}$ ×2	$\begin{cases} 3\times3, 128 \\ 3\times3, 128 \end{cases}$ ×4
conv4_x	14×14	$\begin{cases} 3\times3, 256 \\ 3\times3, 256 \end{cases}$ ×2	$\begin{cases} 3\times3, 256 \\ 3\times3, 256 \end{cases}$ ×6
conv5_x	7×7	$\begin{cases} 3\times3, 512 \\ 3\times3, 512 \end{cases}$ ×2	$\begin{cases} 3\times3, 512 \\ 3\times3, 512 \end{cases}$ ×3
	1×1	average pool, 1000-d fc, softmax	
FLOPs		1.8×10^9	3.6×10^9

Table 2.1 *The representation of model architecture image for ResNet-152, VGG-19, and two-layered feed-forward neural networks.*

learning method that divides each pixel into its appropriate category by using the image blocks surrounding each pixel. The fundamental rationale for using picture blocks is that most classification networks include a fully connected layer and require fixed-size image blocks as input. (Albawi et al., 2017; Zhao et al., 2019). Long et al. (2015) from the University of California, Berkeley proposed a Fully Convolutional Network (FCN), which promoted the original CNN structure and could carry out intensive predictions without a full connection layer. The proposed structure enables the segmentation graph to generate images of any size, and improves the processing speed compared with the image block classification method. Since then, most recent semantic segmentation studies have adopted this structure (Mahabir et al., 2018; Zhao et al., 2019). Different from ordinary photos, earth observation imagery, especially high-resolution earth observation data, include more complex spectral and spatial information, used to display the texture, structure, and geospatial position of terrestrial objects (Song et al., 2019). Due to differences in architectural morphology, semantic segmentation of high-resolution earth observation imagery using CNN has become a major challenge for automatic extraction in the building. However, at present, the segme-

ntation construction based on CNN still pays little attention in relevant fields and needs to be filled urgently. Different from the classification task, building extraction needs to determine the category of each image pixel for accurate segmentation. Traditional CNN models for building extraction is to train a CNN to classify a class per pixel by implementing a patch around the pixel, which is known as patch-based methods (Mnih, 2013; Song et al., 2019). However, a successful prediction of a pixel requires an overlapping patch. This approach causes redundant computations. Since Long et al. (2015) converted the CNN to FCN by removing full connection layers and conducting pixel-wise prediction with bilinear interpolation as the upsampling method, it is more effective than patch-based approaches that require redundant computations as it can classify each pixel directly (Wang et al., 2020). As a result, FCN has been utilized for semantic segmentation extensively and outperforms standard methods.

2.2.1. U-Net

The U-Net convolutional network was proposed on CVPR in 2015 (Ronneberger et al., 2015). U-net is a variant of FCN; the original idea of U-net was to address the problem of biomedical images (Ronneberger et al., 2015). It is a variance of FCN but more effective, requiring only a small amount of labeled data. The improvements include that the network's contextual information can flow to higher-resolution layers by adding more channels, and that the pooling operation is replaced by an upsampling operation (Han & Ye,2018). The entire U-Net network structure is shown in Figure 2.3, which resembles a large U letter: the U-net network structure is symmetric and resembles the letter U. The figure is composed of blue/ white boxes with various colored arrows, where the blue/white boxes represent the feature map, the blue arrows indicate the 3x3 convolution of the extraction, and the

gray arrows represent the skip-connection for feature extraction, red arrows indicate pooling for dimensionality reduction, green arrows indicate upsampling for dimensionality recovery, and cyan arrows indicate 1x1 convolution for outputting results. The gray arrow in copy and crop is used to concatenate, and the crop is used to make the length and width consistent.

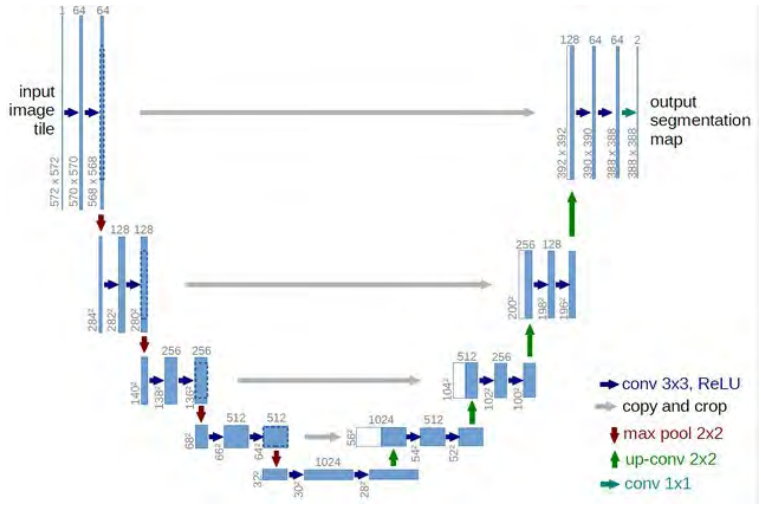


Figure2.3 *U-Net structure (Ronneberger et al., 2015)*

The encoder consists of convolutional and downsampling operations, and the convolutional structure used (Ronneberger et al., 2015) is a uniform 3x3 convolution kernel with 0 padding and one striding. Then the feature map is restored to its original resolution via its decoder. In addition to the convolutional layer, key steps in U-Net also involve upsampling and skip-connection (Song et al., 2019). For the structure of U-Net, the dimensionality of the feature map does not change, but each dimension contains more features, which is an efficient choice for ordinary classification tasks that do not need to recover from the feature map to the original resolution. At the same time, the skip-connection retains more dimensionality and position information, which is more advantageous for semantic segmentation tasks. U-Net is the most commonly used and the simplest segmentation model, which is simple, efficient, easy to understand, easy to build, and can be trained on small data semantic segmentation contests these years

(Khalel & El-Saban,2018).

2.2.2. SegNet

Similar to U-Net, SegNet is also a commonly used semantic segmentation model with an encoder-decoder structure. SegNet, proposed in 2015 (Badrinarayanan et al., 2015), has the characteristics of fast training speed and clear network structure, but many evaluation indicators are inferior to the other state-of-art semantic segmentation models (Song et al., 2019). The core of SegNet is the application of encoder-decoder structure for semantic segmentation. The network structure of the encoder and decoder is the 13 layers of VGG16 (after removing the complete connection), which is completely symmetric. The innovation of SegNet is that it uses an encoder to upsample its low-resolution features. Specifically, the decoder uses the pixel index of the maxpool in the encoder to carry out anti-pooling, thus eliminating the need for learning sampling. Therefore, it can effectively reduce memory footprint and increase computational efficiency during inference (Guo et al., 2018), also has fewer parameters than other models and can be trained end-to-end using stochastic gradient descent (Badrinarayanan et al., 2015). The author firstly discusses some previous work and believes that the existing DL methods are robust mainly because maxpooling and sub-sampling reduce the resolution of feature maps. The SegNet was created to overcome the problem of mapping low-resolution input to the raw input resolution in the semantic segmentation. To de-pool, the decoder utilizes the pixel index of the encoder's maxpool. Unsupervised feature learning has inspired this concept. According to Badrinarayanan et al. (2015), there were several advantages of re-indexing the encoder pool in the decoder:

- Improve the edge of the situation
- Reducing the parameters of the model
- Can be easily integrated into any encoder-decoder structure with only minor modifications (Badrinarayanan et al., 2015).

Like U-Net, SegNet is also an encoder-decoder structure. Its encoder structure removed the full connection layer of the VGG16, thus significantly reducing the size of the model. The decoders and encoders have symmetrical structures, but the pooling is upsampled according to the max-pooling indices in the encoder layer (Guo et al., 2018). In FCN, the feature mapping is directly superimposed on the next layer by bilinear interpolation. Compared with other critical variant networks, the development of SegNet reveals the trade-offs involved in designing the segmentation network, particularly the trade-offs of training time, memory footprint, and accuracy. Architectures that fully store encoder network feature mappings perform best but consume more memory during inference. SegNet, on the other hand, is more efficient because it only stores the maxpool indexes of feature maps and uses them in its decoder network for good performance (Guo et al., 2018).

2.2.3. RefineNet

The Refinenet is a typical semantic segmentation algorithm, proposed by Lin et al. (2017). For popular CNN such as VGG and Resnet, due to the presence of the pooling layers and small resolution of convolutional steps, the loss of some details was accordingly shown (Liu & Huang,2018). As described in Section 2.1, low-level feature maps have rich detailed information, while high-level feature maps have more abstract semantic information (Wang et al.,2017). For pixel-level semantic segmentation, low-

level details are required in addition to high-level semantic features (Chen et al., 2021). Several approaches have been proposed to address this problem. For example, the typical encoder-decoder structure, SegNet structure, use deconvolution to restore the image resolution, but is still difficult to restore the details. Also, structures represented by U-Net use jump connections to produce high-resolution predictions. Moreover, the DeepLab series use atrous convolution to maintain resolution and increase receptive field, but this approach increases computation, and atrous convolution easily loses some high-level semantic information (Lin et al., 2017).

Similar to SegNet, U-net, and DeeplabV3+, Refinenet also has an encoder-decoder FCN (Chen et al.,2018). The network can leverage the features of all layers to make the semantic segmentation more precise and utilize all the information in the subsampling process, using remote residual connections to achieve high-resolution predictions. All components of RefineNet use Residual Connections (Identity Mappings), which makes it easier to transmit short or long to the front, making the segment end-to-end training easier and more efficient. A module called Chained Residual Pooling is also proposed to capture background and context information from a large image region. In this case, fine-grained features can be used directly to reinforce high-level semantic features (Lin et al.,2017). Liu et al.(2019) performed a comparison of these models on the WHU building dataset using confusion matrix Precision, Recall, $F_1 - score$ and IoU. The results show that the quantitative analysis of the baseline models did not vary greatly, probably due to the high resolution and abundance of RS images during training. Another discussion presented by the authors is that implementing pre-training parameters for image classification, such as VGGNet and ResNet, as the backbones of semantic segmentation researches, can enhance the model performance. However, as the

baseline be used for indoor scenes with more class features than remote sensing images. Therefore, the effect of pre-trained backbones applied to RS images may not be as pronounced as the effect on indoor scene segmentation (Liu et al.,2019).

