

博士論文

Semantic Segmentation for Multi-Source Remote Sensing
Imagery based on Convolutional Neural Networks
(Convolutional Neural Networks を用いた多様なリモートセン
シング画像の領域分割に関する研究)

郭 直靈
Guo, Zhiling

THE UNIVERSITY OF TOKYO

DOCTORAL THESIS

Semantic Segmentation for Multi-Source
Remote Sensing Imagery based on
Convolutional Neural Networks

Author:

Guo Zhiling

Supervisor:

Prof. Shibasaki Ryosuke

*A thesis submitted in fulfilment of the requirements
for the degree of Doctor of Philosophy*

in the

Shibasaki Lab

Department of Socio-Cultural Environmental Studies

July 2020

Declaration of Authorship

I, Guo Zhiling, declare that this thesis titled, ‘Semantic Segmentation for Multi-Source Remote Sensing Imagery based on Convolutional Neural Networks’ and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

THE UNIVERSITY OF TOKYO

Abstract

Graduate School of Frontier Sciences
Department of Socio-Cultural Environmental Studies

Doctor of Philosophy

Semantic Segmentation for Multi-Source Remote Sensing Imagery based on Convolutional Neural Networks

by Guo Zhiling

In this dissertation, we creatively investigated the feasibility of applying deep learning methods in different semantic segmentation tasks via multi-source remote sensing imagery. The comprehensive researches including village mapping, urban building extraction, super-resolution integrated method, change detection, slum mapping, map segmentation, etc., are conducted. The proposed methods mentioned above are developed by our open source computer vision package named as GeoVision, which contains subpackage GeoSeg and GeoSR, to facilitate the development of the deep learning based segmentation and super-resolution models, respectively. In village mapping, we present the Ensemble Convolutional Neural Network (ECNN), an elaborate CNN frame formulated based on ensembling state-of-the-art CNN models, to identify village buildings from open high-resolution remote sensing (HRRS) images. First, to optimize and mine the capability of CNN for village mapping and to ensure compatibility with our classification targets, a few state-of-the-art models were carefully optimized and enhanced based on a series of rigorous analyses and evaluations. Second, rather than directly implementing building identification by using these models, we exploited most of their advantages by ensembling their feature extractor parts into a stronger model called ECNN based on the multiscale feature learning method. Finally, the generated ECNN was applied to a pixel-level classification frame to implement object identification. The experimental results obtained from the test area in Savannakhet province, Laos, prove that the proposed ECNN model significantly outperforms existing methods, improving overall accuracy from 96.64% to 99.26%, and kappa from 0.57 to 0.86. As for urban building semantic segmentation part, we investigate the feasibility of applying FCN-based method in conducting map semantic segmentation. Here, high resolution aerial imagery of Tokyo, which provide sufficient information about land features, as a representative sample to perform data source. To mitigate the impact of color difference on segmentation performance, the color transform methods are utilized in image preprocessing as well. In

terms of DCNNs model, a specific deep learning architecture named concatenate feature pyramid networks (CFPN), which is a variant of FCN, is proposed based on feature concatenation and feature pyramid methods. Given the variety of the buildings as well as the limited training dataset, CFPN model is deliberately designed in lightweight structure with relatively few parameters, which could be trained easily. Meanwhile, with the help of feature concatenation and feature pyramid, CFPN is capable of extracting adequate robust feature from complex texture to perform building segmentation with high accuracy, and outperform other baselines by 3.55% to 7.89%. Since multi-source remote sensing imagery has become widely accessible owing to the development of data acquisition systems, we address the challenging task of the semantic segmentation of buildings via multi-source remote sensing imagery with different spatial resolutions. Unlike previous works that mainly focused on optimizing the segmentation model, which did not enable the severe problems caused by the unaligned resolution between the training and testing data to be fundamentally solved, we propose to integrate SR techniques with the existing framework to enhance the segmentation performance. The feasibility of the proposed method was evaluated by utilizing representative multi-source study materials: high-resolution (HR) aerial and low-resolution (LR) panchromatic satellite imagery as the training and testing data, respectively. Instead of directly conducting building segmentation from the LR imagery by using the model trained using the HR imagery, the deep learning-based super-resolution (SR) model was first adopted to super-resolved LR imagery into SR space, which could mitigate the influence of the difference in resolution between the training and testing data. The experimental results obtained from the test area in Tokyo, Japan, demonstrate that the proposed SR-integrated method significantly outperforms that without SR, improving the Jaccard index and kappa by approximately 19.01% and 19.10%, respectively. The results confirmed that the proposed method is a viable tool for building semantic segmentation, especially when the resolution is unaligned. After that, we expand the proposed methods mentioned above to more challenging applications including change detection, slum mapping, and map semantic segmentation. As for change detection, color normalization, super-resolution, and image registration methods are adopted to balance the training and testing datasets, after that, by adopting proposed CFPN model and image difference, the identification of land change can be achieved. In terms of slum mapping, here CFPN is adopted to perform multi-class semantic segmentation, the impact of resolution on slum segmentation is discussed as well. Furthermore, the important GIS-related task: map semantic segmentation, which aims at digitising historical maps is also applied by our deep learning model. The experimental results reveal that our proposed method can serve as a viable tool for semantic segmentation tasks via multi-source remote sensing imagery with high accuracy and efficiency.

Acknowledgements

I would like to thank all the people who contributed in some way to the work described in this dissertation. First and foremost, I thank my academic supervisors, Professor Shibasaki Ryosuke, Shao Xiaowei, and Zheng Yinqiang, for engaging me in new ideas, helping and encouraging me when I face difficulties. Additionally, I would like to thank my vice supervisors for this help in my work.

I would like to acknowledge Shao's group member: Ph.D. Wu Guangming, Shi Xiaodan, Sun Minzhou, Ph.D. Zhao Suwen, Ph.D. Xu Yongwei, and Ph.D. Chen Qi. I greatly benefited from their keen scientific insight and attitude of life.

I also gratefully acknowledge the project sources and financial support that allowed me to pursue my doctoral studies: JSPS (Japan Society for the Promotion of Science) and U-Tokyo Fellowship.

I am grateful for the NTT-GEOSPACE and the National Topographic Office of New Zealand for their kind providing of the training and testing data as well as SAKURA Internet Inc. for providing me the KOUKARYOKU GPU server for the experiment.

I would like to thank the various members with whom I had the opportunity to work and have not already mentioned: Ph.D. Zhang Haoran, Ph.D. Fan Zipei, Ph.D. Yuan Wei, Ph.D. Jiang Renhe, Ph.D. Chen Quanjun, Ph.D. Miyazaki Hiroyuki, Ph.D. Ohira Wataru, Xia Tianqi, Huang Dou, Lian Xinlei, Cheng Qianwei. . .

I would like to acknowledge the University of Tokyo and the Department of Socio-Cultural Environmental Studies. My study experience here would greatly benefit my whole life.

Finally, I would like to acknowledge my dear friends, family, and wife Kaku-saru who always supported me during my time here.

...

Contents

Declaration of Authorship	i
Abstract	ii
Acknowledgements	iv
Contents	v
List of Figures	viii
List of Tables	x
Abbreviations	xi
1 Introduction	1
1.1 Research Background	1
1.2 Related Works	4
1.3 Research Objectives	6
1.3.1 Village Mapping via Patch-based CNNs	7
1.3.2 FCN-based Building Semantic Segmentation in Urban Areas	7
1.3.3 Super-Resolution Integrated Semantic Segmentation	8
1.3.4 Expanding Applications	10
1.3.5 Standard Package	10
1.4 Outline of the Dissertation	11
2 Village Mapping via Patch-based CNNs	12
2.1 Data	14
2.1.1 Study Area	14
2.1.2 Data Source	15
2.2 Methods	16
2.2.1 Convolutional Neural Networks	16
2.2.2 Model Optimization	18
2.2.2.1 Influence of Filter	19
2.2.2.2 Influence of Depth	20
2.2.2.3 Influence of Window Size	21
2.2.3 AlexNet-Like	23

2.2.4	VGGNet-Like	23
2.2.5	GoogLeNet-like	25
2.2.6	SqueezeNet-Like	26
2.2.7	Ensemble Convolutional Neural Networks	27
2.3	Result and Discussion	28
2.3.1	Comparison of Different Models	29
2.3.2	Implementation of CCNs	30
2.4	Conclusions	34
3	FCN-based Building Semantic Segmentation in Urban Areas	35
3.1	Data	37
3.1.1	Study Area	37
3.1.2	Data Source	37
3.2	Methods	38
3.2.1	Preprocessing	38
3.2.2	Concatenate Feature Pyramid Networks	39
3.2.3	Assessment Criteria	41
3.3	Results	42
3.4	Discussion	46
3.4.1	Robustness	46
3.4.2	The Impact of Image Color	48
3.5	Conclusions	49
4	Super-Resolution Integrated Building Semantic Segmentation	50
4.1	Data	52
4.1.1	Study Area	52
4.1.2	Data Source	53
4.2	Methods	54
4.2.1	Data Preprocessing	54
4.2.2	Segmentation Model	55
4.2.3	SR Model	56
4.2.4	Assessment Criteria	58
4.3	Results	58
4.4	Discussion	65
4.5	Conclusions	68
5	Practical Application	69
5.1	Change Detection	69
5.2	Slum Mapping	72
6	Conclusions and Future Works	76
6.1	Conclusions	76
6.2	Future Works	77
A	Appendix I: GeoVision	79
A	GeoSeg	79
1	Related work	80
2	Experiments	81

2.1	Benchmark Dataset	81
2.2	Implementation	81
	Code Organization	81
	Models	82
3	Results and Discussion	84
3.1	Qualitative Result	84
3.2	Quantitative Result	84
3.3	Computational efficiency	85
4	Conclusion	87
B	GeoSR	88
1	Methodology	89
1.1	Workflow	89
1.2	Data	90
	Source	90
	Preprocessing	90
1.3	Models	90
	Model Zoo	91
	Architectures	92
1.4	Logging Tools	92
1.5	Evaluation Metrics and Visualizations Tools	92
2	Results and Discussion	92
2.1	Qualitative Result	92
2.2	Quantitative Result	93
2.3	Computational Efficiency	94
3	Conclusion	94
B	Appendix II: Semantic Segmentation for Urban Planning Maps	95
A	Materials	96
1	Study Area	96
2	Data Source	97
B	Methods	98
C	Results	99
D	Discussion	103
C	Appendix III: List of Publications	105
A	International Journals	105
B	International Conferences	106
	Bibliography	107

List of Figures

1.1	Building information in Google Maps and OpenStreetMap	2
1.2	Buildings change example in urban areas	3
1.3	Visual interpretation	3
1.4	The proposed research framework	6
1.5	Example for building semantic segmentation based on identical training and testing data source	8
1.6	Example for the impact of resolution on segmentation results	9
1.7	Example for super-resolution applied on satellite imagery	9
2.1	Study area	15
2.2	Workflow	17
2.3	Influence of filter amount	20
2.4	Window size: scenario 3 with simple structure	22
2.5	AlexNet-like architecture	24
2.6	VGGNet-like architecture	24
2.7	Reconstruction of CNN activations from different layers of VGGNet-like.	25
2.8	GoogleNet-like architecture	26
2.9	SqueezeNet-like architecture	27
2.10	Ensemble Convolutional Neural Networks.	28
2.11	Identification results of eight small segments in bank regions.	31
2.12	Identification results of eight small segments in mixed-type regions.	32
2.13	Identification results of eight small segments in artificial land regions.	33
3.1	Color balance for preprocessing	39
3.2	Model architecture	40
3.3	Quantitative results for test region 1	43
3.4	Quantitative results for test region 2	44
3.5	Quantitative results for test region 3	45
3.6	Quantitative results for test region 4	46
3.7	Subregion comparison for different models	47
3.8	Average performance of different models	48
3.9	Building outline extraction and denoise	49
4.1	Data	53
4.2	Framework of our building semantic segmentation method	55
4.3	Qualitative results for test region 1	59
4.4	Qualitative results for representative subregions in test region 1	60
4.5	Qualitative results for test region 2	61

4.6	Qualitative results for representative subregions in test region 2	62
4.7	Qualitative results for test region 3	62
4.8	Qualitative results for representative subregions in test region 3	63
4.9	Qualitative results for test region 4	64
4.10	Qualitative results for representative subregions in test region 4	64
4.11	Performance discussion	66
4.12	Results for important land features	67
4.13	Bad results caused by annotation	68
5.1	Semantic segmentation results comparison of 2016	71
5.2	Semantic segmentation results comparison of 2017	72
5.3	Selected semantic segmentation results in 2016 and 2017	73
5.4	Change detection with post-processing	73
5.5	New building examples	74
5.6	New vacant examples	74
5.7	Slum mapping result example in Dhaka	74
5.8	Slum mapping results in different resolution	75
6.1	Catastrophe maps Generalization System	77
A.1	The code organization of Geoseg package	83
A.2	Visualization result	85
A.3	Comparison of segmentation performances	86
A.4	Comparison of computational efficiency	87
A.5	Super-resolution examples generated by GeoSR	88
A.6	The code organization of GeoSR package	90
A.7	SR comparison results of different images generated by different models	93
A.8	SR comparison results of different models	94
B.1	Study areas	97
B.2	Qualitative results for representative subregions in urban planning map of Shibuya district	100
B.3	Segmentation results obtained by different methods for urban planning map of Shinjuku district	100
B.4	Qualitative results for representative subregions in urban planning map of Shinjuku district	101
B.5	Segmentation results obtained by different methods for urban planning map of Taito district	102
B.6	Qualitative results for representative subregions in urban planning map of Taito district	102
B.7	Average performance comparison	103
B.8	Performance discussion	104
B.9	Map denoising and outline extraction	104

List of Tables

2.1	Relationship between filter amount and accuracy	20
2.2	Relationship between depth and accuracy	21
2.3	VGGNet-like architecture	25
2.4	Training result by different CNNs.	29
2.5	Testing result by different structures	29
2.6	Bank regions testing result by different structures	31
2.7	Result 2	32
2.8	Result 3	33
3.1	Quantitative results for test region 1	44
3.2	Quantitative results for test region 2	44
3.3	Quantitative results for test region 3	45
3.4	Quantitative results for test region 4	45
3.5	Average performance of different models	47
3.6	The impact of image color on segmentation results	48
4.1	Quantitative results for test region 1	60
4.2	Quantitative results for test region 2	61
4.3	Quantitative results for test region 3	63
4.4	Quantitative results for test region 4	65
5.1	Training and testing dataset	70
5.2	The quantitative slum mapping results of Dhaka	75
5.3	The impact of resolution on slum mapping	75
A.1	Quantitative comparison of different models	93
A.2	Computational efficiency comparison	94
B.1	Training result	99
B.2	Testing result for Shibuya	99
B.3	Testing result for Shinjuku	101
B.4	Testing result for Taito	101
B.5	Average result for testing	103
B.6	Performance discussion via Std.	103
B.7	Computational efficiency comparison	104
B.8	Memory comparison of different models	104

Abbreviations

SDG	Sustainable Development Goals
GNSS	Global Navigation Satellite System
OSM	OpenStreetMap
TM	Landsat Thematic mapper
ETM+	Enhanced Thematic Mapper Plus
NOAA	National Oceanic and Atmospheric Administration
AVHRR	Advanced Very High Resolution Radiometer
LiDAR	Light Detection And Ranging
HRRS	High Resolution Remote Sensing
GE	Google Earth
RF	Random Forest
Adaboost	Adaptive Boosting
NN	Neural Networks
SVM	Super Vector Machine
NDBI	Normalized Difference Build-up Index
DCNN	Deep Convolutional Neural Networks
CNN	Convolutional Neural Networks
NLP	Natural Language Processing
SGD	Stochastic Gradient Descent
CUDA	Comput Unified Device Architecture
cuDNN	NVIDIA cuDA Deep Neural Network library
UAV	Unmanned Aerial Vehicles
GIS	Geographic Information System
ECNN	Ensemble Nonvolutional Neural Networks
SAR	Synthetic Aperture Radar

ILSVRC	ImageNet Large Scale Visual Recognition Challenge
AlexNet	CNN architecture developed by Alex Krizhevsky
VGGNet	very deep CNN architecture developed by Simonyan
GoogLeNet	CNN architecture developed by Christian Szegedy
SqueezeNet	CNN architecture developed by Forrest
RNN	Recurrent Neural Networks
FCNs	Full Convolutional Networks
UNet	U-shaped Convolutions Networks
FPN	Feature Pyramid Convolutions Networks
ResNet	A Deep Residual Netork
SegNet	A Deep Convolutional Encoder-Decoder for Image Segmentation
ResUNet	Residual UNet
SR	Super-Resolution
ISSR	Single Image Super-Resolution
SRCNN	Super-Resolution based CNN
FSRCNN	Accelerating the Super-Resolution Convolutional Neural Network
ESPCN	Efficient Sub-Pixel Convolutional Neural Network
VDSR	Accurate Image Super-Resolution Using Very Deep Convolutional Networks
DRCN	Deeply-Recursive Convolutional Network for Image Super
DRRN	Deep Recursive Residual Network
LapSRN	Deep Laplacian Pyramid Networks
SRDenseNet	Residual Dense Network for Image Super-Resolution
SRGAN	Single Image Super-Resolution Using a Generative Adversarial

Dedicated to the Peace on Earth...

Chapter 1

Introduction

1.1 Research Background

To achieve Sustainable Development Goals (SDG), the investigation of human settlements is essential, and the maps used to illustrate important land features and their distribution are indispensable and required in a wide range of fields. Important applications include village mapping, change detection, slum mapping, disaster response, and homeland security [1].

Given that accurate building maps are often unavailable or are outdated in undeveloped village areas, building identification in such areas has become a significant research field in remote sensing [1]. Figure 1.1 shows the rural environment maps of the same area in Foxdale Britain by Google maps and OSM respectively, we can find out the building information in Google maps compared with OSM is quite insufficient, which would bring the inconvenience. This is even the case in developed country, and there's no doubt that in developing countries the condition would be more severe. Insufficient building information in village leads to inconvenience and has several negative consequences [2]. First, in the event of a catastrophe, building maps are indispensable [3]. For instance, during catastrophic events such as the aftermath of the 2011 Tohoku earthquake and tsunami [4], land conditions change rapidly with secondary disasters such as landslides, tsunamis, and continual aftershocks [5]. To save victims and provide disaster relief in a convenient way, it is important to swiftly update the locations of residential buildings and information about other land features. Furthermore, in village planning, which aims to benefit village inhabitants, public facilities need to be developed based on information about the distribution of residential buildings [6]. In contrast to densely packed urban buildings, village buildings have distinct characteristics, for instance, they are sparsely scattered, change arbitrarily owing to the lack of regulation, and do not have distinct

architectural features. Moreover, village buildings are usually mixed with complex and diverse land features such as agricultural lands, mountains, and rivers [7]. Such complexity of spatial and structural patterns makes village building identification a fairly challenging problem, and the usage of building maps ensures that the tools used for building identification provide rapid, accurate, efficient, and time-sequenced results.

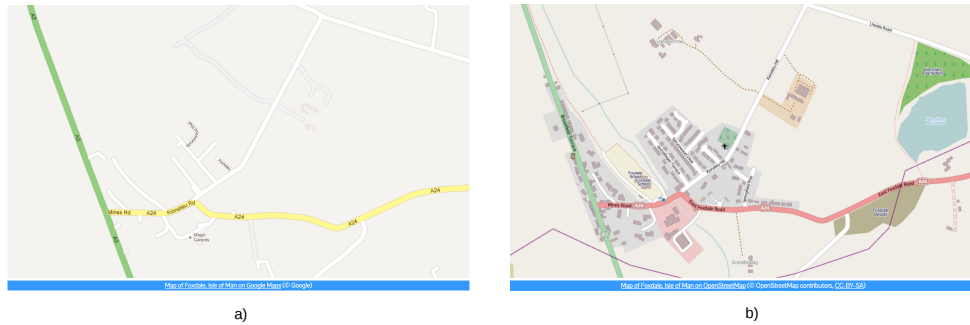


FIGURE 1.1: Map of Foxdale, Isle of Man on Google Maps(a) and OSM(b) respectively (2016.07).

In urban areas, although the maps are often provided, in many cases, the accurate outline of important land features such as buildings and landmarks for land use analysis are still unavailable. Additionally, due to the frequent changing of important land features, especially for rapidly developing cities, it is essential to be able to immediately update such changes for the purposes of urban planning and navigation[8]. Figure 1.2 shows an example of rapid urban change in Kashiwa, Japan. Because of the variety of background, building textures, densely packed features, and imaging conditions, the automatic extraction of building remains is a long-standing important and challenging task [9].

As for large-scale mapping, the slum semantic segmentation and localization is an important task for improving the sanitary condition, humanitarian, and living standard, as well as reducing crime and poverty in developing countries [10]. Since the characteristics of slum regions are very complicated, such as extremely high density, diverse shanty structures, non-uniform patterns and styles, the investigation is still mainly based on census and community survey, which hinders the sustainable development to a considerable extent.

Rather than the fieldwork and ground investigation, the semantic segmentation of land features depending on the multi-source remote sensing imagery would be more convenient and efficient. With the help of remote sensing imagery [11–13], earth-observation

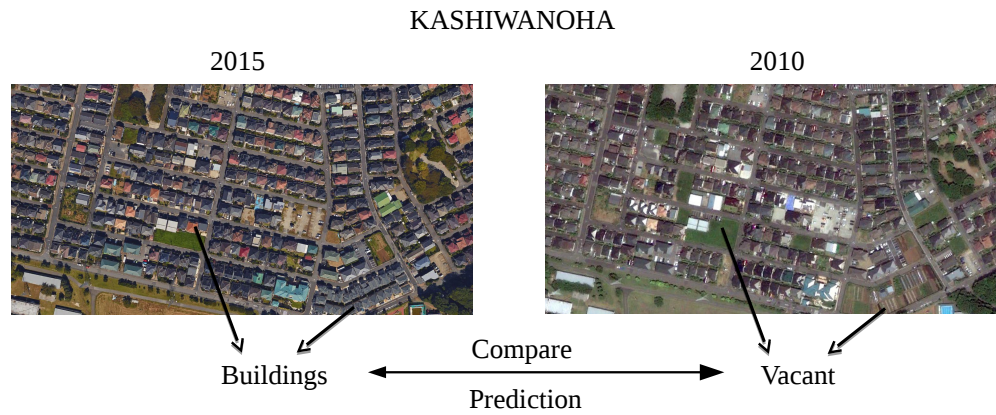


FIGURE 1.2: Buildings change example in urban areas.

activities on regional to global scales can be implemented owing to advantages such as wide spatial coverage and high temporal resolution [14, 15]. By analyzing the tone, texture, and geometric features from the remote sensing image, experts can recognize land features with high confidence [16, 17]. Consequently, we believe that remote sensing imagery can provide as good data sources for land feature semantic segmentation.

In terms of the semantic segmentation technique, many methods have been studied. Note that the traditional visual interpretation of remote sensing images is a very complex and time-consuming process. Although with very high accuracy, it is not suitable to large-scale automation projects. The Figure 1.3, which implemented by a medical group [18] in Nagasaki University, shows the building semantic segmentation result based on visual interpretation in Kenya. The manual visual interpretation work was taken several weeks and the related massive implementation seems impossible.

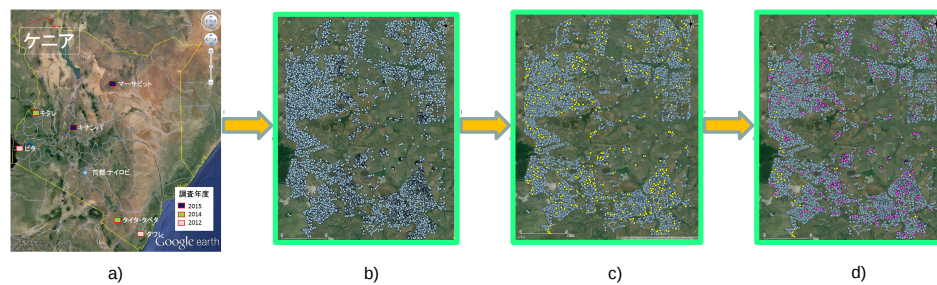


FIGURE 1.3: Identification of buildings in rural Environment based on Google Earth via visual interpretation.

In order to provide automatic and high accuracy segmentation result, with the help of image processing and feature extraction techniques, various machine learning algorithms [19] such as graph theory-based [20], clustering-based [21], Random Forest (RF) [22], Adaptive Boosting (AdaBoost) [23, 24], Neural Networks (NN) [25] and Super Vector Machine (SVM) [26, 27] in remote sensing have been implemented. For instance, Zhang et al. [28] combine the K-means method with AdaBoost to classify buildings, and the overall accuracy is about 90%. Zongur et al. [29] utilize satellite images to detect an airport runway using AdaBoost with a circular-Mellin feature. Using an improved Normalized Difference Build-up Index (NDBI) and remote sensing images, Li et al. [30] dynamically extract urban land. Cetin et al. [31] use textural features such as the mean and standard deviation of image intensity and gradient for building detection. For the identification of forested landslides, Dou et al. [32] utilize a case-based reasoning approach and Li et al. [33] adopt two machine learning algorithms: RF and SVM. When dealing with classifying complex mountainous forests via remote sensing images, Attarchi et al. [34] verify the performances of three machine learning methods: SVM, NN, and RF. For mapping urban areas of DMSP/OLS nighttime light and MODIS data, Jing et al. [35] also utilize SVM.

As shown above, most existing segmentation methods applied for land feature extraction can only generate low- or middle-level image features with limited representation ability, which essentially prevents them from achieving good performance in various scenes. Lately, in image segmentation field, the preponderance of deep learning methods such as convolutional neural networks (CNN) [36] and full convolutional networks (FCN) [37] has been proved owing to its advantages such as efficiently generating high-dimensional abstract feature and high accuracy performance. To facilitate the development of land feature extraction, we believe the comprehensive investigation of semantic segmentation for multi-source remote sensing imagery based on CNN is essential.

1.2 Related Works

Lately, the rapid development of deep convolutional neural networks (DCNN) [38] has led to the construction of several models that have achieved great success with the task of land feature semantic segmentation in terms of both accuracy and computational efficiency. The pioneering work on the topic can be traced to 2015, when Paisitkriangkrai et al. [39] proposed effective semantic pixel labeling using CNN and conditional random fields (CRF) [40] to perform building segmentation with competitive classification accuracy. Subsequently, in 2016, inspired by fully convolutional networks (FCNs) [37], Kampffmeyer et al. [41] designed architecture that allows end-to-end learning of the

pixel-to-pixel semantic segmentation for buildings, and small land features were proven to be detected accurately as well. In 2017, Guo et al. [42] utilized ensemble convolutional neural networks (ECNN) to identify village buildings by using Google's satellite map and Bing Maps with high accuracy. And the development of hourglass-shaped networks (HSNs) such as UNet [43] and SegNet [44] motivated Liu et al. [45] to propose an enhanced HSN. Their model included an inception module, which replaced the typically used convolutional layers, and which results in a network with multi-scale receptive areas with rich context. In contrast to studies that aimed to modify the structure of CNN, Bischke et al. (2017) [46] and Wu et al. (2018) [8] chose to optimize the loss function by applying multi-task loss and multi-constraint loss, respectively. The results demonstrated that optimization of the loss function could significantly improve the performance of classic FCNs in certain building segmentation tasks. In addition, to facilitate the development of parsing the earth through satellite imagery, a challenge named Deepglobe [47] was held during the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) in 2018, and the following is a brief overview of some representative studies. Zhao et al. [48] conducted extraction by using Mask R-CNN [49] with building boundary regularization. Delassus et al. [50] proposed a fusion strategy based on a deep combiner using segmentation of both the results of different CNNs and the input data to segment. By using a new FCN variant named TernaNet V2, Igloukov et al. [51] could extract buildings even at the instance level. Focusing on small buildings, Dickenson et al. [52] utilized CNN to output rotated rectangles for symbolized building footprint extraction. Li et al. [53] used a building extraction method based on ensemble learning to perform the segmentation. Furthermore, a recent study in 2019, Wu et al. [54] utilized stacked fully convolutional networks and a feature alignment framework for multi-label land-cover segmentation with high accuracy.

For other remote sensing imagery based pattern recognition tasks using the CNN method [55], Chen et al. [56] address vehicle detection, Li et al. [57] focus on building pattern classifiers, and Yue et al. [58] use both spectral and spatial features for hyperspectral image classification. To predict geoinformative attributes from large-scale images, Lee et al. [59] also choose CNN, and Sermanet et al. [60] utilize the CNN method to identify house numbers. Other important works such as Marmanis et al. [61] use pretrained CNN model and big dataset to classification land features while Ding et al. [62] add data Augmentation into CNN for SAT based building recognition. In high-resolution image processing, the innovated works conducted by Hu et al. [63], who use transfer learning to enhance obtained model in order to identify land features from HRRS and achieved an overall accuracy of approximately 98%, also Martin et al. [64] classify buildings using multiple CNN layers, pretrained model and K-meaning.

Despite their success in several land feature semantic segmentation tasks, the discussions on extraction via multi-source remote sensing imagery of which the spatial resolution differs are quite inadequate. With the dramatically increasing availability of new large-scale remote sensing data sources, the ever-expanding choices of datasets can be utilized in semantic segmentation tasks [65–67], and the case that training and testing datasets obtained from multiple sources with different resolution would be inevitable and ubiquitous in many practical applications [68].

1.3 Research Objectives

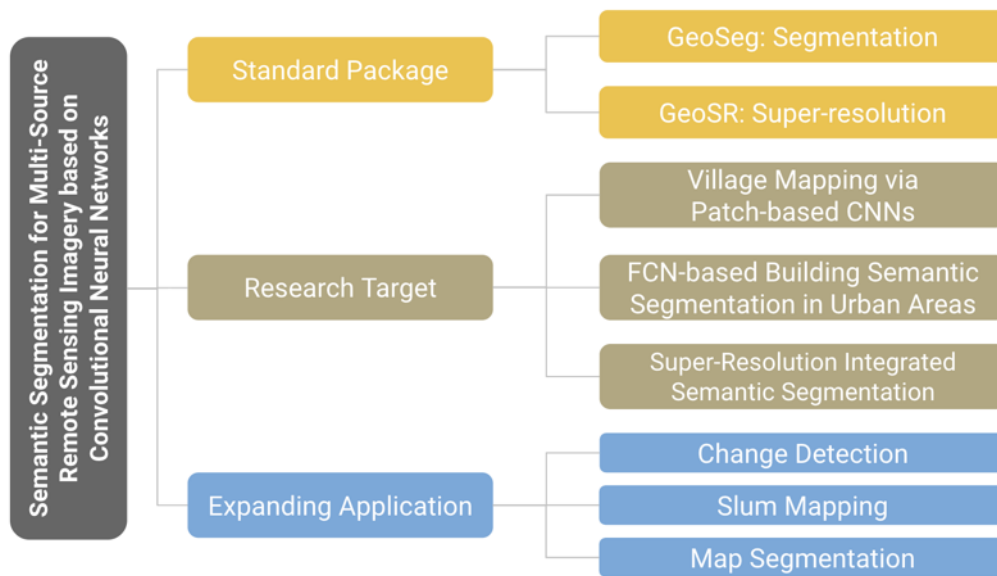


FIGURE 1.4: The proposed research framework.

In this dissertation, as shown in Figure 1.4, we creatively investigated the feasibility of applying deep learning methods in different semantic segmentation tasks via multi-source remote sensing imagery. To the best of our knowledge, there is no existing doctoral dissertation which provides such comprehensive research on village mapping, urban building extraction, change detection, slum mapping, etc. This section we will briefly introduce each task, respectively.

1.3.1 Village Mapping via Patch-based CNNs

We present an elaborate formulated CNN model called Ensemble Convolutional Neural Networks(ECNN). Different from related works, ECNN achieves multiscale feature learning by ensembling the feature extractor part of four optimized state-of-the-art models, and we apply it to implement the patch-based pixel-level village building segmentation task based on satellite imagery.

The main contributions of this part can be summarized as follows:

- We explored how to construct CNN architecture that can adapt to the village building segmentation task based on insightful and in-depth analysis.
- We optimized state-of-the-art CNN models by using rigorous principles to explore their potential for pixel-level building segmentation via HRRS images.
- We presented a novel CNN frame called ECNN based on multiscale feature learning by emsembling parallel optimized state-of-the-art CNN models.
- We implemented the proposed method for village building identification and found that it outperforms the existing state-of-the-art methods, achieving an overall accuracy and kappa coefficient of 99.26% and 0.86 respectively.

1.3.2 FCN-based Building Semantic Segmentation in Urban Areas

To investigate the feasibility of applying DCNNs in conducting map semantic segmentation, in this part, we take high resolution aerial imagery of Tokyo, which provide sufficient information about land features, as a representative sample to perform data source. To mitigate the impact of color difference on segmentation performance, the color transform methods are utilized in image preprocessing as well. In terms of DCNNs model, a specific deep learning architecture named concatenate feature pyramid networks (CFPN), which is a variant of FCN, is proposed based on feature concatenation [69] and feature pyramid [70] methods. Given the variety of the buildings as well as the limited training dataset, CFPN model is deliberately designed in lightweight structure with relatively few parameters, which could be trained easily. Meanwhile, with the help of feature concatenation and feature pyramid, CFPN is capable of extracting adequate robust feature from complex texture to perform building segmentation with high accuracy. The experimental results reveal that the proposed model could outperform other baselines and extract building polygon efficiently.

1.3.3 Super-Resolution Integrated Semantic Segmentation



FIGURE 1.5: Example for building semantic segmentation based on identical training and testing data source.

As show in Figure 1.5, building semantic segmentation based on identical training and testing data source can obtain relatively high accuracy results. However, differences between the resolution of the training and testing datasets would greatly influence building semantic segmentation, an representative example can be found in Figure 1.6, in which a segmentation model trained by 0.16m resolution aerial imagery is directly applied to test satellite image with resolution 2m. By adopting super-resolution methods on low resolution imagery (a qualitative example as shown in Figure 1.7), the results can be enhanced.

In this part, contrary to previous work, we propose to integrate super-resolution (SR) techniques into the existing segmentation framework to address the problem of building semantic segmentation in multi-source remote sensing imagery with different spatial resolution. To validate the feasibility of the proposed method, two high-performance DCNN-based models, namely efficient sub-pixel convolutional neural network (ESPCN) [71] and UNet, are adopted to perform SR and the semantic segmentation operation, respectively. In addition, three-band RGB HR aerial imagery and single-band grayscale LR panchromatic satellite imagery are selected as representative multi-source remote sensing imagery to conduct training and testing, respectively. It is worth emphasizing that, to the best of our knowledge, there has not been any empirical study using SR techniques for the building semantic segmentation from multi-source imagery with different resolution.

The main contributions of this study are three fold:

- We discussed the challenge and limitation of recent deep learning based studies on building semantic segmentation of building while under multi-source imagery with different resolution circumstance.

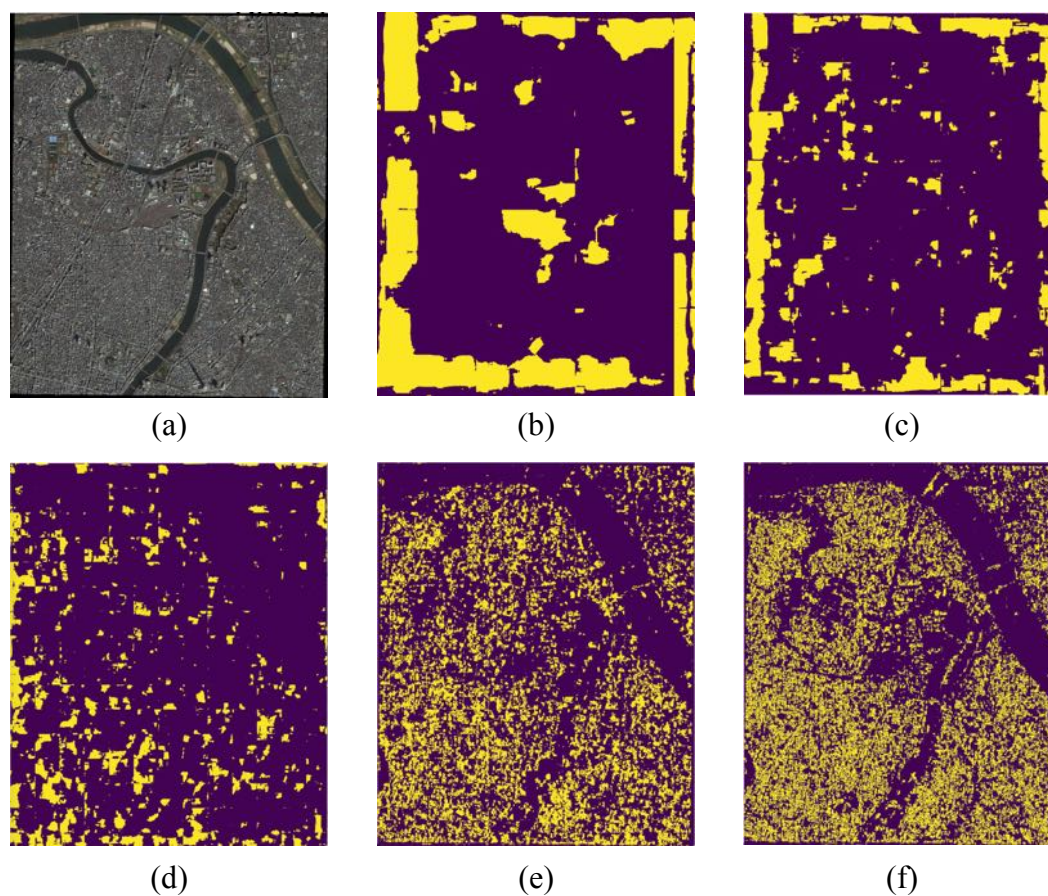


FIGURE 1.6: Example for the impact of resolution on segmentation results. (a) Original satellite image with resolution 2m; (b) building identification results of (a); (c) upscale resolution 2 times by super resolution to 1m; (d) upscale resolution 4 times by super resolution to 0.5m; (e) upscale resolution 8 times by super resolution to 0.25m; (f) upscale resolution 12 times by super resolution to 0.167m(similar with training dataset).

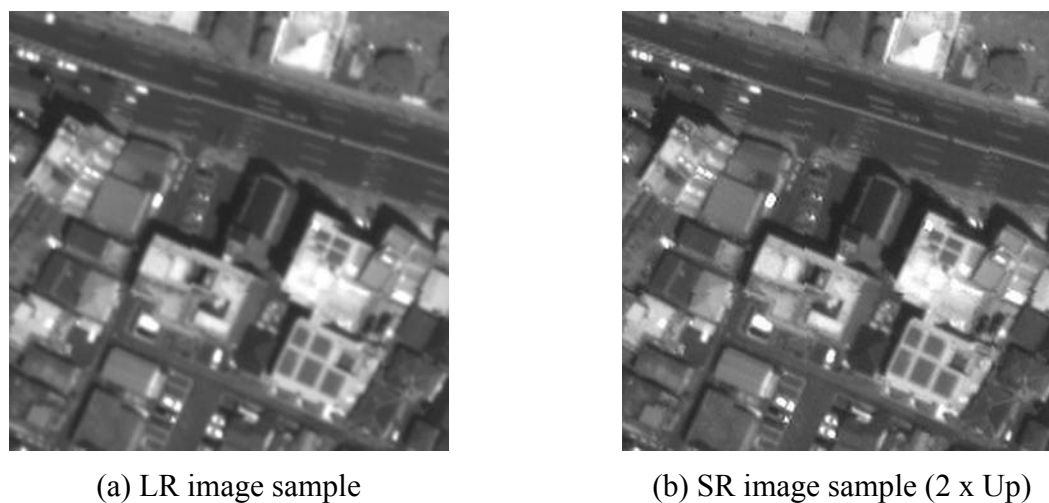


FIGURE 1.7: Example for super-resolution applied on satellite imagery.

- We innovatively presented a novel SR integrated building semantic segmentation framework to tackle the problem caused by the unaligned resolution between training and testing data, and investigated the feasibility of the proposed method based on comprehensive experiments.
- The experimental results demonstrate the proposed method could achieve state-of-the-art performance, and the IoU and Kappa is approximately 19.01% and 19.10% higher than that of the method without SR, respectively. It indicates the effects of SR on segmentation performance in remote sensing imagery, which would benefit the remote sensing community from literature review to future directions.

1.3.4 Expanding Applications

We expand the proposed methods mentioned above to more challenging applications including change detection, slum mapping, and map semantic segmentation. As for change detection, color normalization, super-resolution, and image registration methods are adopted to balance the training and testing datasets, after that, by adopting proposed CFPN model and image difference, the identification of land change can be achieved. In terms of slum mapping, here CFPN is adopted to perform multi-class semantic segmentation, the impact of resolution on slum segmentation is discussed as well. Furthermore, the important GIS-related task: map semantic segmentation, which aims at digitizing historical maps is also applied by our deep learning model.

1.3.5 Standard Package

We present an open source computer vision package named as GeoVision, which contains subpackages GeoSeg and GeoSR, to facilitate the development of the deep learning based segmentation and super-resolution models, respectively. As a unified, simple, and flexible package, GeoVision contains pipeline-like integrated tools from data retrieval to final result evaluation, which enables users to develop self-defined models conveniently; several state-of-the-art models trained through the same high-quality dataset are provided as the baseline in the package as well. Moreover, the proposed package could potentially serve as a viable backend for other related packages such as image segmentation with high efficiency.

