



Deep learning for land use and land cover in aerial images

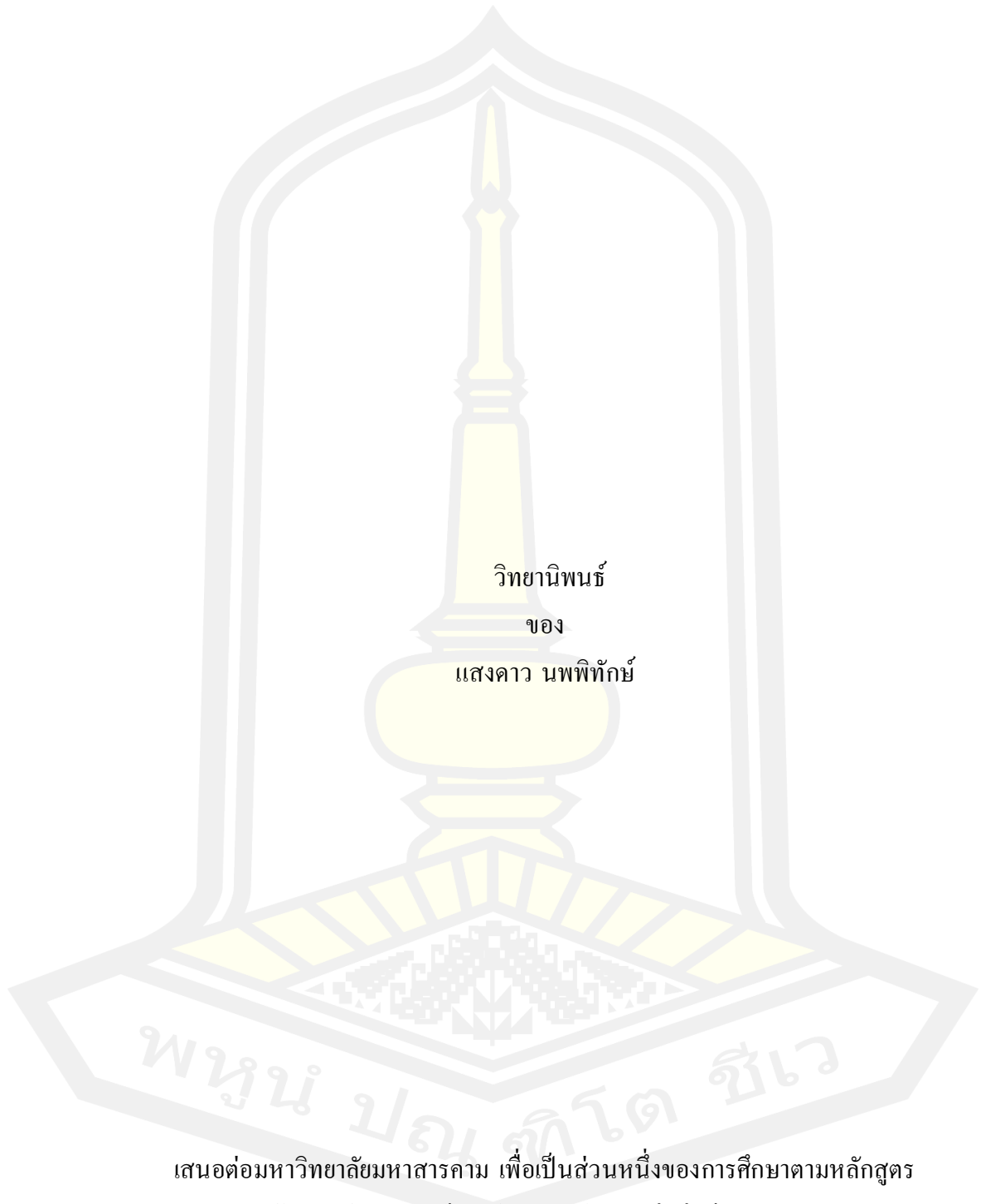
Sangdaow Noppitak

A Thesis Submitted in Partial Fulfillment of Requirements for  
degree of Doctor of Philosophy in Information Technology

March 2022

Copyright of Mahasarakham University

การเรียนรู้เชิงลึกสำหรับการใช้ประโยชน์ที่ดินและสิ่งปกคลุมดินในภาพถ่ายทางอากาศ



วิทยานิพนธ์  
ของ  
แสงดาว นพพิทักษ์

เสนอต่อมหาวิทยาลัยมหาสารคาม เพื่อเป็นส่วนหนึ่งของการศึกษาคตามหลักสูตร  
ปริญญาปรัชญาดุษฎีบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

มีนาคม 2565

ลิขสิทธิ์เป็นของมหาวิทยาลัยมหาสารคาม

Deep learning for land use and land cover in aerial images



Sangdaow Noppitak

A Thesis Submitted in Partial Fulfillment of Requirements  
for Doctor of Philosophy (Information Technology)

March 2022

Copyright of Mahasarakham University



The examining committee has unanimously approved this Thesis, submitted by Miss Sangdaow Noppitak , as a partial fulfillment of the requirements for the Doctor of Philosophy Information Technology at Maharakham University

Examining Committee

- ..... Chairman  
(Prof. Rapeepan Pitakaso , Ph.D.)
- ..... Advisor  
(Asst. Prof. Olarik Surinta , Ph.D.)
- ..... Committee  
(Asst. Prof. Rapeeporn Chamchong ,  
Ph.D.)
- ..... Committee  
(Asst. Prof. Phatthanaphong  
Chompoowises , Ph.D.)
- ..... Committee  
(Asst. Prof. Chatklaw Jareanpon ,  
Ph.D.)