2.2.4. DeepLabV3+

DeepLabV3+ is the latest work of the DeepLab semantic segmentation network series, whose previous works included DeepLab V1, V2 and V3. In the latest work, Chen et al.(2018) reported that the encoder-decoder structure of common semantic segmentation models such as U-Net was applied to integrate multi-scale information. At the same time, the original atrous convolution and Atrous Spatial Pyramid Pooling (ASPP) layer were retained, and the backbone network utilized the Xception model to improve the efficiency of semantic segmentation. DeepLabV3+ achieved 89.0 mIoU on PASCAL Visual Object Classes (PASCAL VOC) and 82.1 on Cityscape (Chen et al.,2018). The following diagram shows the basic structure of DeepLabV3+: Compared with PSPNet, SegNet U-NET, DeepLabV3+ is improved on introducing atrous convolution, which increases the receptivity field

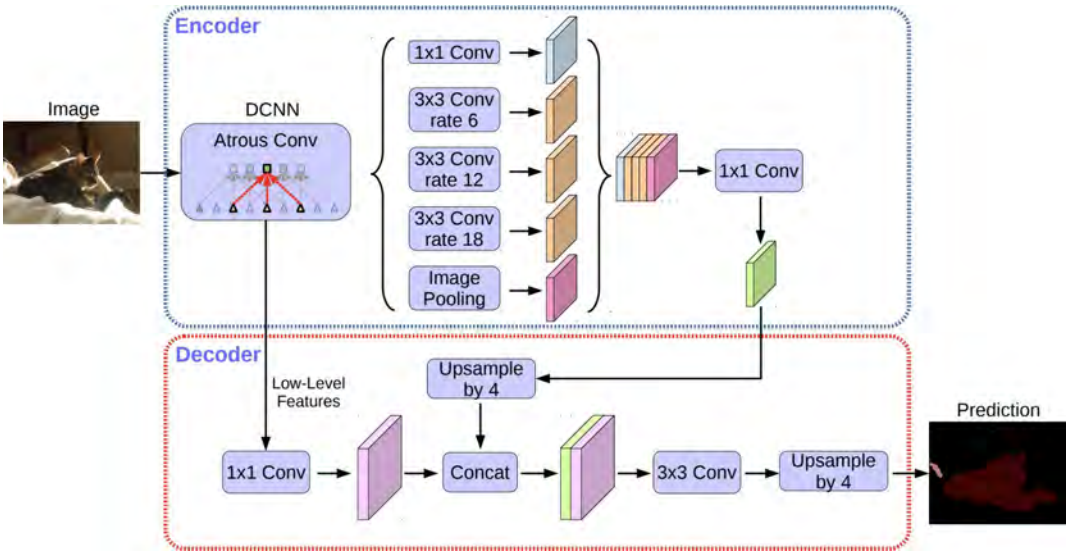


Figure 2.4 *Structure of DeepLabV3+ (Li et al., 2018)*

without loss of information so that each convolutional output contains a large range of information. As a result, a wider range of pixels was selected when extracting feature points. The purpose of atrous convolution is to extract more useful features, and

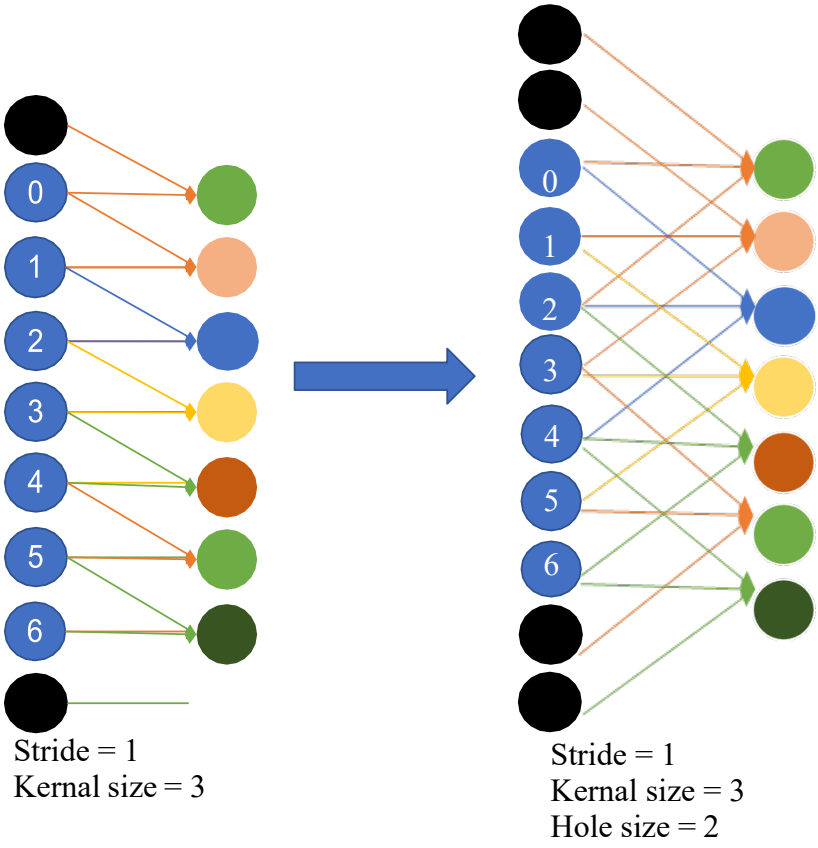


Figure2.5 *2Dconvolutionusinga3kernelwitha dilationrateof2*

thus can be extracted in the Encoder network. The encoder has two core points: 1. Serial atrous convolution was used for the backbone Deep Convolutional Neural Network (DCNN) 2. When the image passes through the backbone DCNN, the results were divided into two parts. One part is directly passed into the decoder, and the other part passes through a parallel atrous convolution structure. Then comes the 1x1 convolution compression feature (Pröve, 2017). The decoder has two inputs, one of the outputs of DCNN, the other of output results of DCNN without parallel atrous convolution. These two results are combined by Concat after processing.

2.2.5. BilateralSegmentationNetwork

The BiSeNet model was released by the Megvii Technology Vision team (Yu et al., 2018). Figure 2.6 shows the architecture of the BiSeNet (Benjdira et al., 2020). BiSeNet is an engineering-oriented network designed to rapidly perform semantic segmentation. The authors summaried three ways of speeding up semantic segmentation:

- Restrict the input by cutting or resizing. The proposed method is simple and effective, but with a great loss of spatial details.
- Reduce the number of network channels, but this weakens the spatial information.
- Abandon the downsampling at the final stage, which would make the insufficient receptive field of the model cover large objects, resulting in poor inference ability (Yu et al., 2018).

All three schemes have insurmountable disadvantages. Semantic segmentation usually uses a U-shaped structure. First, the feature image was reduced to extract the semantic meaning, and then combined with larger feature images to constitute spatial details (Romera et al., 2017; Yu et al., 2018). The complete U-shaped structure itself has a large amount of computational power, reducing the number of channels led to a loss of spatial information, and reducing the number of layers of the U-shaped structure led to a shortage of receptive field (Yu et al., 2018). Therefore, there were some defects in using the U-shaped structure itself, and improvements are necessary to accelerate semantic segmentation. As shown in Figure 2.7, the BiSeNet contains Spatial Path and Context Path. These two components are used to address the reduction of spatial information loss and receptive fields, respectively.

The Spatial Path contains three convolutional layers and the corresponding Batch Normalization layer and ReLU layer. The input image is large, and the output image is 1/8 of the original image. The Context Path contains a pre-trained XCeption39 as a backbone network (can be replaced by ResNet101), a convolutional network for model adjustment, and a series of attention optimization modules. Both the Spatial Path and Context Path finally provide 1/8 features of the original image size. The features of these two parts are fused and output by the FFM module. The FFM module first spliced the two parts features together and then transformed using the "convolution +BN+ReLU", followed by the Resnet-SE module (Yu et al., 2018).

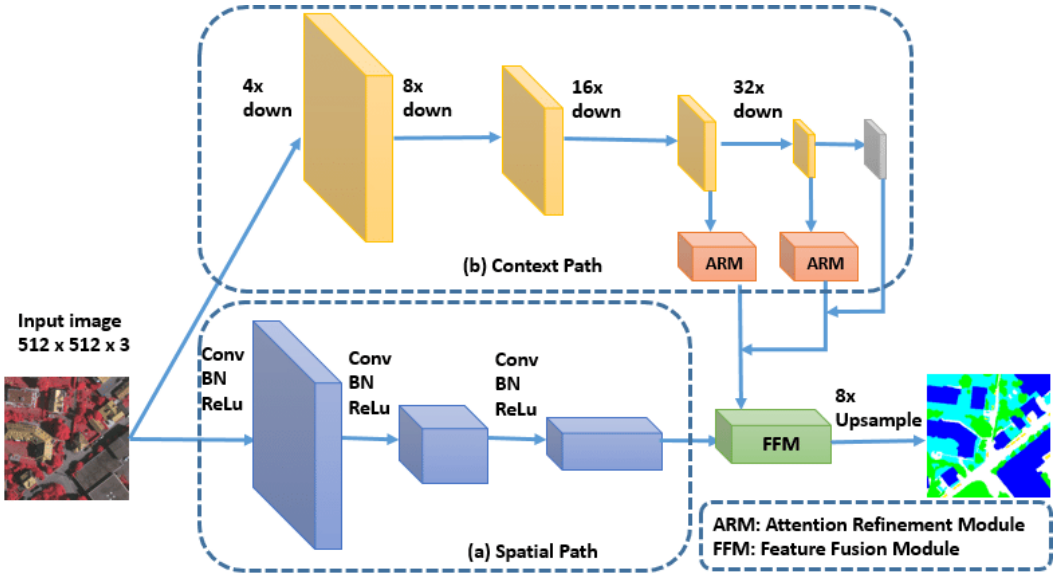


Figure 2.6 *The architecture of BiSeNet (Bilateral Segmentation Network) (Benjdira et al., 2020)*

2.2.6. Pyramid Attention Network

Li et al. (2018) proposed PAN for semantic segmentation based on the Feature Pyramid Attention module (FPA), and Global Attention Upsample module (GAU) and introduced an attention mechanism for semantic segmentation. The authors believe

that the existing semantic segmentation structures were affected by the loss of spatial resolution when encoding high-dimensional information (Li et al., 2018). The paper demonstrated that the use of FCN as a backbone had a poor prediction of small targets with two main challenges:

- Classifying objects is challenging due to their multiple scales. To solve this problem, PSPNet and DeepLab intoduced PSP and ASPP modules to apply multi-scale information. Lin et al. (2018) introduced pixel-level attention to help extract accurate high-level features.

- High-level characteristics easily classify categories but lack spatial information.

The most frequent process to tackle this problem is to utilize a U-shape structure network, such as SegNet, and RefineNet, making low-level information facilitate high-level information recovery of image details. However, all of these methods are time-consuming. The paper proposes an effective decoder structure GAU that can extract high-level global contexts to enable the weighted low-level information.