1.4 Outline of the Dissertation

This chapter formulated the problem of semantic segmentation for multi-source remote sensing imagery based on convolutional neural networks, and described its significance and difficulties. It then continued with a related works in machine learning and deep learning, outlined our approach to the problem, and summarized our main contributions. The remaining chapters are organized as follows:

- Chapter 2: Introduces the method of adopting patch-based CNNs to segment village areas in remote rural environment;
- Chapter 3: Shows the performance of FCN based building semantic segmentation in urban areas;
- Chapter 4: Reveals the effectiveness of applying super-resolution methods in semantic segmentation tasks under resolution difference circumstance;
- Chapter 5: Proposes further application in change detection and slum mapping;
- Chapter 6: Concludes our work and presents proposals for future works;
- Appendices: I: Introduces the framework and function of GeoVision; II: The GIS-related application of map semantic segmentation; III: Our contribution and the list of publications.

Chapter 2

Village Mapping via Patch-based CNNs

With the rapid development of remote sensing satellite imaging techniques in recent years, a considerable number of highly spatially resolved images are available [11–13]. Owing to the high price performance ratio, many remote sensing image classification studies are performed using open high-resolution remote sensing (HRRS) data [72–74]. In this study, three-band HRRS images from Google Earth (GE) [75] and Bing Maps are used as the data source and applied to village building mapping in a large rural region.

Recently, deep convolutional neural networks (CNN) have been successfully applied to many pattern-recognition tasks [36]. Compared with most existing classification methods, which can only generate low- or middle-level image features with limited representation ability, CNN does not require prior manual feature extraction [55, 76]. A large volume of abstract features can be extracted automatically based on gradient descent and back propagation algorithms, thus resulting in higher accuracy and efficiency [77, 78].

CNN-based pixel-level classification is one of the most important and popular topics in the geoscience and remote sensing community, and it can be used to efficiently identify individual land features in greater detail, and significant progress to this end has been achieved in recent years [79–81]. Related works have been introduced in our previous work [82]. In this study, we focus on the use of CNN for pixel-level [83] classification via HRRS images according to our previous work, according to which, pixel-level village building identification is implemented based on a shallow CNN structure that can achieve relatively high accuracy when using GE images compared to other machine learning methods. Although the previous CNN structure proved to be very useful for exploring

features and classification, the unstable performance in some study areas indicates that it might be inadequate for exploiting the full potential capability of CNN.

Identification performance depends highly on the structure of the CNN model [84]. To adapt CNN for village building identification with high accuracy, we can apply state-of-the-art models such as AlexNet [85], VGGNet [86], GoogLeNet [87], SqueezeNet [88] achieved in ImageNet [67] Large-Scale Visual Recognition Challenge (ILSVRC) [84]. The high feasibility of applying the aforementioned models has been proven by many studies in different fields [63, 89–92]. To make the most of these mentioned state-of-the-art models and to ensure compatibility with our experiment, we optimized and enhanced their architectures into four self-designed structures named AlexNet-like, VGGNet-like, GoogLeNet-like and SqueezeNet-like via rigorous experiment while fully considering the characteristics of the input HRRS images and identification targets. The identification capability of an individual optimized CNN model is limited. To make the most of the single feature extraction capability, a promising solution would be to create an ensemble of several CNN models. In this study, we employ multiscale feature learning [93] to achieve the goal.

Multiscale feature learning schemes such as recurrent neural networks (RNNs) [94] and scene parsing using CNNs [95] have been showing tremendous capabilities in different tasks. In multiscale feature learning, several paralleled CNN models of varying contextual input size are implemented to extract features, and thereafter, the output of each CNN is ensembled and concatenated into a classifier. In practice, Martin et al. [64] implemented multi-class land feature classification by using four stacked CNN models. To improve and smooth semantic image segmentation, Marmanis et al. [96] and Farabet et al. [97] implemented multi-scale segmentation-based parallel CNN architectures. Richard et al. [98] and Pedro et al. [99] achieved multiscale feature learning by stacking multiple shallow networks with tied convolution weights on top of each other. Ding et al. [100] combined deep CNN with multiscale feature for intelligent spindle bearing fault diagnosis. In the case of medical image processing, Kiros et al. [101] utilized stacked multiscale feature learning for massive feature extraction and Tom et al. [102] for a deep 3D convolutional encoder. Many studies have used RGB-D images to implement classification and segmentation, [103–106], rather than inputting a four-dimensional image into a single CNN in a directed way; features of information in RGB and depth bands are usually extracted based on well designed parallel CNNs respectively. Finally, the obtained features are merged in a fully-connected layer for implementing different tasks. The multiscale feature learning method can be used effectively not only in the computer vision field but also in other fields such as recommender systems [107], in which different features of data types such as text, image, social relationship, and user

information are extracted using parallel CNNs; the final recommendation is provided based on classification of the ensembled features [108–110].

In this chapter, we present an elaborate formulated CNN model called Ensemble Convolutional Neural Networks (ECNN). Different from related works, ECNN achieves multi-scale feature learning by ensembling the feature extractor part of four optimized state-of-the-art models, and we apply it to implement the pixel-level village building identification task.

The main contributions of this study can be summarized as follows:

- We explored how to construct CNN architecture that can adapt to the village building identification task based on insightful and in-depth analysis.
- We optimized state-of-the-art CNN models by using rigorous principles to explore their potential for pixel-level building identification via HRRS images.
- We presented a novel CNN frame called ECNN based on multiscale feature learning by ensembling parallel optimized state-of-the-art CNN models.
- We implemented the proposed method for village building identification and found that it outperforms the existing state-of-the-art methods, achieving an overall accuracy and kappa coefficient of 99.26% and 0.86 respectively.

The remainder of this chapter is organized as follows. In Section 2.1, we describe the study area and the experimental dataset. Details about the methods are presented in Section 2.2. In Section 2.3, we present the experimental results and discuss the capability of the proposed method in comparison to existing methods. Finally, we present our conclusions and a few proposals for future work in Section 2.4.

2.1 Data

2.1.1 Study Area

To test the feasibility of the proposed method in different regions and by using different data sources, we selected rural areas in developing countries such as Laos and Kenya. One of the study areas is located in Kaysone, Savannakhet province in Laos. Its longitude and latitude range from E104°47'22" to E104°49'54" and from N16°34'28" to N16°36'26", respectively, and it measures approximately 12.08 km². The study area was a complex rural region with many different types of landscape, including abundant

natural components such as mountains, rivers, and vegetation cover, as well as artificial areas such as villages, roads, and cultivated land, which are typical of rural areas. The other study area was Kwale, a small town in the capital of Kwale County, Kenya. It is located at around $S4^{\circ}10'28''$ and $E39^{\circ}27'37''$, 30 km southwest of Mombasa and 15 km inland, and it measures approximately 30.20 km^2 . The area was mainly covered by forest and other desolate landscapes, and the buildings were rather scattered. A few samples from the study area are shown in Figure 2.1.

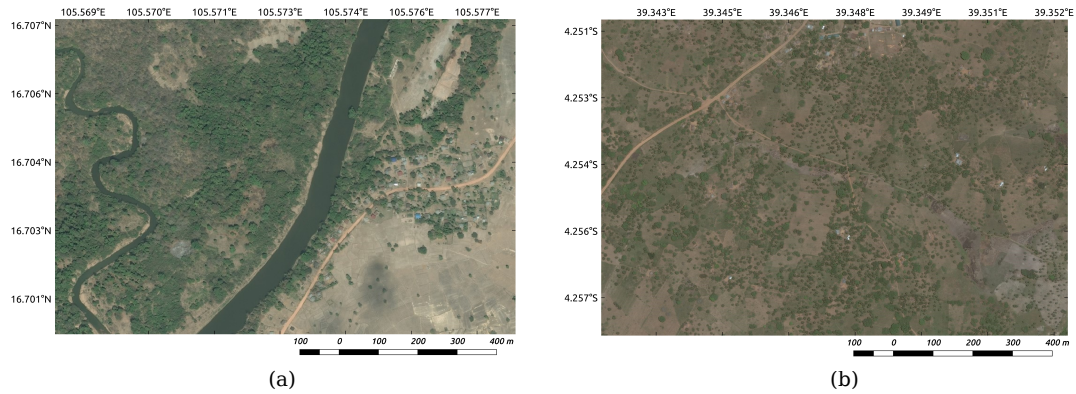


FIGURE 2.1: Study area example (a), located in Savannakhet Province, Laos, shows abundant land features; Study area example (b), located in Kwale Province, Kenya, with relatively desolate land features. The resolution of all images is 1.2 m.

2.1.2 Data Source

The remote top-view RGB image of Kaysone and Kwale, both with a resolution of 1.2 m, were captured from Google’s satellite map in February 2016 and Bing Maps in January 2016, respectively. As the training dataset for Laos, we deliberately selected a few typical village/non-village areas from the data source. In village areas, the training dataset mainly showed land features such as buildings, roads, rivers, and cultivated lands, while in non-village areas, mountains, forests, and vegetation cover are the main features. The ground truth map of the village buildings was manually drawn beforehand by using a polygon-based interaction tool. This ground truth map contained accurate information of the land categories and was chiefly used for sampling and result detection. Similar to Laos, the training dataset in Kenya was also selected considering the characteristics and the diversity of the landscape. The test dataset contained the entire testing area of Laos and Kenya, and several different types of landscape were shown; the land features in different countries and areas showed distinctive characteristics. As shown in Figure 2.1, land features in Laos (Figure 2.1a) are more abundant than those in Kenya (Figure 2.1b).

The diversity and complexity of the images also makes the identification task difficult. This, in turn, warrants that the classification model incorporate all these conditions.

2.2 Methods

Figure 2.2 shows details of the workflow employed in our experiment. First, as introduced in Section 2.1.2, the training dataset in our experiment contains two parts: three-band RGB HRRS images and the corresponding ground truth labels. Importantly, both the complexity and characteristics of the identification target, and the diversity of land features need to be considered when preparing the dataset [111]. Second, to optimize and mine the capability of CNN for rural environmental building identification and ensure compatibility with our classification targets, a few state-of-the-art CNN structures were carefully optimized and enhanced based on a series of rigorous testing results. Then, we generated the ECNN model from the ensembling based on the identification capability of the CNN models. Third, depending on the back propagation and the gradient descent algorithms, the proposed ECNN structure can learn from the training dataset patterns that map the variables to the target and output a trained ECNN model that captures these relationships and can identify buildings in rural environments. Thereafter, cross validation [112] was implemented to verify the feasibility and performance of the CNN models; here, to evaluate the accuracy and reliability of the result, we used the confusion matrix [113], kappa coefficient [114] and overall accuracy. Finally, the generated ECNN model was applied to the prepared testing HRRS dataset to identify village buildings.

2.2.1 Convolutional Neural Networks

The CNN method is more robust and yields better performance than other machine learning methods in image pattern recognition owing to its capability in mining deep representative information from low-level inputs [86]. A single CNN model performs the steps of convolution [115], non-linear activation [116], and pooling [117]. With multilayer networks trained by gradient descent and back propagation algorithms, CNN can learn complex and nonlinear mapping from a high- to low-dimensional feature space [118].

In this experiment, the input dataset $x \in \mathbb{R}^{h \times w \times c}$ refers to multichannel HRRS images, where each dimension represents the height, width, and number of channels. The output classification result $y \in \mathbb{R}^{h' \times w' \times c'}$ generated by $y = H(x, \Theta)$, where Θ denotes a set of parameters called kernels.

In the convolution layer, the input x with bias $\alpha \in \mathbb{R}^{c'}$ is computed by convolutional kernels $\Theta \in \mathbb{R}^{\tilde{h} \times \tilde{w} \times \tilde{c} \times c'}$. This computation can be formulated as follows:

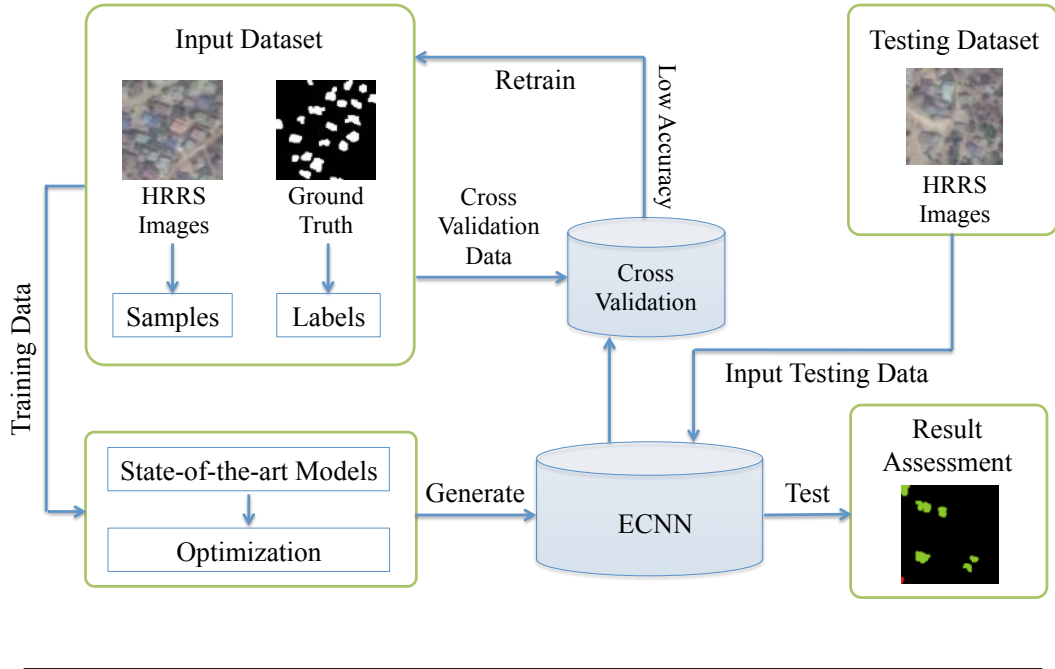


FIGURE 2.2: Workflow.

$$y_{i'j'k'} = H \left(\alpha_{k'} + \sum_{i=1}^{\tilde{h}} \sum_{j=1}^{\tilde{w}} \sum_{d=1}^c \Theta_{ijdk'} \times x_{i'+i, j'+j, d} \right) \quad (2.1)$$

where $H(\cdot)$ denotes a nonlinear function to generate the hypothesis; instead of saturated activation methods, here, we use the rectified linear unit (ReLU):

$$y_{ijk} = \max\{0, x_{ijk}\} \quad (2.2)$$

To implement the subsampling operation, the max-pooling layer [119], which computes the maximum response of each image channel in a $\tilde{h} \times \tilde{w}$ subwindow, is used, and it is calculated as follows:

$$y_{i'j'k} = \max_{1 < i < \tilde{h}, 1 < j < \tilde{w}} x_{i'+i, j'+j, k} \quad (2.3)$$

Finally, the classification result can be generated using the softmax function [120]:

$$y_{ijk} = \frac{\exp(x_{i,j,k})}{\sum_{d=1}^c \exp(x_{i,j,d})} \quad (2.4)$$

2.2.2 Model Optimization

In our previous study [82], the identification task was implemented using a simple CNN structure, in which, the sample window size was 18×18 , and two convolutional layers followed by average pooling were implemented with 6 and 12 filters, respectively. Compared with other machine learning methods, although the preceding CNN structure is very feasible for the purposes of feature exploration and classification, it might not be effective for mining the complete capability of CNN. In this section, we aim to optimize the CNN model to achieve better results.

ILSVRC is an annual competition held by ImageNet since 2010, in which research teams submit programs that classify and detect objects and scenes. It is important to note that in 2012, AlexNet reduced the error rate to 16% from the previous best of 25%, and in the next couple of years, more accurate pattern recognition results were obtained using popular models such as GoogLeNet, VGGNet, SqueezeNet and ResNet [121].

To make the most of these aforementioned state-of-the-art models, we optimized their architectures by considering the characteristics of the input HRRS images and our identification targets. Here, we propose self-designed structures called AlexNet-like, GoogLeNet-like, VGGNet-like and SqueezeNet-like based on rigorous experiments and theories; thereafter, we ensemble these CNN models into ECNN.

The principle of optimizing CNN architecture is highly based on analyzing the learning curves of both the training and the cross validation results [122]. In addition to accuracy, two other important indexes need to be pointed out: bias and variance [123].

In this experiment, both bias and variance lead to severe problems. High bias can cause an algorithm to miss the relevant relationships between features and target outputs. Here are some ways to solve this challenge:

- Optimize the accuracy of the input training data. This means the training HRRS images and the corresponding labels of buildings and other land features must be as accurate as possible.
- Decrease the regularization coefficient λ [124], because doing so can solve under-fitting-related problems.
- Add number of features, such as implementation of higher-level CNN structures, which could extract more features

When facing high variance, which leads to over-fitting [125], the problem can be solved by:

- Adding more training samples would be helpful. Data augmentation such as adding more training HRRS images to the dataset considering the diversity.
- Increase the regularization coefficient λ , which can solve over-fitting problems.
- Decrease the number of features, by using a method such as Dropout [126].

Here, we optimize our model based on the preceding principles. We take the VGGNet-like (introduced in Section 2.2.4) structure as an example to explore how to configure the CNN architecture based on the characteristics of VGGNet. The final promising structure is generated by gradually enhancing and optimizing a simple initial CNN. Considering the experimental requirement, the three parameters to be evaluated in our experiment are number of filters, depth of architecture, and input sample window size. These parameters are connected in a way that determines the total number of units and the weight values of the entire structure.

The initial architecture is based on the basic CNN model utilized in our previous work. To enhance the architecture, the number of filters is configured by multiplying the original number of filters by $f = [3\ 9\ 25\ 100\ 200]$. The number of added convolutional layers is denoted by y , and it ranges from 2 to 12 in steps of 2; the window size s is the area surrounding the pixel to be classified and is set to be between 14 and 50 with an interval of 2. We evaluated the effects of each parameter in terms of accuracy, efficiency, and learning curve; then, we integrated all the optimal settings to obtain a promising VGGNet-like architecture.

2.2.2.1 Influence of Filter

In general, the greater the number of filters, the greater the number of features that can be extracted. Here, we gradually increase the number of features from the original to $f = [3, 9, 25, 100, 200]$ times in each convolutional layer. As shown in Table 2.1, when the number of filters reaches 25 times, the best training and testing results can be generated, and the model can achieve 98.98% and 0.83 in terms of testing accuracy and kappa value, respectively. Moreover, from the learning curve (Figure 2.3), until 200 and 300 times, the model does not encounter the challenge of over-fitting, which means that the number of features has not saturated yet. Upon adding more filters, the model tends to converge faster. However, when considering both accuracy and efficiency, the number of filters that can obtain a good enough result would be suitable.

Although high accuracy could be achieved, the model continued to suffer from unstable convergence, and it was not stable even after adding 200 times the original number of filters. The influence of depth of architecture will be explored in the next section.

TABLE 2.1: Relationship between number of filters and accuracy.

Structure	Para	Acc (%)	Training			Testing		
			Kappa	Epoch (s)	Total (min)	Acc (%)	Kappa	Total (s)
Ori	1669	95.31	0.86	1.41	7.06	97.29	0.60	1.59
×3	11917	98.94	0.97	1.57	7.85	98.15	0.72	1.93
×9	97957	99.19	0.98	2.29	11.46	98.22	0.73	3.31
×25	0.73M	99.73	0.99	7.09	35.44	98.98	0.83	10.16
×100	11.57M	99.69	0.99	83.09	415.46	98.79	0.80	88.00
×200	46.18M	97.54	0.92	299.06	1459.27	98.14	0.70	5.67

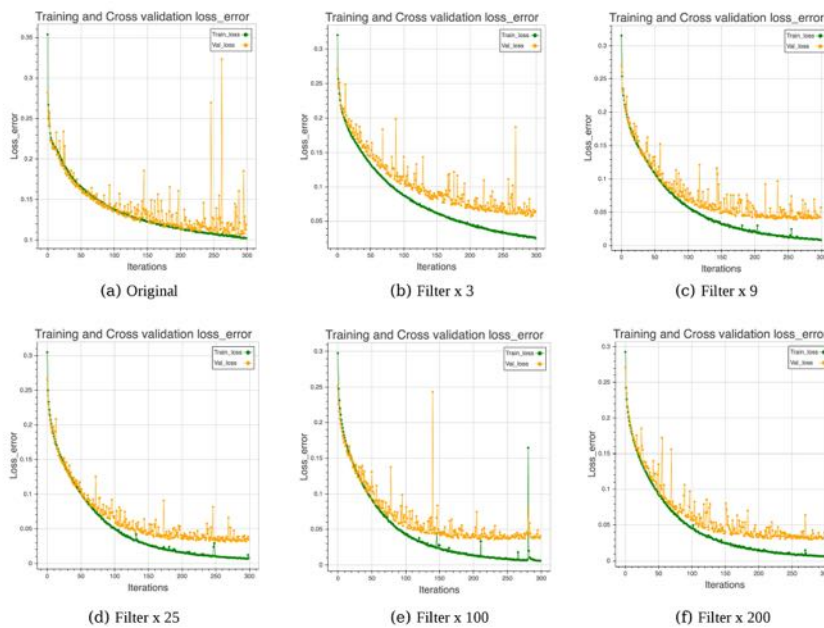


FIGURE 2.3: Influence of number of filters.

2.2.2.2 Influence of Depth

CNNs constitute a very important branch of deep learning. The preponderance of CNNs is highly based on the depth of architecture. By mining deeper and more abstract features and information from an identification target, usually, a very deep network can achieve higher accuracy. In recent years, owing to the improvement in the computational capability and hardware, it has become possible to construct and compute very deep networks. Recent state-of-the-art architectures such as VGGNet and ResNet make the most of this principle.

In this experiment, to explore the effect of depth on the CNN model for identification of buildings in rural environments, we increased the number of convolutional layers from the original 2 to 14 in steps of 2. Both the training testing settings and the results are shown in Table 2.2.

TABLE 2.2: Relationship between depth and accuracy.

Structure	Para	Training				Testing		
		Acc (%)	Kappa	Epoch (s)	Total (min)	Acc (%)	Kappa	Total (s)
Ori	1669	93.02	0.79	1.49	74.29	94.58	0.42	1.73
+2 Conv	4891	96.30	0.89	2.06	103.16	96.84	0.58	2.60
+4 Conv	8133	96.21	0.88	2.73	136.93	98.06	0.68	3.55
+6 Conv	11,335	97.45	0.92	3.44	172.09	98.18	0.70	4.65
+8 Conv	14,557	97.54	0.92	4.16	208.19	98.14	0.70	5.67
+10 Conv	17,779	97.80	0.93	4.94	247.03	97.67	0.65	6.73
+12 Conv	21,001	78.97	0.00	6.18	308.87	97.53	0.00	7.81

At the outset, model accuracy increases as the network depth increases. However, when the number of convolutional layers is higher than 12, the network becomes stocked and even loses its identification capability. After rigorous analysis, we found that this problem is caused by gradient vanishing [127]. As we know, CNN is based on gradient descent and back propagation. When implementing the gradient descent algorithm, the input signal will be activated by activation function in the saturated or diverged region. Thereafter, with propagation processing, this phenomenon will be propagated in the entire model and will cause the corresponding gradient to vanish and explode.

This challenge can be overcome in several ways. For instance, we can use unsaturated activation such as Relu to relieve the problem to a certain degree. Moreover, the batch normalization [128] method, in which feature scaling is performed after convolution can be used; with this method, the result falls into the vanishing and exploding region can be avoided. In this experiment, we selected the simplest solution of adding depth to the most suitable degree, which can yield promising results while avoiding the gradient vanishing problem. Considering efficiency and accuracy, here, we added six convolutional layers into the original structure; as a result, we obtained testing accuracy and kappa value of 98.18% and 0.70 respectively.

2.2.2.3 Influence of Window Size

The size of the input sample is a very significant factor that influences identification capability. Considering the image resolution and the characteristics of village buildings, the ideal window size must be slightly bigger than that for ordinary buildings, while information about a building’s surroundings must be included as well. The input window

size of our original basic architecture is 18×18 , which might be too small to extract enough valuable features.

Herein, we change the window size from 14 to 50 with intervals equal to 2; the parameter amount increases along with the increasing window size. For comparison, the experiment is conducted using a basic and a complex CNN structure, which is constructed based on the previous optimization principle. In particular, we focus on comparing the effect of window size on multiple relations, such as size 14 with 28, 16 and 32, etc., because a double-sized window contains the same information as a small one.

From the testing result (Figure 2.4a), by implementing a simple structure, a double-sized window could yield better results, because it contains more abundant information than a small-sized window. However, if we implement a complex structure, although a double-sized window contains more information, we cannot always obtain better results (Figure 2.4b).

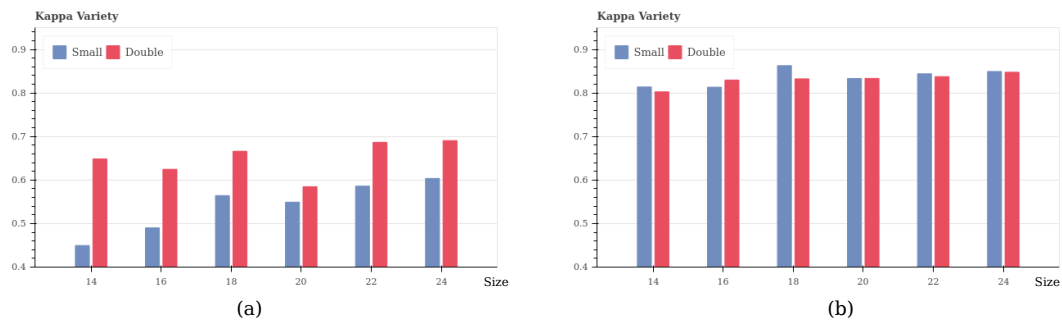


FIGURE 2.4: Window size in multiple relations. (a) with a simple structure; (b) with a complex structure.

With the same CNN structure, bigger window size can obtain more features and parameters, but other methods such as adding filters and depth can also increase the amount of features. If the feature extraction capability of the model is weak, the big-size samples would help it to obtain more information than small-size ones, which would lead to good results. However, when we use a complex structure which can extract sufficient features, bigger window size can no longer yield good results, and extremely big window size might yield redundant and useless features, which lead to bad results. In this experiment, we choose a window size that is 50% bigger than in the ordinary architecture, with an adaptive number of kernels and depth. If the model suffers from over-fitting, herein, we also implement Dropout to address the problem.

In conclusion, to take full advantage of state-of-the-art CNN models, we optimized and enhanced them into new ones that match the village building identification task based on rigorous principles and experiments. Furthermore, we also visualized the representation

of each layer to evaluate the feasibility of the model; here, take features extracted by VGGNet-like as an example. In later sections, we will introduce the self-designed models: AlexNet-like, VGGNet-like, GoogLeNet-like, and SqueezeNet-like.

2.2.3 AlexNet-Like

AlexNet is a revolutionary CNN architecture [85]. The parallel and merged structure of this architecture makes it suitable for extracting two sets of features while sharing information between the two sets. Deep CNN can be formulated elaborately with very high accuracy. Moreover, by running the model on GPUs implemented in CUDA, it becomes feasible to train the CNN model on large-scale datasets.

There are a few tricks of AlexNet in terms of both structure and processing. First, image preprocessing is conducted by only subsampling and feature scaling. Then, instead of the saturated activation method, AlexNet implements Relu, which is very efficient and six times faster than tanh [129], and it can avoid gradient vanishing and exploding to a certain degree. Third, given its parallel structure, AlexNet can be efficiently trained on multiple GPUs, and every GPU shares half kernels. To reduce over-fitting, AlexNet also employs tricks such as data augmentation, Dropout, and overlapping pooling structure. Finally, the stochastic gradient descent (SGD) method [130] is used with configurations such as weight decay, and gradually reducing momentum and learning rate.

In this experiment, we rigorously optimized AlexNet into the AlexNet-like architecture as shown in Figure 2.5. To this end, we reduced the input size to 30×30 , and optimized internal settings such as quantity of filter and kernel size based on the optimization principle, which increased the model's efficiency by reducing the total number of parameters from about 60 million to 67,665.

2.2.4 VGGNet-Like

VGGNet is short for Very Deep Convolutional Networks. As its name suggests, VGGNet addresses the important aspect of CNN architecture design. The depth of this architecture makes it suitable for mining very deep and abstract features [86]. The architecture steadily increases the depth of networks by adding convolutional layers, and the quantity of filters gradually increases from the start to the end. Very small convolutional filters of size 3×3 are used in all layers, and the 1×1 filter can be seen as a linear transformation of the input channels. Other layers such as Zeroppading, Maxpooling, Flatten, Dense and Dropout also increase its identification capability. To avoid over-fitting, we must eliminate redundant features by using Dropout.

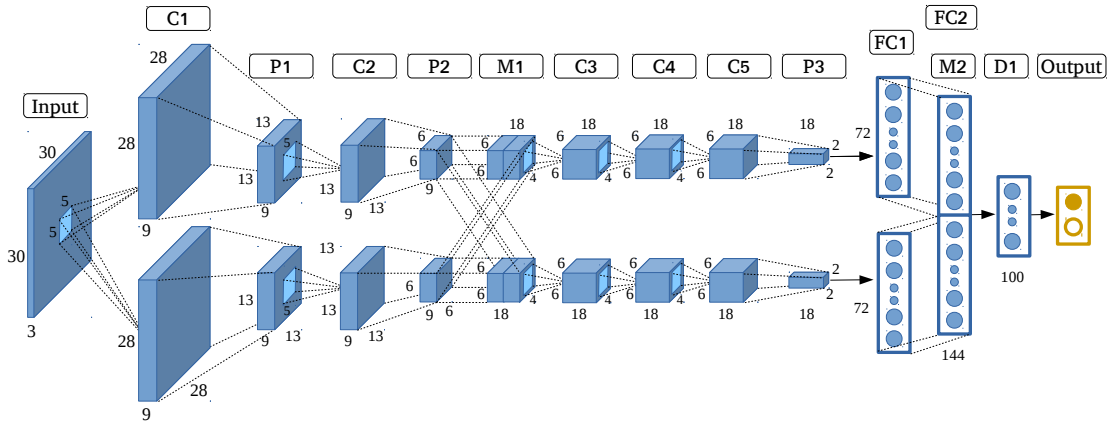


FIGURE 2.5: AlexNet-like architecture.

We propose the VGGNet-like architecture (Figure 2.6) in this experiment, which is very effective for identifying buildings in rural environments based on HRRS images. VGGNet-like is optimized by decreasing the depth quantity and filter size while retaining its original architecture. After optimization, the number of parameters decreases from 140 M to 70,453, which makes the model easy to train. The detailed settings are shown in Table 2.3.

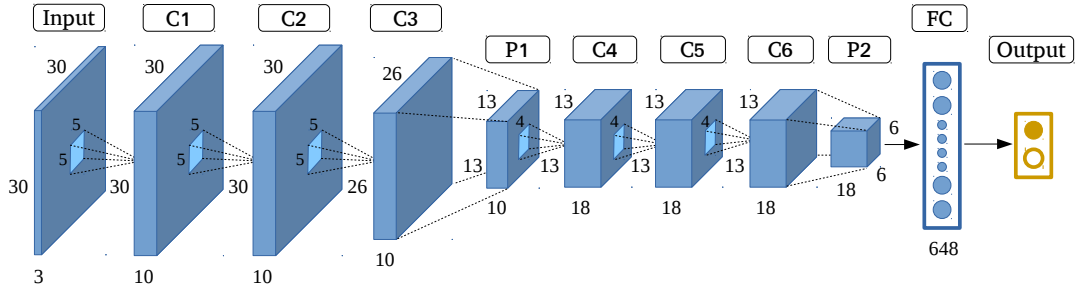


FIGURE 2.6: Very Deep Convolutional Network (VGGNet)-like architecture.

To intuitively understand the CNN activations for village buildings, we visualize the representations of each layer by reconstructing features from simple patterns to complex ones with the technique proposed in [131] using VGGNet-like, shown in Figure 2.7.

Due to the limitation of resolution, the external characteristics of village buildings cannot be shown clearly in some regions. However, the features extracted by convolutional layers still characterize village buildings well and can be reconstructed to images similar to the original image with more abstract information and blurriness as one progresses toward deeper layers. The visualization results also indicate the feature extraction capability of our self-designed models.

TABLE 2.3: VGGNet-like architecture

Layer	Output Shape	Kernel Size	Scale	Para	Connect to
Input	(30, 30, 3)	-	-	-	-
Conv 1	(30, 30, 10)	(5, 5)	-	760	Input
Conv 2	(30, 30, 10)	(5, 5)	-	2510	Conv 1
Conv 3	(26, 26, 10)	(5, 5)	-	2510	Conv 2
Pooling 1	(13, 13, 10)	-	2,2	0	Conv 3
Conv 4	(13, 13, 18)	(4, 4)	-	2898	Pooling 1
Conv 5	(13, 13, 18)	(4, 4)	-	5202	Conv 4
Conv 6	(10, 10, 18)	(4, 4)	-	5202	Conv 5
Pooling 2	(5, 5, 18)	-	2, 2	0	Conv 6
Flatten	(648)	-	-	0	Pooling 2
Output	(1)	-	-	649	Flatten

Total Parameters: 19,731

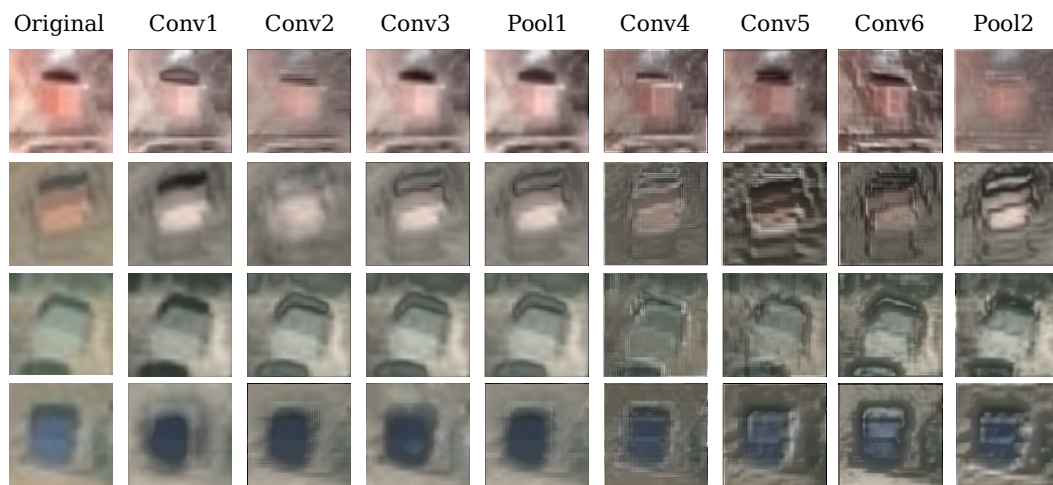


FIGURE 2.7: Reconstruction of Convolutional Neural Network (CNN) activations from different layers of VGGNet-like.

2.2.5 GoogLeNet-like

The main innovation of GoogleNet is its use of an architecture called Inception [87]. In general, Inception is a network in network structure, and the optimal local sparse structure of a vision network is spatially repeated from the start to the end. Three Inception structures used in different circumstances are introduced: typically, 1×1 convolution is used in Inception to compute reductions before the expensive 3×3 and 5×5 convolutions.

GoogleNet provides us with an inspiration of how to build a high-capability architecture. Most of the identification capability progress relies not only on more powerful hardware,

large datasets and bigger models, but also and mainly on new ideas, algorithms, and improved network architectures.

By learning from GoogleNet, in this experiment, we built a GoogleNet-like structure as shown in Figure 2.8. We established the Inception architecture, while optimizing the number and sequence of layers and filters.

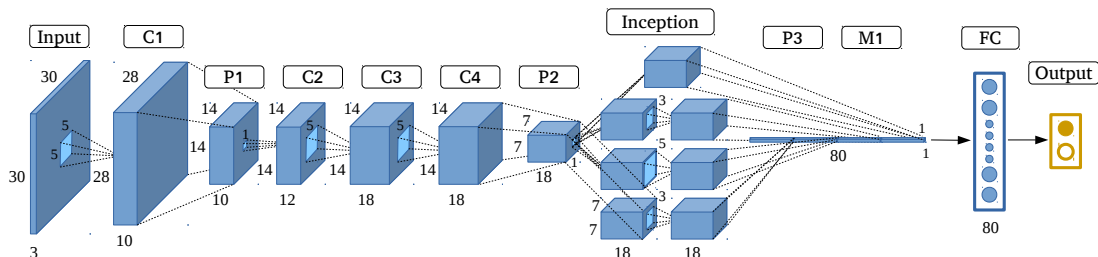


FIGURE 2.8: GoogleNet-like architecture.

2.2.6 SqueezeNet-Like

Compared with other architectures, SqueezeNet has very few parameters while retaining similarly high accuracy [88]. It can achieve AlexNet-level accuracy with 50 times fewer parameters and <0.5 MB model size, in addition to identifying patterns by using very few parameters while preserving accuracy.

There are some tricks associated with its structure. First is the structure called fire, which appears like a fire blazing through a matchstick. Instead of the 3×3 convolutional core used in GoogLeNet, SqueezeNet uses 1×1 filters in a few layers, because 1×1 filters have one-ninth the number of parameters compared to 3×3 filters. The fire module comprises a squeeze convolution layer (consisting of only 1×1 filters), and the aforementioned layer is fed into an expanded layer comprising a mix of 1×1 and 3×3 convolutional filters. Then, the number of parameters can be decreased by decreasing the quantity of input channels. Third, downsampling is performed at a late stage in the network so that convolutional layers can have larger activation maps, which leads to higher classification accuracy. Finally, the output is directly generated by the pooling layer instead of the fully-connected layer, which can decrease the number of filters dramatically. For instance, the final convolutional layer obtains features of size $13 \times 13 \times 1000$, and the pooling layer subsamples these features into size $1 \times 1 \times 1000$, yielding 1000 possibilities in the process.

In this experiment, we designed a SqueezeNet-like architecture (Figure 2.9) starting from a standalone convolutional layer; then, we employed four fire modules. Emulating

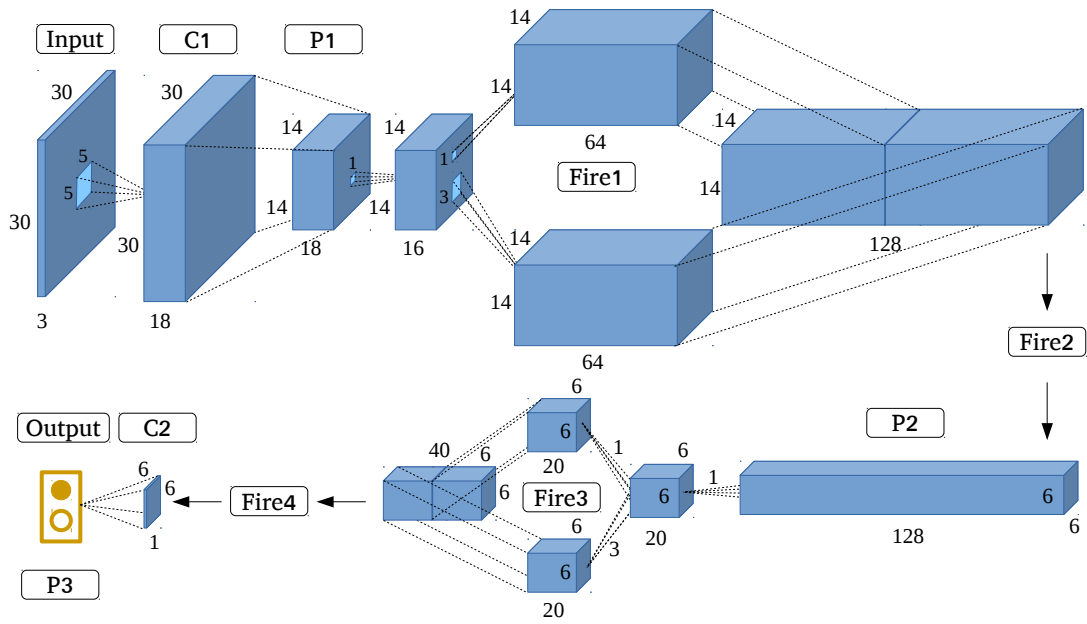


FIGURE 2.9: SqueezeNet-like architecture.

the original SqueezeNet structure, we gradually increased the number of filters per fire module from the start to end. Maxpooling (overlapping pooling) with stride was implemented after *Conv1* and *Merge2*, and the final average pooling layer divides the output into two categories, namely, building and non-building.

2.2.7 Ensemble Convolutional Neural Networks

Very deep CNN structures with strong feature extraction capability are typically used for larger images measuring at least 200×200 pixels [67]. In the case of pixel-level village building identification, as analyzed in Section 2.2.2, small HRRS images are used to avoid redundant noise and information, while very deep structures and a large number of filters are not suitable owing to the problems of efficiency, accuracy, and robustness. Although the optimized state-of-the-art models can mine several features, a few important ones are inevitably lost. The feature extraction capability of an individual model is limited, and a promising solution is ensembling several CNN models into a stronger model by using the multiscale feature learning method.

Here, we present ECNN, shown in Figure 2.10, an elaborate CNN frame formulated based on the ensembling of optimized state-of-the-art CNN models, followed by three layers of neural networks and softmax to implement classification. Instead of varying the contextual input size, multiscale feature learning can be achieved by inputting HRRS

images of the same size to all CNNs. This would also help preserve integrated building information.