Maharakham University has granted approval to accept this Thesis as a partial fulfillment of the requirements for the Doctor of Philosophy Information Technology

..... (Asst. Prof. Sasitorn Kaewman , Ms.c.) (Assoc. Prof. Krit Chaimoon , Ph.D.)  
Dean of The Faculty of Informatics Dean of Graduate School

<b>TITLE</b>	Deep learning for land use and land cover in aerial images		
<b>AUTHOR</b>	Sangdaow Noppitak		
<b>ADVISORS</b>	Assistant Professor Olarik Surinta , Ph.D.		
<b>DEGREE</b>	Doctor of Philosophy	<b>MAJOR</b>	Information Technology
<b>UNIVERSITY</b>	Maharakham University	<b>YEAR</b>	2022

### ABSTRACT

This thesis focuses on two main types of research: classification and segmentation, addressing the challenge of aerial images using deep learning techniques.

Chapter 1 provides a brief general introduction to deep learning for land use and land cover in aerial images, followed by the research questions. Additionally, the objectives of the dissertation and its contributions are described.

In Chapter 2, a convolutional neural network (CNN) method is proposed to classify land use and land cover of five economic crops: rice, sugarcane, cassava, rubber, and longan, from the aerial images. Also, the ensemble learning method is proposed to enhance the performance of the land use classification. The work reported in this thesis used eight CNN architectures to create robust models and classify the aerial images for the classification tasks. Hence, three data augmentation techniques (rotation, height shift, and width shift) are combined when training the CNN models. Moreover, the ensemble CNN model is proposed to enhance the performance of the economic crop classification model.

Chapter 3 propose the snapshot ensemble CNN to improve the performance of the land use classification from the aerial images. The work reported in this thesis experimented with the snapshot ensemble CNN method using various learning schedules. The new drop cyclic cosine learning rate schedule, called dropCyclic, is proposed and compared with two existing learning rate schedules. The proposed learning rate schedule is evaluated on three datasets: UCM, AID, and EcoCropsAID. The results showed that the proposed dripCyclic outperformed the existing learning rate schedules on the UCM dataset. As a result, the ensemble CNN obtains better performance than using only the single CNN.

Chapter 4 proposes the instance segmentation technique to segment the water body from the aerial. The mask region-based CNN (mask R-CNN) is the instance segmentation technique to find the water resource areas for the segmentation task in this thesis. In the experiments, the mask R-CNN model could segment water bodies efficiently. Furthermore, the data augmentation techniques are included in the training process. The experimental results showed that the mask R-CNN method combined with data augmentation techniques when training obtained two times better

performance than without combining data augmentation techniques.

Chapter 5 comprises two main sections: 1) Answers to the research questions 2) future work. This chapter briefly explains the proposed approaches and answers three main research questions in land use and land cover in aerial images using deep learning techniques. Two main approaches are planned to be the focused of future work, as follows. For the classification technique, I plan to replace the unweighted average method with the cost-sensitive probability method in the snapshot ensemble CNN method. For the segmentation technique, I will consider applying the new deep learning methods to enhance the performance of the water resource segmentation and other tasks.

This research makes a significant contribution in classification and segmentation for land use and land cover through deep learning-based innovations and has great potential utility in a wide range of aerial images for geographic information systems and remote sensing.

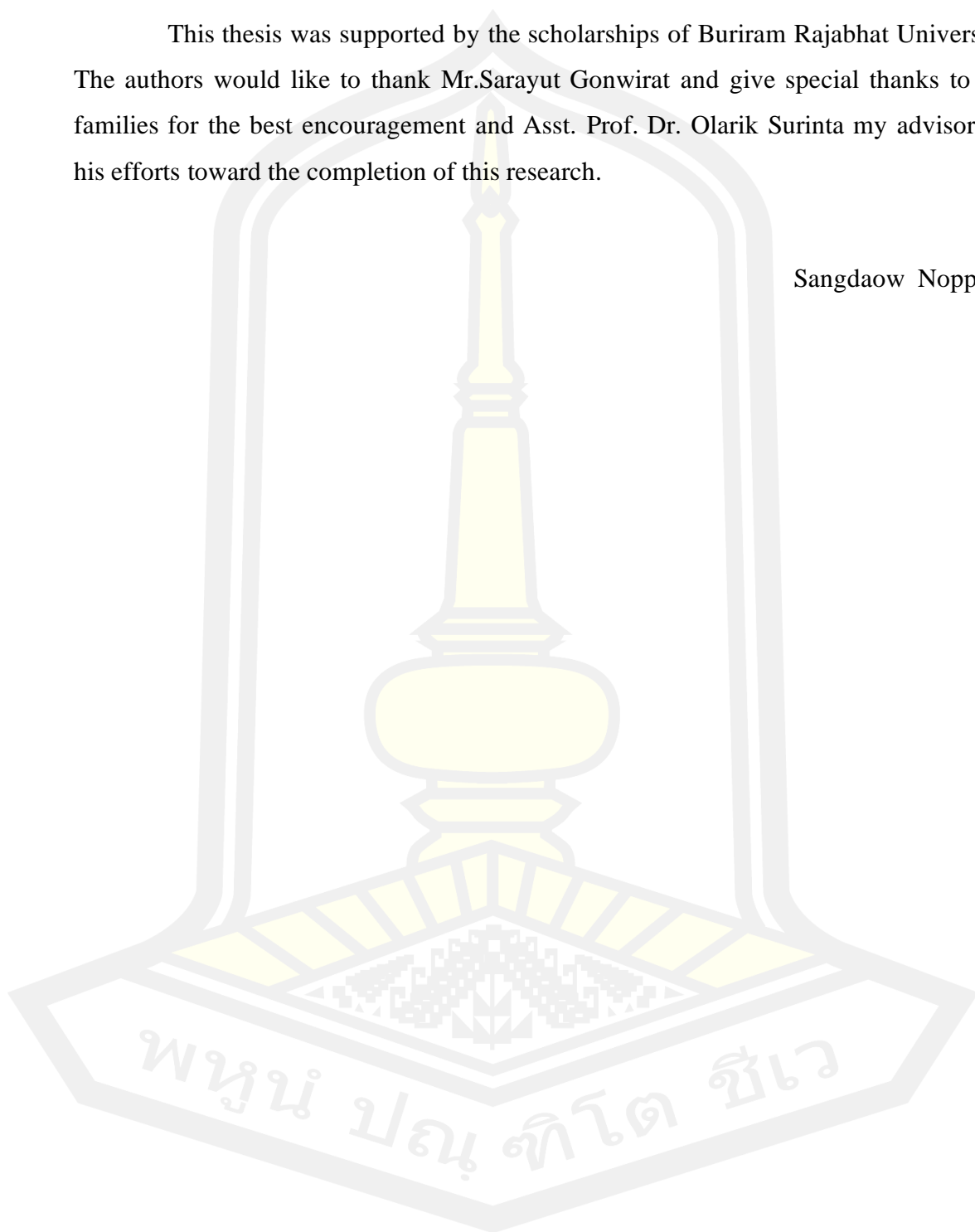
Keyword : Economic crops aerial image, Land use classification, Deep learning architecture, Ensemble convolutional neural network, Ensemble method, Data augmentation, Instance Segmentation, Water Body, Aerial Image, Mask R-CNN, Transfer Learning, Snapshot ensemble