2.3. Related research in FCN building extraction

In recent years, many scholars in the field of semantic segmentation of earth observation imagery have conducted experiments on the networks above, such as FCN, U-Net, SegNet, DeepLab, RefineNet, PAN, and BiSeNet. In these recently famous semantic segmentation models, PAN and BiSeNet are less used in automated building extraction of earth observation imagery. Four other methods have been widely used in model comparisons in recent research. The structure of their proposed semantic segmentation models generally demonstrated its applicability to semantic segmentation

of earth observation imagery, adjusting the structure and integration of the existing baseline model. While proposing their networks, these papers will also train multiple baseline models for comparisons, with the result is that most of the models designed perform best on most of the evaluation metrics on the applied datasets. Currently, there are also many articles reviewing deep learning in the state-of-the-art earth observation. However, they tend to cover a wide range of issues or topics in earth observation, such as land cover classification, and change detection. in recent years many researchers have studied the building extraction of high-resolution earth observation imagery based on CNN models. This section investigates and analyses the development of CNN models in this field, which plays a vital role in earth observation, census survey, disaster monitoring, etc.

When using deep learning to extract buildings from high-resolution earth observation imagery, commonly used state-of-art models were usually applied to train the widely used building imagery datasets and compared to models proposed by researchers. Wang et al. (2020) proposed a novel network, called Efficient Non-local Residual U-shape Network (ENRU-Net), which has the advantage of conserving contextual information and improving existing upsampling methods. The authors pointed out that the main challenges of FCNs building extraction methods first were the original FCN upsampling layers and pooling layers led to the reduction in detailed features, such as small buildings. Secondly, the FCN architectures could not extract sufficient contextual information to reveal the inner pixels of large buildings, even if pooling and convolutional layers were successfully applied (Wang et al., 2020). In order to address these problems, the authors introduced encoder-decoder structures and the non-local block strategies (Badrinarayanan et al., 2015; Ronneberger et al., 2015) into the proposed structure. Then the ENRU-Net model was evaluated by comparing

segmentation models: FCN, U-Net, SegNet and DeepLabv3 on the Massachusetts Buildings Dataset and the Wuhan University (WHU) Aerial Imagery Dataset. Results showed that the proposed method outperformed the state-of-art models in the metrics Overall Accuracy (OA), Inter- sect over Union (IoU) and $F1 - score$ on both datasets. Surprisingly, the performance of DeepLabV3 is inferior to that of FCN, SegNet and U-Net on the Massachusetts building datasets, but outperforms that of other three models on the WHU aerial imagery datasets. Different from FCN, using contextual information for semantic segmentation, as a DCNN proposed in 2017, Deeplabv3 was based on the spatial pyramid pooling in serial modules and spatial pyramid pooling. The addition of the receptive field of filters into the network structure of the SPP module could integrate multi-scale information (Liu et al., 2019). The authors suggested that this was because the WHU aerial Imagery Datasets had a higher spatial resolution and lower image complexity.

Similarly, a U-Net architecture proposed by Chen et al. (2021) has introduced self-attention and reconstruction-bias modules for building extraction tasks, tested on the Massachusetts and WHU building datasets, along with baseline comparison models: U-Net, SegNet, DeepLabV3+, etc. Similar to Wang et al. (2020), the models tested on the WHU building dataset achieved higher evaluation metrics scores than Massachusetts. The proposed method had the highest scores on Recall, IoU, and $F_1 - score$ on the WHU building dataset, at 95.56%, 89.39%, and 94.4%, respectively. However, although the quantitative results of the proposed U-Net model ranked first in the comparative models on the Massachusetts dataset, there was no significant gap between the performance of the proposed structure and the baseline comparisons. In fact, comparing and proposed models performed similarly on Massachusetts dataset by the authors implementation

(Chen et al., 2021). However, the authors pointed out that the proposed model still lacks research contextual information and the entire building structures, especially on large buildings. These are common problems in the current construction of semantic segmentation models in the same research field.

In the recent publication by Cai et al. (2021), the author also performed a comparative study of the state-of-the-art semantic segmentation models on building rooftop extraction tasks using WBD. Different from this research, the authors trained three deep learning models such ass FCN, U-Net, and DeepLabV3+ for one-sixth of the total WBD. Meanwhile, instead of using pre-trained backbone models, Cai et al. (2021) choose to train the models from scratch. The novel approach of this research is to apply two different loss functions to each model: binary cross-entropy loss and focal loss, and then train each model in different portions of, respectively 100%, 75% and 50%, of the WBD. Therefore, each model was trained in a total of six times. This approach can dig deep into the influence of different loss functions on the training of deep learning models and the amount of data during the training process (Cai et al., 2021). The results showed that DeepLabV3+ of training focal loss in the 100% dataset outperformed the other adjustments, with the highest scores on OA at 93.6%, IoU at 65.4%, mIoU at 79.0%, Precision at 77.6% and F_1- score at 79.1%.

3. OPEN-SOURCE DL BUILDING EXTRACTION DATASETS

3.1. Massachusetts Building Datasets

Massachusetts Building Dataset was published by Mnih, (2013). There are 155 RGB color aerial images covering 438.75 km^2 surface of Boston. The spatial resolution of the images is 1500×1500, with 2.25 km^2 per image cover. The labels in this dataset contain the building and background classes. Images and labels required the users to cut themselves to suit their model needs. As one of the first published building datasets for automatic building extraction, the Massachusetts Building Datasets has about 3GB in size, with 1 meter of resolution in the existing high resolution (less than one meter) RS datasets no longer has the advantage, coupled with the relatively low label quantity and quality, Massachusetts dataset is no longer preferred for building dataset today, but is still frequently used in contrast experiment with a different model.

3.2. Inria Aerial Image Labeling Dataset

The Inria Aerial Image Labeling Dataset also addresses the pixel-wise labeling problem of aerial imagery for automatic building extraction (Maggiori et al., 2017). The dataset covers 810 km^2 with a spatial resolution of 0.3 m. There are 360 RGB tiles of 5000×5000 pixels in 10 cities across the globe. The training portion of the dataset is linked to a publicly available ground truth of building footprints. The remainder of the dataset is only utilized for ground truth testing. Photos of the Public domain and official building footprints in the public domain were used to create this dataset.

3.3. WHU Aerial Imagery Dataset

The WHU building dataset was published in 2018 by (Ji et al., 2018). Compared to the former two datasets, the WHU building dataset was composed of multi- source earth observation imagery. The authors team spent nearly 1 year time on manually editing, labeling the high-resolution earth observation database and making it an open-source implementation. The dataset is comprised of aerial building dataset and satellite building dataset. The aerial imagery of 0.075 m resolution covers 450 km^2 of the surface of Christchurch in New Zealand, containing 220,000 buildings. The satellite dataset involves dataset 1 and dataset 2. The former dataset 1 contains images of 204 images, which were collected on different satellite sensors at different resolutions (0.3 m & 2.3 m) and in different cities across five continents, including China, Europe, North America, South America, and Africa. The later contains six adjacent satellite earth observation images with distinct color differences, with a spatial resolution of 0.45m, and an East Asia land area of 860 km^2. The building vector map was manually drawn in ArcGIS, including 34,000 buildings. Similar to the aerial dataset, the entire region was divided into 17388 tiles, facilitating the application of deep learning methods (Ji et al., 2018).

4. METHODS

Chapter 4 is structured as follows: Section 4.1 presents the information associated with the WBD and introduces the general framework of this research. Section 4.2 provides a walkthrough of the implementation details, including hyperparameter settings of each model, the hardware used in this experiment and training strategies applied to optimize the prediction results. Next, Section 4.3 describes the evaluation metrics used for the estimation of the overall performance of the model. Then Section 4.4 briefly compares the structure of the six state-of-the-art semantic segmentation models trained in this research.

4.1. Research Framework

The WBD was manually edited for nearly eight months and provided open source. The database has 241 satellite images with a resolution of 8360×8360 from the City of Waterloo, Ontario Canada, containing 150,000 buildings of different forms, as shown in Figure 4.1. The ground resolution is 0.12 m, covering 136.9 km^2. The City of Waterloo Open Data provides the original vector data and aerial images; how- ever, there are many data errors, such as loss, and dislocation, as shown in Figure 6, that cannot be directly applied. Therefore, the authors use ArcMap and ArcGIS pro to man- ually edit buildings' polygon data and produce high-quality building vector maps. After labeling, the quality was examined by sampling survey, 100 building polygons were randomly chosen for each picture and the percentage of error was observed (number of deviationin 100 units / 100); after multiple times examination, the percentage of error of the entire dataset was controlled below 1 percent meaning that there is less than

one building mislabelled in a hundred units. Examples of cut tiles are shown in Figure 4.4.

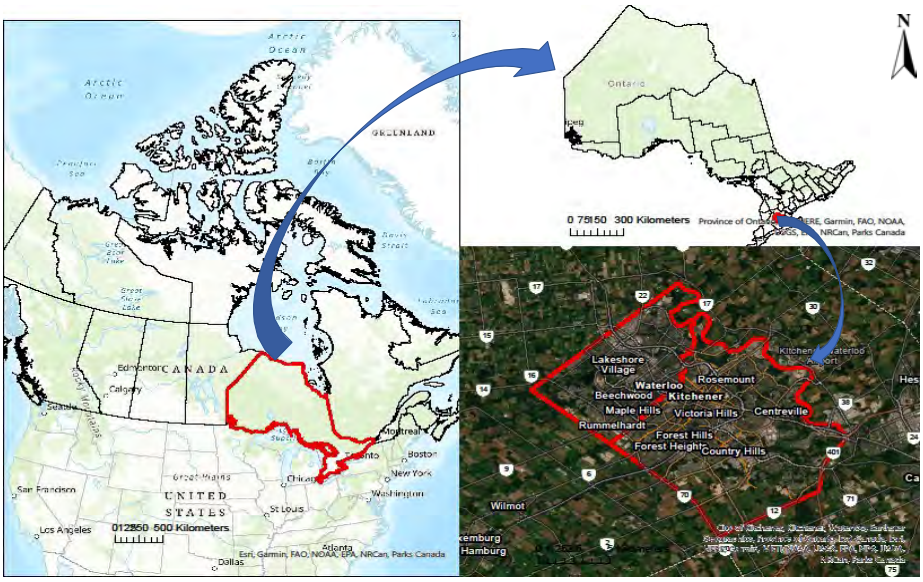


Figure 4.1 *Kitchener-Waterloo Regional Map, the KW area is located at south Ontario, Canada*

The entire dataset was divided into small patches with a dimention of 512×512 pixels as input to deep learning models. Compared with the existing building extraction datasets for deep learning, WBD has the advantage of high RS imagery spatial resolution (0.12 m comparing with Massachusetts' 1 m, WHU's 0.3~2.3 m and IRINA's 0.3 m) and high-quality labels. After manual labeling, the RS images and building shapefiles were cut into the size of 512×512; since the original RS image must be 8360×8360, the "offcut" images, shown in the last three examples in Figure 4.4, were filled with black pixels to 512×512. Then the building shape files were transformed into binary images that can be read by FCN models. There were 10404 small tiles after screening out the blurred and fuzzy images. Figure 4.3 shows three example scenes in the WBD. These three pictures include the scenes of almost all land types, such as buildings, roads, bare soil, urban green space, parking lots, and building types in the WBD dataset such as residential houses, residential apartments, commercial and industrial buildings. The