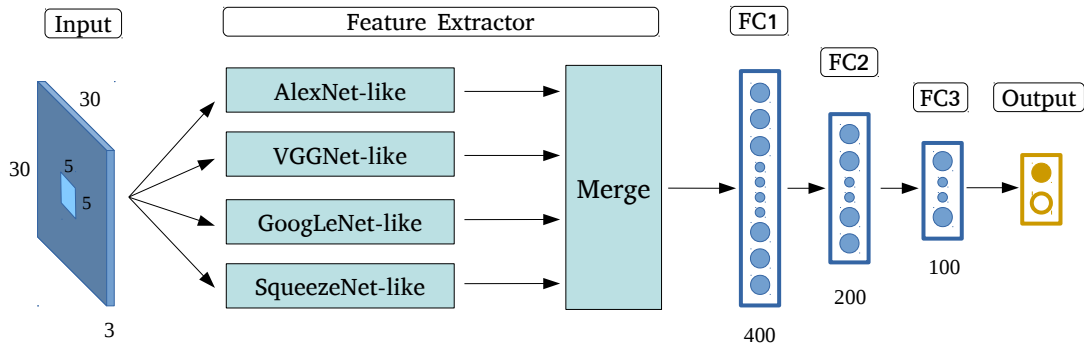


FIGURE 2.10: Ensemble Convolutional Neural Networks.

By taking full advantage of the different optimized state-of-the-art models' feature extraction capabilities, the proposed ECNN structure can achieve better classification results. Moreover, it can solve the problem of remaining small input image size, while avoiding the serious problems caused by very deep CNNs, such as gradient vanishing. To the best of our knowledge, there is no existing related CNN structure to identify village buildings by using HRRS images, and the feasibility of ECNN will be evaluated in the following sections.

2.3 Result and Discussion

We defined the CNN model utilized in our previous study [82] as basic CNN structure, and it cannot achieve a stable, high kappa value in many testing areas. Moreover, the building identification capability of the corresponding model is relatively limited. As shown in the previous section, based on the rigorous CNN model optimization and construction principle, we formulated four types of self-designed structure by using state-of-the-art networks and ensembled them into the ECNN model.

In this section, to compare and discuss the village building identification capability of different models, we first employ the same dataset and study area used in [82]. Thereafter, we use the models to implement village building identification in practice. We discuss and evaluate the feasibility of the model in terms of kappa coefficient, overall accuracy, confusion matrix, standard deviation, and computation efficiency.

2.3.1 Comparison of Different Models

Here, we set the experimental parameters as follows: number of iterations = 300, window size = 30×30 , learning rate = 0.03, activation Relu, and Softmax. In terms of dataset, 50,655 and 12,664 images were selected as the training and cross validation samples respectively. Because the land feature information of non-building areas is much more abundant than building areas in villages, 13,319 are positive samples and 50,000 are negative samples. For the sake of comparison, we selected the same testing area as in our previous study in Laos. The number of filters and depth information were different for each architecture. The employed parameter details and the training results are listed in Table 2.4.

TABLE 2.4: Training result by different CNNs.

Structure	Parameter		Training		
	Original	New	Acc (%)	Kappa	Epoch (s)
ECNN	-	506,288	99.78	0.99	31.21
AlexNet-like	60.97 M	51,249	99.77	0.99	5.42
VGGNet-like	143.67 M	70,453	99.78	0.99	13.81
GoogLeNet-like	7.00 M	37,589	99.71	0.99	6.62
SqueezeNet-like	1.25 M	39,941	99.73	0.99	7.23
Basic	-	4349	96.48	0.90	98.22

The training results show that all proposed self-designed CNN models outperformed the basic ones and achieved very high accuracy of over 99% with much higher efficiency. Thereafter, we implemented the trained models for testing, and the results in terms of overall accuracy, kappa value, and confusion matrix are given in Table 2.5.

Structure	Testing			Confusion Matrix			
	Acc	Kappa	Total (s)	TN	FP	FN	TP
ECNN	99.15	0.85	56.22	522,162	4519	56	13,263
AlexNet-like	98.95	0.82	16.72	521,048	5633	37	13,282
VGGNet-like	98.95	0.82	25.77	52,1058	5623	50	13,269
GoogLeNet-like	98.91	0.81	12.19	520,837	5844	63	13,256
SqueezeNet-like	98.89	0.81	17.93	520,713	5968	45	13,274
Basic	96.64	0.57	180.70	509,295	17,366	799	12,540

TABLE 2.5: Testing result by different CNNs.

From the testing results, the self-designed models performed much better than the basic structure, and the accuracy and kappa coefficient increased by about 2.5% and 0.3, respectively. The confusion matrix shows that TP and TN increased substantially, while

FP and FN decreased, which means misclassification in the cases of building and non-building areas was solved to a certain degree. In particular, ECNN, which can achieve a kappa coefficient of up to 0.85, outperformed other methods. The testing results indicate the feasibility of the model optimization method and the strong capability of the proposed ECNN method, which is based on ensembling the feature extraction parts of the state-of-the-art models for village building identification.

2.3.2 Implementation of CCNs

In this section, we present the building identification results obtained in the study areas in Laos and Kenya. These results were obtained using the optimized state-of-the-art CNN models and ECNN. In addition, we discuss their feasibility in terms of accuracy, stability, and efficiency.

To evaluate and compare the robustness of different CNN models, here, we deliberately selected several representative and typical small-segment areas, where land features and buildings present different characteristics in terms of color, external structure, and texture. The concrete numerical results are presented in terms of kappa coefficient, standard deviation, and mean average overall accuracy, while the intuitive classification results are presented in terms of different colors, where green refers to true positive, that is, the actual buildings are classified correctly as buildings; blue indicates the non-building areas that were incorrectly labeled as buildings; red indicates the buildings that were marked incorrectly as non-buildings; and black indicates true negative, which denotes the correctly classified non-building areas.

Because villages along river banks are representative of the landscape in many countries [132], we selected a few related regions, as shown in the top row of Figure 2.11. Notably, the regular outline of the bank in some regions is quite similar to buildings, which makes identification very challenging in many cases. The testing result obtained using the proposed different CNNs (Figure 2.11; second row to the final row) shows the models' excellent identification capability in such regions, and the majority of buildings are correctly identified, while other land features such as river bank are also well classified. However, in Figure 2.11c,f,h, some regions with vegetation cover are misclassified as buildings, and buildings near the boundary are marked as non-building areas by the optimized state-of-the-art models, while ECNN correctly classified these regions and identified buildings with higher accuracy. In Figure 2.11b, there is a region where non-building areas are misclassified by ECNN; after carefully analyzing the original image, we believe that this was caused by imperfect ground truth.

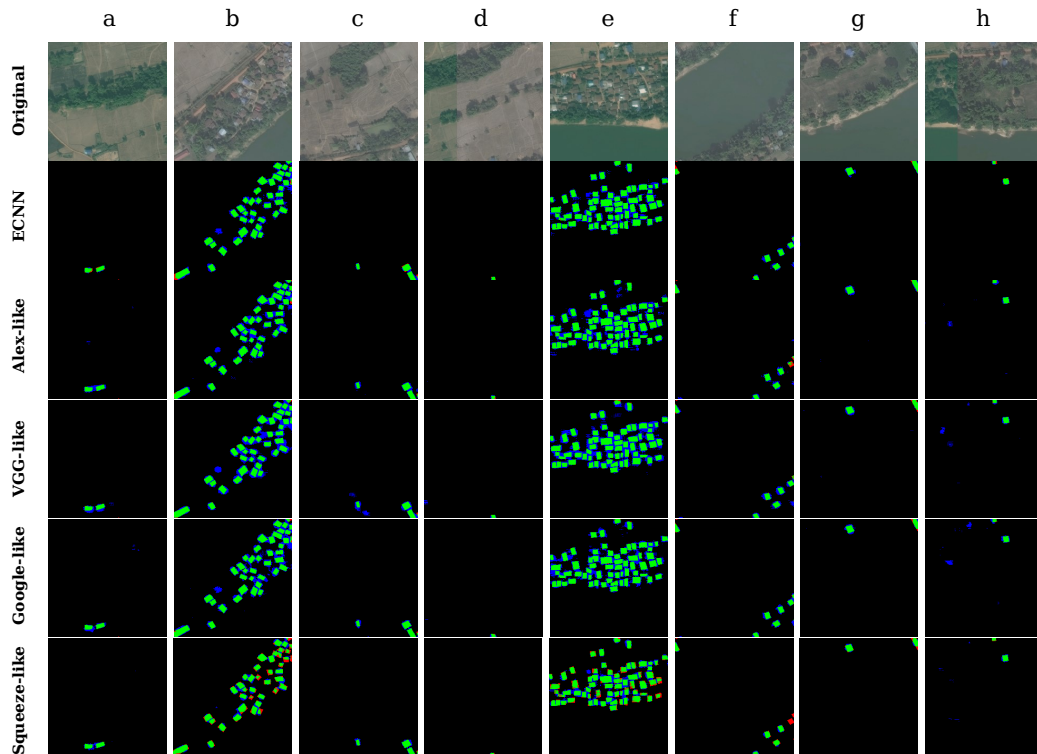


FIGURE 2.11: Identification results of eight small segments in bank regions.

As shown in Table 2.6, the proposed ECNN model outperformed the other models, achieving an average kappa of 0.82 and overall accuracy of 98.34% in regions (a–h). In terms of standard deviation, ECNN is slightly better than a few other models, but the kappa can be relatively unstable when the density of buildings is high in a given region.

Structure	a	b	c	d	e	f	g	h	Mean	Std	Acc_Mean (%)
ECNN	0.86	0.76	0.83	0.91	0.76	0.78	0.86	0.84	0.82	0.05	98.34
AlexNet-like	0.72	0.74	0.80	0.79	0.73	0.73	0.82	0.69	0.75	0.04	98.06
VGGNet-like	0.74	0.73	0.82	0.76	0.73	0.77	0.81	0.53	0.74	0.08	98.00
GoogLeNet-like	0.80	0.76	0.83	0.80	0.76	0.76	0.84	0.77	0.79	0.03	98.30
SqueezeNet-like	0.69	0.68	0.65	0.63	0.68	0.70	0.82	0.61	0.68	0.06	97.49

TABLE 2.6: Testing results in bank regions with different CNNs.

The complex and mixed-type village regions that contain an abundance of terrestrial features such as streams, pools, vacancies, vegetation, and crops were selected for conducting the comparison. As shown in Figure 2.12, ECNN could identify buildings in all cases, and it yielded the least false positive results compared to the other models.

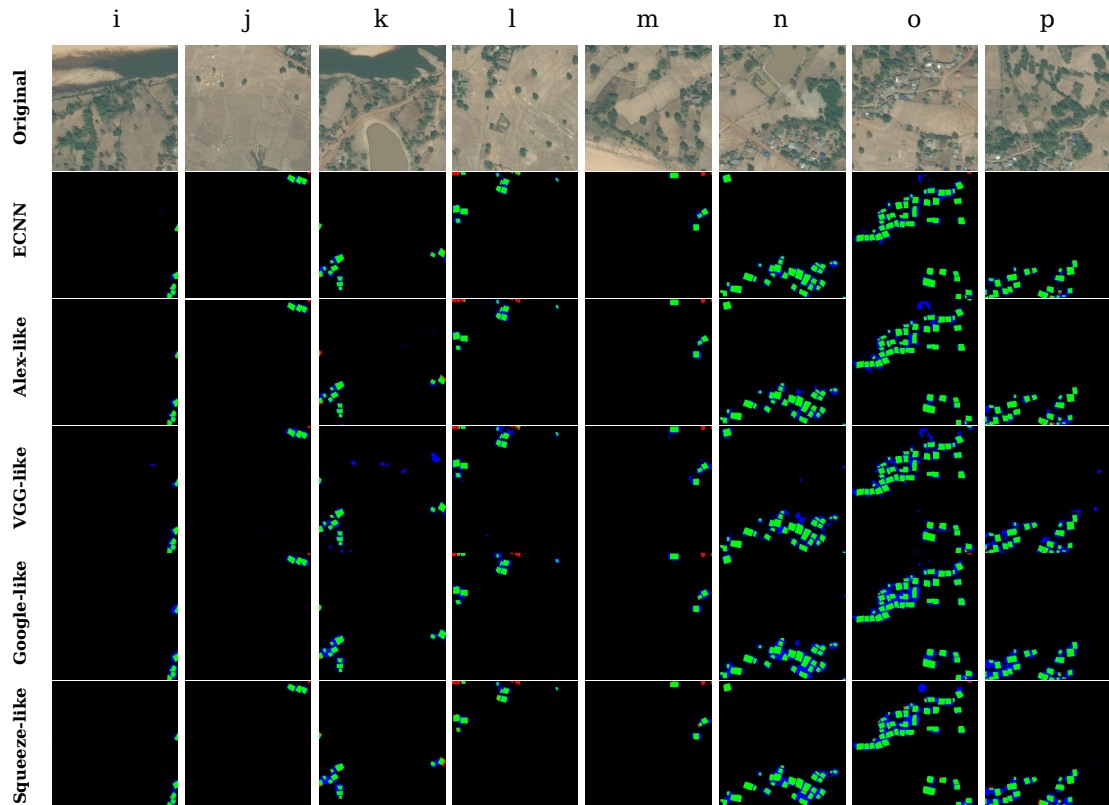


FIGURE 2.12: Identification results of eight small segments in mixed-type regions.

The detailed results are given in Table 2.7. ECNN not only achieved the highest average kappa of 0.77 and overall accuracy of 98.13%, but also the best standard deviation of 0.02. By contrast, the individual optimized state-of-the-art models yield unstable performance with average kappa values ranging from 0.67 to 0.74. This indicates that the proposed ECNN offers higher robustness and better feasibility within complex testing regions compared to individual CNN models.

Structure	i	j	k	l	m	n	o	p	Mean	Std	Acc.Mean(%)
ECNN	0.79	0.81	0.75	0.73	0.76	0.76	0.77	0.78	0.77	0.02	98.13
AlexNet-like	0.79	0.79	0.72	0.69	0.74	0.75	0.74	0.75	0.74	0.03	98.00
VGGNet-like	0.70	0.72	0.62	0.67	0.64	0.70	0.71	0.67	0.68	0.03	97.25
GoogLeNet-like	0.63	0.74	0.68	0.67	0.67	0.66	0.68	0.65	0.67	0.03	97.63
SqueezeNet-like	0.74	0.79	0.65	0.70	0.72	0.71	0.70	0.66	0.71	0.04	97.50

TABLE 2.7: Testing results in mixed-type regions with different CNNs.

As shown in Figure 2.13, finally, we selected typical areas containing plenty of human-built land features such as roads, agricultural fields, and pounds. Owing to the similar textures and external structures to the buildings, artificial land features are prone to misclassification, leading to decreased accuracy of the results along with a large number of false positives. According to the testing results in Figure 2.13 (second row to the final row), although ECNN can achieve better performance than the other models, a

few artificial land features such as roads and yards are inevitably identified as buildings. It indicates that the ECNN model still needs to be enhanced by training it using a more diverse training dataset.

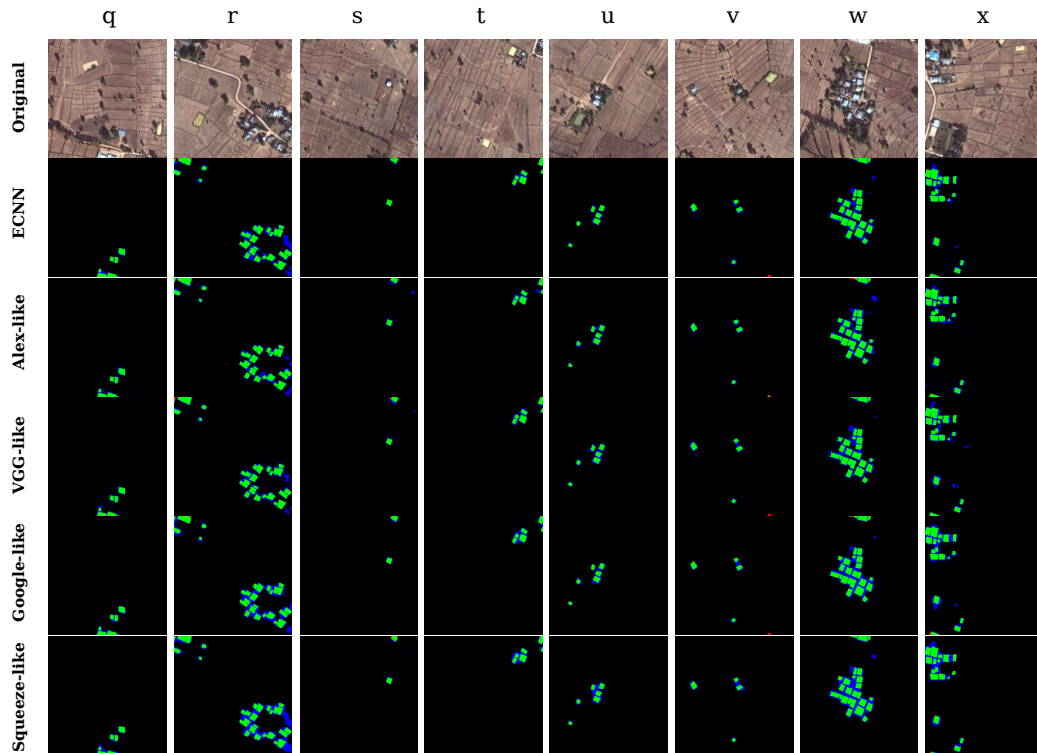


FIGURE 2.13: Identification results of eight small segments in artificial land regions.

The results in Table 2.8 infer that in regions with complex artificial land features, ECNN can achieve a very high kappa of 0.80 and overall accuracy of 98.38%, while the SqueezeNet-like model can achieve a kappa of only 0.72.

Structure	q	r	s	t	u	v	w	x	mean	std	Acc_mean
ECNN	0.84	0.78	0.87	0.84	0.81	0.76	0.78	0.79	0.80	0.04	98.38
AlexNet-like	0.82	0.70	0.81	0.75	0.75	0.75	0.73	0.72	0.75	0.04	98.25
VGGNet-like	0.81	0.78	0.75	0.78	0.72	0.75	0.76	0.77	0.77	0.03	98.32
GoogLeNet-like	0.81	0.74	0.77	0.78	0.73	0.73	0.77	0.75	0.76	0.03	98.04
SqueezeNet-like	0.77	0.65	0.81	0.72	0.69	0.68	0.70	0.70	0.72	0.05	98.30

TABLE 2.8: Testing results of different CNNs in artificial land regions.

It should be noted that a comparison of the models' feasibility in all cases and other regions that are not included in these study areas is very difficult because they differ in terms of resolution, data acquisition methods, reference datasets, and class definitions. However, from the testing results, it can be concluded that the optimized state-of-the-art models, especially ECNN, can achieve comparably efficient village building identification

results to the previous best result in the tested study areas. Moreover, the proposed ECNN model has considerably better accuracy and robustness than the individual optimized CNN model structure in the village building identification task.

2.4 Conclusions

In this study, we proposed a novel CNN frame called ECNN for village building identification using HRRS images. First, we constructed four self-designed CNN structures based on state-of-the-art CNN models and a rigorous optimization principle. Then, to extract most of their identification capabilities, we ensembled the feature extractor parts of each individual optimized model and concatenated them into ECNN based on the multiscale feature learning method. Finally, the generated ECNN was applied to a pixel-level village building identification task in developing countries.

The experimental results show the potential and the capability of the proposed ECNN model and the optimized state-of-the-art models in village building identification. The models achieved considerably higher accuracy than the previous best methods. In particular, the proposed ECNN model achieved considerably higher accuracy, and the kappa value improved from the previous best of 0.57 to 0.86 and overall accuracy from 96.64% to 99.26%. It outperformed the individual optimized CNN models as well, which indicates the feasibility of our proposed method.

More detailed exploration of the method is required in the future. First, to test the robustness of the method, regions of different resolution, as well as various data acquisition methods and reference datasets, need to be tested. Second, multi-class village landscape classification needs to be implemented using the proposed method. Finally, in case there are any limitations in the training data source, transfer learning [133] and the generative model [134] will be applied to enhance the proposed ECNN model.

Chapter 3

FCN-based Building Semantic Segmentation in Urban Areas

Since the achievement of a wide variety of vital tasks such as urban monitoring, demographic modeling, and disaster surveillance strongly rely on the detection of important land features, the semantic segmentation of buildings via remote sensing imagery has become a significant research topic in recent years [135, 136]. To conduct the building segmentation task, the methods such as graph theory-based [20] and clustering-based [21] are usually inappropriate due to the complexity and variety of remote sensing imagery [137]. Furthermore, in terms of conventional classification-based segmentation methods [138–140], which mainly rely on handcrafted features, the concentration on merely a few of the particular and salient features, such as the structure, outline, and color, means that the models inevitably lack strong capability to represent the abstract characteristics of buildings [122]. Thus, the high-performance generalization of building segmentation remains a formidable challenge.

In this study, we propose to adopt deep learning here refer to DCNNs [38] as a alternative strategy to perform automatic building semantic segmentation. Recently, DCNNs are rapidly developing with the help of the dramatically increased availability of large-scale datasets [67] as well as the improvement of computing capability [141]. Instead of artificially designed feature engineering [142], DCNNs can automatically map the data into compact intermediate features and representations which akin to principal components with gradient descent [143] and backpropagation [77]. Different DCNNs architectures have been proposed and successfully applied in various domains [144], more importantly, DCNNs could outperform the state of the art in semantic segmentation tasks [145]. The following is an overview of some representative studies on semantic segmentation oriented DCNNs architectures and corresponding applications.

FCNs method [37], proposed by Long et al. at 2015, is the pioneering study of deep learning based semantic segmentation. FCNs innovatively adopts sequential convolutional operations and learnable upsampling to perform pixel-to-pixel translation. Meanwhile, by applying element-wise addition to the feature map, the result can represent more location and detail information which are destroyed by max-pooling layer. Three variants: FCN32s, FCN16s, and FCN8s, are proposed according to shrinking and upsampling level of different intermediate layers in FCNs. Subsequently, instead of adopting element-wise addition to fusion the information of previous feature and upsampling feature, Ronneberger et al. proposed U-Net [43], which applies multiple skip connections between upper and downer layers, and creatively concatenates each other in channel dimension. The method has been widely utilized in medical image segmentation precisely with high robustness. After that, aiming at solving location information lost problem in the max-pooling process, SegNet method [44] is proposed by Badrinarayanan et al. at 2015. Compared with FCNs, SegNet adopts unpooling which utilizes pooling-indices of corresponding max-pooling operation to perform upsampling. Since deconvolution [146] could reverse the effects of convolution on recorded data with learnable parameters, Noh et al. proposed hourglass-like DeconvNet [147] to enhance SegNet with deconvolution in upsampling layers, with the help of deconvolution and pooling-indices, the location and category details could be well presented to the end. Afterwards, Mon et al. proposed a residual encoder-decoder network named RedNet [148] in 2016. The network utilizes half padding without strides and gets rid of the max pooling directly to avoid losing location information. Also, residual skip connection with element-wise addition is adopted to help the network to go deeper with more feature representations. In 2017, feature pyramid networks (FPN) [149] is presented by Lin et al. which creates a pyramid of feature and use them for segmentation. The features obtained in bottom-up and top-down pathways are fused by residual convolutional blocks [121], and finally concatenate with the help of interpolation. FPN can also be flexibly utilized in object detection field with Region Proposal Network (RPN) [150]. The ResUNet [151] method proposed in 2018 adopts the basic structure of U-Net while replacing the convolutional block in VGG-like [86] structures with Residual block, such operation facilitates the convergence and enhances the representation ability of the model, which lead to gain better segmentation performance. More recent studies like Pyramid Scene Parsing Network(PSPNet) [152], Similarity Group Proposal Network (SGPN) [153], DeepLab [154], PointNet++ [155], FRRN [156], OCNNet [157], etc. also show the tremendous capability in a wide variety of semantic segmentation tasks like scene segmentation [158], panoptic segmentation [159], semi-supervised segmentation [160], etc.

Compared with the segmentation targets in other studies, urban buildings are usually in intense density with diversity, and the features in which such as texture, outline, color,

etc. are quite unique. With regard to texture and outline, the difference would lead the existing deep learning models to face problems such as the failed convergence, overfitting, and low accuracy. About color and noise, the high variability makes the model should be in high robustness. Besides, due to the diversity of building category as well as the difficulty of obtaining large-scale training dataset, developing a model which can be well trained with very limited training dataset fast and precisely is crucial. In this study, a specific deep learning architecture named concatenate feature pyramid networks (CFPN) is proposed based on feature concatenation and [69] and feature pyramid [70] methods. Given the variety of the urban buildings as well as the limited training dataset, CFPN model is deliberately designed in lightweight structure with relatively few parameters, which could be trained easily. Meanwhile, with the help of feature concatenation and feature pyramid, CFPN is capable of extracting adequate robust feature from complex imagery to perform segmentation with high accuracy. Additionally, since image color performs an important role in training [161], we adopted Wallis filter [162] to conduct color balance between training and testing datasets. The experimental results reveal that the proposed model could outperform other baselines by 3.55% to 7.89% and extract building polygon efficiently.

3.1 Data

3.1.1 Study Area

As one of the world's highest density urban areas, Tokyo contains intensely dense buildings with a huge diversity and complexity. In this study, we deliberately selected some representative study areas in downtown Tokyo to demonstrate the feasibility of proposed model in building semantic segmentation. The training areas cover about $17km^2$ and is mainly located in Koto, Taito, and Sumida districts, which include a wide variety of land use categories such as residential, commercial, and industrial areas. In addition, an area of about $12km^2$ in Setagaya district and adjacent areas are selected to perform testing.

3.1.2 Data Source

The remote top-view three-band RGB aerial satellite imagery acquired in March 2016 with a resolution of $0.160m$ were used as the training and testing datasets. In terms of the annotated dataset, to best represent the building footprints, a polygon-based method was used to conduct the annotation, in which, the polygon maximizes the shape of a building from an orthophoto, and any adjoining buildings are marked as a single

building. Owing to the limitations of interpretation based on human-based vision, a few small errors are inevitable.

3.2 Methods

In this section, we present our methods for building semantic segmentation. The three main procedures in the framework are: data processing, model training, and testing with related evaluation. First, aerial imagery of the study area obtained from the source undergoes data preprocessing to generate training data for semantic segmentation. Subsequently, the obtained data is fed into the proposed CFPN model to train the segmentation model. Here, 70% of the training data is used for training, and the remaining 30% is used for cross-validation. To evaluate the quality of the segmentation model, we apply six commonly used evaluation metrics that include precision, recall, overall accuracy [163], F1-score [164], the kappa coefficient [165], and the Jaccard index or intersection over union (IoU) [166]. It should be noted that segmentation models would be retained in case the bad results are generated when conducting cross validation. After testing, the quality of the semantic segmentation results is evaluated by the segmentation assessment criteria mentioned above. To clearly reflect the capability of different models, here, the evaluation metrics are calculated without any post-processing.

3.2.1 Preprocessing

Considering the impact of color difference in model training and testing, balancing the color of data source based on an identical template image can mitigate the variance to a certain degree. As an image filter, Wallis filter [167] scans the image and makes every pixel in the output image have a specified mean and standard deviation. In this experiment we choose Wallis filter to be the color balance tool, and the target image will be converted into a new color space based on the provided template. As shown in the following equation:

$$g(x, y) = [f(x, y) - m_c] \frac{CV_s}{CV_s + (1 - C)V_s} + bm_s + (1 - b)m_c \quad (3.1)$$

Where $g(x, y)$ refers to output result, $f(x, y)$ indicates input, V_c and V_s refer to input and template variance respectively, while m_s and m_c means template and output mean values. When utilizing Wallis filter in our experiment, template HRRS image need to be prepared, in order to calculate its variance and mean value.

Some representative examples as shown in Figure 3.1, the left input images are converted in a new color space which could be balanced with the template image.

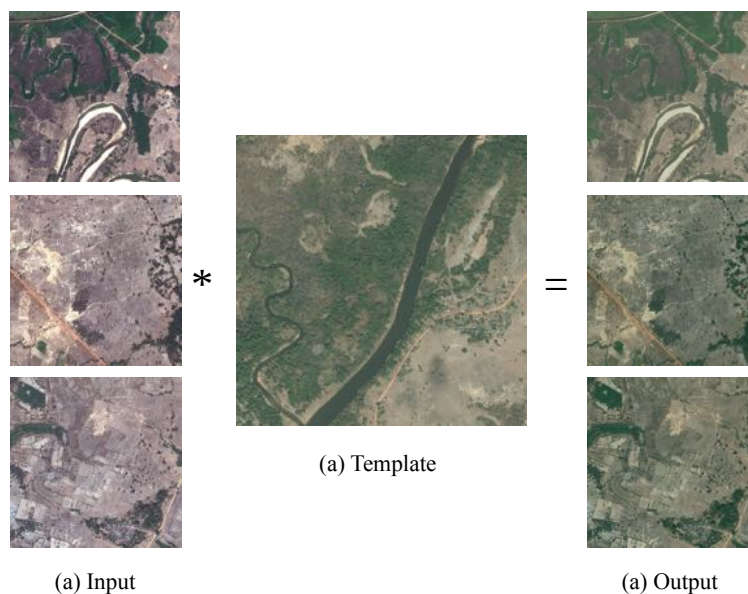


FIGURE 3.1: Color balance for preprocessing via Wallis filter.

After that, the aerial imagery is sliced into patches sized 224×224 pixels by using a random sliding window to generate data for model training and cross-validation purposes. Simultaneously, corresponding ground truth patches with consistent size are generated via the annotation dataset as well. In addition, to reduce the influence caused by data availability, data augmentation techniques [168] like random rotation and flip are also adopted to enrich the training data for both the segmentation.

3.2.2 Concatenate Feature Pyramid Networks

Image pyramid is capable of representing a certain image at vastly different scales with scale-invariant characteristic [169]. Conducting feature extraction over the entire image pyramid in both pyramid and position levels can generate featurized image pyramid, which enables the model to detect objects under multiple scales simultaneously. Given the advantages of multi-scale feature, there has been an increasing awareness of the potential of applying it for detecting objects in different scales, especially for small ones. However, generating multi-scale by adopting image pyramid is too compute and memory intensive to train the model end-to-end. Alternatively, the multi-scale feature in pyramidal shape can also be generated from a single image by utilizing DCNNs with subsampling layers. Due to the depth difference, multi-scale feature maps owning

different resolution have discrepancy in feature representation capability, and feature map which closer to the image layer composed of low-level feature that are not effective for accurate semantic segmentation. In this study, inspired by FPN, we deliberately designed a lightweight structure named as CFPN with relatively few parameters, with the help of feature concatenation and feature pyramid, CFPN is capable of extracting adequate robust feature from complex maps to perform segmentation with high accuracy. The architecture is shown in Figure B.1.

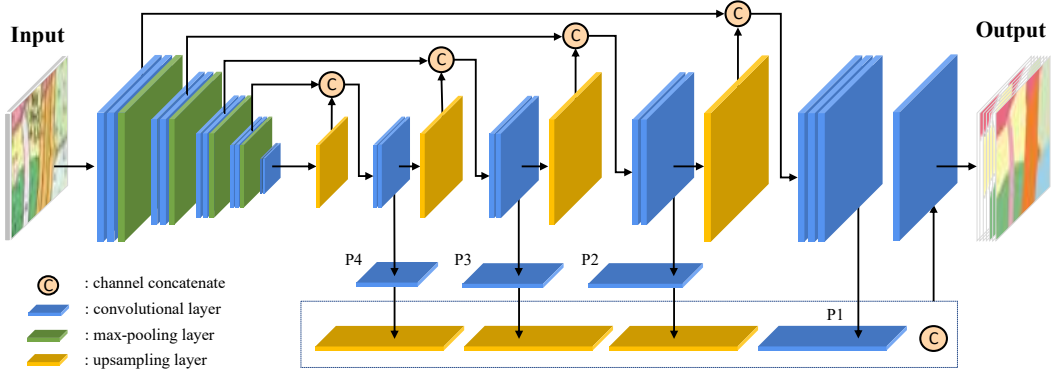


FIGURE 3.2: Concatenate feature pyramid networks for building semantic segmentation

In this study, except for the architecture modification from FPN, we also enhanced the original FPN in some important ways. To avoid dead neurons in the back-propagation step as well as to benefit from initialization, we use leaky ReLU [170] instead of ReLU after each convolution. Concretely, the convolution operation which performs element-wise multiplication via kernels, can be formulated as follows:

$$z = \sum_{i=1}^{h_f} \sum_{j=1}^{w_f} \sum_{d=1}^{c_l} \Theta_{i,j,d,d'} \times x_{i,j,d} + b_{d'} \quad (3.2)$$

where h_f, w_f represent the height and width of the kernel Θ , c_l is the number of channels for input x in layer l , and b in shape $1 \times 1 \times 1 \times d'$ donates the bias.

Then, leaky ReLU ϕ is utilized to generate the hypothesis from z :

$$\phi(z) = \begin{cases} z & \text{if } z > 0 \\ 0.01z & \text{otherwise} \end{cases} \quad (3.3)$$

Subsequently, batch normalization [128] is also added and extensively applied after each non-linearity to accelerate the training and reduce internal covariate shift. The two

parameters in batch normalization, scale γ and shift β , can be learned by:

$$Y_B = \gamma \frac{X_B - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} + \beta \quad (3.4)$$

where X_B and Y_B denote all input and output in mini-batch B . μ_B and σ_B^2 refer to mean and variance of corresponding mini-batch.

Furthermore, to avoid over-fitting, we eliminate the redundant features by adopting dropout [126], and the final binary classification of either building or non-building is predicted by using the sigmoid function. Here, the cross entropy expressed by Equation 3.5 is used to penalize the inconsistency between prediction \hat{Y} and ground truth Y . Further, H and W are the height and width of both the prediction and ground truth, respectively.

$$L(Y, \hat{Y}) = -\frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W \left(Y_{i,j} \times \log(\hat{Y}_{i,j}) + (1 - Y_{i,j}) \times \log(1 - \hat{Y}_{i,j}) \right) \quad (3.5)$$

3.2.3 Assessment Criteria

We assess the properties of the resulting segmentation \hat{Y} with regard to the ground truth Y via six criteria: precision, recall, overall accuracy, F1-score, the kappa coefficient, and the Jaccard index. For the sake of simplicity, tp , fn , fp , and tn , represent the basic terms in the confusion matrix: true positive, false negative, false positive, and true negative, respectively.

Precision and recall are both measures of relevance. Here, Precision (Equation 3.6) measures the proportion of relevant results in the list of all returned search results, and refers to the percentage of correctly predicted buildings to the total number of predicted buildings.

$$Precision = \frac{tp}{tp + fp} \quad (3.6)$$

Contrary to this, recall (Equation 3.7) measures the proportion of the relevant results returned by the segmentation model to the total number of relevant results that could have been returned, and refers to the correctly predicted buildings as a percentage of the exact total number of buildings.

$$Recall = \frac{tp}{tp + fn} \quad (3.7)$$

A trade-off between precision and recall is important. Thus, the F1-score, which takes both precision and recall into account and finds an optimal blend for them, is applied. The formula is as follows:

$$F_1 = \frac{2 * Precision * Recall}{Precision + Recall} \quad (3.8)$$

where the relative contribution of precision and recall to the F1 score are the same.

Overall accuracy, as shown in Equation 3.9, is also an essential metric in semantic segmentation. It refers to the proportion of correctly predicted building and non-building areas of the total number of areas to predict.

$$Overall \ Acc = \frac{tp + tn}{tp + tn + fp + fn} \quad (3.9)$$

To measure the level of agreement between two objective annotators, kappa coefficient is also applied as follows:

$$Po = Overall \ Acc \quad (3.10)$$

$$Pe = \frac{(tp + fp) \times (tp + fn) + (fn + tn) \times (fp + tn)}{(tp + tn + fp + fn)^2} \quad (3.11)$$

$$Kappa = \frac{Po - Pe}{1 - Pe} \quad (3.12)$$

where Po is identical to the overall accuracy and refers to the observed agreement ratio, and Pe is the probability of the expected agreement when both annotators assign building areas randomly.

Moreover, as the most prevalent criterion for segmentation problems, the Jaccard index in Equation 3.13 is used to measure the dissimilarity between the predicted and extracted building areas.

$$Jaccard = \frac{tp}{tp + fp + fn} \quad (3.13)$$

All six of the segmentation assessment criteria mentioned above reach their best value at 1 and worst score at 0.

3.3 Results

To demonstrate the feasibility of the proposed building semantic segmentation framework, we compare it with some state-of-the-art models including FPN, UNet, SegNet, FCNs, etc. To evaluate the robustness of methods, entire testing area is splitted into

four regions, where buildings and other important land features present in different characteristics in terms of structure, density, size, etc.

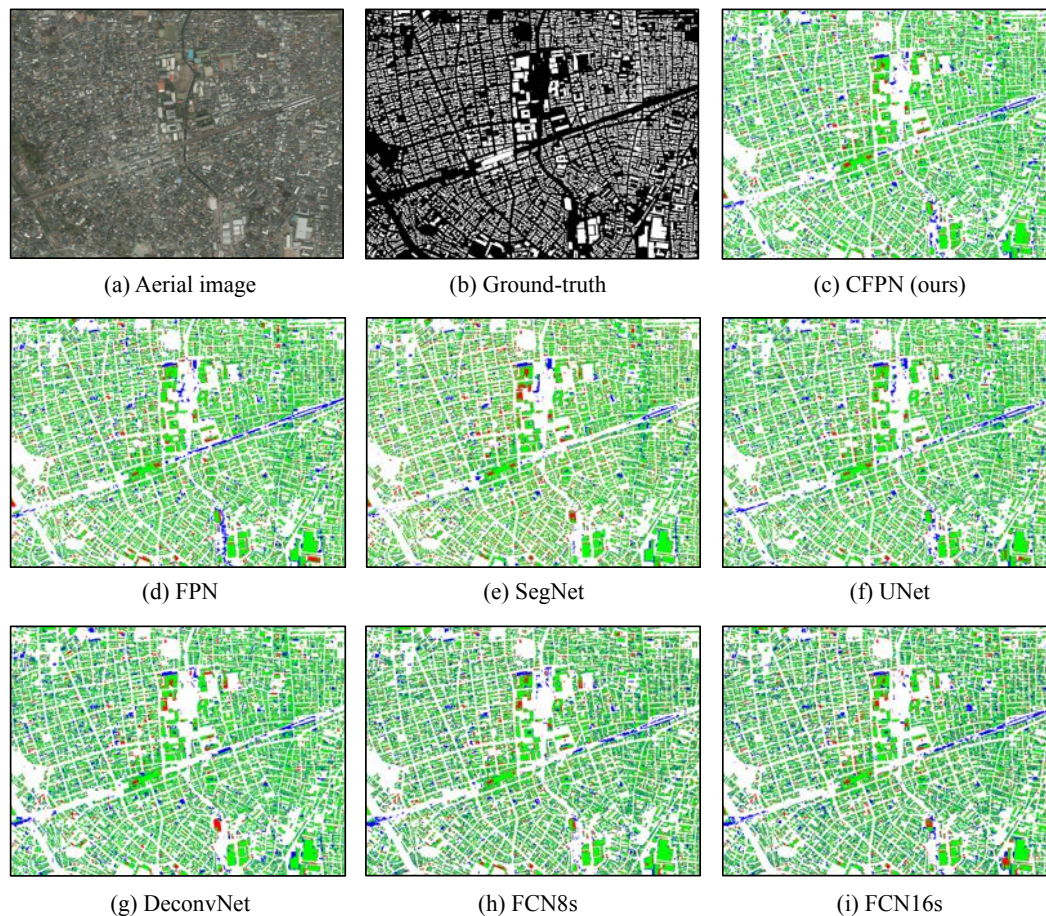


FIGURE 3.3: Quantitative results for test region 1.

This section presents the qualitative and quantitative results of the building semantic segmentation of the four regions via proposed method and baselines. More specifically, with respect to the qualitative results, the assessment criteria introduced in Section 3.2.3 are applied. The quantitative comparison results are shown in Figure 3.3, 3.4, 3.5, and 3.6. The different colors: green, red, blue, and white, are used to indicate the tp , fn , fp , and tn pixels in the segmentation results, respectively. Moreover, the corresponding quantitative results are shown in Table 3.1, 3.2, 3.3, 3.4, respectively.

Since Setagaya-district is mainly a residential area with representative land features like parks, narrow roads, and residential buildings, while the training area in other districts contains much more diverse features like dock, large commercial regions, sea, boats, etc., such discrepancy leads fp especially in high density residential areas, park, and railway. Compared with baselines, both qualitative and quantitative results shown above reveal that the proposed methods outperform others, especially in F1-score, IoU, and Kappa.

TABLE 3.1: Quantitative results for test region 1.