## ACKNOWLEDGEMENTS

This thesis was supported by the scholarships of Buriram Rajabhat University. The authors would like to thank Mr.Sarayut Gonwirat and give special thanks to my families for the best encouragement and Asst. Prof. Dr. Olarik Surinta my advisor for his efforts toward the completion of this research.

Sangdaow Noppitak



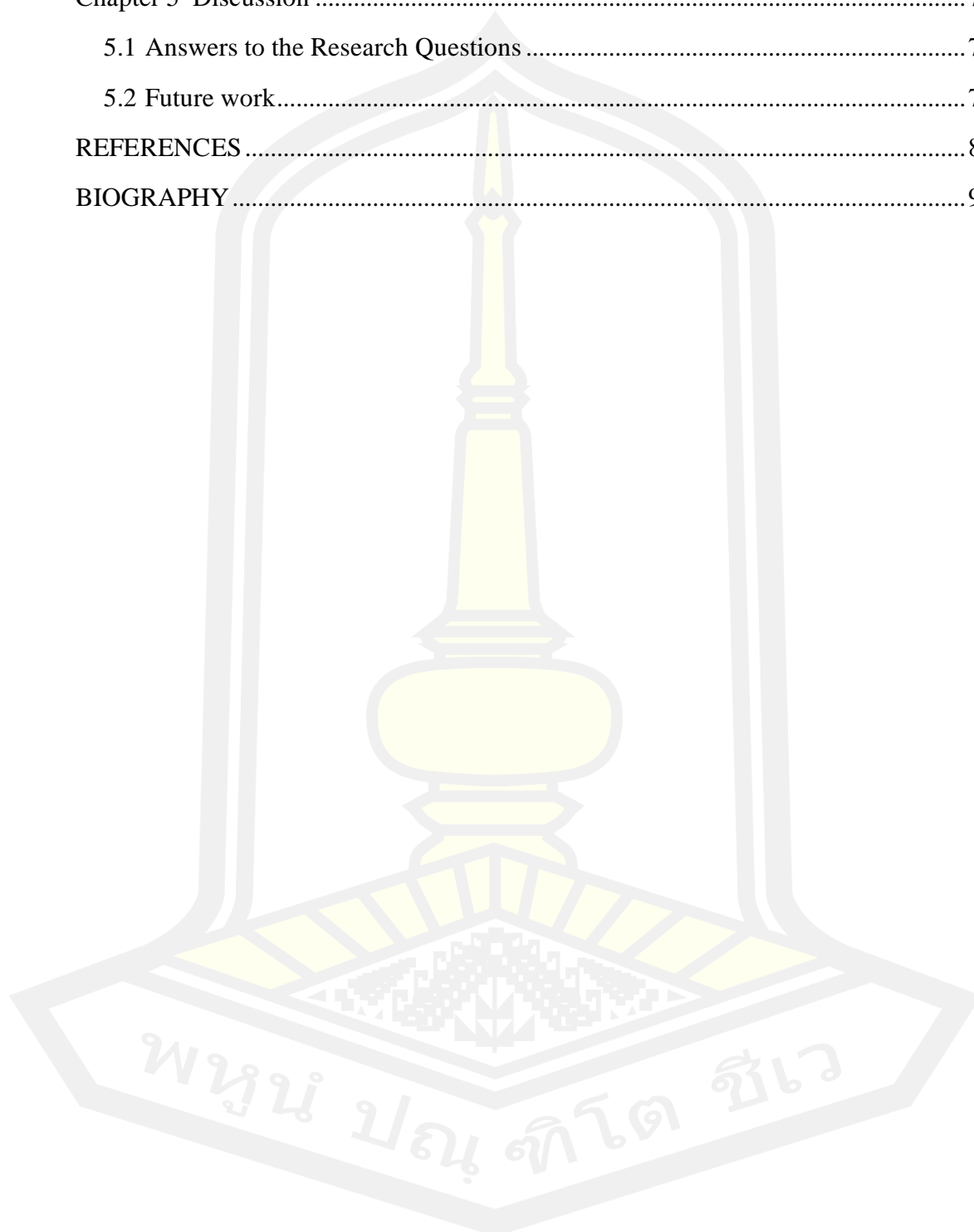
# TABLE OF CONTENTS

	<b>Page</b>
ABSTRACT .....	D
ACKNOWLEDGEMENTS.....	F
TABLE OF CONTENTS .....	G
Table.....	J
Figure .....	K
Acronyms.....	N
Chapter 1 Introduction .....	1
1.1 Research Question.....	3
1.2 The Objective of This Dissertation .....	4
1.3 Contributions .....	4
Chapter 2 Ensemble Convolutional Neural Network .....	7
2.1 Introduction.....	7
2.2 Related Work .....	9
2.3 Proposed Ensemble Convolutional Neural Network Architecture .....	11
2.3.1 Convolutional Neural Network Architectures .....	12
2.3.2 Data Augmentation for Aerial Images .....	15
2.3.3 Ensemble Methods.....	16
2.4 Thailand’s Economic Crops Aerial Image Dataset .....	17
2.5 Experimental Results .....	18
2.5.1 Experiments with Convolutional Neural Network Architectures and Data Augmentation Techniques .....	18
2.5.2 Experiments with Ensemble Methods .....	22
2.6 Conclusions.....	25
Chapter 3 Snapshot Ensemble CNN and New Learning Rate Schedule .....	27
3.1 Introduction.....	27
3.2 Related Work .....	30