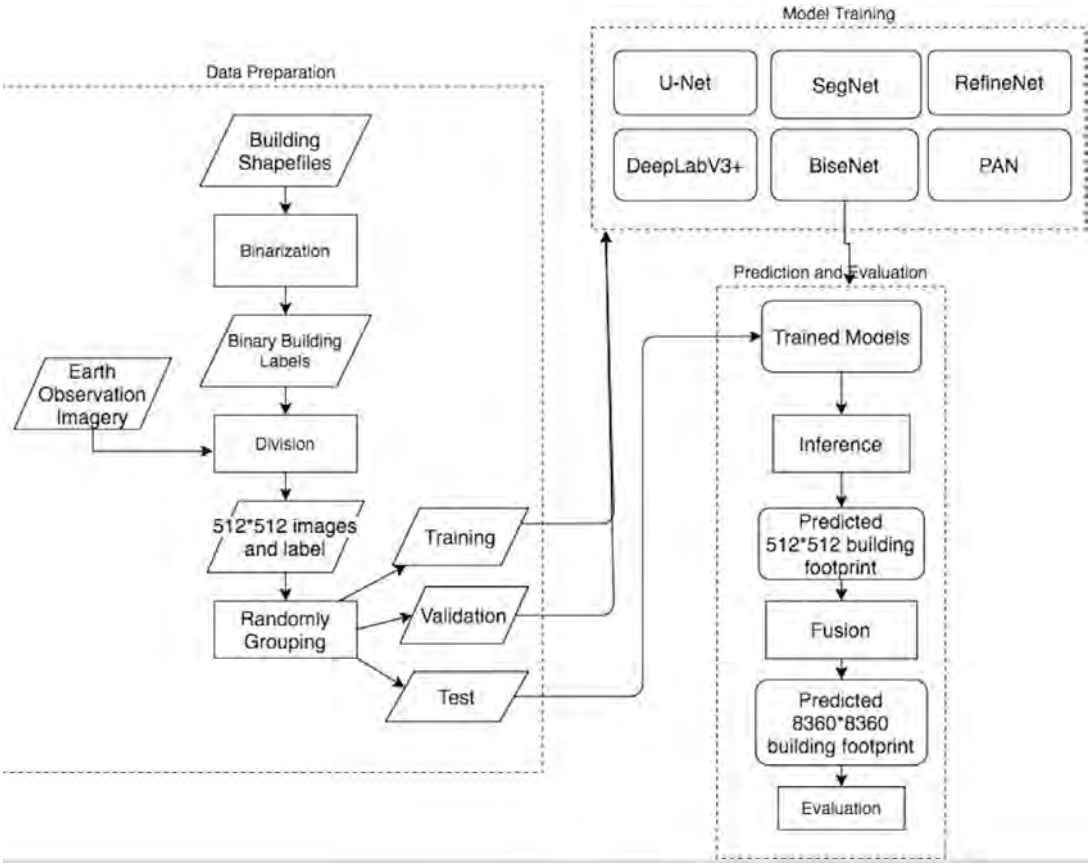


Figure 4.2 *Overall flowchart of building semantic segmentation of high-resolution feature vectors*

height, shape and spectral characteristics of the buildings also vary greatly. Therefore, presenting these three earth observation pictures along with their labels provides a good overview of the entire WBD. Figure 4.2 illustrates the flow chart of the research framework. After removing the blurred patches from the dataset, model training and validation were performed using the training and validation dataset. The optimal hyperparameters can be easily obtained by the implementation of the backbone model. Then, the tested dataset was applied to the trained models for evaluation. Six common evaluation metrics, OA, Precision, Recall, Intersect over Union (IoU), mIoU and F_1 − *score* were recorded for quantitative analysis and model comparison.

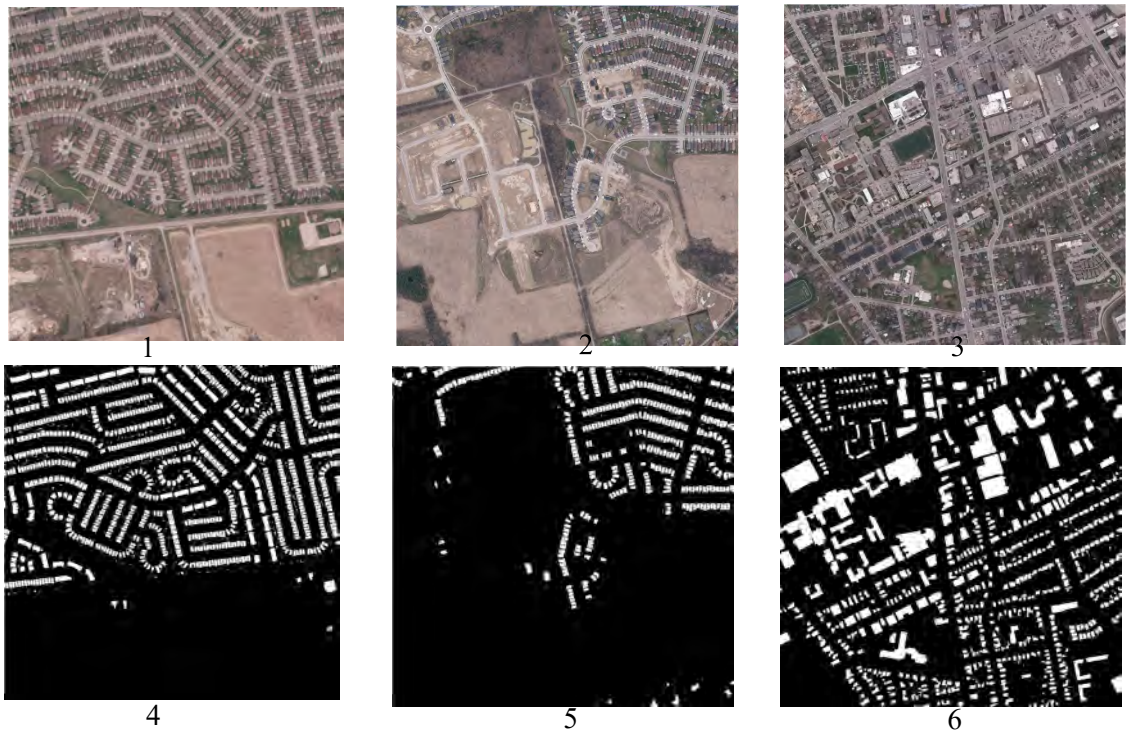


Figure 4.3 *An example scene in the WBD with corresponding label, image 1, 2, 3 are the earth observation images, 4, 5, 6 are the corresponding labels*

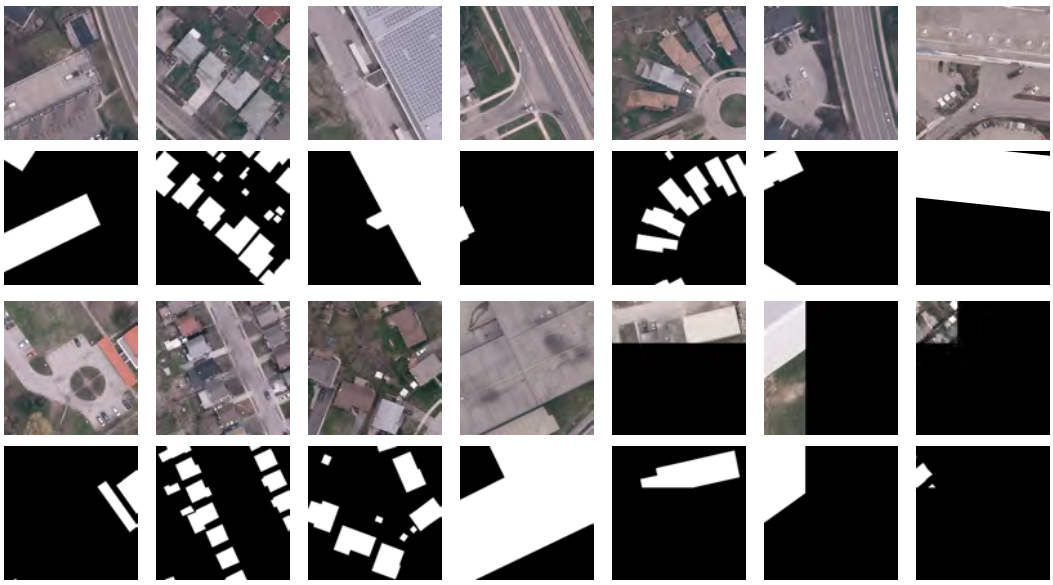


Figure 4.4 *Examples of cut tiles (512×512) from large images and labels, 14 representative scenes covering all building types are chosen. Row a and c represents the earth observation images, b and d represents the corresponding labels.*

The main purpose of this paper is to train the FCN models: U-Net SegNet, RefineNet, DeepLabV3+, BiSeNet and PAN and extract building from the tested dataset, WBD.

The present study aims to give a general development trend of recently proposed semantic segmentation models and provide a conclusive overview of computer vision on building extraction from high spatial resolution earth observation images. The quantitative results and inferences of different models were compared and analyzed by digging deep into reasons of variants, such as the number of parameters per model, the structure of each model, and training strategies ext. Moreover, this paper also intends to discuss the future implementation of computer vision on the semantic segmentation of earth observation imagery and its current limitations.