Model	Overall Acc.	Precision	Recall	F1-score	IoU	Kappa
CFPN(ours)	0.867	0.794	0.843	0.818	0.692	0.714
FPN	0.858	0.790	0.815	0.802	0.669	0.691
SegNet	0.865	0.828	0.778	0.802	0.670	0.700
UNet	0.861	0.790	0.827	0.808	0.678	0.699
DeconvNet	0.853	0.764	0.842	0.801	0.668	0.685
FCN8s	0.841	0.758	0.808	0.782	0.642	0.657
FCN16s	0.835	0.744	0.81	0.776	0.634	0.645

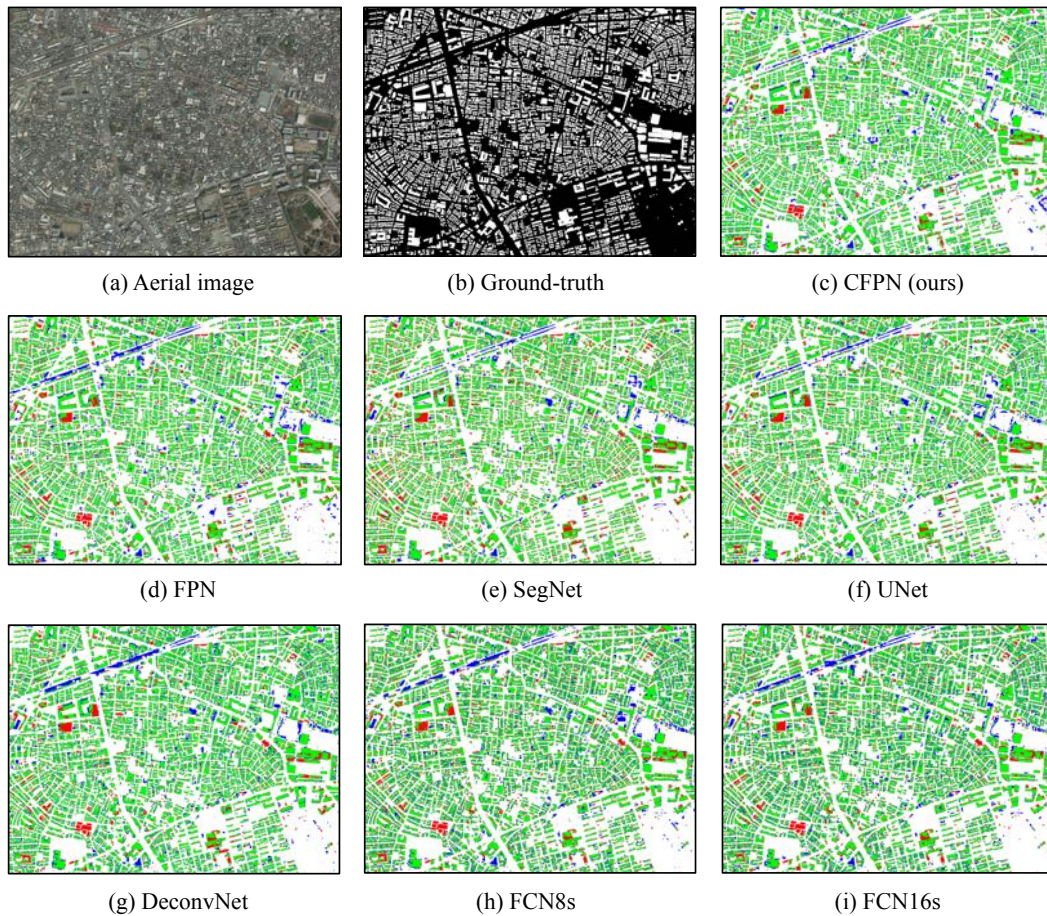


FIGURE 3.4: Quantitative results for test region 2.

TABLE 3.2: Quantitative results for test region 2.

Model	Overall Acc.	Precision	Recall	F1-score	IoU	Kappa
CFPN(ours)	0.875	0.811	0.826	0.818	0.693	0.724
FPN	0.869	0.809	0.804	0.807	0.676	0.708
SegNet	0.869	0.836	0.765	0.799	0.665	0.702
UNet	0.874	0.817	0.809	0.813	0.685	0.717
DeconvNet	0.868	0.792	0.828	0.81	0.68	0.709
FCN8s	0.851	0.77	0.801	0.785	0.647	0.672
FCN16s	0.851	0.767	0.807	0.786	0.648	0.672

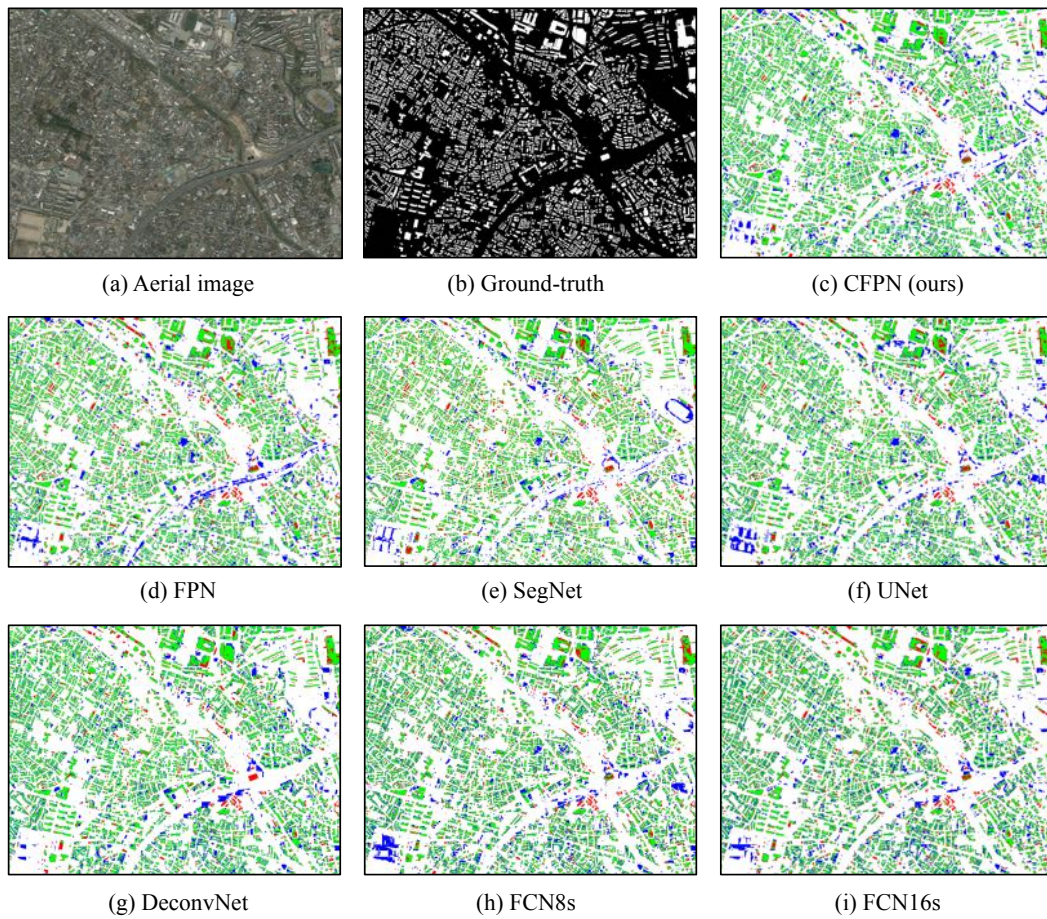


FIGURE 3.5: Quantitative results for test region 3.

TABLE 3.3: Quantitative results for test region 3.

Model	Overall Acc.	Precision	Recall	F1-score	IoU	Kappa
CFPN(ours)	0.885	0.713	0.82	0.763	0.617	0.687
FPN	0.879	0.715	0.774	0.743	0.592	0.664
SegNet	0.886	0.745	0.753	0.749	0.599	0.676
UNet	0.878	0.704	0.797	0.747	0.597	0.668
DeconvNet	0.881	0.71	0.803	0.754	0.605	0.676
FCN8s	0.867	0.681	0.773	0.724	0.567	0.637
FCN16s	0.868	0.684	0.771	0.725	0.568	0.638

TABLE 3.4: Quantitative results for test region 4.

Model	Overall Acc.	Precision	Recall	F1-score	IoU	Kappa
CFPN(ours)	0.900	0.789	0.831	0.809	0.679	0.742
FPN	0.891	0.775	0.803	0.789	0.651	0.715
SegNet	0.897	0.818	0.764	0.79	0.653	0.721
UNet	0.897	0.788	0.813	0.8	0.667	0.731
DeconvNet	0.894	0.772	0.829	0.8	0.666	0.728
FCN8s	0.884	0.754	0.805	0.779	0.638	0.7
FCN16s	0.882	0.752	0.8	0.775	0.633	0.696

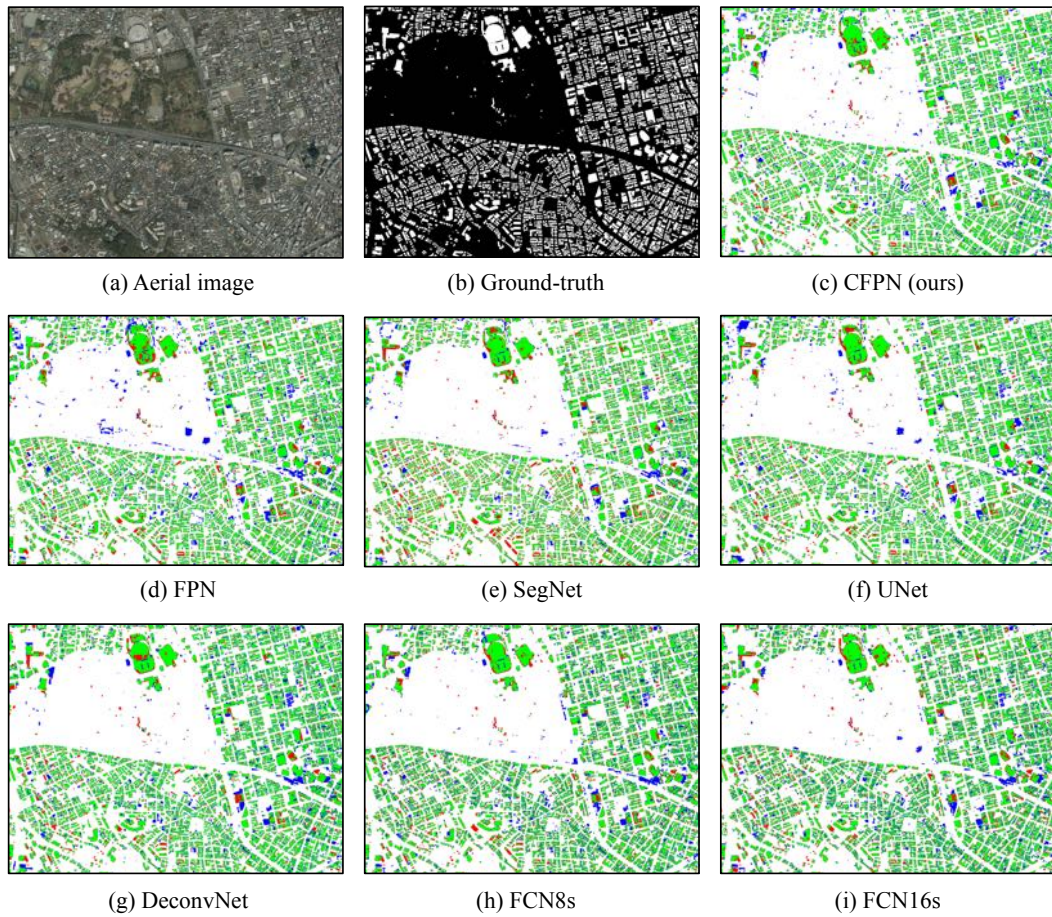


FIGURE 3.6: Quantitative results for test region 4.

Moreover, for better visualization, as shown in Figure 3.7, we randomly selected and enlarged some subregions in details, and they reveal that the proposed method could generate better segmentation with less fp and fn .

3.4 Discussion

3.4.1 Robustness

To demonstrate the competitive robustness of the proposed method, in this part, we show the average segmentation performance obtained by adopting different methods among four testing regions in Table 3.5 and Figure 3.8. Additionally, the feasibility of image denoise and building outline extraction by applying our proposed method is illustrated in Figure 3.9 as well.

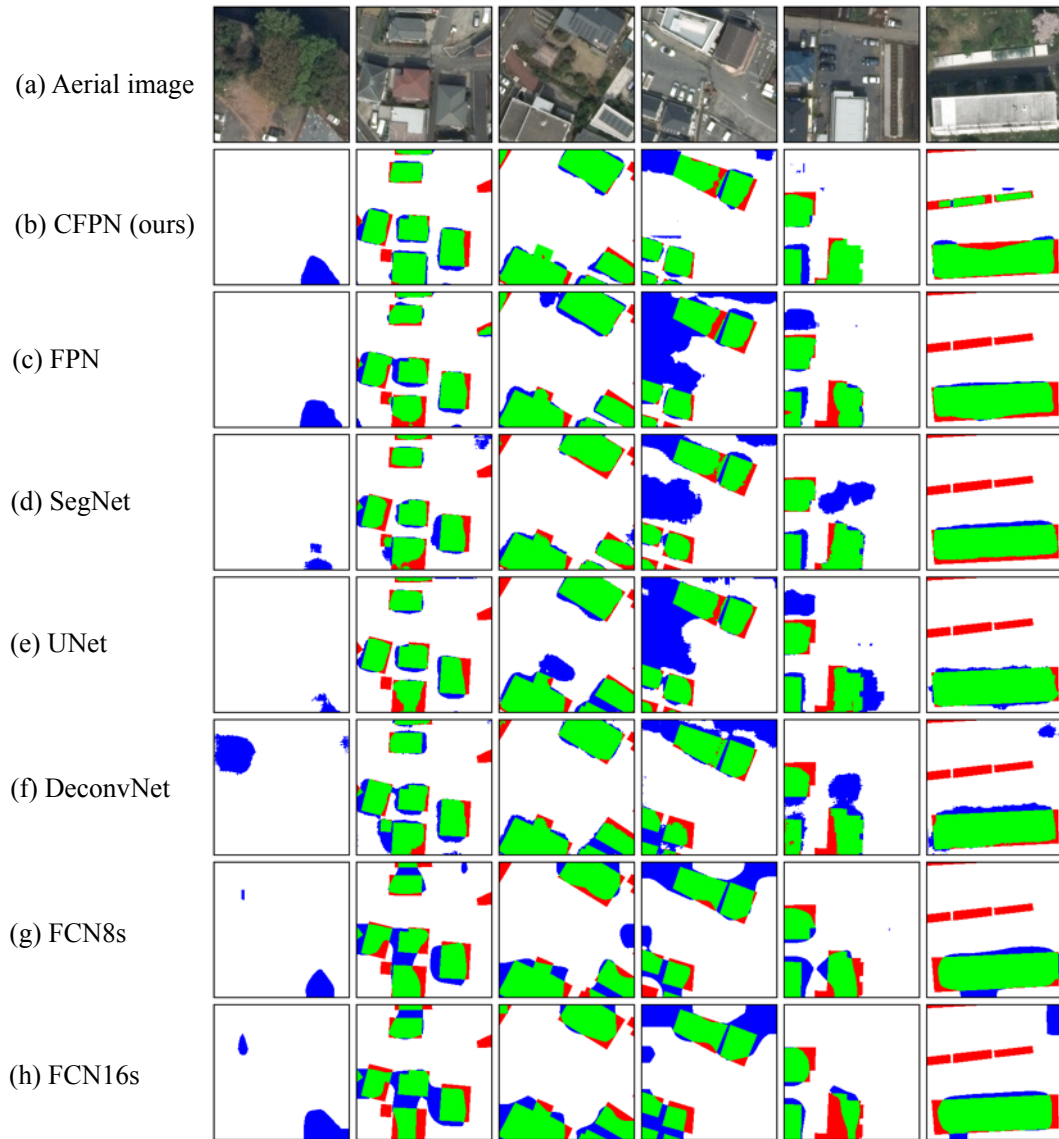


FIGURE 3.7: Subregion comparison for different models.

TABLE 3.5: Average performance of different models among four testing regions.

Model	CFPN(ours)	FPN	SegNet	UNet	DeconvNet	FCN8s	FCN16s
mF1-score	0.802	0.785	0.782	0.792	0.791	0.768	0.766
mIoU	0.670	0.647	0.642	0.657	0.655	0.624	0.621
mKappa	0.717	0.694	0.704	0.704	0.700	0.667	0.663

In general, our CFPN method with outperforms the other methods, improving the mean F1-score from 0.766 to 0.802, mean kappa from 0.621 to 0.670, and the mean Jaccard index from 0.663 to 0.717.

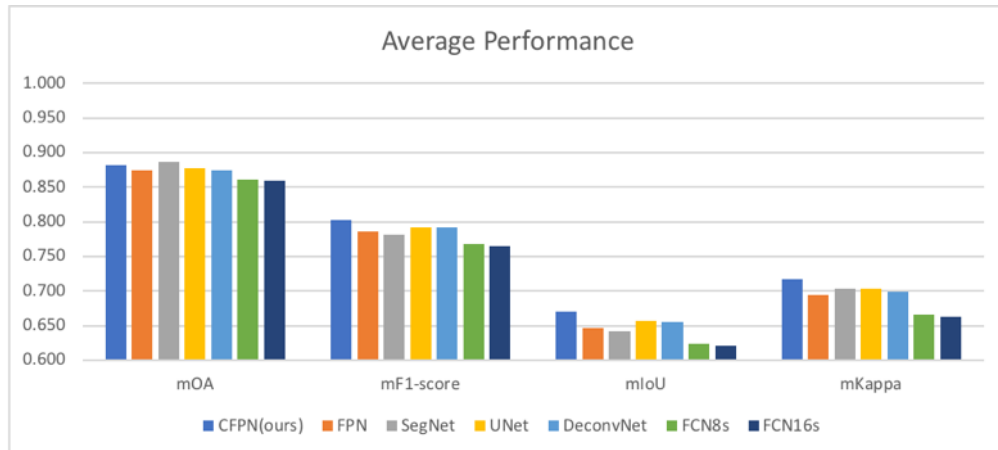


FIGURE 3.8: Average performance of different models among four testing regions.

Figure 3.9 presents six groups of randomly selected visualization results generated by CFPN. From top to bottom rows, there are original images, extracted edges by Canny, building segmentation and outline extraction from CFPN model. In general, the extracted outlines through Canny detector contains pretty much noise (see 2nd Row). Our CFPN can segment the major part of buildings from most of the selected RGB images (see 3rd Row). Building outlines extracted from segmentation results show much fewer false negatives (see 2nd Row vs. 4th Row).

3.4.2 The Impact of Image Color

To investigate the impact of image color on segmentation results, we trained two models based on normalize dataset by Wallis filter and unnormalized dataset, respectively. The quantitative results shown in Table 3.6 reveal that by balancing the color space among training and testing dataset, the variance can be mitigated, which improves the model performance to a certain degree. Since the spectrum difference problem is not serious in this experiment, the further investigation among more diverse dataset and spectrum is essential.

TABLE 3.6: The impact of image color on segmentation results. The comparison between model trained by normalized (Norm) and unnormalized (Unnorm) dataset.

Method	mF1-score	mIoU	mKappa
Norm	0.802	0.670	0.717
Unnorm	0.792	0.660	0.711

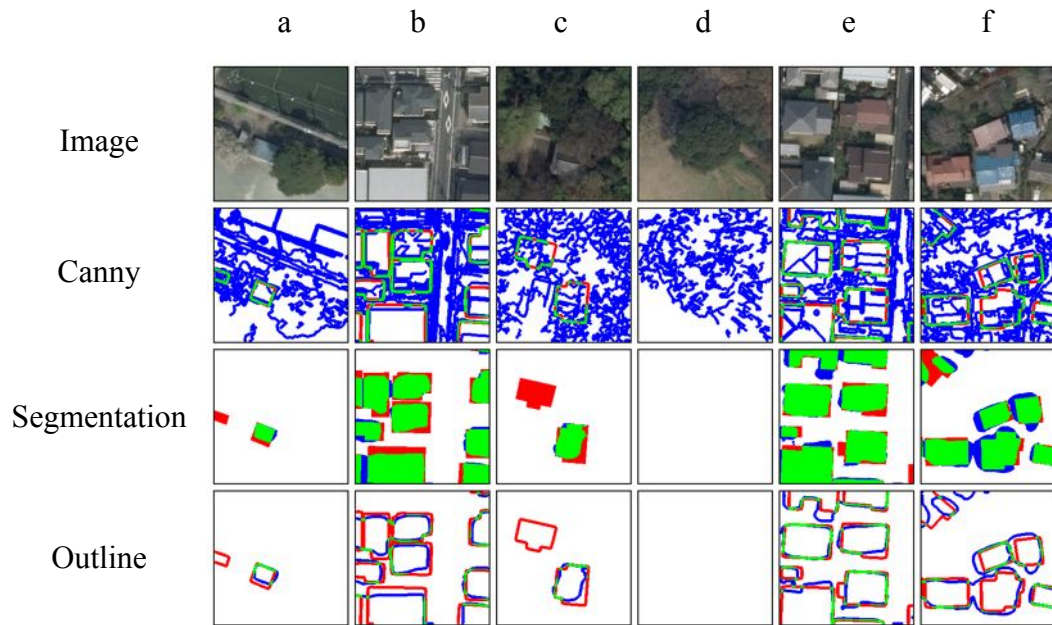


FIGURE 3.9: Building outline extraction and denoise. The 1st row: original aerial images; the 2nd row: canny edge detection results of the original aerial images; the 3rd row: segmentation and denoised results; the 4th row: extracted outline for buildings. The green, red, blue, and white channels in the results represent true positive, false positive, false negative, and true negative predictions, respectively.

3.5 Conclusions

In this chapter, we presented a novel framework for building semantic segmentation from aerial imagery in Tokyo. The experimental results demonstrate the potential and the capability of the proposed CFPN model for semantic segmentation, which could outperform other baselines by 3.55% to 7.89% and extract building polygon efficiently. In addition, although the influence of color and spectrum difference is less serious in the same data source, it is important to carefully consider it especially on multi-source remote sensing imagery, which we aim to study in future.

Chapter 4

Super-Resolution Integrated Building Semantic Segmentation

Despite the success of FCN-based models in several building semantic segmentation tasks, the discussions on building extraction via multi-source remote sensing imagery of which the spatial resolution differs are quite inadequate. With the dramatically increasing availability of new large-scale remote sensing data sources, the ever-expanding choices of datasets can be utilized in semantic segmentation tasks [65–67], and the case that training and testing datasets obtained from multiple sources with different resolution would be inevitable and ubiquitous in many practical applications [68].

In general, differences between the resolution of the training and testing datasets would greatly influence building semantic segmentation. Three factors are mainly responsible for the problems in this regard. First, the resolution defines the ability of a single pixel to cover the Earth’s surface, which would cause the same building to appear to have a different size in multiple remote sensing images of different resolution. A recent study [171] indicated that the factor of building size strongly impacts upon the capability of the DCNN model, and a model trained by using a building of a specific size would find it difficult to detect buildings of a significantly different size. Second, the resolution indicates the ability of the image to represent small objects. Thus, a small-sized land feature would be deformed or ignored in a low-resolution (LR) image due to the limited resolution. Many studies regarding small object detection [172–174] demonstrated the difficulty of solving this problem. Furthermore, as an important indicator, resolution measures the richness of information contained in remote-sensing imagery [175], in which a different resolution represents a different frequency information distribution, which greatly affects the features of the building such as its color, outline, and texture [176]. For the aforementioned reasons, a DCNN model trained at a specific resolution

would find it fundamentally difficult to correctly represent the features of the testing dataset at another resolution, and this would result in a poor generalization of semantic segmentation. Thus, overcoming the constraint of resolution differences among multi-source remote sensing imagery would facilitate the development of building semantic segmentation to a considerable extent.

To deal with the severe problems caused by resolution difference between multi-source remote sensing imagery, the solution can be mainly classified into image transform based [177], data augmentation based [178], and transfer learning based [179] methods. With regard to image transform, the usual approach would be to downscale the high-resolution (HR) imagery into LR space by using downsampling methods [180] or to upscale LR imagery to HR space using a single filter such as bicubic interpolation [181]. The relevant drawbacks are obvious since downsampling would lead to undesired side-effects such as the loss of spatial information whereas interpolation would generate insufficient large gradients along edges and high-frequency regions by simply weighted averaging neighboring LR pixel values [182], small buildings would not be the same as larger ones even if up-scaled. With regard to data augmentation, methods such as color transformation, affine transformation, rotation, and linear scaling could enrich the variety of the training dataset, but could not supplement important features such as high-frequency information effectively at LR. And about transfer learning, although it owns the capability to rebuild the model based on utilizing the knowledge acquired from the previous task, once the feature-space and information distribution changes caused by resolution, the preparation for adequate amount of new training dataset is still unavoidable, which limits the efficiency and scalability in practical applications.

Given the difficulties faced by the methods mentioned above, super-resolution (SR) [183] has emerged as a promising alternative strategy to solve the problem. Aimed at increasing the image resolution while providing finer spatial details than those captured by the original acquisition sensors, SR could balance the size and detail of land features between the training and testing datasets to a certain degree [184]. In addition, as a highly ill-posed problem, SR operation is considered to be a one-to-many mapping from LR to HR space, which can have multiple solutions. Recent studies on DCNN-based SR models have shown tremendous capability in super-resolving an LR image into HR space, showing that generating high-quality SR remote-sensing imagery is achievable. A detailed review of additional DCNN-based SR models and their corresponding applications was recently published [185].

In this study, contrary to previous work, we propose to integrate super-resolution (SR) techniques into the existing segmentation framework to address the problem of building semantic segmentation in multi-source remote sensing imagery with different spatial

resolution. To validate the feasibility of the proposed method, two high-performance DCNN-based models, namely efficient sub-pixel convolutional neural network (ESPCN) [71] and UNet, are adopted to perform SR and the semantic segmentation operation, respectively. In addition, three-band RGB HR aerial imagery and single-band grayscale LR panchromatic satellite imagery are selected as representative multi-source remote sensing imagery to conduct training and testing, respectively. It is worth emphasizing that, to the best of our knowledge, there has not been any empirical study using SR techniques for the building semantic segmentation from multi-source imagery with different resolution.

The main contributions of this study are three fold:

- We discussed the challenge and limitation of recent deep learning based studies on building semantic segmentation of building while under multi-source imagery with different resolution circumstance.
- We innovatively presented a novel SR integrated building semantic segmentation framework to tackle the problem caused by the unaligned resolution between training and testing data, and investigated the feasibility of the proposed method based on comprehensive experiments.
- The experimental results demonstrate the proposed method could achieve state-of-the-art performance, and the IoU and Kappa is approximately 19.01% and 19.10% higher than that of the method without SR, respectively. It indicates the effects of SR on segmentation performance in remote sensing imagery, which would benefit the remote sensing community from literature review to future directions.

The remainder of this chapter is organized as follows. Section 4.1 introduces the area we studied and the data source. Then, the workflow of the proposed method is explained in Section 4.2, where details of the algorithms as well as the evaluation metrics are also presented. After that, the experimental results and discussion appear in Section 4.3 and 4.4. Finally, the conclusions are drawn in Sections 4.5.

4.1 Data

4.1.1 Study Area

As one of the world's highest density urban areas, Tokyo contains intensely dense buildings with a huge diversity and complexity. Such characteristics of urban landscape lead

spatial resolution to play an important role in semantic segmentation task. In this study, we deliberately selected some representative study areas in downtown Tokyo to demonstrate the feasibility of SR in building semantic segmentation. Figure 4.1b shows the detailed study area. We divided the entire area into training and testing areas indicated in purple and green, respectively. The training area covered 33km^2 and is mainly located in the Setagaya, Koto, and Sumida districts, which include a wide variety of land use categories such as residential, commercial, and industrial areas. In addition, an area of 3km^2 in the Koto district with comprehensive land use was selected to perform testing.

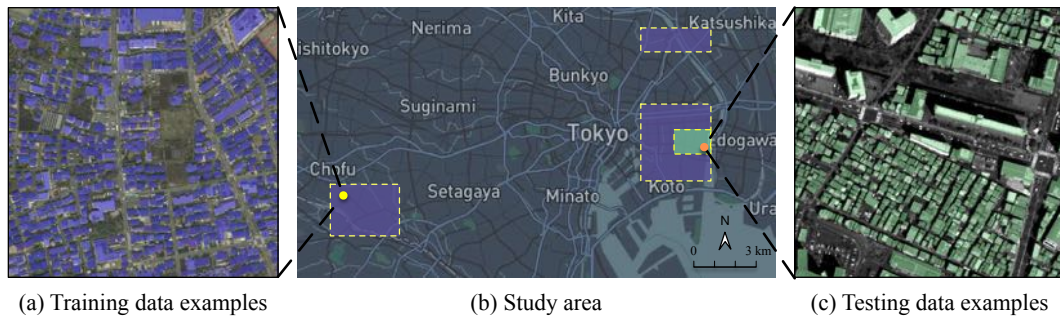


FIGURE 4.1: Materials. (a) and (c) show examples of the training and testing data, including high-resolution aerial imagery in which the corresponding buildings are annotated in purple and low-resolution satellite imagery with the relevant buildings annotated in green. (b) The study area divided into training and testing areas colored purple and green, respectively.

4.1.2 Data Source

The aerial and panchromatic satellite imagery were used as the training and testing datasets, respectively. The remote top-view three-band RGB aerial imagery in the training area was acquired in March 2016 with a resolution of 0.160m , and the source panchromatic imagery in the testing area was captured by the WorldView-2 sensor in May 2016 with spatial and radiometric resolution of 0.500m and 16-bit, respectively. In terms of the annotated dataset, a total of approximately 60,000 and 3,000 building footprints are contained within the training and testing areas, respectively. To best represent the building footprints, a polygon-based method via QGIS was used to conduct the annotation, in which, the polygon maximizes the shape of a building from an orthophoto, and any adjoining buildings are marked as a single building. Owing to the limitations of interpretation based on human-based vision, a few small errors are inevitable especially for high-density areas in LR satellite imagery. Some examples of training and testing imagery and their corresponding annotations are shown in Figure 4.1a in purple and in Figure 4.1c in green, respectively.

4.2 Methods

In this section, we present our novel framework for an SR integrated building semantic segmentation method. As shown in Figure 4.2, the three main procedures in the framework are: data processing, model training, and testing with related evaluation. The two processes that precede the testing stage can be considered as running in parallel in terms of both segmentation and SR model generalization. First, aerial imagery of the study area obtained from the same source undergoes parallel data preprocessing to generate training data for semantic segmentation and SR integration. Subsequently, the obtained data is fed into the proposed upper UNet and lower ESPCN to train the segmentation and SR model, respectively. Here, in both models, 70% of the training data is used for training, and the remaining 30% is used for cross-validation. To evaluate the quality of the segmentation model, we apply six commonly used evaluation metrics that include precision, recall, overall accuracy [163], F1-score [164], the kappa coefficient [165], and the Jaccard index or intersection over union (IoU) [166]. The SR model is assessed by using the peak signal-to-noise ratio (PSNR) [186], which is usually taken as an approximation to human perception of reconstruction quality. It should be noted that both segmentation and SR models would be retained in case the bad results are generated when conducting cross validation. After that, in the testing and evaluation procedure, we first input the processed LR satellite data into the trained SR model to generate related upscaled SR data; then, the trained segmentation model with proper hyperparameters is adopted to enable the generated testing SR satellite data to be used to make predictions. Finally, the quality of the semantic segmentation results is evaluated by the segmentation assessment criteria mentioned above. To clearly reflect the capability of different models, here, the evaluation metrics are calculated without any post-processing for both the semantic segmentation and SR processes.

This section details first the data preprocessing step, followed by the training strategies of the SR and segmentation models. Lastly, the testing method and related assessment criteria are proposed and explained.

4.2.1 Data Preprocessing

Data preprocessing is conducted in parallel to generate training data for both the segmentation and SR models. With respect to the segmentation, the three-band RGB HR aerial imagery is first converted into grayscale to align it with the single-band panchromatic testing LR satellite imagery; then, after applying basic color normalization methods such as adaptive histogram equalization [187], the aerial imagery is sliced into patches sized 224×224 pixels by using a random sliding window to generate data

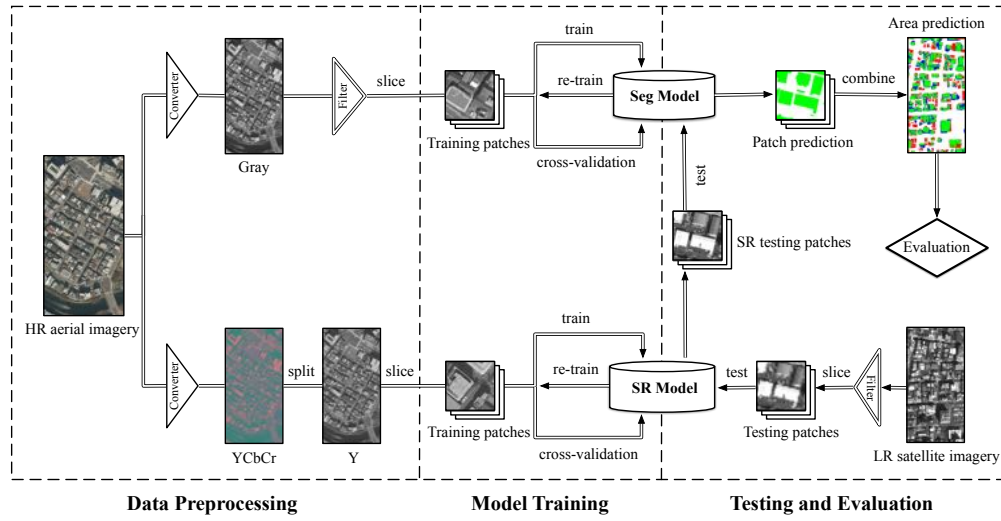


FIGURE 4.2: Framework of our building semantic segmentation method.

for model training and cross-validation purposes. Simultaneously, corresponding ground truth patches with consistent size are generated via the annotation dataset as well. In terms of the SR process, considering humans are more sensitive to luminance changes [188], we convert the aerial imagery from RGB into YCbCr color space, and only take the luminance channel in the YCbCr color space into consideration. Similar to the process of segmentation, the converted aerial imagery in the luminance channel is sliced into small patches sized 224×224 pixels. In addition, to reduce the influence caused by data availability, data augmentation techniques [168] are also adopted to enrich the training data for both the segmentation and SR processes.

4.2.2 Segmentation Model

Several effective segmentation models have been introduced in Section 4, to demonstrate the feasibility of proposed framework, in this study, we propose to adopt UNet architecture as a representative segmentation model to conduct building semantic segmentation.

UNet is one of a state-of-the-art models for image semantic segmentation, and has been successfully applied to perform different tasks with high accuracy and efficiency. The network architecture can be divided into two parts: a contracting path and a symmetric expansive path. The contracting path, which is regarded as a variant of VGG [86], contains five consecutive blocks for feature extraction and downsampling. Each of these blocks consists of two 3×3 unpadded convolutions followed by 2×2 max pooling, which provides the abstracted form of the representation while enlarging the receptive field. The expansive path, which can be considered as the reverse operation of the contraction

path, comprises four blocks and each contains an upsampling of the feature map followed by a 2×2 convolution. Importantly, before feeding the extracted feature map into the next block, the feature map generated in the contraction path with the same shape is integrated inside by concatenation. In addition, the number of feature channels are doubled and divided in half after each downsampling and upsampling, respectively. The non-saturated activator known as a rectified linear unit (ReLU) is adopted after each convolutional operation to perform nonlinear mapping. This architecture makes UNet suitable for mining very deep and abstract features.

Considering the characteristics of UNet, some advantages of adopting UNet architecture as segmentation model to conduct building semantic segmentation can be listed as follows. First, the architecture of UNet performs pixel-to-pixel and end-to-end mapping from input to output, which enables precise localization for the building segmentation result. Second, UNet can generate results in HR space by recovering HR representations. Instead of using pooling operators after successive convolutional layers, the architecture adopts upsampling with a large number of feature channels to increase the output resolution. In addition, the model has the capability to augment feature space by fusing the context from imagery acquired at different resolutions. Because HR features extracted from a contracting path and LR features upsampled by using an expansive path are combined through the process of concatenation, the feature space can be augmented to a certain degree.

4.2.3 SR Model

Aimed at recovering HR imagery from its LR information, SR is an important category of techniques for image processing and offers an excellent opportunity to facilitate the development of different remote sensing applications including building semantic segmentation. In recent years, deep learning based SR methods have been investigated quite intensively and have achieved state-of-the-art performance among various benchmarks of SR, some breakthrough studies such as SRCNN [189], VDSR [190], LapSRN [191], SRGAN [192], etc. To explore the feasibility of integrating SR into the building semantic segmentation task while simultaneously considering the characteristics of the available data source, we propose to adopt a typical deep learning based single-image super-resolution (SISR) method named ESPCN to increase the resolution of LR panchromatic satellite imagery to match that of the HR aerial imagery at some point. Ideally, the reconstructed SR imagery could be augmented with high-frequency information on condition that the spatial resolution is similar to that of the HR aerial imagery.

Instead of upscaling the LR input imagery X^{LR} into HR space before reconstruction, ESPCN directly extracts feature maps from LR space with the help of successive hidden convolutional layers. To generate SR imagery X^{SR} from X^{LR} with an upscaling factor r , X^{LR} with a shape of $h \times w \times c$ would undergo L layers of convolution operations. The first $L - 1$ layers can be described as:

$$f^1(X^{LR}, \Theta_1, b_1) = g(\Theta_1 \times X^{LR} \times b_1) \quad (4.1)$$

$$f^l(X^{LR}, \Theta_{1:l}, b_{1:l}) = g(\Theta_l \times f^{l-1}(X^{LR}) \times b_l) \quad (4.2)$$

where Θ_l and b_l with $l \in (1, L - 1)$ represent the learnable hyperparameters weights and biases, respectively. Function g is the activator ReLU used to perform nonlinear mapping.

The final layer f^L applies an efficient sub-pixel convolution operation, which learns an array of complex upscaling filters to upscale the LR feature maps into the HR output X^{SR} . The formula as follows:

$$X^{SR} = f^L(X^{LR}, \Theta_L, b_L) = PS(\Theta_L \times f^{L-1}(X^{LR}) + b_L) \quad (4.3)$$

where PS is a periodic shuffling operator that reshapes the feature maps of layer $L - 1$ from shape $h \times w \times c \cdot r^2$ into a tensor of shape $rh \times rw \times c$. Weights Θ_L are in the shape $h_f \times w_f \times c_{L-1} \times c \cdot r^2$.

During training, the input LR imagery X^{LR} can be synthesized efficiently by sub-sampling HR aerial imagery X^{HR} from shape $rh \times rw \times c$ to $h \times w \times c$ using a Gaussian filter. After generating the result in each epoch, the loss function pixel-wise mean squared error (MSE) (Equation 4.4) is used to measure the discrepancy between reconstructed X^{SR} and original X^{HR} , both in shape $rh \times rw \times c$. In addition, early stopping is adopted to end the training process once the model performance no longer improves after 100 epochs on cross-validation data.

$$L(X^{HR}, X^{SR}) = \frac{1}{r^2 h w} \sum_{i=1}^{rh} \sum_{j=1}^{rw} (X_{i,j}^{HR} - f_{i,j}^L(X^{LR}))^2 \quad (4.4)$$

In terms of the spatial resolution of the multi-source remote sensing imagery used in this study, the HR aerial imagery is approximately three times higher than LR panchromatic imagery; therefore, three SR models are trained by ESPCN by assigning the values 1, 2, and 3 to the upscaling factor r , respectively.

4.2.4 Assessment Criteria

The quality of the results obtained after semantic segmentation and the use of the SR model is evaluated by applying criteria based on a confusion matrix and image quality assessment (IQA), respectively.

The assessment criteria of the segmentation model are as with Chapter 3. Regarding the quantitative performance of the SR model, the most widely used evaluation criterion, PSNR, is adopted to measure the reconstruction quality of transformation. The PSNR, which is an objective IQA method, is calculated based on the maximum possible pixel value (denoted as MAX) and the pixel-level MSE between HR imagery X^{HR} and super-resolved SR imagery X^{SR} . The corresponding formulas are as follows:

$$MSE = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W (X^{HR} - X^{SR})^2 \quad (4.5)$$

$$PSNR = 10 \times \log_{10}\left(\frac{MAX^2}{MSE}\right) \quad (4.6)$$

We normalize the maximum possible pixel value between multi-source imagery by converting the value of 8-bit aerial imagery and 16-bit panchromatic imagery, and rescale both of them from 0 to 1. Thus, instead of relying on human visual perception, the quality of SR is computationally only related to the MSE.

4.3 Results

To demonstrate the feasibility of proposed SR integrated approach, we employ the same modified UNet model trained by HR aerial imagery as the backbone to test imagery in three main categories: LR, ESPCN based SR, and bicubic based interpolated imagery. Considering the exact resolution of HR aerial and LR panchromatic imagery as well as exploring the influence caused by their resolution difference, we upscale the testing LR panchromatic imagery with resolution $0.500m$ into 2, 3, 4 times by both ESPCN and bicubic interpolation methods. Thus, SR- and bicubic-based interpolated panchromatic imagery with resolution $0.250m$, $0.167m$, and $0.125m$ are generated. Moreover, to evaluate the robustness of methods, we deliberately divide the entire testing area into four regions based on land use, where buildings and other important land features present in different characteristics in terms of grayscale value, texture, structure, density, size, etc.

This section presents the qualitative and quantitative results of the building semantic segmentation of the four regions via different methods. More specifically, with respect to the qualitative results, the assessment criteria introduced in Section 4.2.4 are applied.

The quantitative results are shown in Figure 4.3, 4.5, 4.7, and 4.9. In these figures, (a) and (e) are LR panchromatic imagery and the corresponding segmentation result, (b), (c), and (d) are the segmentation results generated by the ESPCN-based methods with upscale factors of 2, 3, and 4, respectively. Further, (f), (g), and (h) are the segmentation results generated by the bicubic-based methods with upscale factors corresponding to the ESPCN-based methods. The different colors: green, red, blue, and white, are used to indicate the tp , fn , fp , and tn pixels in the segmentation results, respectively. Moreover, for improved visualization, as shown in Figure 4.4, 4.6, 4.8, and 4.10, enlargements of selected representative subregions in each region are displayed in a yellow window to reveal the details, which reflect the effect of applying different methods.