3.2.1	Ensemble learning.....	30
3.2.2	Snapshot ensemble CNN.....	32
3.2.3	The cyclical learning rate for snapshot ensemble CNN .....	33
3.3	Proposed snapshot ensemble CNN for aerial image classification.....	34
3.3.1	Cosine cyclic learning rate schedule.....	34
3.3.2	Snapshot ensemble methods .....	37
3.4	Aerial image datasets .....	37
3.4.1	UC merced land use (UCM) dataset .....	37
3.4.2	Aerial image dataset (AID) .....	38
3.4.3	EcocropsAID dataset .....	39
3.5	Experimental results and discussion.....	41
3.5.1	Evaluation metrics .....	42
3.5.2	Training setting .....	43
3.5.3	Classification results on the UCM dataset .....	46
3.4.4	Classification results on the AID dataset .....	50
3.4.5	Classification results on the EcocropsAID dataset .....	55
3.6	Discussion .....	57
3.6.1	Loss error curve of the different learning rate schedules .....	57
3.6.2	Cosine Cyclic Learning Rate Schedule with Max and Min Values .....	59
3.7	Conclusion and future work .....	59
Chapter 4 Instance Segmentation.....		62
4.1	Introduction.....	62
4.2	Mask R-CNN Architecture .....	64
4.2.1	Backbone Architecture .....	65
4.2.2	Head Architecture .....	66
4.3	Aerial Image Water Resources Dataset .....	66
4.4	Experiment and discussion .....	68
4.4.1	Model Evaluation.....	70
4.4.2	Result of Instance Segmentation of Water Bodies .....	71

4.5 Conclusion .....	74
Chapter 5 Discussion .....	75
5.1 Answers to the Research Questions .....	76
5.2 Future work.....	79
REFERENCES .....	81
BIOGRAPHY .....	95



## Table

	<b>Page</b>
Table 1 The brief detail of the aerial image datasets that used for land use and land cover classification.....	10
Table 2 The best performances of the convolutional neural network architectures on the EcoCropsAID dataset. ....	19
Table 3 Test accuracy (%) of the CNN architectures and data augmentation techniques on the EcoCropsAID dataset. ....	22
Table 4 Performances of the ensemble CNN methods on the EcoCropsAID dataset.	23
Table 5 The summary of three aerial image datasets: UCM, AID, ECOcropsAID..	41
Table 6 Summarize the hyperparameter settings of the CNN model and snapshot ensemble methods. ....	45
Table 7 Classification performances (LD, mean validation accuracy, standard deviation and test accuracy) of the snapshot ensemble CNN using different learning rate schedules: CCA, MMCCLP, dropCyclic and training with different state-of-the-art CNNs: MobileNetV2, VGG16, VGG19, on the UCM dataset. ....	48
Table 8 The performance comparison of the snapshot ensemble CNN using the proposed dropCyclic learning rate schedule with existing techniques on the UCM dataset. ....	50
Table 9 Classification performances of the snapshot ensemble CNN using different learning rate schedules and training with different state-of-the-art CNNs on the AID dataset. ....	52
Table 10 The performance comparison of the snapshot ensemble CNN using the proposed dropCyclic learning rate schedule with existing techniques on the AID dataset. ....	54
Table 11 Classification performances of the snapshot ensemble CNN using different learning rate schedules and training with different state-of-the-art CNNs on the EcoCropsAID dataset.....	55
Table 12 The result of the experiment using mask R-CNN with the AIWR Dataset.	72