4.2. Implementation Details

The implementations of the baseline models are based on the deep learning framework TensorFlow. The experimental settings are shown in Table 4.1. All models were trained from scratch for up to 50 epochs using the Adam optimizer with a learning rate of 0.0001. The hardware is composed of 2 × Nvidia GTX 1080Ti. The deep learning framework was Tensorflow 1.4.0. Adam (Kingma & Ba, 2014) was selected as the optimizer. Meanwhile, the initial batch size is adjusted empirically for each model individually, also considering the GPU memory demand and availability. The batch size was set to be 16 for U-Net, SegNet, and RefineNet, 8 for BiSeNet and PAN, as well as 5 for DeepLabV3+. Adam Optimizer is one of the most commonly used optimization algorithms and has been verified in a large number of deep neural network experiments (Paszke et al., 2016). Polynomial decay is applied to ensure model convergence. It is well known that the learning rate is the most important hyper-parameter in the parameter transformation of deep neural networks (Smith, 2017). The learning of parameters in the deep neural network is mainly found by the gradient descent method for parameters that can minimize structural risk. In gradient descent, the model

may not converge, if the learning rate value is too large, and the convergence rate will be slow if too small (Smith & Leslie, 2017; Smith & Leslie, 2018). However, the lower the learning rate is, the slower the change in the loss function is, and the model is easy to be overfitted. While using a low learning rate ensures no local minima are lost, it also led to a longer convergence time. However, the high learning rate is prone to gradient explosion, large loss vibration amplitude, and the model is difficult to converge. Thus, it is important to choose an appropriate learning rate. The usual strategy uses a large learning rate at the beginning to ensure convergence and uses smaller rates near the optimal point to avoid oscillations back and forth. Therefore, relatively simple and direct Learning Rate adjustment can be achieved by Learning Rate Decay (You et al., 2019). The loss function of the experiments is a focal loss for binary classification since there are only two classes in construction extraction tasks. As proposed by Lin et al. (2017) focal loss addressed the rapid object detection issue, the imbalance of categories, and differences in pixel classification difficulty in the classification problem, which was improved based on cross-entropy loss. Taking the binary classification as an example, the loss value of the original classification is the sum of cross-entropy of each training sample, indicating the same weight for each sample. Equation 4.1 shows the cross-entropy loss function.

$$CE(p, y) = \begin{cases} -log(p) & \text{if y = 1,} \\ -log(1-p) & \text{otherwise} \end{cases} \tag{4.1}$$

Since it is a binary classification, p represents the probability that the predicted sample belongs to 1 ranging from 0 to 1. y represents the label, and the value of y is 1 or -1. As shown in Equation 4.2, when y=1, assuming the probability of a class with a sample X predicted of 1 equals 0.6, which is p = 0.6, then the loss is -log(0.6). Note that the loss is greater than or equal to 0. If p=0.9, the loss is -log(0.9), so the loss of p=0.6 is greater than that of p=0.9.

$$p_t = \begin{cases} p & \text{if y = 1,} \\ 1-p & \text{otherwise} \end{cases} \tag{4.2}$$

The focal loss can be expressed by Function 4.3, where γ is a focusing parameter with a value greater than 0. $(1- p_t)^{\gamma}$ is a modulating factor. From Equation 4.3, when a sample is misclassified, p_t is very small. Therefore, the modulating factor is close to 1, focal loss equals cross-entropy loss in this case. When the classification is correct and the samples are easy to classify, p_t is close to 1, the modulating factor is close 0, with a small contribution to the total loss. Meanwhile, another property of focal loss is that when γ=0, Focal loss is the traditional cross-entropy loss, and when γ increases, the modulating factor will also increase.

$$FL(p_t) = -(1-p_t)^{\gamma} log(p_t) \tag{4.3}$$

Table 4.1 *Experiment Settings*

System	Ubuntu18.04
HPC Resource	2 * NVIDA GTX 1080-Ti
DL Framework	Tensorflow 2.0
Program	Python 3.
Optimizer	Adam
LR Policy	polynomial decay
Loss Function	Binary Focal Loss
LR	0.0001

4.3. Evaluation Metrics

Some commonly used evaluation metrics for binary classification tasks are introduced in this section, including OA, Precision, Recall, $F_1 - score$, Intersect over Union and Mean Intersect over Union.

Confusion Matrix is a N×N matrix, N is the number of predicted classes. In this experiment, N = 2. Table 4.2 represents the confusion matrix table. Where True Positive (TP) indicates that the number of pixels belonging to a positive class predicted as a positive class, which in this case, can be interpreted as the number of predicted building pixels belonging to the building class. False Positive (FP) indicates the number of pixels belonging to the negative class predicted as positive. False Negative (FN) indicates the number of pixels belonging to the positive class predicted as the negative, which in this case, can be understood as the number of pixels belonging to the building class is incorrectly classified as none. Here True

Negative (TN) indicates the number of pixels belonging to negative class predicted as negative class.

Table 4.2 *Binary Classification Confusion Matrix*

Confusion Matrix		Target		
		Positive	Negative	
Model	Positive	TP	FP	Positive Predictive Value=TP/(TP+FP)
	Negative	FN	TN	Negative Predictive Value=TN/(FN+TN)
		Recall TP/(TP+FN)	Specificity TN/(FP+TN)	Accuracy=(TP+TN)/(TP+FP+FN+TN)

Overall Accuracy (OA) indicates the percentage of correctly inferred pixels in all samples. Accuracy is generally used to evaluate the global accuracy of a model, which cannot contain too much information to comprehensively evaluate the performance of a model. Can be expressed by:

$$Accuracy = \frac{TP + TN}{P + N} \tag{4.4}$$

Precision, also known as the precision ratio, is the ability of the model to find only relevant targets, while precision is directed against our prediction results, which shows how many predicted positive samples are positive samples.

$$Precision = \frac{TP}{TP + FP} \tag{4.5}$$

Recall refers to the ability of the model to find all relevant targets, the maximum number of real targets covered by the predicted results given by the model. The recall rate is for our original sample, which shows how many positive examples were

correctly predicted. The key to distinguish Recall from Precision is Recall refers to all positive samples in the data set, while Precision refers to all positive examples inferred by the model.

$$Recall = \frac{TP}{TP + FN} \tag{4.6}$$

$F_1 - score$ is an index that combines precision and recalls with values ranging from 0 to 1, where 1 represents the most accurate prediction of the model and 0 represents the worst output result of the model. The F value is the harmonic value of the accuracy and recall rate, closer to the smaller of the two numbers, so the F value is maximum when the accuracy and recall rate is close.

$$F_1 = \frac{2}{\frac{1}{\text{recall}} \times \frac{1}{\text{precision}}} \tag{4.7}$$

Intersection over Union (IoU) is an index that measures the accuracy of corresponding objects in a specific dataset. It has been applied to many objects detection challenges such as PASCAL VOC Challenge (Fu et al., 2019). IoU is a simple measurement standard, any task with a predictive range or bounding box in the output can be measured with IoU. Two criteria are needed to use IoU to measure object detection of any size and shape:

- Ground-truth bounding boxes (the approximate range of objects to be detected is marked in the image of the training set).
- The range of results obtained by the algorithm.

In other words, this criterion is used to measure the correlation between the ground truth and prediction, and the higher correlation between ground truth and prediction, the

higher the value. IoU can be expressed by:

$$IoU = \frac{TP}{TP + FP + FN} \tag{4.8}$$

Mean Intersection over Union (mIoU) calculates the ratio of the intersection and union of two sets of real and predictive values. Can be expressed by the formula:

$$mIoU = \frac{TP}{FP + FN + TP} \tag{4.9}$$

4.4. Baseline Models

As mentioned in Section 2.2, the models trained on the WBD involve the encoder-decoder structure models such as SegNet, U-Net, RefineNet, and DeeplabV3+, as well as dual-path models such as Bisenet and PAN. U-Net was created to segment neural structures. The encoder block is made up of two convolutional layers and a max-pooling layer. To achieve precise localization, the decoder concatenates the feature maps in the matching encoder before convolution (Dragan et al., 2019). SegNet was created to segment roads and the internal scenes. Its encoder is VGG16's convolutional layers, and is symmetrically built in accordance with the encoder. Besides, The max-pooling indices provided to the decoder are preserved by SegNet, allowing for a faster retrieval of spatial information during the upsampling (Liu et al.,2019). While the goal of RefineNet is to use the knowledge gained through downsampling to make high-resolution predictions. Multi-path refinement uses short- and long-range residual connections, multi-level fusion, and chained residual pooling. RefineNet- Res50 is the version of RefineNet that I utilized in this experiment (Lin et al.,2017). DeepLabV3+ is a

model in the DeepLab series. The atrous convolution, which extends the receptive fields without downsampling and losing spatial information is a creative design in the DeepLabs model. DeepLab v3+ uses a novel encoder-decoder structure that uses atrous convolution to encode multi-scale features and recover fine details in the decoder. Both RefineNet and DeepLab v3+ performed well on the PASCAL VOC 2012 and Cityscapes datasets (Fu et al., 2019). BiSeNet was proposed to address two challenges of semantic segmentation including a small receptive field and a lack of spatial information. In the convolutional neural network CNN, the Receptive field is determined by defining the area size of the input layer that matches an element in the output result of a certain layer. The image information should be fully considered to make the segmentation result complete and accurate when a big reception field exists. BiSeNet designed the Context Path and adopted the ResNet as the backbone network to increase the depth and expand the receptive field. Spatial information mainly refers to the local details of the images, especially for images with edges. Due to the large scale of the convolutional network, the input image is generally required to be of small size, so Crop or Resize of the original image is required. This will lead to the loss of detailed spatial information. By setting up a Spacial Path containing only three networks, Bisenet can retain a wealth of spatial information and then integrate spatial details at low latitudes with information at high latitudes. PAN combines the attention mechanism with the spatial pyramid structure to extract precise and dense characteristics, and investigate the impact of global context information on semantic segmentation. Specifically, PAN introduces high-level features to the FPA module and uses a spatial Pyramid Attention structure paired with global context information to learn a better feature representation.

5. RESULTS AND DISCUSSION

In this section, some state-of-the-art semantic segmentation models are trained and compared, including U-NET, SEGNET, RefineNet, Deeplabv3+, BiseNet and PAN, with the same implementation framework and settings as mentioned in Chapter 4. This section presents the experiment results from both quantitative and qualitative perspectives. The results are summarized in Table 5.1. Due to the abundant training data and the high spatial resolution of WBD, DeepLabV3+ can outperform the other encoder-decoder models resulting from its excellent feature extraction and maintained sufficient feature information even without skip-connection.

5.1. Quantitative Analysis

Table 5.1 shows results of the evaluation on the WBD test set. For the convenience of analysis, the highest result in each category has been highlighted in bold. From the quantitative analysis percentage, DeepLabV3+ outperformed the other models, leading almost all evaluation indexes, except that accuracy was about 2% lower than Bisenet; and FPS was 7.66 images/seconds smaller than U-Net. Deeplabv3+ achieved the highest value on OA at 98.21 percent, and defeated other models on Recall, Kappa, mIoU, IoU and $F1-$ score, which are respectively 88.34%, 89.03%, 87.3%, 76.96%, 90.01%. Surprisingly, RefineNet was the worst-performing model, with the lowest percentage of 80.43%, 68.47%, 71.59%, 75.86%, 56.86% and 73.97% in Precision, Recall, Kappa, mIoU, IoU and F_1-score, respectively. RefineNet had complex structures than the other models, the reason for this phenomenon might be caused by model complexity which can be reflected by a number of parameters.

The pre-trained parameters VGG16 in U-Net and ResNet50 in RefineNet introduced a large number of parameters into the model. As Liu et al. (2019) suggested, pre-trained models enabled the structure of the RefineNet complex, which even had nearly double parameters than DeepLabv3+. Therefore, the heavy structure of RefineNet has hindered its performance and is arduous in training.