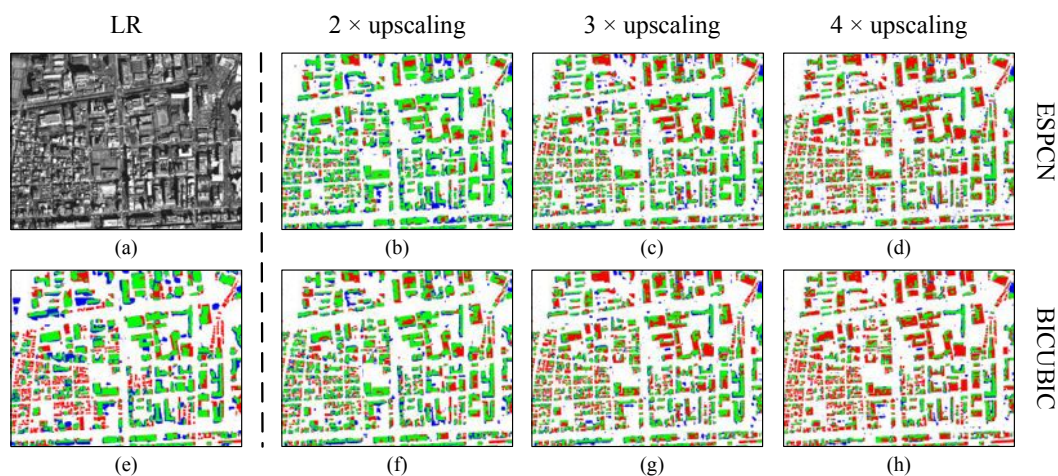


FIGURE 4.3: Qualitative results for test region 1.

Figure 4.3 shows the qualitative results for test region 1, which mainly contains commercial and residential areas, in which the types of buildings are particularly diverse, whereas the non-building areas include several open sided car parks and sports grounds. The corresponding quantitative results generated by the different methods are provided in Table 4.1, and indicate that the proposed ESPCN method with an upscale factor of 2 outperformed other models in terms of the recall, overall accuracy, F1-score, kappa, and Jaccard index. With respect to precision, the results are worse than those of other upscaled imagery but are still more accurate than those obtained for the original LR imagery.

Notably, as shown in the first row of Figure 4.4, in some residential areas, the size of building as well as the separation distance between adjacent buildings is quite small in LR imagery, which considerably increases the challenge of segmentation, and makes it difficult to identify buildings at all. The use of ESPCN not only enlarges the size of building and distance between adjacent buildings like bicubic interpolation does but

TABLE 4.1: Quantitative results for test region 1.

Model	Scale	Resolution	Precision	Recall	Overall Acc	F1-score	Kappa	Jaccard
LR	1	0.500	0.693	0.621	0.803	0.655	0.518	0.487
ESPCN	2	0.250	0.728	0.772	0.844	0.749	0.637	0.599
BICUBIC	2	0.250	0.752	0.645	0.829	0.694	0.576	0.532
ESPCN	3	0.167	0.737	0.606	0.816	0.665	0.540	0.499
BICUBIC	3	0.167	0.749	0.527	0.804	0.619	0.492	0.448
ESPCN	4	0.125	0.735	0.462	0.788	0.568	0.436	0.396
BICUBIC	4	0.125	0.755	0.388	0.777	0.512	0.387	0.344

also enrich the texture information. The effect can easily be seen in the enlarged views and related segmentation results, where all buildings are well segmented by adopting ESPCN with an upscale factor of 2, and, ESPCN outperforms the simple interpolation methods for every respective upscale factor. Similar to the residential areas, as shown in the third row of Figure 4.4, the external outlines of buildings in the commercial area are particularly clear when using ESPCN, which produces more accurate segmentation results.

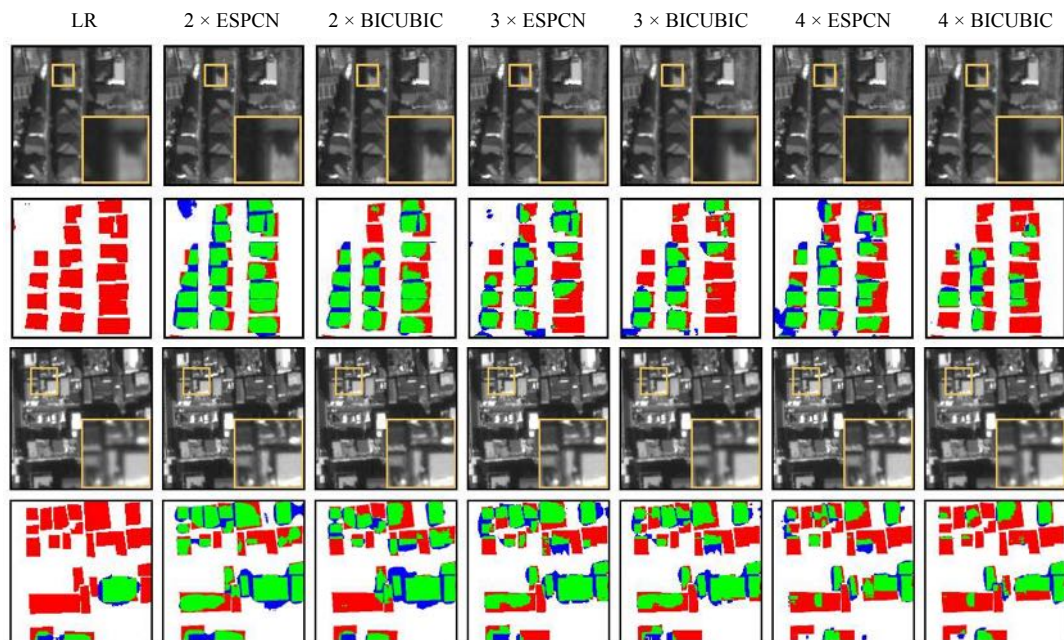


FIGURE 4.4: Qualitative results for representative subregions in test region 1.

Region 2 is a mainly residential area, and the related qualitative results are shown in Figure 4.5. As with the results of region 1, because of the misalignment in resolution between the training and testing data, the majority of small and detached houses in LR imagery are misclassified as non-building areas. Apart from the prevalence of detached houses, the residential buildings in region 2 also include medium-rise mansions and apartments with a comparatively larger distance separating them, and the region also

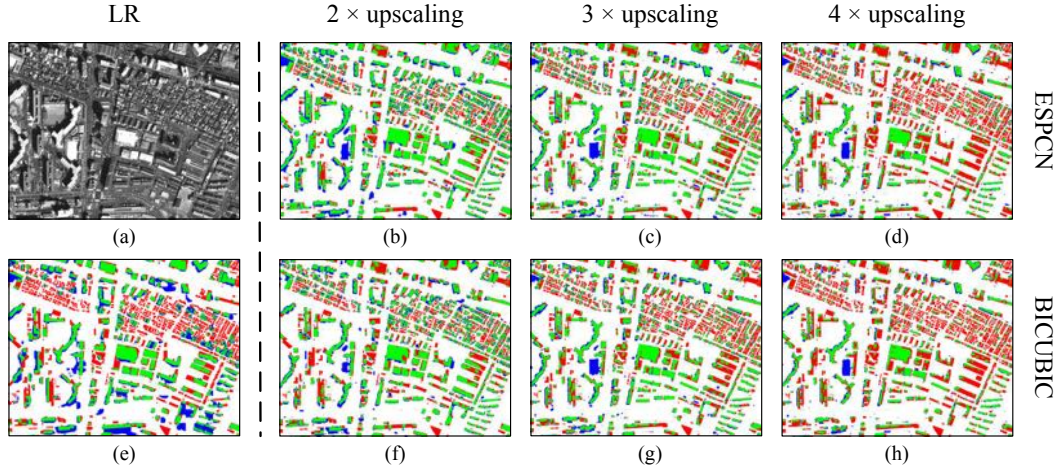


FIGURE 4.5: Qualitative results for test region 2.

TABLE 4.2: Quantitative results for test region 2.

Model	Scale	Resolution	Precision	Recall	Overall Acc	F1-score	Kappa	Jaccard
LR	1	0.500	0.645	0.514	0.787	0.572	0.432	0.400
ESPCN	2	0.250	0.727	0.680	0.840	0.703	0.594	0.542
BICUBIC	2	0.250	0.748	0.577	0.828	0.651	0.540	0.483
ESPCN	3	0.167	0.749	0.560	0.826	0.641	0.529	0.472
BICUBIC	3	0.167	0.756	0.509	0.818	0.608	0.495	0.437
ESPCN	4	0.125	0.751	0.406	0.798	0.527	0.413	0.358
BICUBIC	4	0.125	0.765	0.373	0.794	0.501	0.390	0.335

contains several small parks. Both the qualitative and quantitative results shown in Figure 4.5 and Table 4.2 demonstrate the effect of ESPCN on the semantic segmentation of residential buildings in the different categories.

Figure 4.6 shows some representative mansions and apartments as well as related segmentation details. Except for a few tiny accessory buildings and protruding architectural contours, buildings are correctly segmented with a low fp value by adopting ESPCN with an upscale factor of 2. In contrast, the use of LR imagery leads to the misclassification of many roads and areas containing vegetation as buildings, whereas buildings are incorrectly detected.

As shown in Figure 4.7a, region 3 mainly consists of quasi-industrial zones occupied by light industrial and service facilities, with non-building land features such as a river and large-scale transport system also included. Intuitively, the qualitative results seem to suggest that ESPCN with an upscale factor of 2 outperformed LR and the other methods with fewer fp and fn results, especially in areas bordering the railway line and high-density building areas. Some representative results are presented in Figure 4.8.

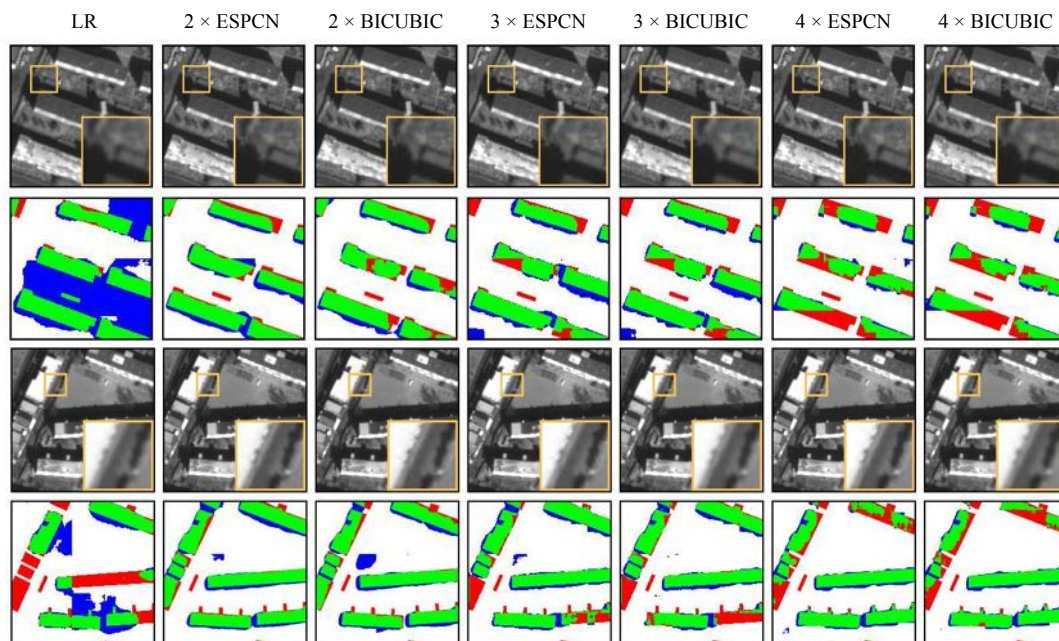


FIGURE 4.6: Qualitative results for representative subregions in test region 2.

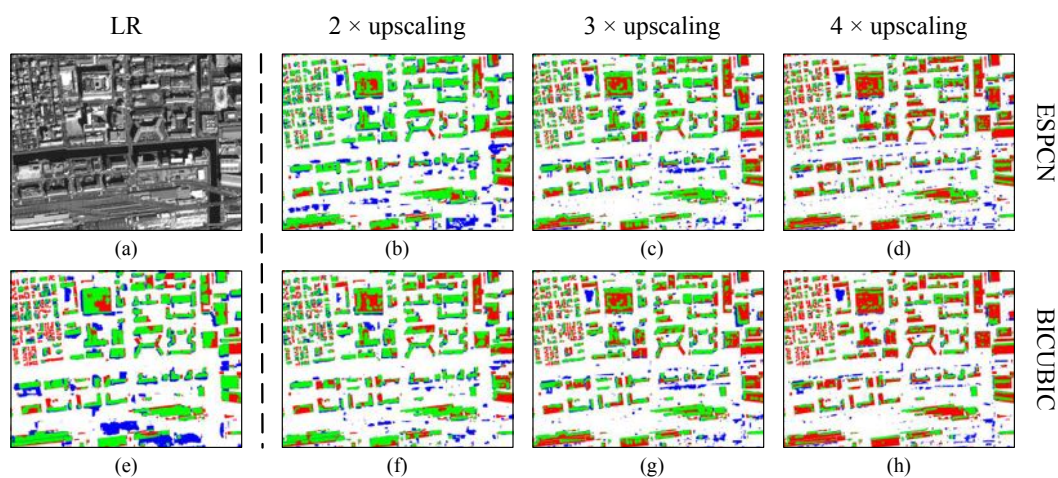


FIGURE 4.7: Qualitative results for test region 3.

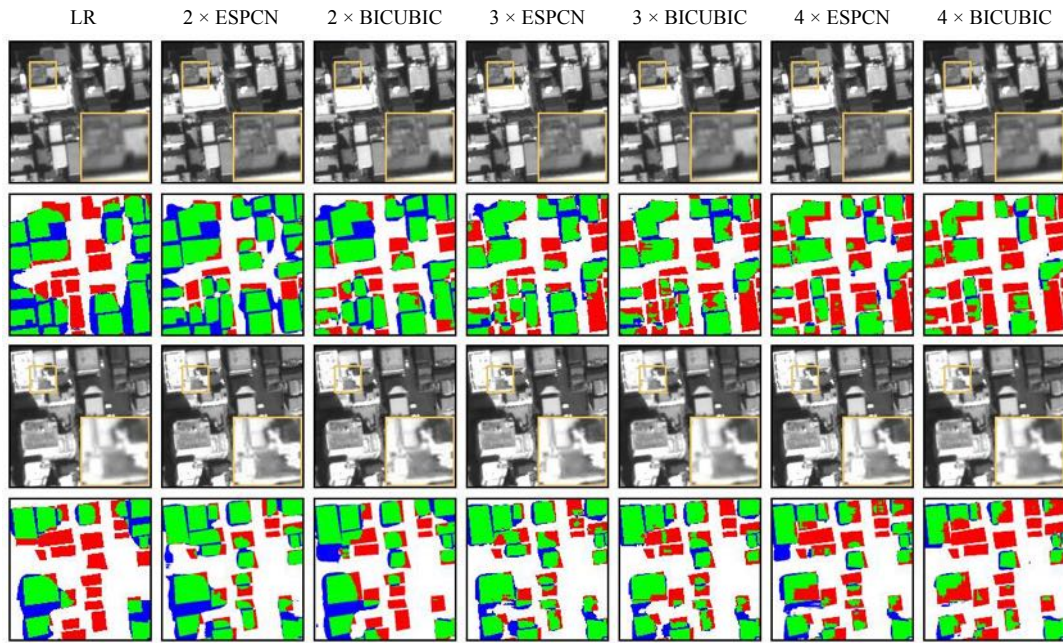


FIGURE 4.8: Qualitative results for representative subregions in test region 3.

TABLE 4.3: Quantitative results for test region 3.

Model	Scale	Resolution	Precision	Recall	Overall Acc	F1-score	Kappa	Jaccard
LR	1	0.500	0.689	0.699	0.827	0.694	0.573	0.531
ESPCN	2	0.250	0.715	0.749	0.846	0.732	0.624	0.577
BICUBIC	2	0.250	0.769	0.640	0.845	0.699	0.595	0.537
ESPCN	3	0.167	0.741	0.609	0.830	0.668	0.556	0.502
BICUBIC	3	0.167	0.747	0.541	0.819	0.627	0.512	0.457
ESPCN	4	0.125	0.712	0.447	0.794	0.550	0.425	0.379
BICUBIC	4	0.125	0.731	0.380	0.786	0.500	0.381	0.333

The quantitative results in Table 4.3 also infer that SR imagery obtained with an appropriate upscale factor can achieve performance superior to that attainable with LR imagery in regions with comprehensive land features.

Region 4 shown in Figure 4.9a is situated in the vicinity of the Tokyo Bay estuary. This highly particular location consists of industrial areas with large factories and storage buildings as well as docks spread over the entire region. Land features particular to this location, such as containers, are widely distributed in the port, while barges are moored in the harbor. The quantitative results shown in Figure 4.9b to h and indicate that ESPCN with an upscale factor of 2 can segment large buildings with the lowest fn .

Qualitative results for test region 4.

The impact of the resolution on the segmentation of large buildings was analyzed in greater detail by selecting a few representative large buildings with a simple roof texture

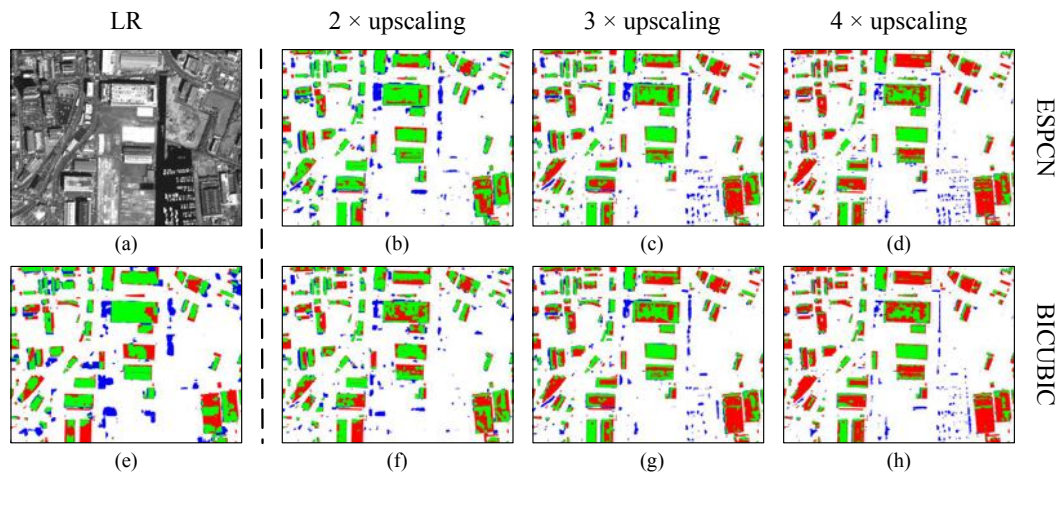


FIGURE 4.9: Qualitative results for test region 4.

and that are surrounded by wide open areas for comparison purposes. As shown in Figure 4.10, although the size of large buildings in LR imagery is comparable with that of small buildings in the training HR imagery, the unclear contour of buildings in LR imagery is prone to misclassification and produces results with a large fn value. Such results reflect the importance of aligning the resolution between training and testing data from the side, as well as the effects of SR integrated method on semantic segmentation in satellite imagery.

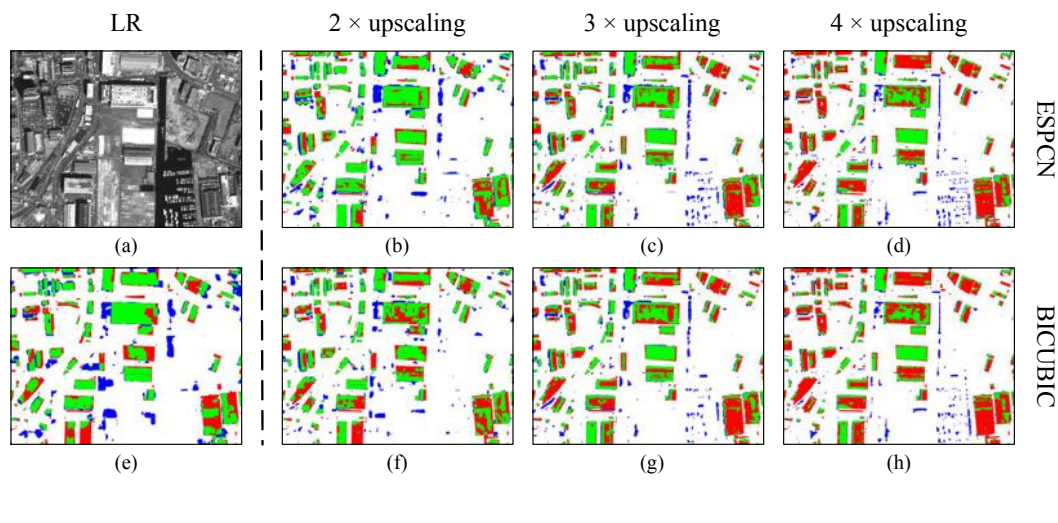


FIGURE 4.10: Qualitative results for representative subregions in test region 4.

The detailed results provided in Table 4.4 confirm the aforementioned conclusion. Large buildings in LR imagery can be detected by the model trained on HR imagery with relatively high accuracy; however, in contrast with the ESPCN integrated method, which contains high-frequency information, the performance remains poor.

TABLE 4.4: Quantitative results for test region 4.

Model	Scale	Resolution	Precision	Recall	Overall Acc	F1-score	Kappa	Jaccard
LR	1	0.500	0.711	0.655	0.862	0.682	0.594	0.517
ESPCN	2	0.250	0.766	0.711	0.886	0.738	0.665	0.584
BICUBIC	2	0.250	0.781	0.569	0.867	0.659	0.578	0.491
ESPCN	3	0.167	0.752	0.551	0.858	0.636	0.55	0.466
BICUBIC	3	0.167	0.766	0.500	0.853	0.605	0.520	0.434
ESPCN	4	0.125	0.748	0.422	0.838	0.540	0.450	0.370
BICUBIC	4	0.125	0.766	0.369	0.832	0.498	0.412	0.332

4.4 Discussion

Section 4.3 presented comprehensive qualitative and quantitative results of the segmentation of buildings, which are located in various areas and which differ in terms of their density, shape, texture, size, and usage. The discussion we provide in this section aims to further demonstrate the feasibility of SR-integrated segmentation methods. First, the average quantitative results for the four regions are used to indicate the robustness of the proposed method. Then, we take reconstruction quality as a reference to show the relationship between segmentation and SR. It should be noted that since the HR satellite imagery is not available, we utilize the reconstruction quality generated in training procedure by HR aerial imagery to represent that of SR satellite imagery. Finally, selected qualitative results of important land features other than buildings are shown. Besides, poor results are briefly analyzed and discussed.

The average segmentation performance obtained by adopting different methods is shown in Figure 4.11. In general, the ESPCN-integrated method with an upscale factor of 2 significantly outperforms the other methods including LR imagery, improving the overall accuracy from 0.802 to 0.854, the F1-score from 0.651 to 0.730, kappa from 0.529 to 0.630, and the Jaccard index from 0.484 to 0.576. These quantitative results indicate that the model trained by HR imagery cannot detect small buildings with high accuracy and that increasing the resolution of LR imagery could enlarge the building size by providing more pixels, which would align the size of buildings in the training data to a certain degree. This point of view would also be supported by the results generated via bicubic interpolation with an upscale factor of 2. Apart from increasing the image resolution, compared with simple interpolation, SR-based methods would also reconstruct finer spatial details with higher PSNR, which would yield improved segmentation results for the same upscale factor.

In principle, regarding the alignment of the resolution of HR with that of LR imagery, upscaling the resolution of LR imagery with a factor of 3 to $0.167m$ by ESPCN would match that of HR imagery to a great extent to generate the best segmentation results.

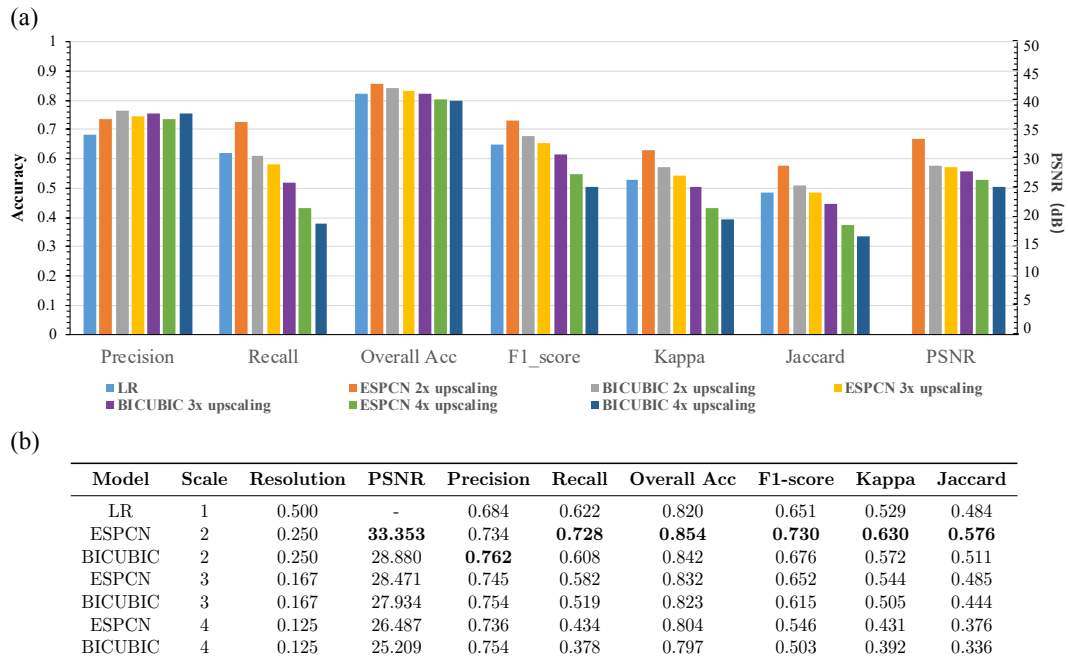


FIGURE 4.11: Average performance of SR reconstruction and segmentation for the four test regions using different methods. (a) Bar diagram for performance comparison. The x- and y-axis represent the assessment criteria and corresponding values, respectively. (b) Table for performance comparison. For each assessment criterion, the highest values are highlighted in **bold**.

However, because of the ill-posed problem, reconstructing high-quality SR imagery from LR space with a large upscale factor would be highly challenging. According to the IQA criteria shown in Figure 4.11b, an increase in the upscale factor from 2 to 4 causes the PSNR of ESPCN-based SR imagery to drastically decline from 33.353 to 28.471 to 26.487. Although the resolution could match that of the training data, the reconstruction quality also severely impacts the correct representation of high-frequency information. Thus, low-quality SR imagery even with an appropriate resolution would worsen the segmentation performance. Ultimately, maintaining a balance between resolution and reconstruction quality is of great importance.

Figure 4.12 shows the semantic segmentation results of other representative land features. The first row shows a parking lot on which cars of different categories are distributed. As shown in the enlarged yellow window, by adopting SR, the shape and textural information of each car becomes much more refined. Because cars are common land features in HR aerial imagery, abundant training data for cars would enable the value of fp to be effectively decreased. In contrast, as shown in the third and fourth rows, the barges moored in the harbor are likely to be misclassified as buildings after the resolution is upscaled. This problem is caused by insufficient training samples for boats

in HR aerial imagery. In terms of railways, trains and tracks are presented by a simple stripe-like feature, and increasing the resolution would enlarge the distance between adjacent strips to improve the performance. Finally, as shown in the last two rows, some polygon-like land features with simple textures such as playgrounds are prone to be misclassified as buildings at a different resolution, indicating that the problem is caused by the UNet model rather than the proposed SR integrated method.

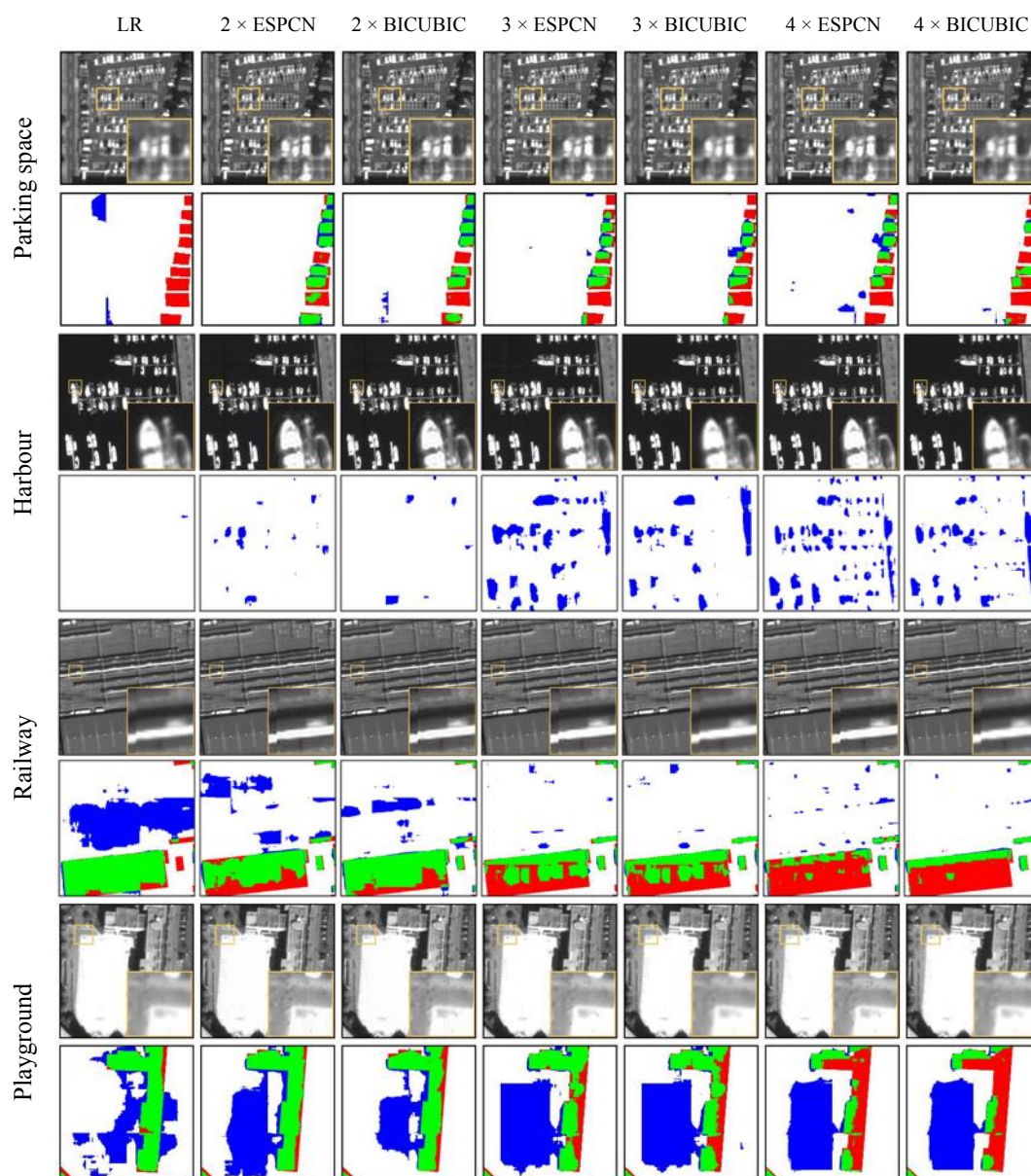


FIGURE 4.12: Results for important land features.

Especially, Figure 4.13 shows a large region in which all methods misclassify non-building areas; after carefully analyzing the original LR image, we believe that the problem is caused by an imperfect ground truth. The large building, which could be well segmented

by adopting ESPCN with an upscale factor of 2, further demonstrates the feasibility of the proposed method.

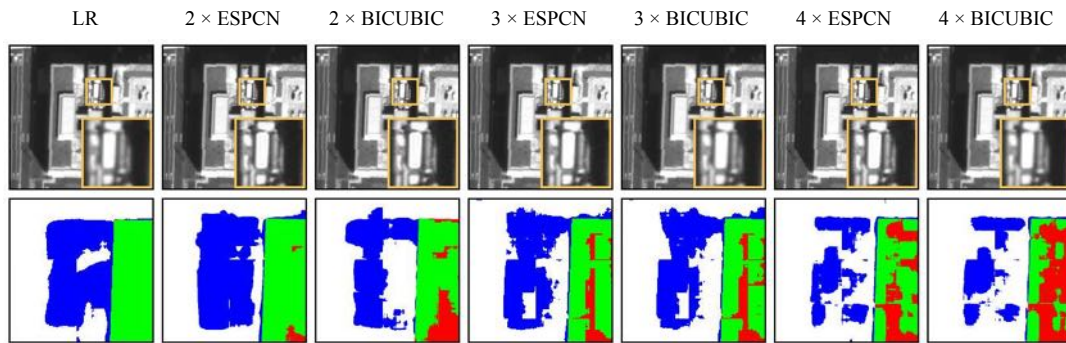


FIGURE 4.13: Bad results caused by annotation.

It should be noted that the investigation of the feasibility of the SR-integrated method for processing multi-source remote sensing imagery is difficult because these images differ in terms of data acquisition methods, resolution, and color space. However, the testing results confirm that the accuracy and robustness of the proposed SR-integrated method is considerably higher than those of the other methods, and that it can achieve comparably accurate building semantic segmentation results using the provided study materials.

4.5 Conclusions

In this chapter, we presented a novel SR-integrated method for building semantic segmentation of multi-source remote sensing imagery of different resolution. The experimental results demonstrate the potential and the capability of the proposed method to solve the problem caused by the resolution of the training data being unaligned with that of the testing data. In particular, the proposed SR-integrated method could achieve considerably higher accuracy and more precise segmentation results than the other methods, which also indicates the feasibility of our proposed method. In addition, it is important to carefully consider the color influence on multi-source remote sensing imagery, investigate the method of balancing resolution and reconstruction quality to enhance the segmentation to a maximum extent, optimize the robustness of both segmentation and SR models, and explore the effectiveness of proposed method in other study areas with buildings in different types, which we aim to study in future.

Chapter 5

Practical Application

In this chapter, we expand the proposed methods introduced in Chapter 2, 3, 4 to more challenging applications including change detection, slum mapping. As for change detection, color normalization, super-resolution, and image registration methods are adopted to balance the training and testing datasets, after that, by adopting proposed CFPN model and image difference, the identification of land change can be achieved. In terms of slum mapping, here CFPN is adopted to perform multi-class semantic segmentation, the impact of resolution on slum segmentation is discussed as well.

5.1 Change Detection

Change detection is one of the most significant tasks in urban planning and urban monitoring. Instead of the traditional time-consuming fieldwork and survey, the detection from remote sensing imagery by deep learning methods [193] has merged as an effective measure. Given the different characteristics, here mainly refer to color space and resolution among different data source, training a general model that can be transferred to all testing conditions is difficult.

In this preliminary study, we investigate the change detection based on multi-source remote sensing imagery with different resolution and color space. To balance resolution and color, super-resolution and color normalization methods, here refer to ESPCN and histogram equalization, are integrated to previous deep learning framework. The preprocessing of training and testing data as shown in Table 5.3.

The aerial RGB training data with resolution 0.16m, is preprocessed and trained in four method as follows;

TABLE 5.1: Training and testing dataset.

Data	Name	Band	Color	Normalize	Resolution	Year	Source
Training	Unnorm_3C	3	RGB	No	0.16m	2016	Aerial Image
	Norm_3G	3	GRAY	Yes			
	Unnorm_1G	1	GRAY	No			
	Norm_1G	1	GRAY	Yes			
Testing	NonSR	-	GRAY	No	0.50m	2016, 2017	Satellite Image
	SR	-	GRAY	Yes	0.16m		

- Unnorm_3C: original three-band RGB color imagery without color normalization;
- Norm_3G: convert original three-band RGB imagery into one-band grayscale imagery with color normalization, then stack three same grayscale images into three-band grayscale one;
- Unnorm_1G: convert original three-band RGB imagery into one-band grayscale imagery without color normalization;
- Norm_1G: convert original three-band RGB imagery into one-band grayscale imagery with color normalization.

The one-band satellite testing data with resolution 0.50m, is preprocessed and tested in two methods as follows:

- NonSR: without super-resolution; the normalization method and band number match that with paired training dataset
- SR: without super-resolution into 0.16m; the normalization method and band number match that with paired training dataset

Figure 5.1 and Figure 5.2 show the building semantic segmentation results of 2016 and 2017 by applying different training and testing methods, respectively.

Although the ground-truth is unavailable, by visual interpretation, we find that the segmentation results generated via method Norm_3G_SR can achieve best results in both 2016 and 2017. The selected results and comparison of 2016 and 2017 can be found in Figure 5.3.

Given the misalignment of satellite imagery taken from different time is inevitable, we usually observe 20-150 pixels shift in a processed pair depending on the scene geometry. To address this issue, we employ projective transformation to wrap the 2017 image based on scale invariant feature transform (SIFT) features [194] of corresponding paired 2016 image. After image registration, the morphological transformations is further adopted to mitigate noise. Such post-processing can be found in 5.4, where gray means unchanged areas; white is new building areas; black refers to new vacant areas.

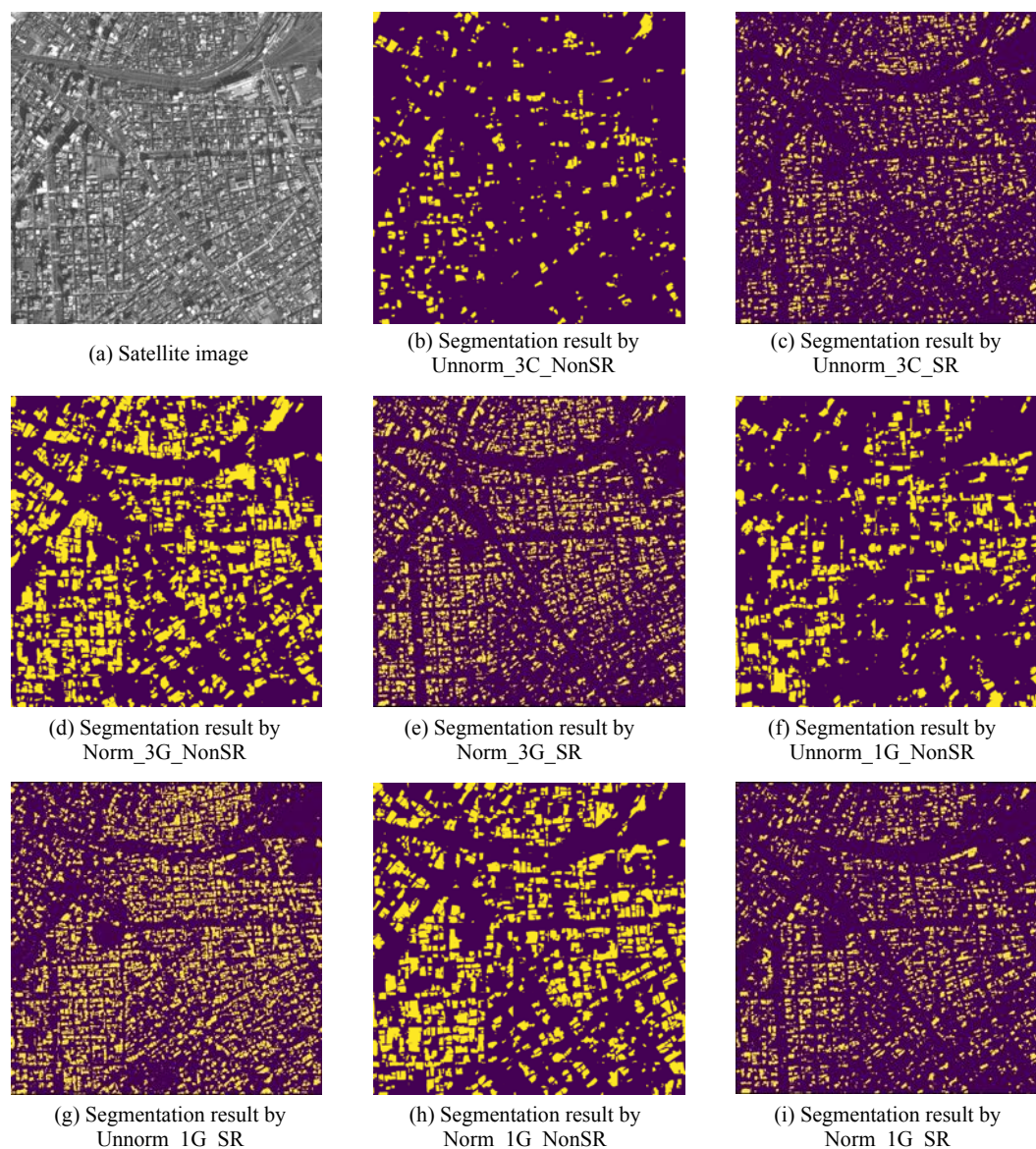


FIGURE 5.1: Semantic segmentation results comparison of 2016. Norm refers to the pre-processing with color normalization; integer means the number of image band; C refers to color image; G means grayscale image; SR is super-resolution integrated.

Figure 5.5 and 5.6 illustrate some new building and new vacant examples generated by proposed method. It reveals that the proposed method can serve as a viable tool for change detection high efficiency. Moreover, the deep learning based end-to-end change detection method can be investigated in case the ground-truth of changed areas is provided.