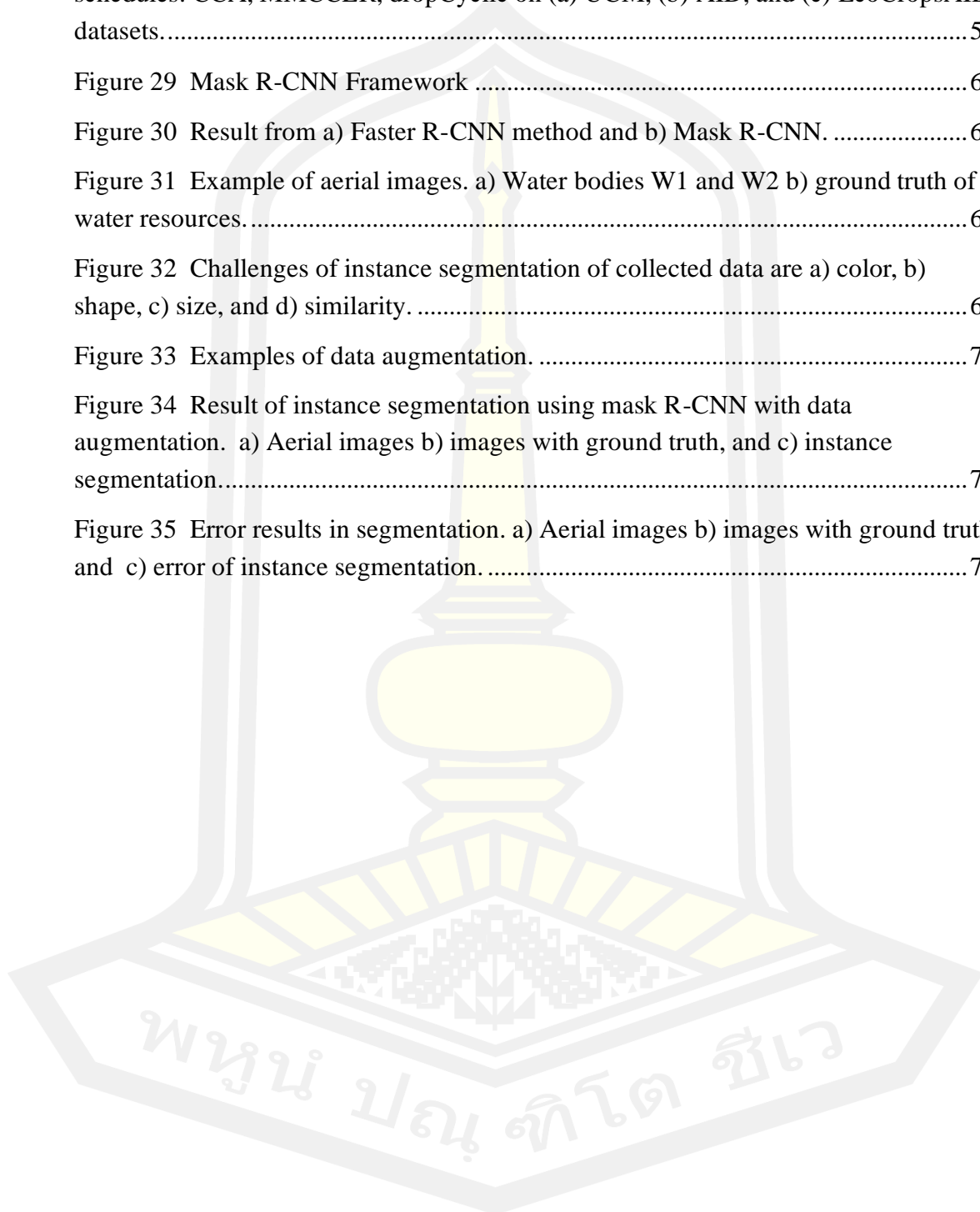
## Figure

## Page

Figure 1 Illustration of land use changes that were captured on (a) April 20, 2012, (b) January 05, 2014, and (c) June 21, 2012. ....	1
Figure 2 The framework of ensemble convolutional neural networks for land use classification in economic crop aerial images. ....	12
Figure 3 Network architectures of VGG16 and VGG19 (Bold). ....	13
Figure 4 Illustration of a) the normal cell and b) the reduction cell, which is generated by the RNN method (Zoph et al., 2018). ....	14
Figure 5 Example of data augmentation techniques. a) Original image, b) rotation, c) width shift, d) height shift, and e) combination between rotation and width shift. ....	15
Figure 6 Example of economic crops aerial images. a) Cassava, b) Longan, c) Rice, d) Rubber, and e) Sugarcane. ....	17
Figure 7 The of confusion matrix of a) VGG16, b) VGG19, and c) NASNet models. ....	20
Figure 8 Visualization of three different feature maps, taken from the VGG16 (block1_conv2, block3_conv3, block5_conv3). Blue pixels activate a unit, red pixels decrease the activation. ....	21
Figure 9 Illustration the confusion matrix of (a) the VGG16 architecture when using the data augmentation technique with the height shift technique and (b) the ensemble CNN with the weight average method on the EcoCropsAID dataset. ....	24
Figure 10 The results indicate that (a) several samples of (~29) cassava class are misclassified as sugarcane class and (b) (~13) rice class are misclassified as cassava class. ....	25
Figure 11 Illustration of the cyclic cosine annealing curve training with 100 epochs and using five cycles. ....	35
Figure 12 Illustration of the step decay schedule. ....	36
Figure 13 Examples of the UCM dataset: a) agricultural, b) beach, c) buildings, d) river, and e) tennis court. ....	38
Figure 14 Illustration of the (a) sparse residential, (b) medium residential, and (c) dense residential. ....	38

Figure 15 Example aerial images of the AID dataset: a) airport, b) bridge, c) desert, d) pond, and e) storage tanks.....	39
Figure 16 Examples of the EcoCropsAID dataset: a) cassava, b) sugarcane c) longan d) rubber, and e) rice. ....	40
Figure 17 Illustration of the similarity patterns between two classes: (a) longan and (b) rubber and (c) cassava and (d) rice, of the EcoCropsAID dataset. ....	40
Figure 18 Illustration of the diversity in the patterns of the EcoCropsAID dataset. Examples of (a) longan and (b) rubber images. ....	41
Figure 19 Illustration of the dropCyclic learning rate curve when the parameters were set as; M=5 cycles (20 epochs per cycle), drop=1.0,0.95,0.85,0.75,0.65,0.50 and , (a) c=5, (b) c=10, (c) c=15. ....	44
Figure 20 The validation accuracy of the snapshot ensemble CNN using MobileNetV2 and the dropCyclic method as a learning rate schedule with different learning rate values (0~0.001) on the UCM dataset. ....	46
Figure 21 Loss error (%) of snapshot ensemble CNN with different learning rate schedule methods: CCA (first column), MMCCLR (second column), dropCyclic (third column) and CNN architectures: (a) MobileNetV2 (b) VGG16, (c) VGG19, on the UCM dataset. Each snapshot ensemble CNN was trained with M=5 cycles. ....	47
Figure 22 The confusion matrix of residential classes: sparse, medium, dense that has a similar pattern. ....	49
Figure 23 Loss error (%) of snapshot ensemble CNN with different learning rate schedule methods: CCA (first column), MMCCLR (second column), dropCyclic (third column) and CNN architectures: (a) MobileNetV2 (b) VGG16, (c) VGG19, on the AID dataset. Each snapshot ensemble CNN was trained with M=5 cycles. ....	51
Figure 24 Class distribution of the AID dataset. ....	53
Figure 25 Illustration of the ROC curve for snapshot ensemble CNN models. The highlighted area is zoomed in at the upper left area of the curve.....	54
Figure 26 Illustrated loss error (%) of snapshot ensemble CNN with different learning rate schedule methods: CCA (first column), MMCCLR (second column), dropCyclic (third column) and CNN architectures: (a) MobileNetV2 (b) VGG16, (c) VGG19, on the EcoCropsAID dataset. Each snapshot ensemble CNN was trained with M=5 cycles. ....	56
Figure 27 The confusion matrix of the snapshot ensemble CNN based on dropCyclic and MobileNetV2 on the EcoCropsAID dataset. ....	57