Table 5.1 *Results evaluated by OA(%), Precision(%), Recall(%), IoU(%),mIoU(%),* $F_1 - score(\%)$ *and FPS.*

Method	OA	Precision	Recall	IoU	mIoU	$F_1 - score$	FPS
Unet	96.68	80.38	84.05	68.9	82.37	82.17	**24.51**
SegNet	96.77	83.08	80.97	66.69	81.34	82.02	9.33
BiSegNet	98.03	**93.78**	83.92	73.85	85.67	88.58	18.90
PAN	97.78	88.4	87.01	73.32	85.23	87.70	17.99
RefineNet	95.61	80.43	68.47	56.86	75.86	73.97	21.92
DeeplabV3+	**98.21**	91.75	**88.34**	**76.96**	**87.3**	**90.01**	16.85

5.2. Qualitative Analysis

To analyze the qualitative results, I selected some representative figures with land categories, including different sizes of buildings, parking lots, roads and bare soil. These lands might have similar characteristics to buildings, and the pixels within large building polygons usually cannot be classified properly by deep learning methods. Figure 5.1 presents some detailed 512×512 tiles that have been input to the model and makes predictions from the dataset tested. Images from left to right are respectively referred to RS image, label, inference of U-Net, SegNet, RefineNet, BiSeNet, PAN and DeepLabV3 +. The images I chose include the scenes of small buildings like the garage, residential buildings, large industrial/commercial buildings, roads, bare soil and parking lots. Figure 5.1 shows that U-Net had some errors which misclassified road to buildings. From the picture comparison, it is more intuitive to see that the learning from input data of

Refinenet is the most needed improvement among the several models. There are not only FP but also FN, a lot of pixels belonging to building class that is not identified as building, and some pixels in the parking lots are mistakenly predicted as building. Compared with the highest-scoring DeepLabV3+ in the quantitative analysis, the prediction of DeepLabV3+ was closest to the ground truth. It can be seen that it can correctly identify some small garages that are not achieved by other models, and that the edges and internal pixels of large buildings outperformed other models. However, there are still some misclassifications when the textures of the building are too close to its surrounding environment.

As mentioned in Section 4, the original image size of the uncut WBD was 8350×8350. Due to the limitation of GPU memory, large images and labels were cut into small patches with size 512×512. Figure 5.2 shows some representative scenes of large images and the joining of small inference patches. The 5 example scenes from top to bottom are RS image, labels, U-Net, SegNet, RefineNet, PAN, BiSeNet and DeepLabV3+ respectively. We can observe that DeepLabV3+ and BiSeNet are the two models whose predictions are closest to the ground truth from a global perspective. They have a better ability to extract the contours of small buildings. Most models cannot adequately identify the inner pixels of large buildings because they cannot obtain competent global contextual information from training data. Meanwhile, the comparison shows that the RefineNet predictions lack detailed information and do not match the results of other models with simpler structures such as U-Net and SegNet .

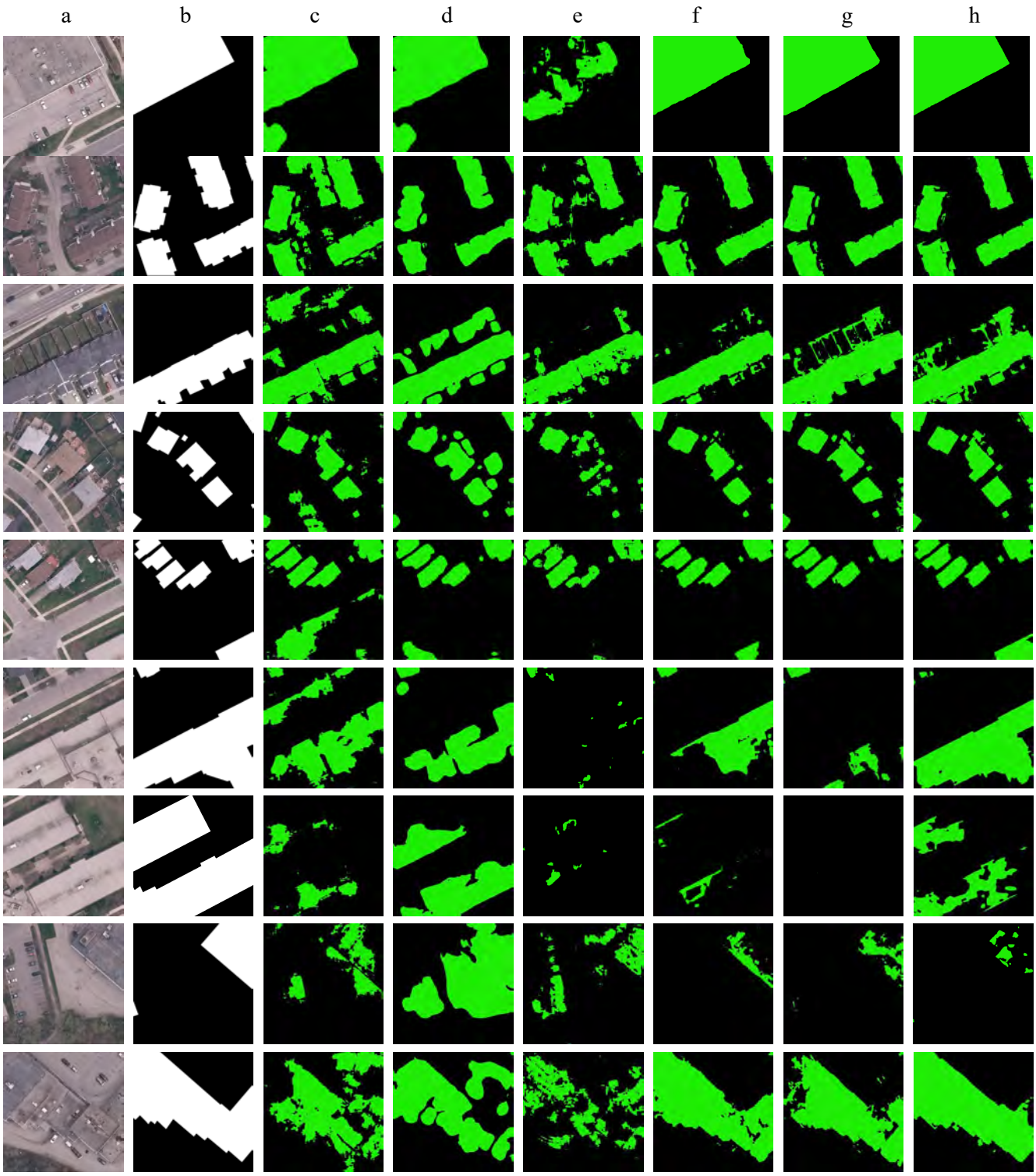
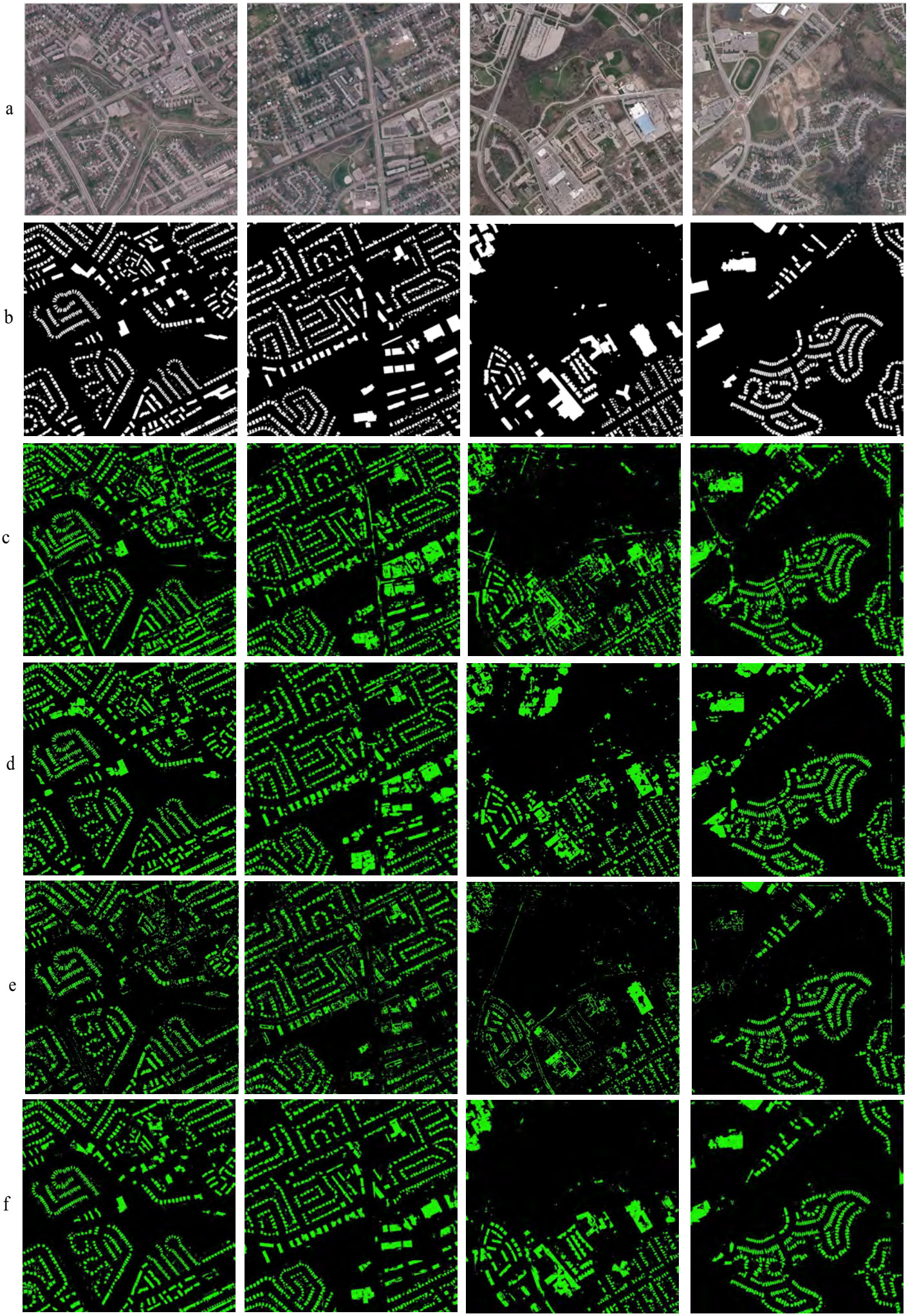