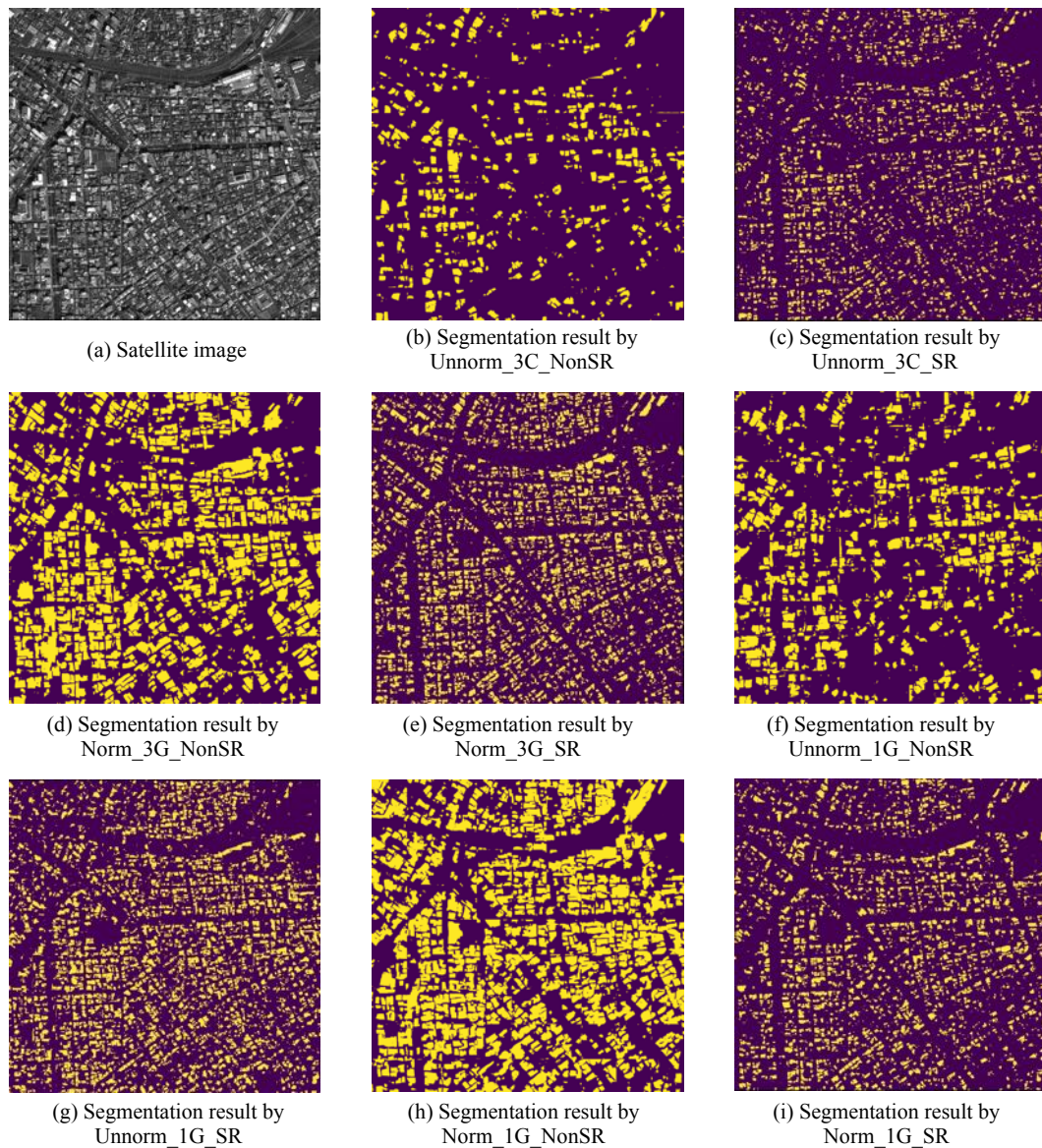


FIGURE 5.2: Semantic segmentation results comparison of 2017. Norm refers to the pre-processing with color normalization; integer means the number of image band; C refers to color image; G means grayscale image; SR is super-resolution integrated.

5.2 Slum Mapping

The management of slum areas is an important task for improving the sanitary condition, humanitarian, and living standard, as well as reducing crime and poverty in developing countries. Since the characteristics of slum areas are very complicated, such as extremely high density, disordered land use, diverse architecture and shanty structures, non-uniform patterns and styles, etc, the traditional methods which measure the

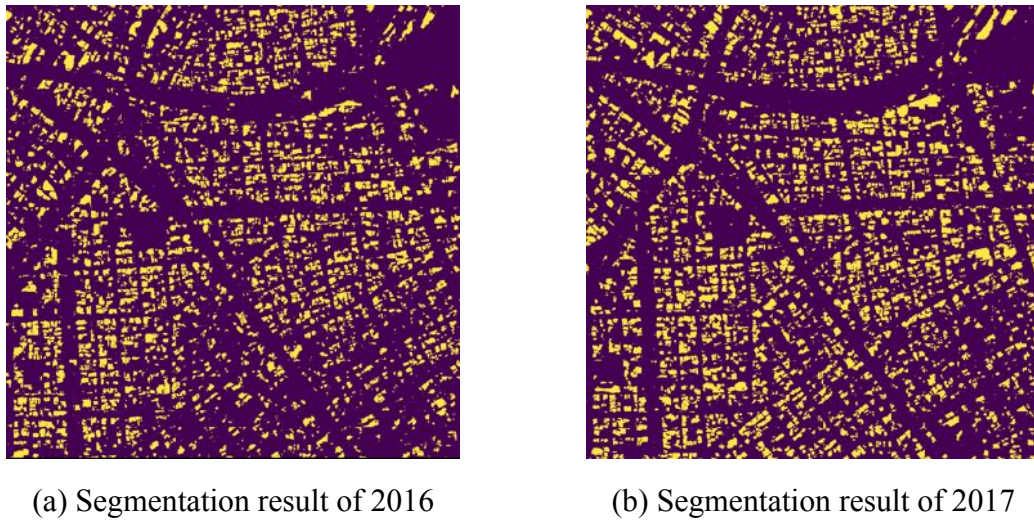


FIGURE 5.3: Selected semantic segmentation results in 2016 and 2017 with the method of Norm_3G_SR.

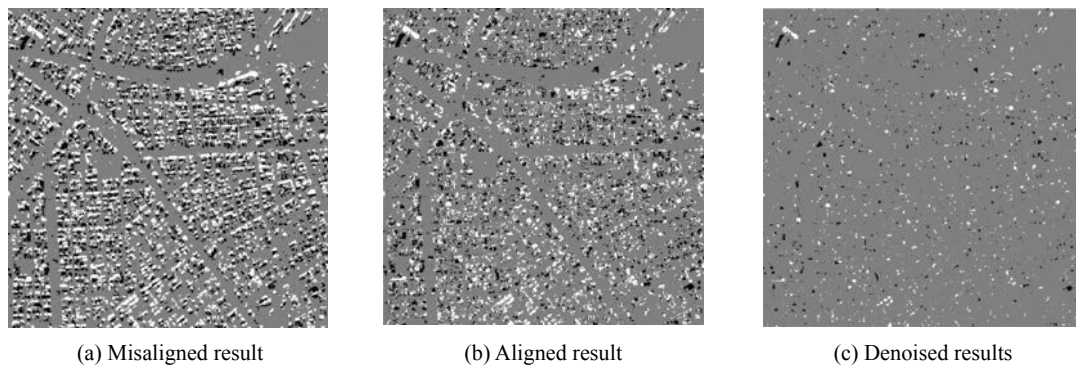


FIGURE 5.4: Change detection with post-processing. Gray means unchanged areas; white is new building areas; black refers to new vacant areas.

scale as well as localize the concrete region of slum areas based on census and community survey inevitably hinder the development to a considerable extent.

In this study, we investigate the way of slum mapping based on remote sensing imagery and deep learning frameworks. Thanks to the annotation data provided by Cheng et al. [195], in which the physical environment of a city was categorized by a two-dimension approach: diversity of buildings and street pattern (aggregate) in vertical and horizontal axis, respectively. The related RGB remote sensing dataset is downloaded from google maps.

Here, we take the annotated ground-truth in Guangzhou and related satellite imagery with resolution of 5m to train the deep learning model, and the workflow is as with

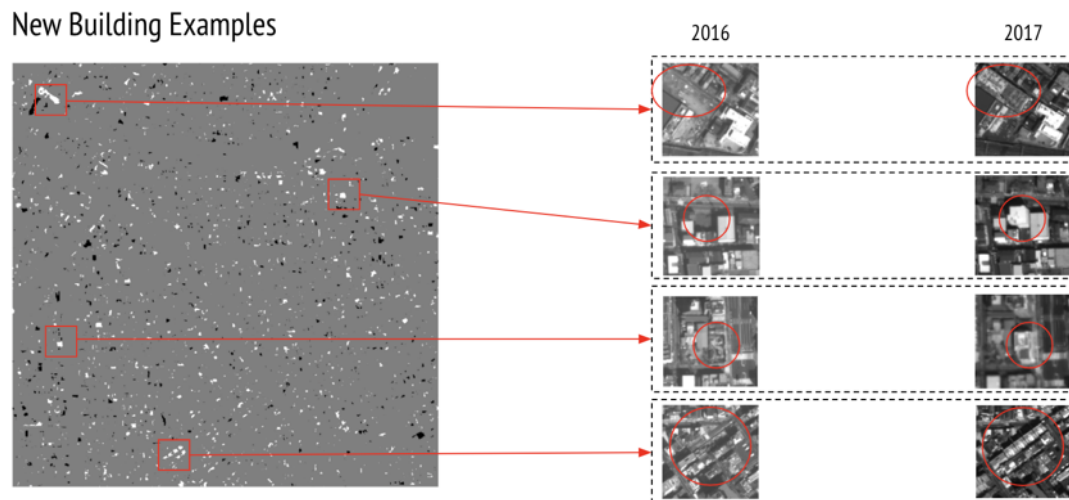


FIGURE 5.5: New-building examples.

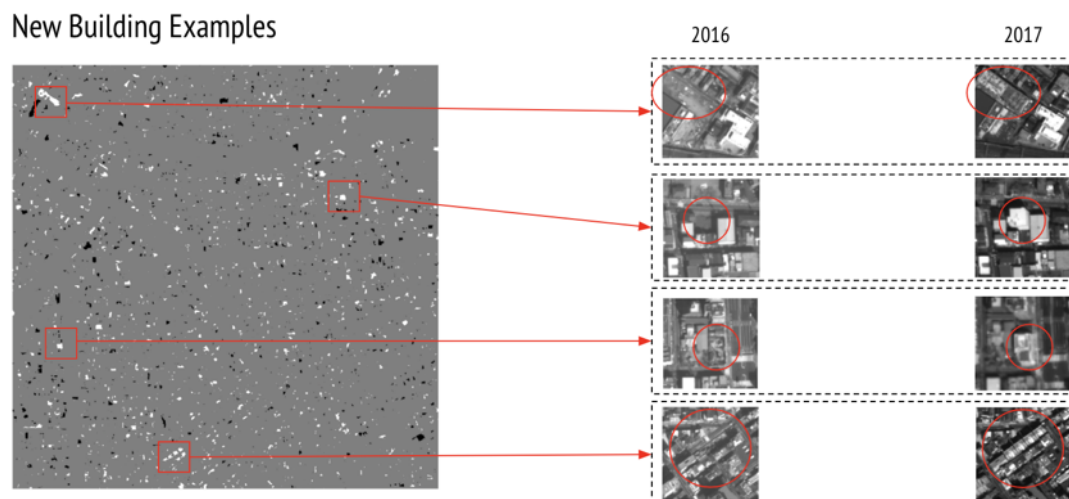


FIGURE 5.6: New-vacant examples.

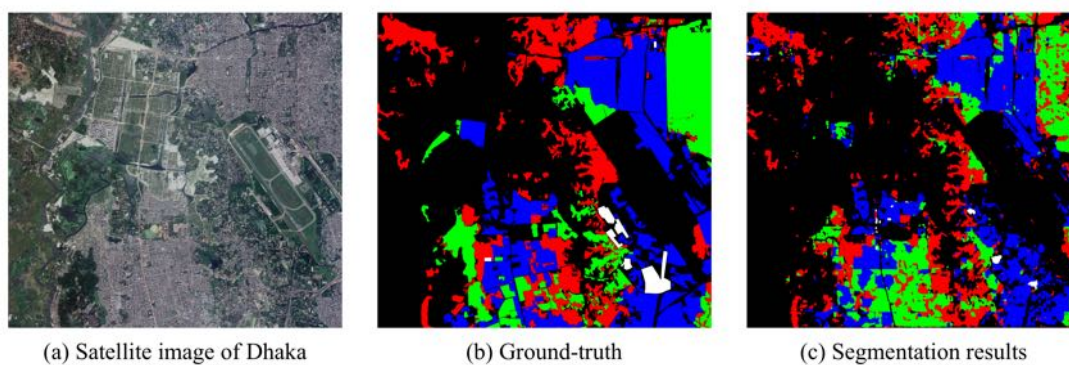


FIGURE 5.7: Slum mapping result example in Dhaka.

TABLE 5.2: The quantitative slum mapping results of Dhaka.

Recall	Precision	Overall Acc	Kappa	IoU	F1-score
0.878	0.879	0.878	0.624	0.814	0.874

Chapter 3. After that, the trained model is utilized to conduct semantic segmentation in Dhaka, Bangladesh. The corresponding qualitative and quantitative results are shown in 5.7 and 5.2, respectively. Since the exterior characteristics of slum differ in different countries, red regions in 5.7 mainly refer to the high degree of informality.

The impact of resolution on building semantic segmentation has been discussed in Chapter 4, to further investigate the effectiveness of the resolution on other land features and data source, we conduct an experiment by training models with resolution from 1m to 5m using slum dataset mentioned above. The qualitative and quantitative results from Figure 5.8 and 5.3 reveal that the model with resolution in 3m outperform others. The quality of annotation dataset and the demand for high-frequency feature may lead such results; some researches [196, 197] which focus on investigating the impact of resolution on vegetation and land use semengtation also obtained alike results. The detailed investigation study will be part of the future works.

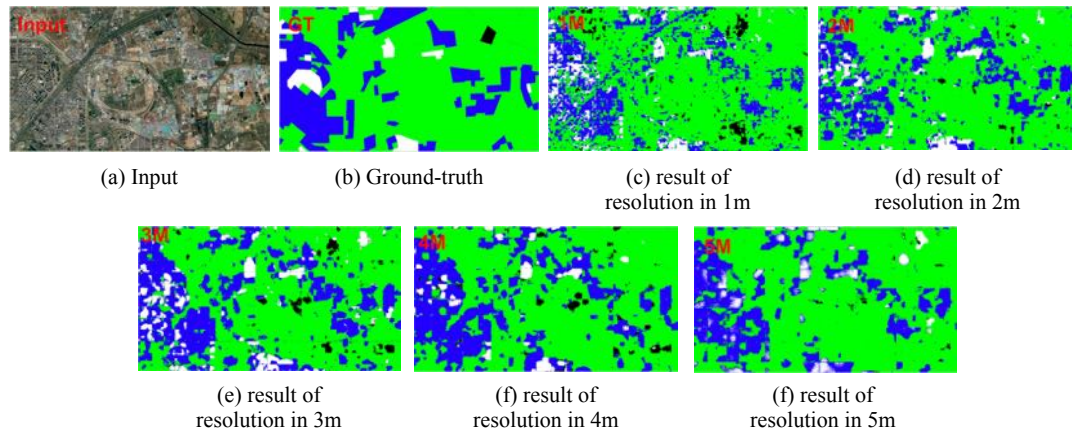


FIGURE 5.8: Slum mapping results in different resolution.

TABLE 5.3: The impact of resolution on slum mapping.

Resolution	Recall	Precision	Overall Acc	Kappa	IoU	F1-score
1m	0.499	0.648	0.499	0.245	0.347	0.471
2m	0.522	0.703	0.522	0.300	0.373	0.500
3m	0.556	0.724	0.556	0.348	0.410	0.554
4m	0.521	0.644	0.521	0.277	0.367	0.493
5m	0.526	0.684	0.526	0.296	0.371	0.497

Chapter 6

Conclusions and Future Works

6.1 Conclusions

In this dissertation, we creatively investigated the feasibility of applying deep learning methods in different semantic segmentation tasks via multi-source remote sensing imagery. The comprehensive researches including village mapping, urban building segmentation, slum mapping, super-resolution integrated method for model transfer, change detection, etc., are conducted. In village mapping study, we explored how to construct CNN architecture that can adapt to the village building identification task and presented a novel CNN frame called ECNN based on multiscale feature learning by emsembling parallel optimized state-of-the-art CNN models. The model outperformed others with high accuracy. And in urban building semantic segmentation, we presented a FCN based model named CFPN. The proposed model is further applied in tasks including slum mapping and change detection. The experimental results demonstrate the proposed model outperforms the existing state-of-the-art methods, and can serve as a viable tool for mapping tasks with high accuracy and efficiency. After that, we discussed the challenge and limitation of recent deep learning based studies on model transfer problems. We innovatively presented a novel SR integrated building semantic segmentation framework to tackle the problem caused by the unaligned resolution between training and testing data, and investigated the feasibility of the proposed method based on comprehensive experiments. Change detection is conducted by integrating deep learning based semantic segmentation framework as well as resolution and color transfer. The experimental results mainly show the feasibility of the proposed method in detecting urban changes. Moreover, to facilitate the development of the deep learning based segmentation and super-resolution models, we developed an open source computer vision package named

as GeoVision, which contains subpackage GeoSeg and GeoSR to perform semantic segmentation and super-resolution, respectively.

6.2 Future Works

Although this study indicates the proposed method could be efficiently used in semantic segmentation for multi-source remote sensing imagery, further and more detailed exploration on the method is required in the future. First, to test the method's stability, more extended and sophisticated areas need to be tested. Second, to further evaluate the feasibility of the proposed change detection method, ground-truth data will be prepared. Third, given the rapid development of deep learning techniques, the proposed methods can not keep the long-term state-of-the-art. With the help of GeoVision introduced in Appendix A, the upgrade of our framework and model will be achieved. Fourth, to enhance the robustness of model in different cases, more comprehensive investigation on model transfer is still essential. Fifth, the limitation of training dataset hindered the segmentation quality in a considerable extent, applying other deep learning techniques like weak-supervised learning and meta learning is important. Finally, To achieve higher quality segmentation in tasks like change detection and slum mapping, adopting remote sensing imagery from more data sources including DSM and multispectral imagery will be investigated.

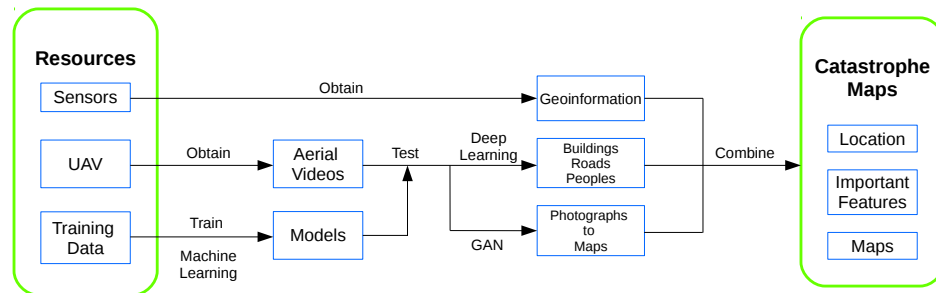


FIGURE 6.1: System for Automatic and Real-time Generalization of Catastrophe maps.

There are still many important applications which can be implemented in the future. Japan is one of the countries most affected by natural disasters; the catastrophes such as earthquake, Tsunami and landslide usually cause enormous losses. As an indispensable resource, maps used to illustrate land conditions and changes after catastrophe are quite significant. In this research plan, a system for automatic and real-time generalization of catastrophe maps is proposed. Rather than existing methodologies, which highly depend on human beings, here, by considering the characteristics and importance of catastrophe

maps, we propose a brand new map generalization system based on combining GIS and machine learning methods, which would be capable to automatically provide accurate, efficient and time-sequenced catastrophe maps. As shown in Figure 6.1, By implementing satellite imagery, machine learning methods and geographic sensors, the important land features such as safe roads, broken buildings can be identified; meanwhile, the digitalized map with accurate geographic coordinates can be generated as well. Base on the obtained results, life and property would be saved. Furthermore, not only in catastrophe, this system could also be used in many other map generalization conditions.

Moreover, we believe the combination of proposed method with Internet of Things (IoT) can achieve more promising and interesting researches in the future.

Appendix A

Appendix I: GeoVision

A GeoSeg

Automatic, robust, and accurate image segmentation is a long existing challenge in computer vision. Over the past decades, many supervised or unsupervised methods are proposed to handle this task [198, 199]. However, due to the limitations of both the quality of dataset and processing algorithm, the precision level of these methods are quite limited [42]. Recent years, thanks to the rapid development of deep convolutional neural networks (DCNNs) as well as the dramatically increased availability of large-scale datasets, the performances show significant improvement in many image segmentation tasks [200, 201].

Differ to ordinary images, because of cost, technical requirement and sensitivity of national defense, it is rather difficult to get very high-resolution (VHR) aerial imagery in the field of remote sensing. And, the lack of large-scale, high-resolution dataset limits the development of accurate building segmentation and outline extraction. Recently, due to rapid evolution of imaging sensors, the availability and accessibility of high-quality remote sensing datasets have increased dramatically [202, 203]. On the basis of these datasets, many well-optimized and innovative methods, including different variants of fully convolutional networks(FCNs), have been developed for the purpose of accurate building segmentation [204]. Generally, these methods achieve the state-of-the-art accuracy or computational efficiency under corresponding datasets. However, since these methods are trained and evaluated through different datasets, it is hard to have an in-depth comparison of performances of various models. Additionally, although the datasets are open-access, the implemented models or algorithms are usually not revealed in details by the authors.

Facing this problem, we introduce Geoseg (<https://github.com/huster-wgm/geoseg>), a computer vision package that is focus on implementing the state-of-the-art methods for automatic and accurate building segmentation and outline extraction. The Geoseg package implements more than 9 FCN-based models including FCNs [37], U-Net [43], SegNet [44], FPN [149], ResUNet [205], MC-FCN [8], and BR-Net [9]. For in-depth comparison, balanced and unbalanced evaluation metrics, such as precision, recall, overall accuracy, f1-score, Jaccard index or intersection over union (IoU) [206] and kappa coefficient [207], are implemented.

The main contributions of this study are summarized as follows:

- We build a computer vision package that implemented several state-of-the-art methods (i.e., BR-Net) for building segmentation and outline extraction of very high-resolution aerial imagery;
- We have carefully trained and evaluated different models using the same dataset to produce a performance benchmark of various models.
- The package is optimized and opened to the public that other researchers or developers can easily adopt for their own researches.

The rest of the study is organized as follows: the related work is presented in Section ???. The benchmark dataset and implementation details of the experiments are described in Section 2. In Section 3, the results and discussion of different models are introduced. Conclusions regarding our study are presented in Sections 4, respectively.

1 Related work

To assist deep learning researches or applications, there are several deep learning frameworks. According to the compiling mechanism, these frameworks can be categorized into two groups: static and dynamic framework. Static frameworks, such as Caffe¹ and TensorFlow², construct and compiled completed model before training and updating parameters. For dynamic frameworks, such as Chainer³ and PyTorch⁴, at every iteration, only executed part of the model is compiled. Compared with static frameworks, the dynamic frameworks are less efficient but much flexible.

For different frameworks, there are "Model Zoo" packages that implemented with various pre-trained deep learning models. However, most of the implemented models are focused

¹<http://caffe.berkeleyvision.org/>

²<https://www.tensorflow.org/>

³<https://chainer.org/>

⁴<https://pytorch.org/>

on methods for image classification. Even for image segmentation packages such as ChainerCV⁵, the implemented methods are quite limited, and datasets are not relevant to aerial imagery.

As far as we know, Geoseg is the first computer vision package that implemented with abundant deep learning models for automatic building segmentation and outline extraction.

2 Experiments

2.1 Benchmark Dataset

Thanks to the trend of open source, more and more high-quality aerial imagery datasets are available. Among them, a very high-resolution (VHR) aerial image dataset called Aerial Imagery for Roof Segmentation (AIRS) (<https://www.airs-dataset.com/>) is published most recently [208]. The spatial resolution of the dataset reaches 0.075 cm. The original orthophotos and corresponding building outlines are provided by Land Information of New Zealand (LINZ). For the purpose of accurate roof segmentation, the vectorized building outlines are carefully adjusted to ensure that all building polygons are strictly aligned with their corresponding roofs. The similar dataset with the resolution in 0.16m is provided as well.

To have a fair comparison of different methods, a study area of AIRS that covers 32 km² in Christchurch is chosen [9]. The study area is evenly divided into two regions: training and testing. For each area, there are 28,786 and 26,747 building objects, respectively. Before experiments, both regions are processed by a sliding window of 224 × 224 pixels to generate image slices (without overlap). After filter out image slices with low building coverage rates from training region, the number of samples in training, validation, and testing data are 27,912 11,952 and 71,688, respectively.

2.2 Implementation

Code Organization Geoseg is built on top of PyTorch with version == 0.3.0 (updating to the latest version is scheduled). The whole package is organized as Figure A.1. There are 5 sub-directories including dataset/, logs/, models/, result/ and utils/. The dataset/ directory contains all samples for training, validating and testing. The logs/ directory records learning curves, training and validating performance during model iterations. The models/ directory contains scripts implemented with various network

⁵<https://github.com/chainer/chainercv>

architectures of the models. The visualization results are saved in `result/` directory. The `utils/` directory implements scripts for handling dataset, running instruction, evaluation metrics and visualization tools.

For scripts (e.g., `FCNs.py`, `FPN.py`, and `UNet.py`) at root directory of Geoseg, demo codes for training, logging and evaluating specific models are presented.

For scripts starting with "vis" (e.g., `visSingle.py` and `visSingleComparison.py`), demo codes for result visualization of a single model or various models comparison are implemented.

Models In Geoseg, we implemented over 9 FCN-based models according to the reports from original papers. Since the original methods were implemented in various platform and used for various sizes of input, Geoseg introduces few modifications on several models for unification. The details of the implemented models are listed as follows:

1. FCNs. The classic FCNs method is proposed by Long et al. at 2015. This method innovatively adopts sequential convolutional operations and bilinear upsampling to performance pixel-to-pixel translation. According to fusion and upsampling level of different intermediate layers, the FCNs methods have three variants: FCN32s, FCN16s, and FCN8s.
2. U-Net. The U-Net method is proposed by Ronneberger et al. at 2015. This method adopts multiple skip connections between upper and downer layers.
3. FPN. The FPN method is published on *CVPR2017*. Similar to U-Net, this method adopts multiple skip connections. Besides, the FPN model generates multi-scale predictions for final output.
4. SegNet. The SegNet method is proposed by Badrinarayanan et al. at 2017. As compared with FCNs, SegNet adopts unpooling which utilizes pooling index of corresponding max-pooling operation to perform upsampling.
5. ResUNet. The ResUNet method adopts the basic structure of U-Net and replaces the convolutional block of VGG-16 [86] with Residual block [121]. This architecture enhances the representation ability of the model and gains better model performance.
6. MC-FCN. The MC-FCN method is proposed by Wu et al. at 2018. The MC-FCN adopts the U-Net as backend and introduces multi-constraints of corresponding outputs.

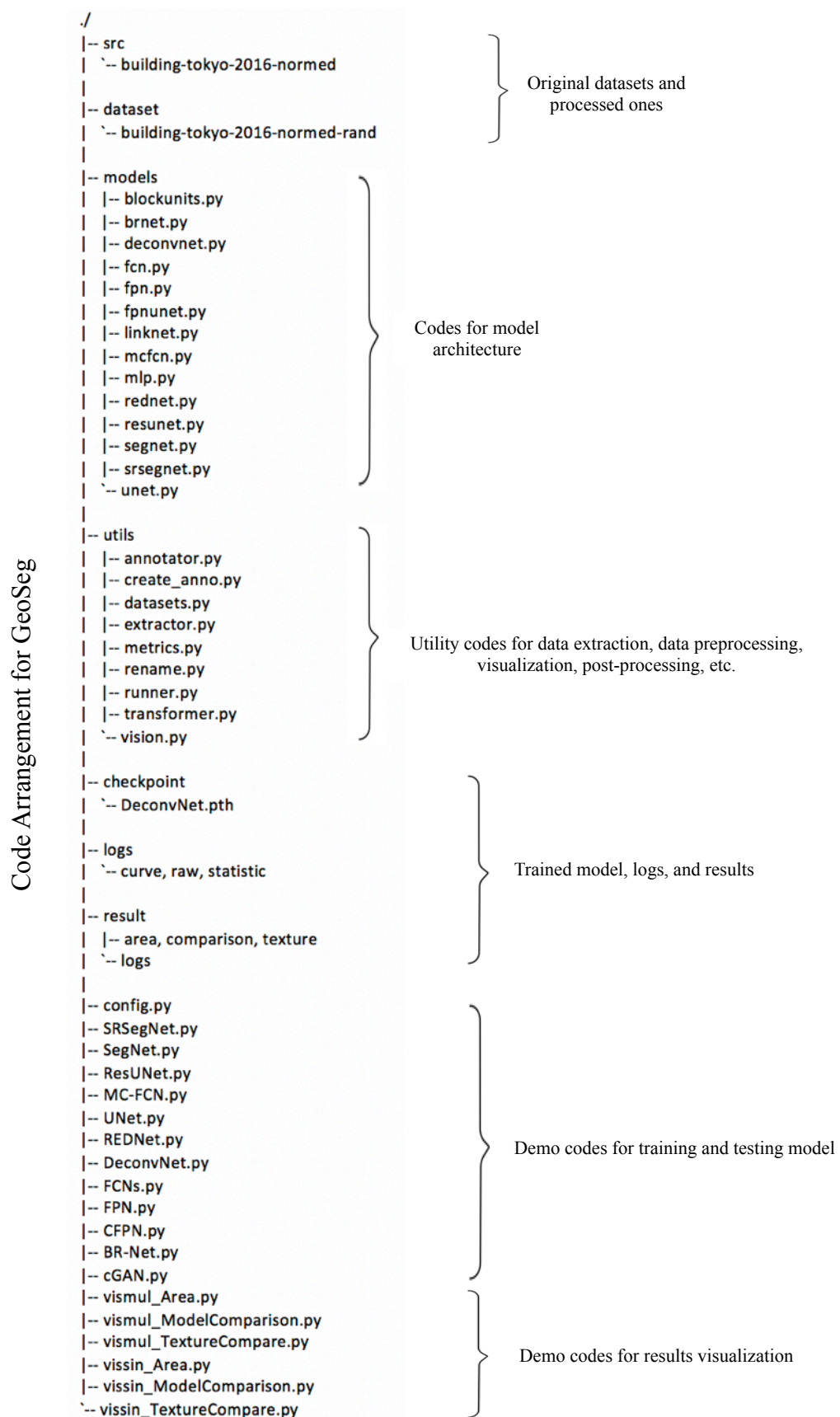


FIGURE A.1: The code organization of Geoseg package. The package implements model constructing, training, logging, evaluating and result visualization modules.

7. BR-Net. The BR-Net method is published by *Remote Sensing* at 2018. The method utilizes a modified U-Net, which replaces traditional ReLU with LeakyReLU (with $\alpha = 0.1$), as shared backend. Besides, extra boundary loss is proposed to regulate the model.

Because of the effectiveness of batch normalization (BN) [128], advanced models, including FPN, SegNet, ResUNet, MC-FCN, and BR-Net, heavily adopt BN layers after each convolutional operations to increase training speed and prevent bias.

3 Results and Discussion

Three FCN variants (FCN8s, FCN18s, and FCN32s), SegNet, U-Net, FPN, ResUNet, MC-FCN, and BR-Net model are adopted as baseline models for comparisons. These models are trained and evaluated utilizing the same dataset and processing platform.

3.1 Qualitative Result

Figure A.2 presents six groups of randomly selected visualization results generated by FPN. From top to bottom rows, there are original images, extracted edges by Canny, building segmentation and outline extraction from FPN model. In general, the extracted outlines through Canny detector contains pretty much noise (see 2nd Row). The FPN can segment the major part of buildings from most of the selected RGB images (see 3rd Row). Building outlines extracted from segmentation results show much fewer false negatives (see 2nd Row vs. 4th Row).

3.2 Quantitative Result

For model evaluations, two imbalanced metrics of precision and recall, and four general metrics of overall accuracy, F1 score, Jaccard index, and kappa coefficient are utilized for quantitative evaluations. Figure A.3 presents comparative results between FCN8s, FCN16s, FCN32s, U-Net, FPN, ResUNet, MC-FCN and BR-Net for the testing samples.

For the imbalanced metrics of precision and recall, the BR-Net method achieves the highest value of precision (0.743) which indicates that the method performs well in terms of suppressing false positives. And, the MC-FCN method gains the highest value of recall (0.824) among nine implemented methods.

For the four general metrics, the BR-Net model achieves the highest values for overall accuracy, F1 score, Jaccard index, and kappa coefficient. Compared with the weakest

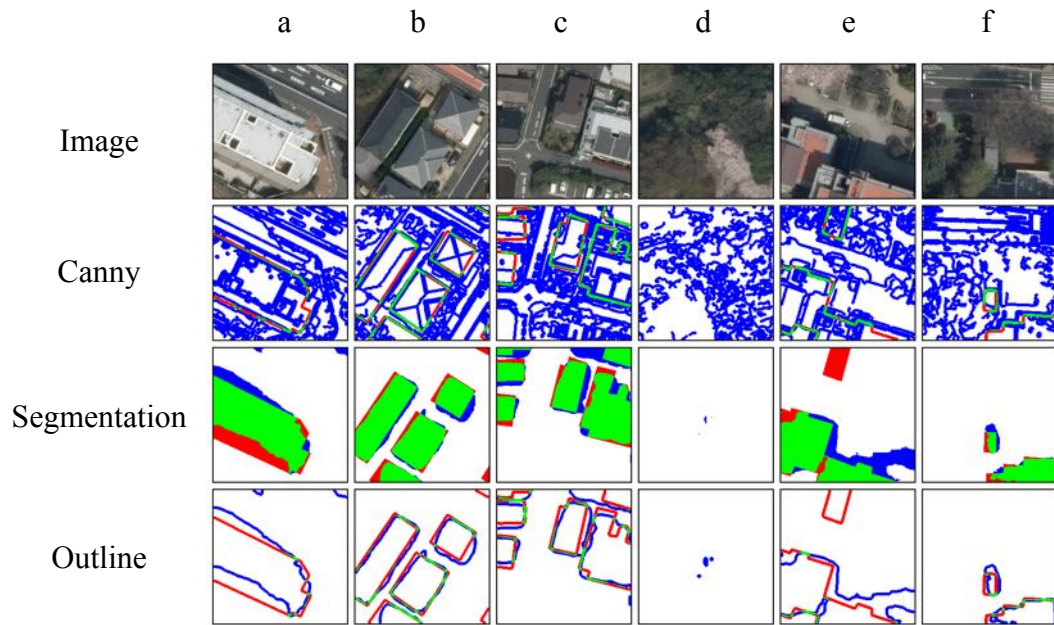


FIGURE A.2: Randomly selected eight samples of visualization result. The green, red, blue, and white channels in the results represent true positive, false positive, false negative, and true negative predictions, respectively.

model (FCN32s), the best model (BR-Net) achieves improvement of approximately 7.2% (0.949 vs. 0.885) on overall accuracy. For F1 score, the best model achieves improvement of about 17.8% (0.766 vs. 0.650) over FCN32s. Compared to the FCN32s method, the BR-Net method achieves improvements of 29.4% (0.686 vs. 0.530) and 25.8% (0.737 vs. 0.586) for Jaccard index and kappa coefficient, respectively. Considering the fact that all these models are proposed within three years, we can imagine the evolution speed within the research field.

3.3 Computational efficiency

The nine models are all implemented in PyTorch and tested on a 64-bit Ubuntu system equipped with an NVIDIA GeForce GTX 1070 GPU ⁶. During iterations, the Adam optimizer [209] with a learning rate of $2e-4$ and betas of (0.9, 0.999) is utilized. To ensure a fair comparison of the different methods, the batch size and iteration number for training are fixed as 24 and 5,000, respectively.

The computational efficiencies of the different methods during different stages are listed in Figure A.4. During the training stage, the slowest models (FCN8s and FCN16s) process approximately 29.2 FPS, while the fastest model (U-Net) reaches 91.3 FPS. Because

⁶<https://www.nvidia.com/en-us/geforce/products/10series/geforce-gtx-1070-ti/>

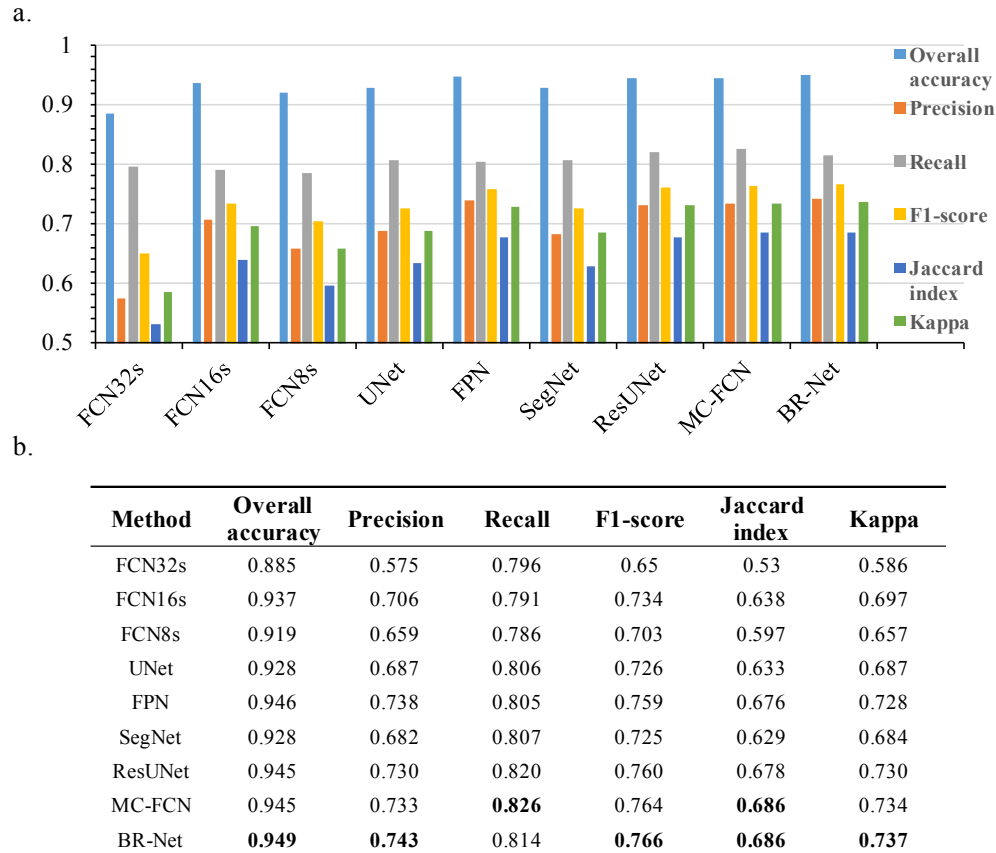


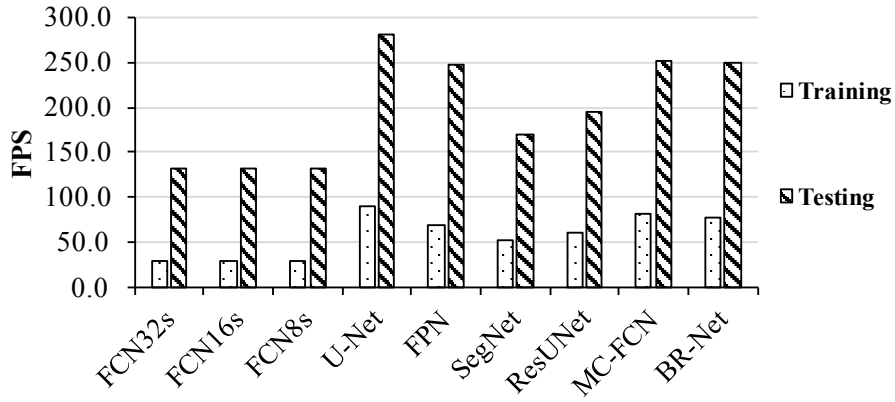
FIGURE A.3: Comparison of segmentation performances of implemented models across the entire testing data. (a) Bar chart for performance comparison. The x- and y-axis represent the implemented methods and corresponding performances, respectively. (b) Table of performance comparisons of methods. For each evaluation metric, the highest values are highlighted in **bold**.

of fewer computational operations, at the testing stage, the slowest model(FCN32) and the fastest model (U-Net) reach 131.6 and 280.4 FPS, respectively.

Even with slight differences in their architectures, three FCNs variants (FCN32s, FCN16s, and FCN8s) show almost identical computational efficiency at both training and testing stages. Consider the huge differences in their performances (see details in Figure A.3 b), it is better to avoid applying FCN32s model.

Compared with U-Net, more complex models such as FPN, ResUNet, MC-FCN and BR-Net adopt extra computation layers that lead to a slightly slower processing speed at both training and testing stages. The SegNet model, which is slower and weaker than U-Net, is also not a good option for robust building segmentation and outline extraction.

a.



b.

Stage	FCN32s	FCN16s	FCN8s	U-Net	FPN	SegNet	ResUNet	MC-FCN	BR-Net
Training (FPS)	29.4	29.2	29.2	91.3	70.1	53.6	61.0	81.8	78.6
Testing (FPS)	131.6	131.4	131.2	280.4	247.0	169.4	195.3	252.0	249.8

FIGURE A.4: Comparison of computational efficiency of the nine implemented methods. (a) Bar chart for computational efficiency comparison. The x- and y-axis represent the implemented methods and corresponding processing speed of frames per second (FPS), respectively. (b) Table of performance comparisons of methods. For each stage, the highest values are highlighted in **bold**.