Figure 28 Illustration of the training loss values when training with the snapshot ensemble CNN using the MobileNetV2 and evaluated with various learning rate schedules: CCA, MMCCLR, dropCyclic on (a) UCM, (b) AID, and (c) EcoCropsAID datasets.....	58
Figure 29 Mask R-CNN Framework .....	65
Figure 30 Result from a) Faster R-CNN method and b) Mask R-CNN. ....	65
Figure 31 Example of aerial images. a) Water bodies W1 and W2 b) ground truth of water resources.....	67
Figure 32 Challenges of instance segmentation of collected data are a) color, b) shape, c) size, and d) similarity. ....	69
Figure 33 Examples of data augmentation. ....	70
Figure 34 Result of instance segmentation using mask R-CNN with data augmentation. a) Aerial images b) images with ground truth, and c) instance segmentation.....	73
Figure 35 Error results in segmentation. a) Aerial images b) images with ground truth and c) error of instance segmentation. ....	74



## Acronyms

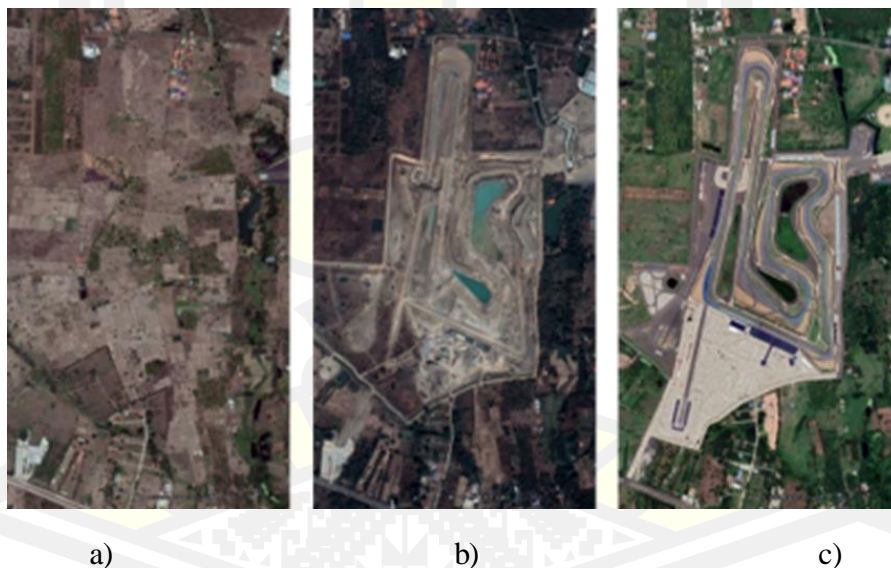
AID	Aerial Image Dataset
AIWR	Aerial Image Water Resources Dataset
CCLR	Cosine Cyclic Learning Rate
CNN	Convolutional Neural Network
dropCyclicLR	Drop Cosine Cyclic Learning Rate Schedule
DSM	Digital Surface Model
EcoCropsAID	Economic Crops Aerial Image Dataset
EWLoss	Edges Weighting Loss
GIS	Information Systems
IoU	Intersection Over Union
LCM	Land Change Modeler
LogLR Test	Learning Rate Range Estimation Method
LULC	Land Use and Land Cover
mAP	Mean Average Precision
Mask R-CNN	Mask Region-Based Convolutional Neural Network
MLP	Multilayer Perceptron
MMCCLR	Max-Min Cosine Cyclic Learning Rate Scheduler
POIs	Points of Interest
ROI	Region of Interest
RPN	Region Proposal Network
RRF DeconvNet	Restricted Receptive Field Deconvolution Network
RS	Remote Sensing
SEL	Snapshot Ensemble Learning Rate
UAV	Unmanned Aerial Image
UCM	UC Merced Land Use Dataset
VHR	High-Resolution Satellite Imagery
W1	Natural Water Bodies
W2	Artificial Water Bodies



# Chapter 1

## Introduction

In the last ten years, the land use and land cover have changed so much (Rujoiu-Mare & Mihai, 2016), so researchers are interested in researching this domain (Caldas, Goodin, Sherwood, Campos Krauer, & Wisely, 2015). Examples of the change of land use and land cover are shown in Figure 1. It shows the rice area changed to the Chang International Circuit, Buriram province. I captured the land use between 2012 to 2020 using the google earth program that changed from crop field to race track. Figure 1a), I captured the aerial image on April 20, 2012. It was a rice area. Figure 1b) captured on January 5, 2014. The construction of the race track. Figure 1c) captured on June 21, 2020. The land use changed to the race track.



**Figure 1** Illustration of land use changes that were captured on (a) April 20, 2012, (b) January 05, 2014, and (c) June 21, 2012.

Suppose I have an accurate and up-to-date knowledge of land use and land cover. In that case, I will be able to apply the information of land use and land cover to many domains, such as urban planning, environmental monitoring and assessment, national policy formulation (Treitz & Rogan, 2004), biodiversity studies, climate change models, climate change, and the design and review of land-use policies (Almeida et al., 2016).



Numerous studies have investigated land use and land cover analysis to plan and manage the quality of life (Mishra, Rai, & Rai, 2020). For example, In Ho Chi Minh City, Vietnam, Due to environmental problems, such as rising urban temperatures, air pollution, and flooding, Schaefer & Thinh (2019) created systems for land use and land cover using geographic information systems (GIS) and remote sensing (RS) techniques (Schaefer & Thinh, 2019). The GIS and RS are also proposed to address the problems of the changing environment in the Halda river caused by the rapid development of urbanization (Chowdhury, Hasan, & Abdullah-Al-Mamun, 2020), as well as, people in the Nile river, Egypt, lack basic security caused by unplanned urban expansion (Elagouz, Abou-Shleel, Belal, & El-Mohandes, 2020). Elagouz et al. (2020) analyzes the information on land use and land cover using GIS and RS techniques to monitor the people activity that may affect the land use and land cover, such as the agricultural use, expansion of urban areas, and land use in the overlapping areas (Elagouz et al., 2020) (Rujoiu-Mare & Mihai, 2016).