Figure 5.1 *Example inference of cut tiles (512×512) from large images and labels. Colume a, b, c, d, e, f, g h are respectively image, label, the prediction of U-Net, SegNet, RefineNet, BiseNet, PAN and DeepLabv3+*


a
b
c
d
e
f

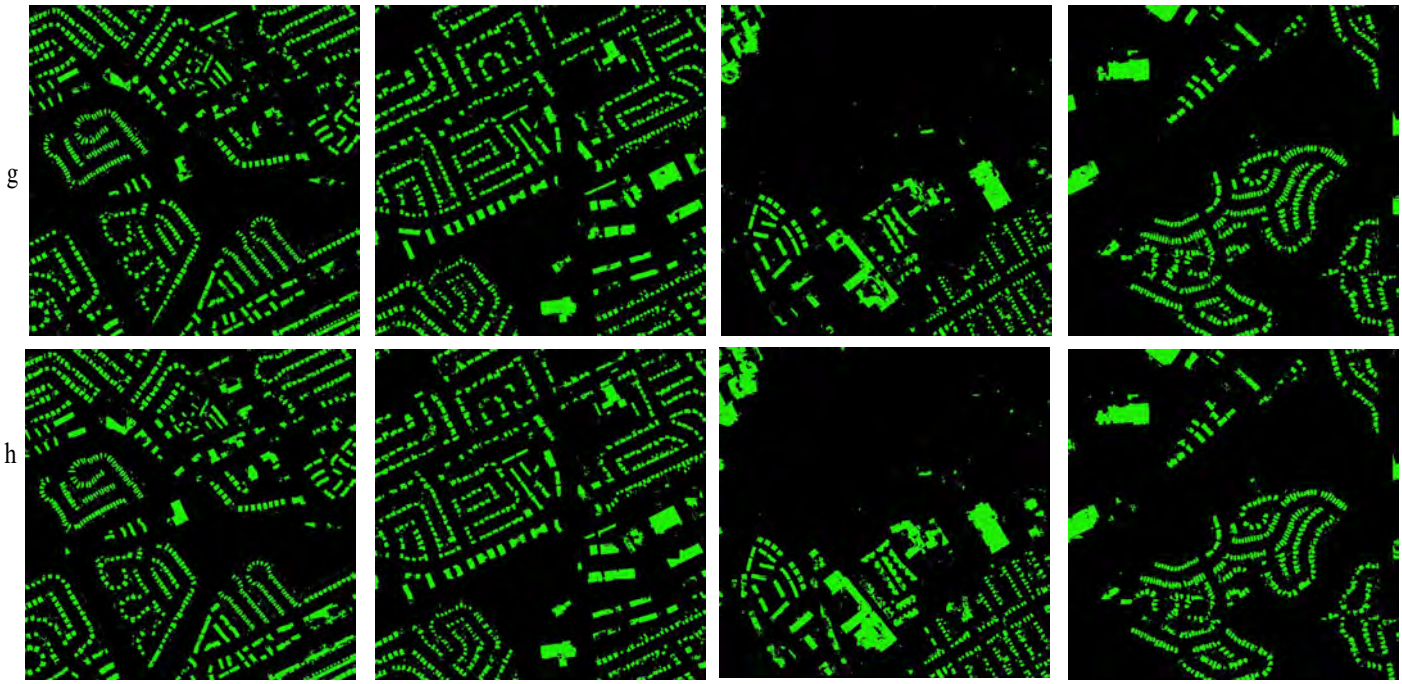


Figure 5.2 *Five examples predictions from testing data (8360×8360), Row a, b, c, d, e, f, g, h are respectively image, label, the inference of: U-Net, SegNet, RefineNet, PAN, BiSeNet and DeepLabV3+*

5.3. Discussion

Osawa et al. (2019) suggest that the model training of deep learning was a practical experiment. An amount of model verification can only be completed through training. At the same time, deep learning had many network structures and hyper-parameters, so it needs repeated attempts. Moreover, training deep learning models required GPU hardware support and more training time. How to effectively train deep learning models has gradually become a subject of knowledge. With the application of existing hardware facilities introduced in Section 4.2, some difficulties were overcome in training these deep learning models. This chapter pointed out that, errors and difficulties in the training process and presented some strategies for hyperparameter tuning to address these problems.

5.3.1. Loss Function

All algorithms in deep learning rely on the minimization or maximization of the loss function. The loss function is a measure of the effect of the predictive model in predicting the expected outcome. Building recognition is a binary classification problem in computer vision, and found that implementing a multi-class classification loss does not work well in the training process of the state-of-the-art models under the Tensorflow framework. Firstly attempt to use the multi-class of cross-entropy loss and focal loss in training, but found the model did not converge well but the loss value remains high. At this point, adjusting the learning rate and batch size did not solve the problem. Then focal loss softmax was changed to focal loss sigmoid, substituting the loss function with binary focal loss by making the training process successful, and the models successfully converged around epoch 17 to 20, as shown in Figure 5.3. Also, from the figure, the loss

value for RefineNet while the the model was more converged than the other models. This fact also explained why RefineNet produced the most inaccurate predictions among the other models.

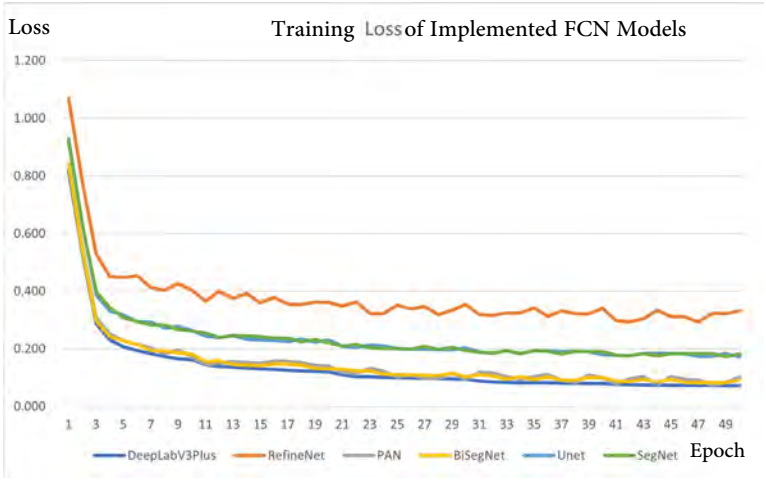


Figure 5.3 *The training Loss curve on training set and validation set of implemented deep learning models*

5.3.2. Hyper-parameter tuning

The loss value of the neural network model was not decreased, which is a very troublesome problem when we trained a neural network model, leading to training failure and poor model performance in the subsequent inference steps. From the previous section 5.3.1,although some models have successfully converged, the loss values on the training and validation datasets were not adequately low and still showed the decreased trend in the epochs 50. There was still a margin for further optimization of model performance through hyperparameter tuning. Due to the limited GPU memory, increasing the batch size crushed the program in the middle of training for some models with larger trainable parameters. The total parameters (million) for each model with the implemented backbone models are shown in Table 5.2. It could be seen that RefineNet had the largest number of parameters, with about 58.8 million parameters. This indicates that the RefineNet with a pre-trained ResNet50 as the backbone was the most complex model in

the experimental framework, implying that it was the most difficult to train. In the process of training RefineNet on WBD I found that increasing the batch size to 16 caused the program to crash at the third epoch. I solved the problems that loss value does not drop for a long time or the training process broke down by decreasing the batch size and increasing the initial learning rate to reduce the fluctuation range of weight parameters, thus reducing the possibility of increasing weight (Zulkifli, 2018). In general, the larger the batch size is, the more accurate the gradient descent direction in a particular range is, and the smaller training vibrations are. However, in the experiment, when the batch size is large to a certain extent, the number of iterations required to run an epoch (the full data set) reduced, but the time it required to achieve the same accuracy significantly increased, implying a slower correction for parameters. This paper reports several fundamental research progresses on the building

Table 5.2 *Complexity comparison of the state-of-art semantic segmentation models*

Model	Parameters(m)
U-Net	25.8
SegNet	45.3
RefineNet	58.8
PAN	23.7
BiSeNet	29.2
DeepLabV3+	40.4

extraction from high-resolution remote sensing images. Moreover, the open-source WBD was used to train and compare several state-of-the-art semantic segmentation models. The deficiency of this experiment was that the model cannot be trained for a long time due to the limitation of experimental conditions, and the adjustment of parameters is relatively rough. From Figure 5.3 there still have the trend of going down the loss values, if the training epoch was increased to 100 or even more, the models would be more properly trained and the prediction results would be better. Meanwhile, this experiment

can also be improved by setting up different training groups, such as 50, 100, and 200, and compare the results. Implementing different loss functions and backbone models can also facilitate model comparison.

6. CONCLUSION

The research in this paper shows that the convolutional neural network has great potential for the automatic extraction of ground object information from high-resolution earth observation images. Currently, there is an increasing number of FCN based semantic segmentation models that have been introduced to the high-resolution field of earth observation imagery for building rooftop extraction. This paper studies the evolution of six selected FCN semantic segmentation models: U-Net, SegNet, RefineNet, DeeplabV3+, PAN, and BiSeNet; the former four models have been widely applied for automatic building extraction tasks. However, there is a lack of studying on PAN, BiSeNet in the field of research since they are newly proposed models in recent years. The result showed that under the implementation details used in this experiment, DeeplabV3+ outperformed the other models in OA, Recall, IoU, mIoU, and $F_1{-}score$ reaching 98.21%, 88.34% 76.96%, 87.3%, 90.01%, respectively. Furthermore, U-Net is the fastest model with the highest FPS 24.5 in the training process due to its simple structure and fewer trainable parameters. Meanwhile, the qualitative analysis revealed defects of the selected baseline models. For example, the internal pixels of large buildings cannot consistently be predicted. There were also misclassifications due to the spectral characteristics of small buildings such as garages with their surrounding environment.

The state-of-art semantic segmentation models trained in this research paper could serve as a baseline model in the research and still had some shortcomings. In future

work, a backbone deep network structure such as ResNet with more than 100 network layers should be added. While searching for new architectures, backbones of different complexity can be tested for speed and performance. Meanwhile, deriving a more adaptable activation function or a higher-level loss function will improve the accuracy of the model, which is something that future work will explore. It is hoped that the open-source database WBD and the Fully Connected Networks can promote the further development of building extraction research, and finally achieve the automation and intelligence of building semantic segmentation.

REFERENCES

Aasen, H., Honkavaara, E., Lucieer, A., & Zarco-Tejada, P. J. (2018). Quantitative remote sensing at ultra-high resolution with UAV spectroscopy: A review of sensor technology, measurement procedures, and data correction workflows. *Remote Sensing*, *10*(7), 1091.

Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017, August). Understanding of a convolutional neural network. In *2017 International Conference on Engineering and Technology (ICET)* (pp. 1-6). IEEE.

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

Attarzadeh, R., & Momeni, M. (2018). Object-based rule sets and its transferability for building extraction from high resolution satellite imagery. *Journal of the Indian Society of Remote Sensing*, *46*(2), 169-178.