4 Conclusion

In this paper, we introduce a computer vision package termed Geoseg that focus on accurate building segmentation and outline extraction. The Geoseg is built on top of PyTorch, a dynamic deep learning framework. In Geoseg, we implement nine models as well as utilities for handling dataset, logging, training, evaluating and visualization. Through a large-scale aerial image dataset, we evaluate performances and computational efficiency of implemented models including FCN32s, FCN16s, FCN8s, U-Net, FPN, SegNet, ResUNet, MC-FCN, and BR-Net. In comparison to the weakest model (FCN32s), the best model (BR-Net) achieves increments of 17.8% (0.766 vs. 0.650), 29.4% (0.686 vs. 0.530), and 25.8% (0.737 vs. 0.586) in F1-score, Jaccard index, and kappa coefficient, respectively. In future studies, we will further optimize our network architecture to achieve better performance with less computational cost.

B GeoSR

The single-frame super-resolution (SR) [210], which brings an excellent opportunity to improve a wide range of important remote sensing applications, such as land segmentation, scene classification, and features detection, is one of the most challenging problems in computer vision. Over the past decades, several methods in group of image reconstruction (RE) based [183], image learning (LE) based [211], and hybrid (HY) based [212] have been studied to solve the problem. However, due to the limitations of reconstruction and learning capability, the conventional methods could only show their feasibility under the specific condition. Recently, with the great evolution of deep learning algorithms [38] and increased availability of large-scale datasets, the performance ceiling of single-frame SR has been continuously rising, which enables SR to be a more potential and promising technique in several cutting-edge fields.










Upscale Factor	Low Resolution	BICUBIC / PSNR		Super Resolution / PSNR	
2			27.081		30.486
4			23.723		25.370
8			20.277		22.737

FIGURE A.5: Super-resolution examples generated by GeoSR. The size of the original low-resolution images are upscaled into 2, 4, and 8 times, respectively.

Although deep learning based SR methods could achieve state-of-the-art performance in accuracy or computational efficiency, the different source of datasets, deep learning

platforms, and computational environments in training and evaluation procedures make it hard to have an in-depth comparison of performances for various models.

Facing aforementioned problems, we present GeoSR⁷ (example as shown in figure A.5), an open source computer vision package for deep learning based single-frame remote sensing imagery super-resolution. The GeoSR package implements over 10 deep learning based SR models as the baseline, such as SRCNN [189], ESPCN [71], VDSR [190], DRRN [213], and SRGAN [192]. For in-depth comparison, benchmark datasets, logging tools, evaluation metrics, and visualization tools are implemented as well.

The main contributions of this study can be highlighted as follows. First, to the best of our knowledge, it is the first computer vision package for deep learning based single-frame remote sensing imagery super-resolution. Second, it innovatively integrates well-established data retrieval, model development, visualization, evaluation, and logging tools as a pipeline. Third, several state-of-the-art models, which carefully trained and evaluated using the same dataset, are available in the package to achieve a reliable benchmark. Moreover, with a scalable API, the package could be potentially utilized in other remote sensing tasks, such as image segmentation, classification, and object detection. Last but not least, to facilitate the development of SR community, the proposed package is provided as an open source to the public, which enables researchers and developers to adopt for their researches efficiently.

1 Methodology

1.1 Workflow

GeoSR is built on top of deep learning library PyTorch⁸ with version == 0.4.1. Figure A.6 shows the directory structure of the package. There are 7 sub-directories including `src/`, `dataset/`, `logs/`, `model_zoo/`, `archs/`, `utils/`, and `result/`. Some detailed information of the sub-directory will be introduced in later parts. The scripts (e.g., “SRCNN.py”, “ESPCN.py”, and “VDSR.py”) at the root directory of GeoSR are used for training, logging, and evaluating specific models.

⁷<https://github.com/Chokurei/geosr>

⁸<https://pytorch.org/>

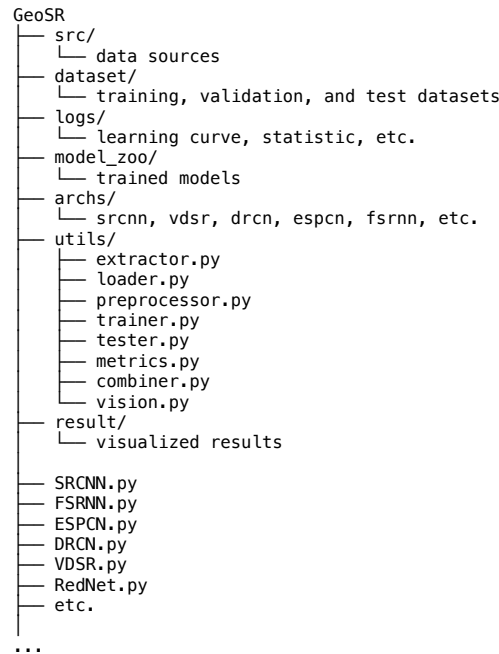


FIGURE A.6: The directory structure and code organization of GeoSR package. The package integrates model construction, training, logging, evaluation, and result visualization modules.

1.2 Data

Source We provide high-quality data sources in remote sensing field like UC-Merced⁹, WHU-RS19¹⁰, NWPU-RESISC45¹¹ in `src/` as the benchmark in GeoSR. The training patches stored in `dataset/` with specific size can be generated from data source in three ways: random sliding, stride sliding, and random allocation, by the code in “`extractor.py`”.

Preprocessing In the SR reconstruction process, different algorithms are designed to test with which color space could obtain better image reconstruction quality. In GeoSR, we currently involve two color spaces: RGB and YCbCr. More coordinate systems are scheduled in future works.

Regarding data augmentation, instead of utilizing given methods in “`preprocessor.py`”, users can self-define suitable one for preprocessing with scalable API.

1.3 Models

⁹<http://weege.vision.ucmerced.edu/datasets/landuse.html>

¹⁰<http://www.xinhua-fluid.com/people/yangwen/WHU-RS19.html>

¹¹<http://www.escience.cn/people/JunweiHan/NWPU-RESISC45.html>

Model Zoo GeoSR currently contains over 10 trained deep learning based SR models according to the reports from original papers. Since the original methods were implemented in the various platform and used for various sizes of input, GeoSR implements few modifications on several models for unification. Here we briefly introduce a few of the representative models as follows:

1. SRCNN [189]. As a disruptive model, SRCNN is the first deep learning model to perform sample-based SR. Although it only contains three parts: patch extraction and representation, Non-linear mapping, and reconstruction, the model could outperform other conventional models.
2. FSRCNN [214]. Compared with SRCNN model, both accuracy and efficiency are improved in FSRCNN. Instead of applying bicubic interpolation as input data, FSRCNN can directly extract feature from small size LR image. After that, the obtained feature will get through procedures such as shrinking, mapping, expanding, and deconvolution, and finally restore to SR image.
3. ESPCN [71]. The innovative structure invented by ESPCN is called sub-pixel. The ESPCN model could always perform feature learning procedures in a small size, and finally get SR result with the help of sub-pixel.
4. VDSR. Inspired by ResNet, VDSR applies very deep residual networks to achieve SR. The model shows high robustness owing to its nonunified size input training dataset.
5. DRCN [215]. The DRCN method is proposed by Kim, Jiwon et al. at 2016. The global residual and recursive-supervision can make model avoid gradient vanishing and exploding with high performance.
6. DRRN [213]. The core vision of DRRN contains low residual learning, global residual learning, and multiple-weight learning. The structure enables DRRN to obtain better performance with a deeper network.
7. LapSRN [191]. The authors of LapSRN point out the problems of previous deep learning based SR models and use the loss in pyramid structure to achieve large upscale factor SR.
8. SRDenseNet [216]. Since DenseNet has shown the tremendous capability in feature extraction, SRDenseNet [216] is invented upon the structure of DenseNet.
9. SRGAN [192]. The SRGAN model is the first model which successfully integrates GAN structure in SR model. Although the existing SR models could get high PSNR, some important details are inevitably lost. Except for GAN structure, SRGAN also focuses on optimize the loss function.

Architectures The directory `archs/` contains the PyTorch version source code of different models introduced in 1.3 and some basic structures like residual block and inception block. Users can optimize the existing structure to match own requirement or build self-defined architectures conveniently. The trained model will be stored in directory `model.zoo/`.

1.4 Logging Tools

GeoSR logs the detailed information of model hyperparameter settings, model statistic performance, and corresponding learning curves in `logs/`. With the help of logging tools, SR can be easily applied by trained models according to the requirement of users. Moreover, quantitative results and computational efficiency will be saved in `result/` as well.

1.5 Evaluation Metrics and Visualizations Tools

To compare the performance of the obtained SR image with regard to the original remote sensing image HR, several evaluation metrics have been used, such as signal-to-noise ratio (PSNR), structure similarity (SSIM), normalized root mean square error (NRMSE), spectral angle mapper (SAM), and erreur relative globale adimensionnelle de synthèse (ERGAS).

Currently, GeoSR can automatically generate visualized comparison results of different models or iterations for single or multi images. As shown in Figure A.5, comparison results of different upscale factors can also be obtained. To easily understand the performance details, a certain region can be selected and enlarged by users conveniently.

2 Results and Discussion

2.1 Qualitative Result

Figure A.7 presents six groups of randomly selected visualization results generated by different SR models with an upscale factor of 2. From top to bottom rows, there are high-resolution images, obtained SR images by bicubic interpolation, SRCNN, ESPCN, and VDSR model, respectively. Detailed information of a specific area can be enlarged as shown in red sub-region. In general, results of different methods can be easily visualized and compared by “`vision.py`” in GeoSR. Compared with results generated by interpolation (see 2nd row), deep learning methods (3rd row to the final) can show fewer distortion, aliasing, blur, and noise especially around the outline of land features.

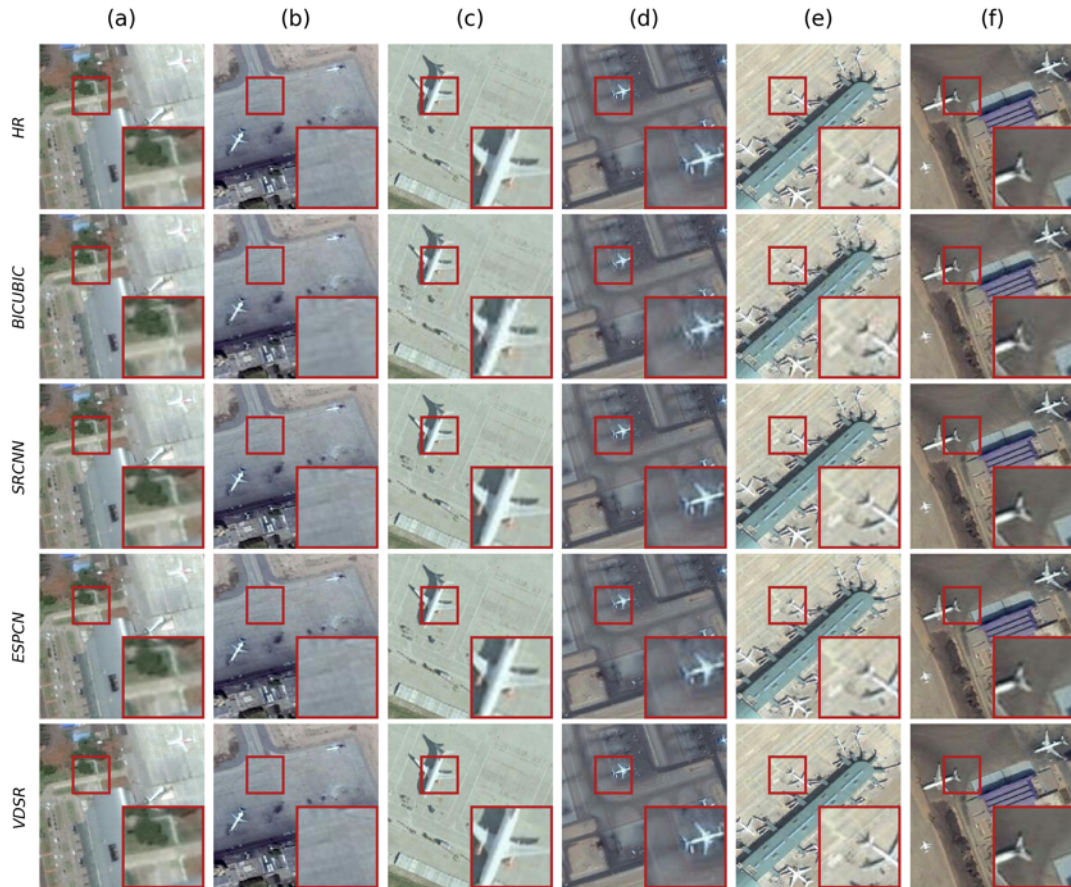


FIGURE A.7: SR comparison results of different images generated by different models.

2.2 Quantitative Result

TABLE A.1: Quantitative comparison of different models

Metrics	BICUBIC	SRCNN	FSRCNN	ESPCN	VDSR
PSNR	27.232	33.034	24.642	30.666	59.742
SSIM	0.937	0.994	0.906	0.987	0.998

As introduced in section 1.5, we provided several metrics for quantitative model evaluation. The related results can be intuitively illustrated via table and histogram respectively by “vision.py”. Here we take two representative evaluators: PSNR and SSIM, as an example to show the performance of different models. Table A.1 and figure A.8 are SR results generated via airport remote sensing imagery with upscale factor of 2.

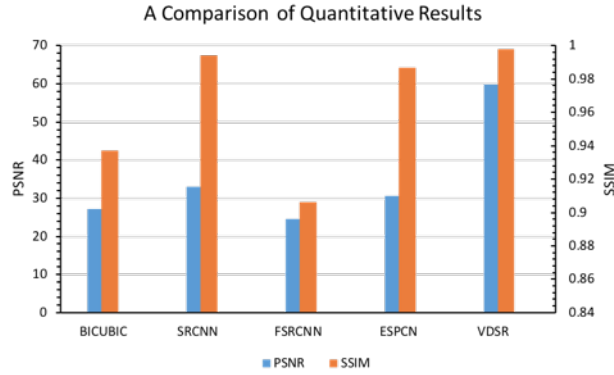


FIGURE A.8: SR comparison results of different models.

2.3 Computational Efficiency

As an important performance indicator, computational efficiency shows the properties of a model which related to the amount of computational resource used by the algorithm. With the help of logging tools, GeoSR logs the related information in training, validation, and testing procedures. Table A.2 illustrates an example of computational efficiency comparison with different models by using GPU NVIDIA TITAN X¹², and the corresponding histogram result can be generated by “vision.py” as well.

TABLE A.2: Computational efficiency comparison

Stage	SRCNN	FSRCNN	ESPCN	VDSR	SRDenseNet
Training / fps	484.9	639.0	627.6	235.0	309.8
Testing / fps	452.9	539.8	641.1	216.7	295.5

3 Conclusion

In this study, we innovatively present an open source computer vision package for deep learning based single-frame remote sensing imagery super-resolution. The package enables researchers and developers to adopt for their researches with high efficiency, which could potentially facilitate the development of the SR in remote sensing field. In future works, scalable API for integrating other important tasks such as land feature classification, segmentation, and detection with SR technique will be improved.

¹²<https://www.nvidia.com/en-us/geforce/products/10series/titan-x-pascal/>

Appendix B

Appendix II: Semantic Segmentation for Urban Planning Maps

Map digitization is one of the most important tasks for research and practical application in fields such as remote sensing, urban planning, and geographic information system (GIS) [217, 218]. More specifically, with the help of map digitization, the digital version of many old and historical maps have not been digitized can be generated, and applied as significant resources for a wide variety of researches like urban sprawl, transportation patterns, diminishing woodlots, shoreline erosion [219], etc. Evidently, the achievement of automatic map digitization can definitely facilitate the sharing of map series online with the public at large, and augments map use in teaching, application, and public service mission to a considerable extent.

Aiming at partitioning imagery into semantically meaningful parts and classify each part into one of the pre-determined classes, the semantic image segmentation techniques [220] emerge as a viable tool for achieving automatic map digitization. However, due to the characteristics of maps, which usually contain complex texture as well as diverse color and noise, the traditional semantic segmentation methods such as graph theory-based [20], clustering-based [137], and classification-based methods [138] are hard to represent adequate and proper patterns for maps based on graphic and hand-crafted textural features, and lead to poor generalization. Given the difficulties faced by the methods mentioned above, the semantic segmentation of maps has been still mainly relying on manual visual interpretation [221], which would be very time consuming and inevitably causes many severe problems. Thus, it remains a challenge to achieve automatic semantic segmentation for maps with high accuracy and efficiency.

In this study, we take urban planning maps, which provide sufficient information about land use, as a representative sample map to perform data source. In terms of DCNNs model, CFPN proposed in Chapter 3 is adopted. The main contributions of this study can be summarized as follows:

- We innovatively investigated the feasibility of automatic semantic segmentation of urban planning maps via deep learning methods.
- We introduced a new dataset. As far as we know, it is the first one for urban planning map semantic segmentation with ground truth.
- The experimental results demonstrate the proposed CFPN outperforms the existing state-of-the-art methods, achieving a mIoU and mF1score of 0.872 and 0.928, respectively.

The remainder of this section is organized as follows. In Section A, we describe the study area and the experimental dataset. Details about the methods are presented in Section B. In Section C, we present the experimental results and discuss the capability of the proposed method in comparison to existing methods.

A Materials

1 Study Area

To demonstrate the feasibility of map segmentation via deep learning, we deliberately selected the urban planning maps of some representative study areas in downtown Tokyo to conduct the experiment. Figure B.1 illustrates the detailed study areas information of this study. As shown in Figure B.1a, three districts in Tokyo are separated into training and testing area colored in red-orange and yellow, respectively. Considering the characteristics of data, we choose Shibuya district (Figure B.1b) as the main area for both training and testing whereas taking Shinjuku and Taito district as additional testing area (Figure B.1c) to evaluate the robustness of model. According to the urban planning map, there are ten land use categories in Shibuya district, such as residential land, commercial land, quasi-residential land, etc. Besides, two other categories: blank space and marginal areas which belong to adjacent districts, are included as background. In contrast with training area, including blank space and marginal areas, the testing area of Shinjuku and Taito district consists of eleven and nine categories, respectively.

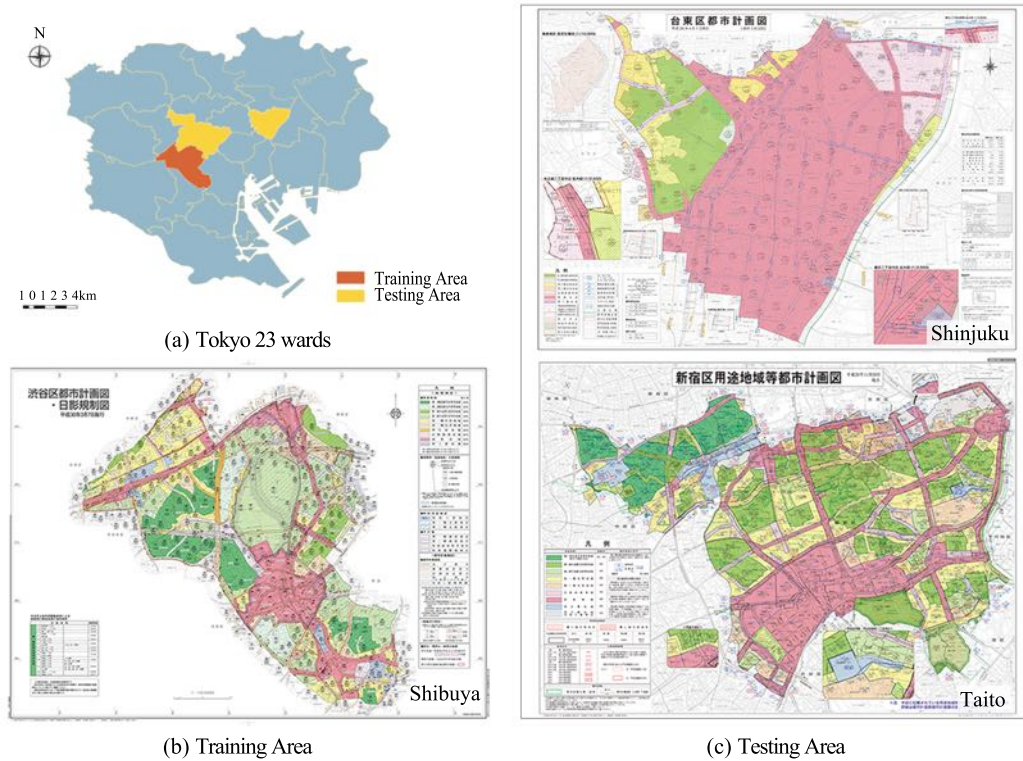


FIGURE B.1: Study areas. (a) Tokyo 23 wards, study area is split into two categories: training and testing, colored in red-orange and yellow, respectively. (b) Urban planning map of Shibuya district which represents the training area. (c) Urban planning map of Shinjuku and Taito district which refer to the testing area.

2 Data Source

To our knowledge, there is no existing published dataset for the purpose of semantic segmentation for urban planning maps. To conduct map semantic segmentation, obtaining the map imagery, especially the corresponding paired ground truth is challenging. In this study, we propose a dataset contains three images of different size (on average 5000×4000 pixels), with all fully annotated ground truth. The original paper version urban planning maps of Shibuya, Shinjuku, and Taito district were released by the government of Japan in March 2018, November 2016, and April 2014, respectively. The imagery of urban planning maps is obtained by scanning paper version ones directly. It should be noted that the scanned imagery not only contains the original features in the map, such as text, symbol, and watermark, but also includes a variety of the noise information caused by scanning procedure, which warrants the segmentation model adapts all these conditions robustly. Each map imagery is paired with a ground truth image beforehand for land use annotation. To best present each land category, polygon-based segmentation masks are created manually by utilizing QGIS [222]. The ground truth for each district is basically an RGB image with different classes, which could align each land use category in map imagery precisely while indicating the boundary between adjacent

land use. To assure that all land use categories have enough representation, any instance of a category larger than an approximately 20×20 pixels is annotated. Here, we intentionally did not annotate explanatory notes, enlarged view, and some redundant background because they are not essential for segmentation task. Due to the density of annotations as well as the variety of land cover categories, some small human error is inevitable.

B Methods

In this study, the patches in the upper two-thirds of the Shibuya district map imagery and the corresponding ground truth are utilized to perform training and validation. Concretely, a sliding window method with a stride of 224 pixels is applied to slice the imagery mentioned above into small patches sized 224×224 pixels. After that, several spatial augmentations such as random rotation and scale transformation are adopted to increase the diversity of data. Here, the total number of patches in training and validation is augmented to 1200 pieces, which are shuffled and divided into training and validation data with ratio of 70% and 30%, respectively. After training, the hyperparameters in proposed multi-class segmentation model: CFPN would be determined. Attempting to quantify the quality of the model, we apply the trained model with proper hyperparameters to conduct semantic segmentation on the remaining one-third of the Shibuya district map imagery as well as on Shinjuku and Taito district. The segmentation results are evaluated via six evaluation metrics generated from multi-class confusion matrix [223], including intersection over union (IoU) [224] for each class, mean intersection over union (mIoU), mean precision (mPrecision), mean recall (mRecall), mean F1-score (mF1-score) [164], and overall accuracy (OA). To clearly reflect the segmentation capability of the model and better assess the experimental results, no post-processing is adopted for computing evaluation metrics. The section details first the feature pyramid methodology, followed by the proposed CFPN structure, lastly, evaluation metrics are proposed.

Attempt to quantify the quality of the multi-class model, we apply the trained model with proper hyperparameters to make semantic segmentation on the testing data and evaluate via six evaluation metrics generated from multi-class confusion matrix [223], including mean precision, mean recall, overall accuracy, mean F1-score [164], mean intersection over union (mIoU) or mean Jaccard index [224], and standard deviation of IoU. Here, we trained multiple models (one hundred) instead of a single model via each algorithm to explore the confidence and variance. Thus, all of the presented scores are

the average score generated from successfully converged models, and four more evaluation metrics: max-IoU, min-IoU, standard deviation of mIoU, and success rate of convergence, are applied to assess the robustness.

C Results

In this section, we show the training quantitative results (Table B.1), testing quantitative results (Table B.2, Table B.3, Table B.4), and testing qualitative results (Table B.3, Table B.5) in different districts. Additionally, qualitative results for representative subregions in corresponding districts (Figure B.2, Figure B.4, Figure B.6) will be illustrated as well. Both quantitative and qualitative results from all districts demonstrate the comparative performance of our proposed model and framework.

TABLE B.1: Training result.

Class No.	Class Name	CFPN (ours)	FPN	Residual-UNet	UNet	FCN-8	FCN-16	FCN-32
1	Blank	0.996	0.985	0.996	0.992	0.989	0.989	0.988
2	Cml.	0.988	0.982	0.988	0.973	0.939	0.945	0.928
3	Cat. 1 Resi.	0.975	0.966	0.974	0.936	0.875	0.851	0.857
4	Cat. 2 M&H. Excl. Resi.	0.986	0.983	0.986	0.967	0.942	0.931	0.922
5	Cat. 1 L. Excl. Resi.	0.985	0.979	0.982	0.965	0.926	0.921	0.900
6	Quasi-industrial	0.968	0.955	0.964	0.885	0.881	0.876	0.849
7	Vicinity	0.984	0.950	0.983	0.964	0.944	0.942	0.930
8	Quasi-residential	0.966	0.964	0.958	0.857	0.796	0.834	0.741
9	Cat. 2 L. Excl. Resi.	0.987	0.982	0.983	0.951	0.940	0.940	0.922
10	Cat. 1 M&H. Excl. Resi.	0.979	0.971	0.973	0.915	0.890	0.848	0.880
11	Cat. 2 Resi.	0.967	0.953	0.961	0.891	0.840	0.825	0.810
12	Nbhd. Cml.	0.951	0.930	0.943	0.850	0.756	0.662	0.707
mIoU	-	0.978	0.967	0.974	0.929	0.893	0.880	0.870
mPrecision	-	0.987	0.984	0.986	0.962	0.949	0.934	0.921
mRecall	-	0.990	0.982	0.988	0.963	0.937	0.936	0.936
mF1-score	-	0.989	0.983	0.987	0.963	0.942	0.934	0.928
OA	-	0.994	0.987	0.993	0.983	0.970	0.966	0.964

TABLE B.2: Testing result for Shibuya.

Class No.	Class Name	CFPN (ours)	FPN	Residual-UNet	UNet	FCN-8	FCN-16	FCN-32
1	Blank	0.993	0.984	0.991	0.987	0.982	0.984	0.984
2	Cml.	0.984	0.980	0.984	0.970	0.914	0.933	0.897
3	Cat. 1 Resi.	0.965	0.955	0.960	0.920	0.815	0.762	0.804
4	Cat. 2 M&H. Excl. Resi.	0.979	0.976	0.976	0.955	0.895	0.868	0.871
5	Cat. 1 L. Excl. Resi.	0.980	0.975	0.977	0.960	0.880	0.896	0.866
6	Quasi-industrial	0.966	0.955	0.953	0.887	0.762	0.840	0.733
7	Vicinity	0.975	0.950	0.967	0.951	0.923	0.926	0.897
8	Quasi-residential	0.946	0.945	0.927	0.732	0.502	0.666	0.554
9	Cat. 2 L. Excl. Resi.	0.984	0.977	0.978	0.944	0.848	0.892	0.844
10	Cat. 1 M&H. Excl. Resi.	0.973	0.966	0.966	0.919	0.813	0.720	0.821
11	Cat. 2 Resi.	0.953	0.936	0.941	0.871	0.716	0.687	0.680
12	Nbhd. Cml.	0.940	0.920	0.926	0.846	0.684	0.616	0.558
mIoU	-	0.970	0.960	0.962	0.912	0.811	0.816	0.792
mPrecision	-	0.983	0.981	0.980	0.954	0.891	0.886	0.864
mRecall	-	0.987	0.978	0.981	0.951	0.890	0.903	0.895
mF1-score	-	0.985	0.979	0.981	0.952	0.890	0.894	0.878
OA	-	0.991	0.985	0.989	0.979	0.955	0.952	0.947

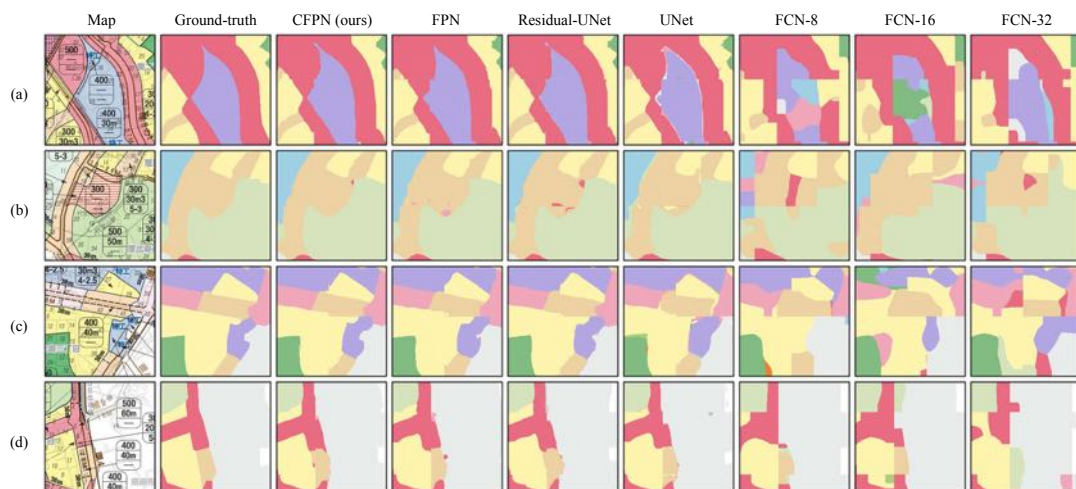


FIGURE B.2: Qualitative results for representative subregions in urban planning map of Shibuya district.



FIGURE B.3: Segmentation results obtained by different methods for urban planning map of Shinjuku district.

TABLE B.3: Testing result for Shinjuku.

Class No.	Class Name	CFPN (ours)	FPN	Residual-UNet	UNet	FCN-8	FCN-16	FCN-32
1	Blank	0.899	0.895	0.902	0.897	0.906	0.896	0.901
2	Cml.	0.934	0.934	0.942	0.932	0.823	0.869	0.695
3	Cat. 1 Resi.	0.829	0.835	0.802	0.815	0.586	0.454	0.452
4	Cat. 2 M&H. Excl. Resi.	0.650	0.513	0.474	0.338	0.139	0.134	0.084
5	Cat. 1 L. Excl. Resi.	0.923	0.949	0.909	0.815	0.273	0.714	0.486
6	Quasi-industrial	0.797	0.837	0.799	0.554	0.088	0.094	0.210
7	Vicinity	0.609	0.591	0.592	0.592	0.567	0.552	0.469
8	Special-industrial	0.734	0.798	0.735	0.592	0.221	0.103	0.011
9	Cat. 1 M&H. Excl. Resi.	0.912	0.871	0.865	0.889	0.283	0.291	0.828
10	Cat. 2 Resi.	0.742	0.699	0.737	0.608	0.512	0.343	0.329
11	Nbhd. Cml.	0.731	0.656	0.558	0.643	0.307	0.150	0.191
mIoU	-	0.796	0.780	0.756	0.698	0.428	0.418	0.423
mPrecision	-	0.889	0.890	0.854	0.831	0.621	0.567	0.564
mRecall	-	0.885	0.865	0.859	0.808	0.578	0.560	0.549
mF1-score	-	0.883	0.869	0.852	0.808	0.554	0.532	0.539
OA	-	0.916	0.909	0.907	0.900	0.804	0.791	0.811

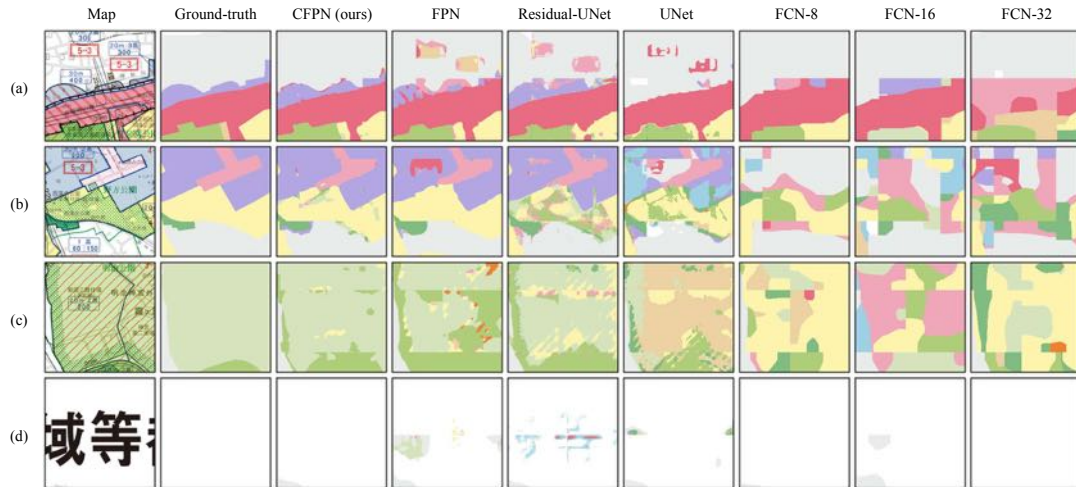


FIGURE B.4: Qualitative results for representative subregions in urban planning map of Shinjuku district.

TABLE B.4: Testing result Taito.

Class No.	Class Name	CFPN (ours)	FPN	Residual-UNet	UNet	FCN-8	FCN-16	FCN-32
1	Blank	0.923	0.909	0.873	0.930	0.878	0.911	0.899
2	Cml.	0.983	0.980	0.981	0.975	0.949	0.962	0.932
3	Cat. 1 Resi.	0.900	0.862	0.891	0.841	0.705	0.530	0.507
4	Quasi-industrial	0.649	0.604	0.575	0.212	0.020	0.182	0.000
5	Vicinity	0.685	0.605	0.468	0.688	0.462	0.590	0.458
6	Cat. 2 M&H. Excl. Resi.	0.905	0.909	0.799	0.795	0.236	0.169	0.000
7	Cat. 1 M&H. Excl. Resi.	0.954	0.888	0.942	0.915	0.788	0.490	0.887
8	Cat. 2 Resi.	0.755	0.585	0.662	0.709	0.027	0.000	0.121
9	Nbhd. Cml.	0.901	0.831	0.862	0.796	0.395	0.442	0.062
mIoU	-	0.851	0.797	0.784	0.763	0.496	0.475	0.429
std-IoU	-	0.115	0.146	0.166	0.215	0.334	0.307	0.378
mPrecision	-	0.948	0.908	0.895	0.860	0.605	0.639	0.543
mRecall	-	0.893	0.879	0.867	0.837	0.589	0.575	0.488
mF1-score	-	0.915	0.879	0.868	0.843	0.588	0.581	0.499
OA	-	0.948	0.934	0.914	0.944	0.884	0.898	0.875



FIGURE B.5: Segmentation results obtained by different methods for urban planning map of Taito district.

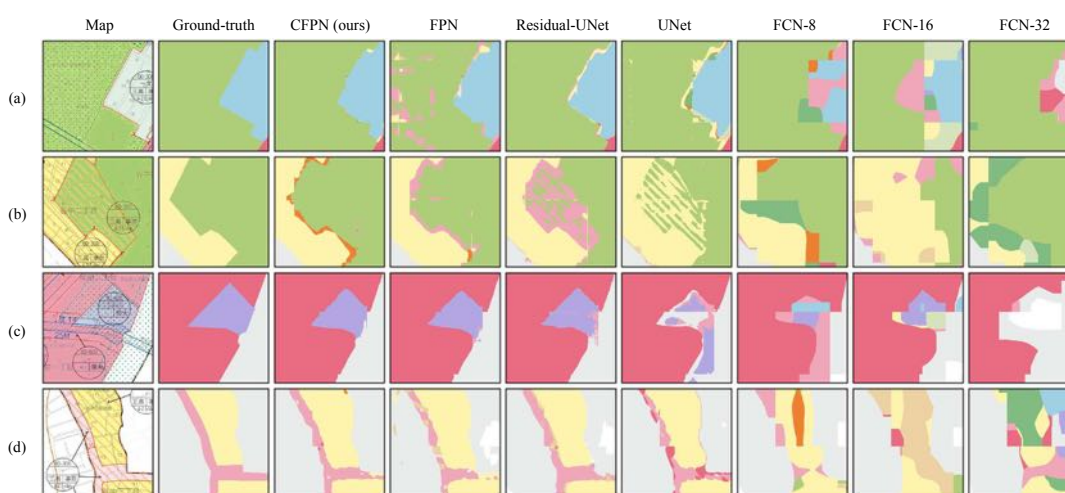


FIGURE B.6: Qualitative results for representative subregions in urban planning map of Taito district.

D Discussion

In this section we show the average performance (Figure B.7, Table B.5) and robustness (Figure B.8, Table B.6) comparison of different models. To demonstrate the efficiency of our proposed method, the computational efficiency and memory comparison are shown in Table B.7 and Table ??, respectively. Furthermore, the feasibility of map denoise and outline extraction by applying our proposed method is illustrated in Figure B.9.

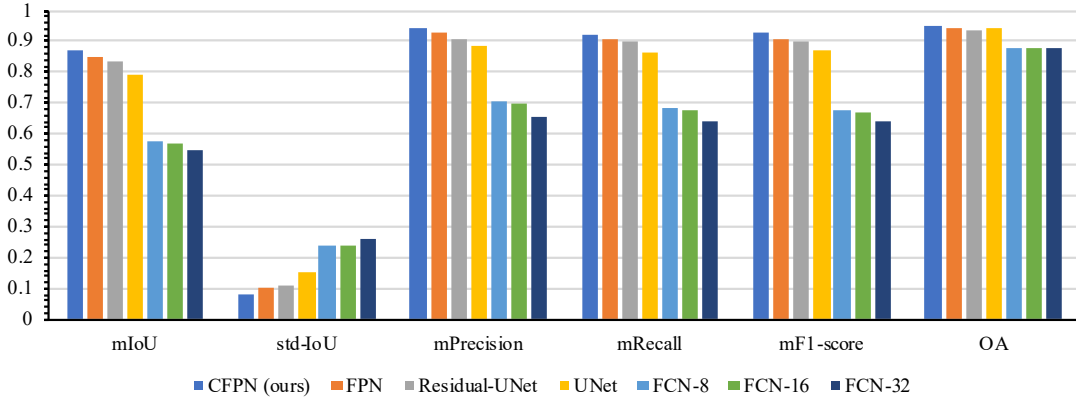


FIGURE B.7: Average performance comparison for different models.

TABLE B.5: Average result for testing amount three districts.

Model	mIoU	std-IoU	mPrecision	mRecall	mF1-score	OA
CFPN (ours)	0.872	0.080	0.940	0.922	0.928	0.952
FPN	0.846	0.101	0.926	0.907	0.909	0.943
Residual-UNet	0.834	0.112	0.910	0.902	0.900	0.937
UNet	0.791	0.153	0.882	0.865	0.868	0.941
FCN-8	0.578	0.239	0.706	0.686	0.677	0.881
FCN-16	0.570	0.237	0.697	0.679	0.669	0.880
FCN-32	0.548	0.263	0.657	0.644	0.639	0.878

the models are further evaluated by standard deviation of IoU as shown in Table B.6.

TABLE B.6: Performance discussion via standard deviation of IoU.

Stage	Region	CFPN (ours)	FPN	Residual-UNet	UNet	FCN-8	FCN-16	FCN-32
Training	Shibuya	0.012	0.016	0.015	0.046	0.065	0.083	0.079
	Shibuya	0.016	0.019	0.021	0.068	0.125	0.116	0.13
Testing	Shinjuku	0.108	0.138	0.148	0.176	0.259	0.289	0.281
	Taito	0.115	0.146	0.166	0.215	0.334	0.307	0.378

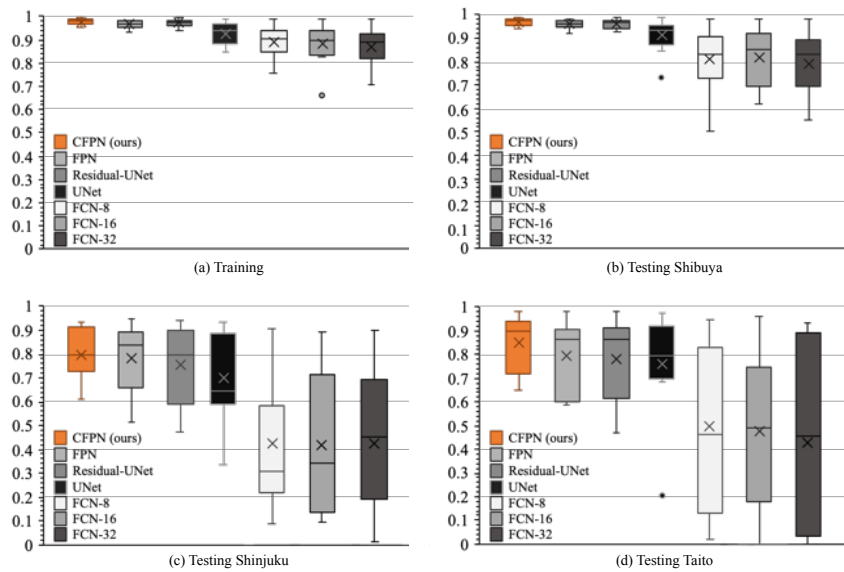


FIGURE B.8: Performance discussion. The box and whisker chart shows distribution of results into quartiles, here highlighting the mean and outliers. The boxes have lines extending vertically, which indicate variability outside the upper and lower quartiles, any point outside those lines is considered an outlier.