In recent years, the deep learning method has been more popular in land use and land cover classification, such as the classification of agricultural land cover (cultivation, water resources, grassland, and wood areas) on Landsat 5/7 images (Storie & Henry, 2018). (Kussul, Lavreniuk, Skakun, & Shelestov, 2017) used a convolutional neural network (CNN) including 1-D and 2-D CNN architectures to classify land cover and crop types on the Landsat-8 and Sentinel-1A satellite images. (Deepan, 2019) proposed the ensemble CNNs architecture. The ensemble CNNs contained three CNN architectures; CNN, VGG-16, and Inception-ResNet. The output probabilities of each CNN were then sent to the unweighted average ensemble method to predict the output. (Al-Najjar et al., 2019) proposed the CNN architectures to classify land cover on the unmanned aerial image (UAV) and digital surface model (DSM) datasets. In their approach, the UAV and DSM images were divided into small patches. The small patches were sent to the CNN model to extract the features. Both features from UAV and DSM images were fused, called fused features. The CNN model also classified the patches into seven classes; waterbody, vegetable/tree, grassland, shadow, paved road, building, and bare land.

The main objective of this thesis is to propose land use and land cover classification and segmentation systems using deep learning methods. This thesis will create results that improve GIS which can then be used for better monitoring the change of land use and cover, and water management in urban and agricultural areas.

## 1.1 Research Question

In Thailand, land use and land cover are changing all the time. During the year 2011-2013, the computation of the land change modeler (LCM) found that the number of human, aquatic breeding grounds, and forest areas were increased. On the other hand, agricultural areas and water resources were decreased (Suwanlertcharoen, Prukpitikul, Buakaew, & Kaewpoo, 2013). Therefore, if I can analyse the change of land use and land cover quickly and accurately, I will plan to manage the land use and land cover more effectively.

**RQ1:** In Thailand, the cultivation of economic crops, including rice, sugarcane, cassava, rubber, and longan, throughout the country. It is not simple to analyze the area of the economic crops, because I need the expert in geographic information systems (GIS) and remote sensing (RS) areas to analyze the economic crops area. It requires information from other government sections to compute and evaluate. It requires more time to collect the relevant information from all sections and incident delays in data analysis by experts will occur. From this issue, how can I reduce the waiting time by collecting all information from government sections? Could convolutional neural networks (CNNs), a type of deep learning method, classify the areas of economic crops from the aerial images? If possible, applying the economic crops classification will help the planner, the policy-maker monitor the land use and land cover of the economic crops faster. In addition, can the ensemble learning method improve the land use and land cover classification performance?

**RQ2:** Many types of ensemble learning methods, such as bagging ensemble (Duin & Tax, 2005), boosting ensemble (Choi et al., 2019), stacked ensemble (R. Sun, 2019), and snapshot ensemble (Bunrit, Kerdprasop, & Kerdprasop, 2019), are proposed to solve the image classification problem. Furthermore, the snapshot ensemble CNN method was designed to find the optimal CNN models. The cyclic cosine annealing method was proposed to decrease the learning rate value while training the CNN model. It made the learning loss of the CNN drop very fast compared to the other CNN methods. Consequently, the outputs of optimal models were then combined and given to the weighted ensemble learning method to predict the output. Then, can the snapshot ensemble CNN method improve the performance of the economic crops classification on the aerial images? I am also concerned with the learning algorithm that proposes to decrease the learning loss and also increase the performance of the snapshot ensemble CNN.

**RQ3:** When using the LCM model to calculate land use, I found that the water resources were reduced each year (Suwanlertcharoen et al., 2013). It may affect agriculture because water resources are essential for cultivation. To find the water resources areas, the expert always uses geographic information systems (GIS) and remote sensing systems to analyze the areas from the high-resolution satellite image. It spent much computation time and more money as well. Importantly, is it possible to employ the deep learning method to find the water resources from the aerial images? If possible, it will reduce the cost of computation. Also, everyone can download aerial images from the general application, such as Google maps and analyze the water resources using the deep learning method. Furthermore, if it is accurate, I can also analyze the amount of water and plan to manage water usage.

In order to answer all of these questions (RQ1 to RQ3), Chapter 2 to Chapter 4 describe the research done in this thesis. Finally, Chapter 5 provides concrete answers to research questions.

## 1.2 The Objective of This Dissertation

This study will focus on three detailed objectives:

- 1) I aim to classify land use and land cover of five economic crops, including rice, sugarcane, cassava, rubber, and longan, from the aerial images using convolutional neural network (CNN) methods. I also enhance the performance of the land use classification method using ensemble learning methods.
- 2) I proposed to use the snapshot ensemble CNN technique to improve the performance of the land use classification from the aerial images.
- 3) I propose the instance segmentation technique to segment the water body from the aerial images.

## 1.3 Contributions

The main contribution of this thesis is a novel method for the learning rate schedule, which is appropriate for land use classification on aerial images. The new learning method is designed to decrease the training loss while training the convolutional neural network (CNN) models. I also combined the new learning method with the snapshot ensemble CNN technique. I performed experiments on four aerial image datasets consisting of UCM (Y. Yang &

Newsam, 2010), AID (G. S. Xia et al., 2017), EcoCropsAID (Noppitak & Surinta, 2021), and AIWR (Noppitak, Gonwirat, & Surinta, 2020) datasets. I perform the ensemble convolutional network architectures for land use classification. In the ensemble method, the grid-search method is proposed to optimize the weighted parameters of the CNN models.

I apply the mask R-CNN technique to segment the water body from aerial images in the segmentation task. The mask R-CNN technique includes two parts; CNN backbone architecture and network head. In this thesis, I use the ResNet-101 for extracting the feature from the whole image. In the training process, I also use the data augmentation technique to increase the amount of training data while training, including affine image transformations and color modification.