Badrinarayanan, V., Handa, A., & Cipolla, R. (2015). SegNet: A deep convolutional encoder-decoderarchitecture for robust semantic pixel-wise labelling. *arXiv preprint arXiv:1505.07293*.

Benjdira, B., Ammar, A., Koubaa, A., & Ouni, K. (2020). Data-efficient domain adaptation for semantic segmentation of aerial imagery using generative adversarial networks. *Applied Sciences*, *10*(3), 1092.

Cai, Y., He, H., Yang, K., Fatholahi, S. N., Ma, L., Xu, L., & Li, J. (2021). A Comparative Study of Deep Learning Approaches to Rooftop Detection in Aerial Images. *Canadian Journal of Remote Sensing*, 1-19.

Chen, L. C., Zhu, Y., Papandreou, G., Schroff, F., & Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 801-818).

Chen, Z., Li, D., Fan, W., Guan, H., Wang, C., & Li, J. (2021). Self-Attention in Reconstruction Bias U-Net for Semantic Segmentation of Building Rooftops in Optical Remote Sensing Images. *Remote Sensing*, *13*(13), 2524.

Cui, W. H., Feng, X., & Qin, K. (2014, March). The pixel rectangle index used in object based building extraction from high resolution images. IOP Conf. Series: Earth and Environmental Science (2014), p. 012233.

Dargan, S., Kumar, M., Ayyagari, M. R., & Kumar, G. (2020). A survey of deep learning and its applications: a new paradigm to machine learning. *Archives of Computational Methods in Engineering*, *27*(4), 1071-1092.

Fu, J., Liu, J., Tian, H., Li, Y., Bao, Y., Fang, Z., & Lu, H. (2019). Dual attention network for scene segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 3146-3154).

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., ... & Chen, T. (2018). Recent advances in convolutional neural networks. *Pattern Recognition*, *77*, 354-377.

Guo, J., Pan, Z., Lei, B., & Ding, C. (2017). Automatic color correction for multisource remote sensing images with Wasserstein CNN. *Remote Sensing*, *9*(5), 483.

Guo, Y., Liu, Y., Georgiou, T., & Lew, M. S. (2018). A review of semantic segmentation using deep neural networks. *International Journal of Multimedia Information Retrieval*, *7*(2), 87-93.

Han, Y., & Ye, J. C. (2018). Framing U-Net via deep convolutional framelets: Application to sparse-view CT. *IEEE Transactions on Medical Imaging*, *37*(6), 1418-1429.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770-778).

Huang, J., Zhang, X., Xin, Q., Sun, Y., & Zhang, P. (2019). Automatic building extraction from high-resolution aerial images and LiDAR data using gated residual refinement network. *ISPRS Journal of Photogrammetry and Remote Sensing*, *151*, 91-105.

Huang, X., & Zhang, L. (2011). Morphological building/shadow index for building extraction from high-resolution imagery over urban areas. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *5*(1), 161-172.

Ji, S., Wei, S., & Lu, M. (2018). Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set. *IEEE Transactions on Geoscience and Remote Sensing*, *57*(1), 574-586.

Jin, X., & Davis, C. H. (2005). Automated building extraction from high-resolution satellite imagery in urban areas using structural, contextual, and spectral information. *EURASIP Journal on Advances in Signal Processing*, *2005*(14), 1-11.

Khalel, A., & El-Saban, M. (2018). Automatic pixelwise object labeling for aerial imagery using stacked u-nets. *arXiv preprint arXiv:1803.04953*.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, *25*, 1097-1105.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278-2324.

Lenc, K., & Vedaldi, A. (2015). Understanding image representations by measuring their equivariance and equivalence. In *Proceedings of the IEEE Conference on Computer Vvision and Pattern Recognition* (pp. 991-999).

Li, H., Xiong, P., An, J., & Wang, L. (2018). Pyramid attention network for semantic segmentation. *arXiv preprint arXiv:1805.10180*.

Liasis, G., & Stavrou, S. (2016). Building extraction in satellite images using active contours and colour features. *International Journal of Remote Sensing*, *37*(5), 1127-1153.

Lin, G., Milan, A., Shen, C., & Reid, I. (2017). RefineNet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1925-1934).

Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 2980-2988).

Liow, Y. T., & Pavlidis, T. (1990). Use of shadows for extracting buildings in aerial images. *Computer Vision, Graphics, and Image Processing*, *49*(2), 242-277.

Liu, G., & Zou, J. (2018). Level set evolution with sparsity constraint for object extraction. *IET Image Processing*, *12*(8), 1413-1422.

Liu, H., Luo, J., Huang, B., Hu, X., Sun, Y., Yang, Y., ... & Zhou, N. (2019). DE-Net: Deep encoding network for building extraction from high-resolution remote sensing imagery. *Remote Sensing*, *11*(20), 2380.

Liu, S., & Huang, D. (2018). Receptive field block net for accurate and fast object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 385-400).

Liu, W., Yang, M., Xie, M., Guo, Z., Li, E., Zhang, L., ... & Wang, D. (2019). Accurate building extraction from fused DSM and UAV images using a chain fully convolutional neural network. *Remote Sensing*, *11*(24), 2912.

Liu, Y., Fan, B., Wang, L., Bai, J., Xiang, S., & Pan, C. (2018). Semantic labeling in very high resolution images via a self-cascaded convolutional neural network. *ISPRS Journal of Photogrammetry and Remote Sensing*, *145*, 78-95.

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 3431-3440).

Ma, W., Wan, Y., Li, J., Zhu, S., & Wang, M. (2019). An automatic morphological at- tribute building extraction approach for satellite high spatial resolution imagery. *Remote Sensing*, *11*(3), 337.

Maggiori, E., Tarabalka, Y., Charpiat, G., & Alliez, P. (2017). Can semantic labeling methods generalize to any city? the inria aerial image labeling benchmark. *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 3226–3229.

Mahabir, R., Croitoru, A., Crooks, A. T., Agouris, P., & Stefanidis, A. (2018). A critical review of high and very high-resolution remote sensing approaches for detecting and mapping slums: Trends, challenges and emerging opportunities. *Urban Science*, *2*(1), 8.

Masci, J., Meier, U., Cireşan, D., & Schmidhuber, J. (2011, June). Stacked convolutional auto-encoders for hierarchical feature extraction. In *International conference on artificial neural networks* (pp. 52-59).

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

Mohamed, A. R., Dahl, G., & Hinton, G. (2009). Deep belief networks for phonerecognition. In *Nips Workshop on Deep Learning for Speech Recognition and Related Applications* (Vol. 1, No. 9, p. 39).

Osawa, K., Swaroop, S., Jain, A., Eschenhagen, R., Turner, R. E., Yokota, R., & Khan, M. E. (2019). Practical deep learning with Bayesian principles. *arXiv preprint arXiv:1906.02506*.

O'Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*.

Papageorgiou, C., & Poggio, T. (2000). A trainable system for object detection. *International Journal of Computer Vision*, *38*(1), 15-33.

Paszke, A., Chaurasia, A., Kim, S., & Culurciello, E. (2016). Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv preprint arXiv:1606.02147*.

Pröve, P. L. (2017). An introduction to different types of convolutions in deep learning. *Online URL: https://towardsdatascience. com/types-ofconvolutions-in-deep-learning-717013397f4d*.

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., Prabhat, (2019). Deep learning and process understanding for data-driven earth system science. *Nature*, *566*(7743), 195–204.

Romera, E., Alvarez, J. M., Bergasa, L. M., & Arroyo, R. (2017). Erfnet: Efficient residual factorized convnet for real-time semantic segmentation. *IEEE Transactions on Intelligent Transportation Systems*, *19*(1), 263–272.

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computingand Computer-assisted Intervention* (pp. 234-241).

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, *115*(3), 211–252.

Schroff, F., Criminisi, A., & Zisserman, A. (2008, September). Object Class Segmentation using Random Forests. In *BMVC* (pp. 1-10).

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Smith, L. N. (2017, March). Cyclical learning rates for training neural networks. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)* (pp. 464-472). IEEE.

Smith, L. N. (2018). A disciplined approach to neural network hyper-parameters: Part 1– learning rate, batch size, momentum, and weight decay. *arXiv preprint arXiv:1803.09820*.

Song, J., Gao, S., Zhu, Y., & Ma, C. (2019). A survey of remote sensing image classification based on CNNs. *Big Earth Data*, *3*(3), 232-254.

Sozzi, M., Marinello, F., Pezzuolo, A., & Sartori, L. (2018). Benchmark of satellites image services for precision agricultural use. In *Proceedings of the AgEng Conference, Wageningen,The Netherlands* (pp. 8-11).

Sun, C., Shrivastava, A., Singh, S., & Gupta, A. (2017). Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 843-852).

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1-9).

Transon, J., d'Andrimont, R., Maugnard, A., & Defourny, P. (2018). Survey of hyperspec- tral earth observation applications from space in the sentinel-2 context. *Remote Sensing*, *10*(2), 157.

Wang, H., Wang, Y., Zhang, Q., Xiang, S., & Pan, C. (2017). Gated convolutional neural network for semantic segmentation in high-resolution images. *Remote Sensing*, *9*(5), 446.

Wang, S., Hou, X., & Zhao, X. (2020). Automatic building extraction from high-resolution aerial imagery via fully convolutional encoder-decoder network with non-local block. *IEEE Access*, *8*, 7313–7322.

Wang, W., Ooi, B. C., Yang, X., Zhang, D., & Zhuang, Y. (2014). Effective multi-modal retrieval based on stacked auto-encoders. *Proceedings of the VLDB Endowment*, *7*(8), 649–660.

You, K., Long, M., Wang, J., & Jordan, M. I. (2019). How does learning rate decay help modern neural networks? *arXiv preprint arXiv:1908.01878*.

Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., & Sang, N. (2018). BiseNet: Bilateral segmentation network for real-time semantic segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 325-341).

Zhang, Q., Huang, X., & Zhang, G. (2016). A morphological building detection frame- work for high-resolution optical imagery over urban areas. *IEEE Geoscience and Remote Sensing Letters*, *13*(9), 1388–1392.

Zhang, X., Han, L., Han, L., & Zhu, L. (2020). How well do deep learning-based methods for land cover classification and object detection perform on high resolution remote sensing imagery?. *Remote Sensing*, *12*(3), 417.

Zhang, Z., & Wang, Y. (2019). Jointnet: A common neural network for road and building extraction. *Remote Sensing*, *11*(6), 696.

Zhao, Z. Q., Zheng, P., Xu, S. T., & Wu, X. (2019). Object detection with deep learning: A review. *IEEE Transactions on Neural Networks and Learning Systems*, *30*(11), 3212-3232.

Zulkifli, H. (2018). Understanding learning rates and how it improves performance in deep learning. *Towards Data Science*, *21*, 23.