TABLE B.7: Computational Efficiency of the model.

Stage	CFPN (ours)	FPN	Residual-UNet	UNet	FCN-8	FCN-16	FCN-32
Training (FPS)	7.718	6.860	6.505	8.314	15.677	12.877	7.542
Testing (FPS)	7.039	7.091	6.635	8.212	14.507	12.258	7.617

TABLE B.8: Memory comparison of different models.

Size	CFPN (ours)	FPN	Residual-UNet	UNet	FCN-8	FCN-16	FCN-32
Parameters (M)	2.759	5.803	7.996	2.745	98.302	98.301	98.299
Memory (MB)	11.113	23.334	32.087	11.020	393.238	393.233	393.223



FIGURE B.9: Map denoising and outline extraction. First row: the original scanned map; second row: canny edge detection results of the original scanned map; third row: denoised map; fourth line: extracted outline for different region.

Appendix C

Appendix III: List of Publications

A International Journals

- **Guo, Zhiling**, et al. "Super-Resolution Integrated Building Semantic Segmentation for Multi-Source Remote Sensing Imagery." *IEEE Access* 7 (2019): 99381-99397.
- Wu, Guangming, Yimin Guo, Xiaoya Song, **Zhiling Guo**, Haoran Zhang, Xiaodan Shi, Ryosuke Shibasaki, and Xiaowei Shao. "A Stacked Fully Convolutional Networks with Feature Alignment Framework for Multi-Label Land-cover Segmentation." *Remote Sensing* 11, no. 9 (2019): 1051.
- Wu, Guangming, **Zhiling Guo**, Xiaodan Shi, Qi Chen, Yongwei Xu, Ryosuke Shibasaki, and Xiaowei Shao. "A boundary regulated network for accurate roof segmentation and outline extraction." *Remote Sensing* 10, no. 8 (2018): 1195.
- Chen, Qi, Lei Wang, Yifan Wu, Guangming Wu, **Zhiling Guo**, and Steven L. Waslander. "Aerial imagery for roof segmentation: A large-scale dataset towards automatic mapping of buildings." *ISPRS journal of photogrammetry and remote sensing* 147 (2019): 42-55.
- Wu, Guangming, Xiaowei Shao, **Zhiling Guo**, Qi Chen, Wei Yuan, Xiaodan Shi, Yongwei Xu, and Ryosuke Shibasaki. "Automatic building segmentation of aerial imagery using multi-constraint fully convolutional networks." *Remote Sensing* 10, no. 3 (2018): 407.
- **Guo, Zhiling**, Qi Chen, Guangming Wu, Yongwei Xu, Ryosuke Shibasaki, and Xiaowei Shao. "Village building identification based on ensemble convolutional neural networks." *Sensors* 17, no. 11 (2017): 2487.

- **Guo, Zhiling**, et al. "Identification of village building via Google Earth images and supervised machine learning methods." *Remote Sensing* 8, no. 4 (2016): 271.

B International Conferences

- **Guo, Zhiling**, et al. "Geosr: A Computer Vision Package for Deep Learning Based Single-Frame Remote Sensing Imagery Super-Resolution." *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2019.
- Wu, Guangming, **Zhiling Guo**, Xiaowei Shao, and Ryosuke Shibasaki. "Geoseg: A Computer Vision Package for Automatic Building Segmentation and Outline Extraction." In *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*, pp. 158-161. IEEE, 2019.
- Dwivedi, Uttam Kumar, **Zhiling Guo**, Hiroyuki Miyazaki, Mohamed Batran, and Ryosuke Shibasaki. "Development of Population Distribution Map and Automated Human Settlement Map Using High Resolution Remote Sensing Images." In *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*, pp. 7224-7227. IEEE, 2018.
- **Guo, Zhiling**, et al. "Semantic Segmentation for Urban Planning Maps Based on U-Net." *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2018.
- Miyazaki, Hiroyuki, Kentaro Kuwata, Wataru Ohira, **Zhiling Guo**, Xiaowei Shao, Yongwei Xu, and Ryosuke Shibasaki. "Development of an automated system for building detection from high-resolution satellite images." In *2016 4th International Workshop on Earth Observation and Remote Sensing Applications (EORSA)*, pp. 245-249. IEEE, 2016.

Bibliography

- [1] Nicolas H. Younan and Selim Aksoy. Foreword to the special issue on pattern recognition in remote sensing. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 5:1331–1333, 2012.
- [2] J. Choi, J. Lee, D. Kim, G. Soprani, P. Cerri, A. Broggi, and K. Yi. Environment-detection-and-mapping algorithm for autonomous driving in rural or off-road environment. *IEEE Trans. Intell. Trans. Syst*, 2012:974–982.
- [3] Huilin Xing and Xiwei Xu. *M8. 0 Wenchuan Earthquake*, volume 123. Springer, 2010.
- [4] N. Mori, T. Takahashi, T. Yasuda, and H. Yanagisawa. Survey of 2011 tohoku earthquake tsunami inundation and run-up. *Geophys. Res. Lett*, 2011.
- [5] Jonathan J. Davies, Alastair R. Beresford, and Andy Hopper. Scalable, distributed, real-time map generation. *IEEE Pervasive Computing*, 5:47–54, 2006.
- [6] Nick Gallent, Meri Juntti, Sue Kidd, and Dave Shaw. *Introduction to rural planning*. Routledge, 2008.
- [7] J. Davidson and G. Wibberley. *Planning and the Rural Environment: Urban and Regional Planning Series*. Amsterdam, Netherlands, ; Elsevier, 2016.
- [8] Guangming Wu, Xiaowei Shao, Zhiling Guo, Qi Chen, Wei Yuan, Xiaodan Shi, Yongwei Xu, and Ryosuke Shibasaki. Automatic building segmentation of aerial imagery using multi-constraint fully convolutional networks. *Remote Sensing*, 10(3):407, 2018.
- [9] Guangming Wu, Zhiling Guo, Xiaodan Shi, Qi Chen, Yongwei Xu, Ryosuke Shibasaki, and Xiaowei Shao. A boundary regulated network for accurate roof segmentation and outline extraction. *Remote Sensing*, 10(8):1195, 2018.
- [10] Monika Kuffer, Karin Pfeffer, and Richard Sliuzas. Slums from space—15 years of slum mapping using remote sensing. *Remote Sensing*, 8(6):455, 2016.

-
- [11] Thomas Lillesand, Ralph W. Kiefer, and Jonathan Chipman. *Remote sensing and image interpretation*. Wiley, John & Sons, 2014.
- [12] J. Richards and X. Jia. *Remote Sensing Digital Image Analysis: An Introduction*. Berlin/Heidelberg, Germany, ; Springer, 1999.
- [13] Robert A. Schowengerdt. *Remote sensing: models and methods for image processing*. Academic press, 2006.
- [14] Agnès Bégué, Elodie Vintrou, Denis Ruelland, Maxime Claden, and Nadine Dessay. Can a 25-year trend in soudano-sahelian vegetation dynamics be interpreted in terms of land use change? a remote sensing approach. *Global Environmental Change*, 21:413–420, 2011.
- [15] Wenbin Wu, Ryosuke Shibasaki, Peng Yang, Qingbo Zhou, and Huajun Tang. Remotely sensed estimation of cropland in china: A comparison of the maps derived from four global land cover datasets. *Canadian Journal of Remote Sensing*, 34:467–479, 2008.
- [16] Qiong Hu, Wenbin Wu, Tian Xia, Qiangyi Yu, Peng Yang, Zhengguo Li, and Qian Song. Exploring the use of google earth imagery and object-based methods in land use/cover mapping. *Remote Sensing*, 5:6026–6042, 2013.
- [17] Qian Yu, Peng Gong, Nick Clinton, Greg Biging, Maggi Kelly, and Dave Schirokauer. Object-based detailed vegetation classification with airborne high spatial resolution remote sensing imagery. *Photogrammetric Engineering & Remote Sensing*, 72:799–811, 2006.
- [18] Chisa Shinsugi, Masaki Matsumura, Mohamed Karama, Junichi Tanaka, Mwatasa Changoma, and Satoshi Kaneko. Factors associated with stunting among children according to the level of food insecurity in the household: a cross-sectional study in a rural community of southeastern kenya. *BMC public health*, 15, 2015.
- [19] Sankar K. Pal and Amita Pal. *Pattern recognition: from classical to modern approaches*. World Scientific, 2001.
- [20] Yan Yan, Gaowen Liu, Sen Wang, Jian Zhang, and Kai Zheng. Graph-based clustering and ranking for diversified image search. *Multimedia Systems*, 23(1):41–52, 2017.
- [21] Yanfeng Wei, Zhongming Zhao, and Jianghong Song. Urban building extraction from high-resolution satellite panchromatic image using clustering and edge detection. In *IGARSS 2004. 2004 IEEE International Geoscience and Remote Sensing Symposium*, volume 3, pages 2008–2010. Ieee, 2004.

- [22] Mariana Belgiu and Lucian Drăguț. Random forest in remote sensing: A review of applications and future directions. *ISPRS Journal of Photogrammetry and Remote Sensing*, 114:24–31, 2016.
- [23] Fei Lv, Min Han, and Tie Qiu. *Remote sensing image classification based on ensemble extreme learning machine with stacked autoencoder*. IEEE Access, 2017.
- [24] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. *In*, 2001, 2001.
- [25] Peter M. Atkinson and Arl Tatnall. Introduction neural networks in remote sensing. *International Journal of remote sensing*, 18:699–709, 1997.
- [26] Farid Melgani and Lorenzo Bruzzone. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Transactions on geoscience and remote sensing*, 42:1778–1790, 2004.
- [27] Alessia Mammone, Marco Turchi, and Nello Cristianini. Support vector machines. In *Interdisciplinary Reviews: Computational Statistics*, 1(3):, pages 283–289. 2009.
- [28] Jian Zheng, Zhazhong Cui, Anfei Liu, and Yu Jia. A k-means remote sensing image classification method based on adaboost. *In*, 2008:27–32, 2008.
- [29] Ugur Zongur, Ugur Halici, Orsan Aytakin, and Ilkay Ulusoy. Airport runway detection in satellite images by adaboost learning. *In*, 7477, 2009.
- [30] Rui Li, Jiulin Sun, Juanle Wang, Lijun Zhu, and Rui Liu. The study on dynamic extraction of urban land use cover with remote sensing image based on adaboost algorithm. In *Sixth International Symposium on Multispectral Image Processing and Pattern Recognition*. pages 74981U–74981U. International Society for Optics and Photonics, 2009.
- [31] Melih Cetin and Ugur Halici and. and örsan aytakin. building detection in satellite images by textural features and adaboost. *In*, 2010:1–4, 2010.
- [32] Jie Dou, Kuan-Tsung Chang, Shuisen Chen, Ali P. Yunus, Jin-King Liu, Huan Xia, and Zhongfan Zhu. Automatic case-based reasoning approach for landslide detection: integration of object-oriented image analysis and a genetic algorithm. *Remote Sensing*, 7:4318–4342, 2015.
- [33] Xianju Li, Xinwen Cheng, Weitao Chen, Gang Chen, and Shengwei Liu. Identification of forested landslides using lidar data, object-based image analysis, and machine learning algorithms. *Remote Sensing*, 7:9705–9726, 2015.

- [34] Sara Attarchi and Richard Gloaguen. Classifying complex mountainous forests with l-band sar and landsat data integration: A comparison among different machine learning methods in the hyrcanian forest. *Remote Sensing*, 6:3624–3647, 2014.
- [35] Wenlong Jing, Yaping Yang, Xiafang Yue, and Xiaodan Zhao. Mapping urban areas with integration of dmsp/ols nighttime light and modis data using machine learning techniques. *Remote Sensing*, 7:12419–12439, 2015.
- [36] Yann LeCun et al. *Lenet-5, convolutional neural networks*. lecun.com/exdb/lenet, URL, 2015.
- [37] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.
- [38] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.
- [39] Sakrapee Paisitkriangkrai, Jamie Sherrah, Pranam Janney, Van-Den Hengel, et al. Effective semantic pixel labelling with convolutional networks and conditional random fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 36–43, 2015.
- [40] John Lafferty, Andrew McCallum, and Fernando CN Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. 2001.
- [41] Michael Kampffmeyer, Arnt-Borre Salberg, and Robert Jenssen. Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 1–9, 2016.
- [42] Zhiling Guo, Qi Chen, Guangming Wu, Yongwei Xu, Ryosuke Shibasaki, and Xiaowei Shao. Village building identification based on ensemble convolutional neural networks. *Sensors*, 17(11):2487, 2017.
- [43] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.
- [44] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.

- [45] Yu Liu, Duc Minh Nguyen, Nikos Deligiannis, Wenrui Ding, and Adrian Munteanu. Hourglass-shapenetwork based semantic segmentation for high resolution aerial imagery. *Remote Sensing*, 9(6):522, 2017.
- [46] Benjamin Bischke, Patrick Helber, Joachim Folz, Damian Borth, and Andreas Dengel. Multi-task learning for segmentation of building footprints with deep neural networks. *arXiv preprint arXiv:1709.05932*, 2017.
- [47] Ilke Demir, Krzysztof Koperski, David Lindenbaum, Guan Pang, Jing Huang, Saikat Basu, Forest Hughes, Devis Tuia, and Ramesh Raska. Deepglobe 2018: A challenge to parse the earth through satellite images. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 172–17209. IEEE, 2018.
- [48] Kang Zhao, Jungwon Kang, Jaewook Jung, and Gunho Sohn. Building extraction from satellite images using mask r-cnn with building boundary regularization. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 242–2424. IEEE, 2018.
- [49] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.
- [50] Remi Delassus and Romain Giot. Cnns fusion for building detection in aerial images for the building detection challenge. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 237–2374. IEEE, 2018.
- [51] Vladimir Iglovikov, Selim Seferbekov, Alexander Buslaev, and Alexey Shvets. Ternausnetv2: Fully convolutional network for instance segmentation. *arXiv preprint arXiv:1806.00844*, 2018.
- [52] Matt Dickenson and Lionel Gueguen. Rotated rectangles for symbolized building footprint extraction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA*, pages 18–22, 2018.
- [53] Weijia Li, Conghui He, Jiarui Fang, and Haohuan Fu. Semantic segmentation based building extraction method using multi-source gis map datasets and satellite imagery. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA*, pages 18–22, 2018.
- [54] Guangming Wu, Yimin Guo, Xiaoya Song, Zhiling Guo, Haoran Zhang, Xiaodan Shi, Ryosuke Shibasaki, and Xiaowei Shao. A stacked fully convolutional networks

- with feature alignment framework for multi-label land-cover segmentation. *Remote Sensing*, 11(9):1051, 2019.
- [55] Jake Bouvrie. Notes on convolutional neural networks, 2006.
- [56] Xueyun Chen, Shiming Xiang, Cheng-Lin Liu, and Chun-Hong Pan. Vehicle detection in satellite images by parallel deep convolutional neural networks. *In*, 2013(2):181–185, 2013.
- [57] Bao-Qing Li and Baoxin Li. Building pattern classifiers using convolutional neural networks. *In*, 1999:3081–3085, 1999.
- [58] Jun Yue, Wenzhi Zhao, Shanjun Mao, and Hui Liu. Spectral–spatial classification of hyperspectral images using deep convolutional neural networks. *Remote Sensing Letters*, 6:468–477, 2015.
- [59] Stefan Lee, Haipeng Zhang, and David J. Crandall. Predicting geo-informative attributes in large-scale image collections using convolutional neural networks. *In*, 2015:550–557, 2015.
- [60] Pierre Sermanet, Soumith Chintala, and Yann LeCun. Convolutional neural networks applied to house numbers digit classification. *In*, 2012(21):3288–3291, 2012.
- [61] Dimitrios Marmanis, Mihai Datcu, Thomas Esch, and Uwe Stilla. Deep learning earth observation classification using imagenet pretrained networks. *IEEE Geoscience and Remote Sensing Letters*, 13:105–109, 2016.
- [62] Jun Ding, Bo Chen, Hongwei Liu, and Mengyuan Huang. Convolutional neural network with data augmentation for sar target recognition. *IEEE Geoscience and Remote Sensing Letters*, 13:364–368, 2016.
- [63] Fan Hu, Gui-Song Xia, Jingwen Hu, and Liangpei Zhang. Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. *Remote Sensing*, 7:14680–14707, 2015.
- [64] Martin Långkvist, Andrey Kiselev, Marjan Alirezaie, and Amy Loutfi. Classification and segmentation of satellite orthoimagery using convolutional neural networks. *Remote Sensing*, 8, 2016.
- [65] Gui-Song Xia, Xiang Bai, Jian Ding, Zhen Zhu, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. Dota: A large-scale dataset for object detection in aerial images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3974–3983, 2018.

- [66] Qi Chen, Lei Wang, Yifan Wu, Guangming Wu, Zhiling Guo, and Steven L Waslander. Aerial imagery for roof segmentation: A large-scale dataset towards automatic mapping of buildings. *ISPRS Journal of Photogrammetry and Remote Sensing*, 147:42–55, 2019.
- [67] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.
- [68] Juan Mario Haut, Ruben Fernandez-Beltran, Mercedes E Paoletti, Javier Plaza, Antonio Plaza, and Filiberto Pla. A new deep generative network for unsupervised remote sensing single-image super-resolution. *IEEE Transactions on Geoscience and Remote Sensing*, (99):1–19, 2018.
- [69] Long D Nguyen, Dongyun Lin, Zhiping Lin, and Jiuwen Cao. Deep cnns for microscopic image classification by exploiting transfer learning and feature concatenation. In *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–5. IEEE, 2018.
- [70] Piotr Dollár, Ron Appel, Serge Belongie, and Pietro Perona. Fast feature pyramids for object detection. *IEEE transactions on pattern analysis and machine intelligence*, 36(8):1532–1545, 2014.
- [71] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1874–1883, 2016.
- [72] A. Romero, C. Gatta, and G. Camps-Valls. Unsupervised deep feature extraction for remote sensing image classification. *IEEE Trans. Geosci. Remote Sens*, 2016:1349–1362.
- [73] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez. Convolutional neural networks for large-scale remote-sensing image classification. *IEEE Trans. Geosci. Remote Sens*, 2017:645–657.
- [74] Q. Wang, W. Shi, P. M. Atkinson, and E. A Pardo-Igúzquiza. new geostatistical solution to remote sensing image downscaling. *IEEE Trans. Geosci. Remote Sens*, 2016:386–396.
- [75] J. Holt. Using google earth™: Bring the world into your classroom level 6–8 (epub 3). ; *Shell Education: Huntington Beach, CA, USA*, 1, 2017.

- [76] E. G. Nestler, M. M. Osqui, and J. G. Convolutional Neural Network Bernstein. *U. S. Patent*, 114:14, December 2016.
- [77] Yann LeCun, Bernhard E. Boser, John S. Denker, Donnie Henderson, Richard E. Howard, Wayne E. Hubbard, and Lawrence D. Jackel. Handwritten digit recognition with a back-propagation network. *In*, pages 396–404, 1990.
- [78] P. Convolutional neural network Kim. *In*. In *MATLAB Deep Learning; : /Heidelberg, Germany*, pages 121–147. Springer, Berlin, 2017.
- [79] L. Zhang, L. Zhang, and B. Du. Deep learning for remote sensing data: A technical tutorial on the state of the art. *IEEE Geosci. Remote Sens. Mag*, 2016:22–40.
- [80] X. Ma, J. Geng, and H. Wang. Hyperspectral image classification via contextual deep learning. *EURASIP J. Image Video Process*, 2015, 2015.
- [81] X. Chen, S. Xiang, C. Liu, and C. Pan. Vehicle detection in satellite images by hybrid deep convolutional neural networks. *IEEE Geosci. Remote Sens. Lett*, 2014:1797–1801.
- [82] Z. Guo, X. Shao, Y. Xu, H. Miyazaki, W. Ohira, and R. Shibasaki. Identification of village building via google earth images and supervised machine learning methods. *Remote Sens*, 2016.
- [83] K. Yu, Y. Lin, and J. Lafferty. Learning image representations from the pixel level via hierarchical sparse coding. *In Proceedings of the*, 2011:1713–1720, June 2011.
- [84] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.
- [85] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *In*, pages 1097–1105, 2012.
- [86] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [87] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.
- [88] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Squeezenet Keutzer. Alexnet-level accuracy with 50x fewer parameters and 0.5 mb model size. *arXiv*, 2016.

- [89] Jifeng Dai, Kaiming He, and Jian Sun. Convolutional feature masking for joint object and stuff segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3992–4000, 2015.
- [90] S. Guo, Y. Luo, and Y. Song. Random forests and vgg-net: An algorithm for the isic 2017 skin lesion classification challenge. *arXiv*, 2017.
- [91] T. Sercu, C. Puhersch, B. Kingsbury, and Y. LeCun. Very deep multilingual convolutional neural networks for lvcsr. In *Proceedings of the 2016 IEEE International Conference on Acoustics*, pages 20–25, Shanghai, China, pp. 4955–4959, March 2016. Speech and Signal Processing (ICASSP).
- [92] Y. Sun, D. Liang, X. Wang, and X. Tang. Face recognition with very deep neural networks. *arXiv*, 2015.
- [93] N. Audebert. le saux. In *Asian Conference on Computer Vision*, pages 180–196, B.; Lefèvre, S. Semantic segmentation of earth observation data using multimodal and multi-scale deep networks. In ; Springer, 2016. Berlin/Heidelberg, Germany.
- [94] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
- [95] Clément Farabet, Camille Couprie, Laurent Najman, and Yann LeCun. Scene parsing with multiscale feature learning, purity trees, and optimal covers. *arXiv preprint arXiv:1202.2160*, 2012.
- [96] Dimitrios Marmanis, Konrad Schindler, Jan Dirk Wegner, Silvano Galliani, Mihai Datcu, and Uwe Stilla. Classification with an edge: Improving semantic image segmentation with boundary detection. *ISPRS Journal of Photogrammetry and Remote Sensing*, 135:158–172, 2018.
- [97] Clement Farabet, Camille Couprie, Laurent Najman, and Yann LeCun. Learning hierarchical features for scene labeling. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1915–1929, 2012.
- [98] Richard Socher, Brody Huval, Bharath Bath, Christopher D Manning, and Andrew Y Ng. Convolutional-recursive deep learning for 3d object classification. In *Advances in neural information processing systems*, pages 656–664, 2012.
- [99] P. Pinheiro and R. Collobert. Recurrent convolutional neural networks for scene labeling. In *Proceedings of the International Conference on Machine Learning*, pages 21–26, China, pp. 82–90, June 2014. Beijing.

- [100] X. Ding and Q. He. Energy-fluctuated multiscale feature learning with deep convnet for intelligent spindle bearing fault diagnosis. *IEEE Trans. Instrum. Meas.*, 2017:1926–1935.
- [101] R. Kiros, K. Popuri, D. Cobzas, and M. Jagersand. Stacked multiscale feature learning for domain independent medical image segmentation. In *International Workshop on Machine Learning in Medical Imaging; : /Heidelberg, Germany*, pages 25–32. Springer, Berlin, 2014.
- [102] Tom Brosch, Lisa YW Tang, Youngjin Yoo, David KB Li, Anthony Traboulsee, and Roger Tam. Deep 3d convolutional encoder networks with shortcuts for multiscale feature integration applied to multiple sclerosis lesion segmentation. *IEEE transactions on medical imaging*, 35(5):1229–1239, 2016.
- [103] C. Couprie, C. Farabet, L. Najman, Le Cun, and Y. Indoor. semantic segmentation using depth information. *arXiv*, 2013.
- [104] A. Eitel, J. T. Springenberg, L. Spinello, M. Riedmiller, and W. Burgard. Multi-modal deep learning for robust rgb-d object recognition. In *Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 681–687, Germany, 28 September–2, October 2015. Hamburg.
- [105] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik. Learning rich features from rgb-d images for object detection and segmentation. In *European Conference on Computer Vision; : /Heidelberg, Germany*, pages 345–360, Berlin, 2014. Springer.
- [106] Anran Wang, Jiwen Lu, Jianfei Cai, Tat-Jen Cham, and Gang Wang. Large-margin multi-modal deep learning for rgb-d object recognition. *IEEE Transactions on Multimedia*, 17(11):1887–1898, 2015.
- [107] F. Ricci, L. Rokach, and B. Shapira. Introduction to recommender systems handbook. In *Recommender Systems Handbook; : /Heidelberg, Germany*, pages 1–35. Springer, Berlin, 2011.
- [108] H. Wang, N. Wang, and D. Yeung. Collaborative deep learning for recommender systems. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 10–13, Australia, pp. 1235–1244, August 2015. Sydney.
- [109] A. M. Elkahky, Y. Song, and X. A He. multi-view deep learning approach for cross domain user modeling in recommendation systems. In *Proceedings of the 24th International Conference on World Wide Web*, pages 18–22, Italy, pp. 278–288, May 2015. Florence.

- [110] H. Cheng, L. Koc, J. Harmsen, T. Shaked, T. Chandra, H. Aradhye, G. Anderson, G. Corrado, W. Chai, M. Ispir, et al. Wide & deep learning for recommender systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, , MA, USA, 15, pages 7–10. 2016.
- [111] J. Quionero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence. *Dataset Shift in Machine Learning*. Cambridge, MA, USA, ; MIT Press, 2009.
- [112] G. H. Golub, M. Heath, and G. Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 1979:215–223.
- [113] J. Davis and M. Goadrich. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 25–29, PA, USA, pp. 233–240, June 2006. Pittsburgh.
- [114] Jean Carletta. Assessing agreement on classification tasks: the kappa statistic. *Computational linguistics*, 22:249–254, 1996.
- [115] Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. *arXiv preprint arXiv:1312.4400*, 2013.
- [116] K. M. Sammut and S. R. Jones. Implementing nonlinear activation functions in neural network emulators. *Electron. Lett*, 1991:1037–1038.
- [117] W. Lin and L. Dong. Adaptive downsampling to improve image compression at low bit rates. *IEEE Trans. Image Process*, 2006:2513–2521.
- [118] Y. LeCun and Y. Bengio. *Convolutional networks for images, speech, and time series*, volume 3361. Cambridge, MA, USA, In *The Handbook of Brain Theory and Neural Networks*; MIT Press, 1995.
- [119] A. Giusti, D. C. Ciresan, J. Masci, L. M. Gambardella, and J. Schmidhuber. Fast image scanning with deep max-pooling convolutional neural networks. In *Proceedings of the 2013 20th IEEE International Conference on Image Processing (ICIP)*, pages 15–18, Australia, pp. 4034–4038, September 2013. Melbourne.
- [120] D. Heckerman and C. Models and Meek. and selection criteria for regression and classification. In *Proceedings of the Thirteenth conference on Uncertainty in artificial intelligence*, pages 1–3, RI, USA, pp. 223–228, August 1997. Providence.
- [121] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [122] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

- [123] T. G. Dietterich and E. B. Kong. Machine learning bias, statistical bias, and statistical variance of decision tree algorithms. ; Technical Report; Department of Computer Science, Oregon State University: Corvallis, OR, USA, 1995.
- [124] Bernhard Scholkopf and Alexander J. Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2001.
- [125] Douglas M. Hawkins. The problem of overfitting. *Journal of chemical information and computer sciences*, 44:1–12, 2004.
- [126] Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 2014.
- [127] Sepp Hochreiter. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 6:107–116, 1998.
- [128] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International Conference on Machine Learning, pages 448–456, 2015.
- [129] Engui Fan. Extended tanh-function method and its applications to nonlinear equations. *Physics Letters A*, 277:212–218, 2000.
- [130] Léon Bottou. Stochastic gradient descent tricks. In Neural networks: Tricks of the trade, pages 421–436. 2012.
- [131] M. D. Zeiler, R. Visualizing Fergus, and Understanding Convolutional Networks. In. In European Conference on Computer Vision; : /Heidelberg, Germany, pages 818–833, Berlin, 2014. Springer.
- [132] Y. Liu, Y. Mei, and C. Chen. Village planning methods under new countryside construction background. *City Plan. Rev*, 2008:74–78.
- [133] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22:1345–1359, 2010.
- [134] JM Bernardo, MJ Bayarri, JO Berger, AP Dawid, D Heckerman, AFM Smith, and M West. Generative or discriminative? getting the best of both worlds. *Bayesian statistics*, 8(3):3–24, 2007.
- [135] Maria Vakalopoulou, Konstantinos Karantzalos, Nikos Komodakis, and Nikos Paragios. Building detection in very high resolution multispectral data with deep learning features. In *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1873–1876. IEEE, 2015.

- [136] Shunta Saito and Yoshimitsu Aoki. Building and road detection from large aerial imagery. In *Image Processing: Machine Vision Applications VIII*, volume 9405, page 94050K. International Society for Optics and Photonics, 2015.
- [137] Brian Fulkerson, Andrea Vedaldi, and Stefano Soatto. Class segmentation and object localization with superpixel neighborhoods. In *2009 IEEE 12th international conference on computer vision*, pages 670–677. IEEE, 2009.
- [138] Mingjun Song and Daniel Civco. Road extraction using svm and image segmentation. *Photogrammetric Engineering & Remote Sensing*, 70(12):1365–1371, 2004.
- [139] Olivier Teboul, Loic Simon, Panagiotis Koutsourakis, and Nikos Paragios. Segmentation of building facades using procedural shape priors. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3105–3112. IEEE, 2010.
- [140] Aparajithan Sampath and Jie Shan. Segmentation and reconstruction of polyhedral building roofs from aerial lidar point clouds. *IEEE Transactions on geoscience and remote sensing*, 48(3):1554–1567, 2010.
- [141] John D Owens, Mike Houston, David Luebke, Simon Green, John E Stone, and James C Phillips. Gpu computing. 2008.
- [142] C Reid Turner, Alfonso Fuggetta, Luigi Lavazza, and Alexander L Wolf. A conceptual basis for feature engineering. *Journal of Systems and Software*, 49(1):3–15, 1999.
- [143] Yoshua Bengio, Patrice Simard, Paolo Frasconi, et al. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994.
- [144] Li Deng, Dong Yu, et al. Deep learning: methods and applications. *Foundations and Trends® in Signal Processing*, 7(3–4):197–387, 2014.
- [145] Alberto Garcia-Garcia, Sergio Orts-Escolano, Sergiu Oprea, Victor Villena-Martinez, and Jose Garcia-Rodriguez. A review on deep learning techniques applied to semantic segmentation. *arXiv preprint arXiv:1704.06857*, 2017.
- [146] Varun Bhagwan, Timothy Liu, Justin Ormont, and Heather Underwood. Deconvolution of digital images, November 27 2018. US Patent App. 10/140,495.
- [147] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 1520–1528, 2015.

- [148] Xiao-Jiao Mao, Chunhua Shen, and Yu-Bin Yang. Image restoration using convolutional auto-encoders with symmetric skip connections. *arXiv preprint arXiv:1606.08921*, 2016.
- [149] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, volume 1, page 4, 2017.
- [150] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.
- [151] Zhengxin Zhang, Qingjie Liu, and Yunhong Wang. Road extraction by deep residual u-net. *IEEE Geoscience and Remote Sensing Letters*, 15(5):749–753, 2018.
- [152] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.
- [153] Weiyue Wang, Ronald Yu, Qianguai Huang, and Ulrich Neumann. Sgpn: Similarity group proposal network for 3d point cloud instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2569–2578, 2018.
- [154] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2018.
- [155] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems*, pages 5099–5108, 2017.
- [156] Tobias Pohlen, Alexander Hermans, Markus Mathias, and Bastian Leibe. Full-resolution residual networks for semantic segmentation in street scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4151–4160, 2017.
- [157] Yuhui Yuan and Jingdong Wang. Ocnet: Object context network for scene parsing. *arXiv preprint arXiv:1809.00916*, 2018.
- [158] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014.

- [159] Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersin Yumer, and Raquel Urtasun. Upsnet: A unified panoptic segmentation network. *arXiv preprint arXiv:1901.03784*, 2019.
- [160] Qiang Wang, Li Zhang, Luca Bertinetto, Weiming Hu, and Philip HS Torr. Fast online object tracking and segmentation: A unifying approach. *arXiv preprint arXiv:1812.05050*, 2018.
- [161] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):60, 2019.
- [162] Dongjie Tan. Image enhancement based on adaptive median filter and wallis filter. In *2015 4th National Conference on Electrical, Electronics and Computer Engineering*. Atlantis Press, 2015.
- [163] Russell G Congalton. A review of assessing the accuracy of classifications of remotely sensed data. *Remote sensing of environment*, 37(1):35–46, 1991.
- [164] Yutaka Sasaki et al. The truth of the f-measure. *Teach Tutor mater*, 1(5):1–5, 2007.
- [165] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.
- [166] Mark Polak, Hong Zhang, and Minghong Pi. An evaluation metric for image segmentation of multiple objects. *Image and Vision Computing*, 27(8):1223–1227, 2009.
- [167] Armin Gruen and Haihong Li. Road extraction from aerial and satellite images by dynamic programming. *ISPRS Journal of Photogrammetry and Remote Sensing*, 50:11–20, 1995.
- [168] Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*, 2017.
- [169] Edward H Adelson, Charles H Anderson, James R Bergen, Peter J Burt, and Joan M Ogden. Pyramid methods in image processing. *RCA engineer*, 29(6):33–41, 1984.
- [170] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li. Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*, 2015.
- [171] Ryuhei Hamaguchi and Shuhei Hikosaka. Building detection from satellite imagery using ensemble of size-specific detectors. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 223–2234. IEEE, 2018.

- [172] Mate Kisantal, Zbigniew Wojna, Jakub Murawski, Jacek Naruniec, and Kyunghyun Cho. Augmentation for small object detection. *CoRR*, abs/1902.07296, 2019.
- [173] Yancheng Bai, Yongqiang Zhang, Mingli Ding, and Bernard Ghanem. Sod-mtgan: Small object detection via multi-task generative adversarial network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 206–221, 2018.
- [174] Guo X Hu, Zhong Yang, Lei Hu, Li Huang, and Jia M Han. Small object detection with multiscale features. *International Journal of Digital Multimedia Broadcasting*, 2018, 2018.
- [175] JM Haut, ME Paoletti, R Fernandez-Beltran, J Plaza, A Plaza, and Jun Li. Remote sensing single-image superresolution based on a deep compendium model. *IEEE Geoscience and Remote Sensing Letters*, 2019.
- [176] Hasan Demirel and Gholamreza Anbarjafari. Satellite image resolution enhancement using complex wavelet transform. *IEEE geoscience and remote sensing letters*, 7(1):123–126, 2009.
- [177] Philippe Thévenaz, Thierry Blu, and Michael Unser. Image interpolation and resampling. *Handbook of medical imaging, processing and analysis*, 1(1):393–420, 2000.
- [178] Sylvia Frühwirth-Schnatter. Data augmentation and dynamic linear models. *Journal of time series analysis*, 15(2):183–202, 1994.
- [179] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.
- [180] Yongbing Zhang, Debin Zhao, Jian Zhang, Ruiqin Xiong, and Wen Gao. Interpolation-dependent image downsampling. *IEEE Transactions on Image Processing*, 20(11):3291–3296, 2011.
- [181] Robert Keys. Cubic convolution interpolation for digital image processing. *IEEE transactions on acoustics, speech, and signal processing*, 29(6):1153–1160, 1981.
- [182] Chih-Yuan Yang, Chao Ma, and Ming-Hsuan Yang. Single-image super-resolution: A benchmark. In *European Conference on Computer Vision*, pages 372–386. Springer, 2014.
- [183] Sung Cheol Park, Min Kyu Park, and Moon Gi Kang. Super-resolution image reconstruction: a technical overview. *IEEE signal processing magazine*, 20(3):21–36, 2003.

- [184] Hong Zhu, Xinming Tang, Junfeng Xie, Weidong Song, Fan Mo, and Xiaoming Gao. Spatio-temporal super-resolution reconstruction of remote-sensing images based on adaptive multi-scale detail enhancement. *Sensors*, 18(2):498, 2018.
- [185] Zhihao Wang, Jian Chen, and Steven CH Hoi. Deep learning for image super-resolution: A survey. *arXiv preprint arXiv:1902.06068*, 2019.
- [186] Alain Hore and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In *2010 20th International Conference on Pattern Recognition*, pages 2366–2369. IEEE, 2010.
- [187] Stephen M Pizer, E Philip Amburn, John D Austin, Robert Cromartie, Ari Geselowitz, Trey Greer, Bart ter Haar Romeny, John B Zimmerman, and Karel Zuiderveld. Adaptive histogram equalization and its variations. *Computer vision, graphics, and image processing*, 39(3):355–368, 1987.
- [188] Samuel Schulter, Christian Leistner, and Horst Bischof. Fast and accurate image upscaling with super-resolution forests. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3791–3799, 2015.
- [189] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *European conference on computer vision*, pages 184–199. Springer, 2014.
- [190] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1646–1654, 2016.
- [191] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 624–632, 2017.
- [192] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017.
- [193] Unal Okayay, Jennifer Telling, Craig L Glennie, and William E Dietrich. Airborne lidar change detection: An overview of earth sciences applications. *Earth-Science Reviews*, page 102929, 2019.
- [194] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

- [195] QianWei Cheng. Study on distribution and transition tendency of informal areas in cities : Through physical environment mapping, mar 2019.
- [196] T Kavzoglu and H Tonbul. Segmentation quality assessment for varying spatial resolutions of very high resolution satellite imagery. *RESURSLAR*, page 109.
- [197] Nika Mesner and Kristof Ostir. Investigating the impact of spatial and spectral resolution of satellite images on segmentation quality. *Journal of Applied Remote Sensing*, 8(1):083696, 2014.
- [198] Prasanna K Sahoo, SAKC Soltani, and Andrew KC Wong. A survey of thresholding techniques. *Computer vision, graphics, and image processing*, 41(2):233–260, 1988.
- [199] Sedat Ozer, Deanna L Langer, Xin Liu, Masoom A Haider, Theodorus H van der Kwast, Andrew J Evans, Yongyi Yang, Miles N Wernick, and Imam S Yetik. Supervised and unsupervised methods for prostate cancer segmentation with multispectral mri. *Medical physics*, 37(4):1873–1883, 2010.
- [200] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.
- [201] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.
- [202] Lei Ma, Manchun Li, Xiaoxue Ma, Liang Cheng, Peijun Du, and Yongxue Liu. A review of supervised object-based land-cover image classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 130:277–293, 2017.
- [203] Shunping Ji, Shiqing Wei, and Meng Lu. Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set. *IEEE Transactions on Geoscience and Remote Sensing*, (99):1–13, 2018.
- [204] Lin Li, Jian Liang, Min Weng, and Haihong Zhu. A multiple-feature reuse network to extract buildings from remote sensing imagery. *Remote Sensing*, 10(9):1350, 2018.
- [205] Yongyang Xu, Liang Wu, Zhong Xie, and Zhanlong Chen. Building extraction in very high resolution remote sensing imagery using deep learning and guided filters. *Remote Sensing*, 10(1):144, 2018.

- [206] Raimundo Real and Juan M Vargas. The probabilistic basis of jaccard's index of similarity. *Systematic biology*, 45(3):380–385, 1996.
- [207] Steve Stehman. Estimating the kappa coefficient and its variance under stratified random sampling. *Photogrammetric Engineering and Remote Sensing*, 62(4):401–407, 1996.
- [208] Qi Chen, Lei Wang, Yifan Wu, Guangming Wu, Zhiling Guo, and Steven L Waslander. Aerial imagery for roof segmentation: A large-scale dataset towards automatic mapping of buildings. *arXiv preprint arXiv:1807.09532*, 2018.
- [209] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [210] William T Freeman, Thouis R Jones, and Egon C Pasztor. Example-based super-resolution. *IEEE Computer graphics and Applications*, 22(2):56–65, 2002.
- [211] Wei Wu, Zheng Liu, and Xiaohai He. Learning-based super resolution using kernel partial least squares. *Image and Vision Computing*, 29(6):394–406, 2011.
- [212] Daniel Glasner, Shai Bagon, and Michal Irani. Super-resolution from a single image. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 349–356. IEEE, 2009.
- [213] Ying Tai, Jian Yang, and Xiaoming Liu. Image super-resolution via deep recursive residual network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3147–3155, 2017.
- [214] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *European Conference on Computer Vision*, pages 391–407. Springer, 2016.
- [215] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1637–1645, 2016.
- [216] Tong Tong, Gen Li, Xiejie Liu, and Qinquan Gao. Image super-resolution using dense skip connections. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4799–4807, 2017.
- [217] Andreas Illert. Automatic digitization of large scale maps. In *AUTOCARTO-CONFERENCE-*, volume 6, pages 113–113. Citeseer, 1991.
- [218] Bai Hongye. Digitization of ancient maps based on gis technology: the yu ji tu map. In *World Library and Information Congress: 75th IFLA General Conference*

- and Assembly” Libraries Create Futures: Building on Cultural Heritage”, Milan, Italy, 2009.*
- [219] Maggi Kelly, Barbara Allen-Diaz, and Norma Kobzina. Digitization of a historic dataset: the wieslander california vegetation type mapping project. *Madroño*, 52(3):191–202, 2005.
- [220] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.
- [221] Vladimir I Pavlovic, Rajeev Sharma, and Thomas S Huang. Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (7):677–695, 1997.
- [222] QGIS Development Team. *QGIS Geographic Information System*. Open Source Geospatial Foundation, 2018.
- [223] Tom Fawcett. An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874, 2006.
- [224] Pang-Ning Tan. *Introduction to data mining*. Pearson Education India, 2018.