Additionally, I present two new aerial image datasets, including EcoCropsAID and AIWR datasets, for classification and segmentation tasks, respectively. The EcoCropsAID dataset describes data collected of the economic crops area of Thailand, which includes five categories; rice, sugarcane, cassava, rubber, and longan. The AIWR dataset represents the area of the water body. It comprises two types of data; natural and artificial water bodies. The contributions of the thesis are as follows.

Chapter 2 proposed an ensemble convolutional neural networks method for land use classification, namely the ensemble CNN method. In addition, a new aerial image dataset, namely the EcoCropsAID dataset, is presented for the classification problem. This dataset contains 5,400 aerial images of five categories; rice, sugarcane, cassava, rubber, and longan. I collect the aerial images between the years 2014 and 2018 using the Google Earth program. In the experiments, first, I discovered the robust CNN models from 8 state-of-the-art architectures, including InceptionResNetV2, MobileNetV2, DenseNet201, Xception, ResNet152V2, NASNetLarge, VGG19, and VGG16. Second, I chose only three CNN architectures (NASNetLarge, VGG19, and VGG16) that provided better performance. Third, three data augmentation techniques were applied while training the CNN models, including rotation, width shift, and height shift. I found that training the CNN models with the data augmentation techniques yielded high accuracy performance. Fourth, I created the ensemble CNN models that combined three CNN models with the data augmentation techniques. Finally, the results from each CNN model were computed using three ensemble learning methods; unweighted majority vote, unweighted average, and weighted average. The experimental results showed

that when using the weighted average learning method, the ensemble CNN outperformed other ensemble learning methods. This chapter is based on the following publication:

Noppitak, S., and Surinta, O. (2021). Ensemble convolutional neural network architectures for land use classification in economic crops aerial images. *ICIC Express Letters*, 15(6), 531–543. doi: 10.24507/icicel.15.06.531

In chapter 3, a new learning rate schedule method is proposed to find the optimal range of learning rates in the snapshot ensemble convolutional neural network (CNN) method, called the dropCyclic learning rate schedule. The dropCyclic allows training the CNN model from the new starting point of the learning rate in each cycle. That means the starting point of the learning rate value in each cycle is decreased depending on the drop parameter. The benefit of dropCyclic is that it has fewer control parameters. Moreover, it does not need to adjust the upper and lower boundary of the learning rate. I perform the dropCyclic learning rate schedule on three aerial image datasets, including UCM, AID, and EcoCropsAID. I also compare the proposed dropCyclic learning rate schedule with two state-of-the-art learning rate schedules; cyclic cosine annealing and max-min cosine cyclic learning rate scheduler. The experimental results showed that the proposed learning rate schedule outperformed two state-of-the-art learning rate schedules on the UCM dataset.

In chapter 4, I proposed to use the mask region-based CNN technique, namely mask R-CNN, to segment the water body from aerial images. The mask R-CNN contained two architectures consisting of CNN backbone and head architectures. In my experiment, I attached the ResNet-101 architecture as for the CNN backbone architecture. The backbone architecture was used to extract the robust feature from the aerial images. In the head network, the fully connected layers were attached to the ResNet-101. The head architecture was designed to predict the region of interest (ROI) of the water body. Subsequently, I performed two data augmentation techniques: affine image transformations and color modification while training the mask R-CNN. The performance of the mask R-CNN combined with data augmentation techniques increased almost two times compared to the mask R-CNN without using data augmentation techniques. This chapter is based on the following publication:

Noppitak, S., Gonwirat, S., & Surinta, O. (2020). Instance segmentation of water body from aerial image using mask region-based convolutional neural network. *Proceedings of the 3rd International Conference on Information Science and System (ICISS)*, 61–66. doi: 10.1145/3388176.3388184.

## Chapter 2

### Ensemble Convolutional Neural Network

The analysis of land use and land cover is a task of remote sensing and geographic information systems. Nowadays, deep learning techniques can analyze land use and land cover with high performance. In this paper, I focus on the classification of land use for Thailand's economic crops based on the convolutional neural network (CNN) technique. I evaluated the ensemble CNN framework on Thailand's economic crops aerial image dataset called the EcoCropsAID dataset. Five economic crop categories, were rice, sugarcane, cassava, rubber, and longan, and images were collected using the Google Earth program. Economic crops aerial images obtained between 2014 and 2018 were considered. There were 5,400 images with approximately 1,000 images per class. Due to the ensemble CNN framework, I first proposed to use eight pre-trained CNN models consisting of InceptionResNetV2, MobileNetV2, DenseNet201, Xception, ResNet152V2, NasNetLarge, VGG16, and VGG19 to discover the best baseline CNN model. Second, three simplistic data augmentation techniques (rotation, width shift, and height shift) are applied to increase the accuracy of the CNN models. Finally, I created an ensemble CNN framework that consisted of 3 CNNs based on the best CNN models. I also compared three ensemble methods, that were weighted average, unweighted average, and unweighted majority vote.

#### 2.1 Introduction

Thailand is a country that mainly exports agricultural products that are economic crops, including rice, corn, cassava, sugar, rubber, palm oil, tapioca, and longan (Office of Agricultural Economics, 2019b). Hence, the government sector has to analyze the information and forecast the world economy, especially in agriculture, which requires the consideration of many factors outside the country, such as the world agricultural economy, crude oil price, etc. The domestic factors include the amount of water in reservoirs, rainfall, land use, etc. (Office of Agricultural Economics, 2019a). Without appropriate planning of land use, negative consequences might ensue, such as selection of the wrong plant products and quantities inconsistent with export to international markets.



Land use and land cover can be interrogated by remote sensing and geographic information system (GIS) tasks that can be used to analyze, plan, and manage the quality of human life in respect of many issues. For example, the rapidly growing population causes environmental problems, air pollution, and temperature rise in urban areas (Mishra et al., 2020). Since the flooding in Ho Chi Minh City, Vietnam, Schaefer and Thinh (Schaefer & Thinh, 2019) proposed using GIS and RS methodologies to evaluate the changes of land cover and agricultural protection sites. In the future, planners can use the proposed methods in decision support for managing land use. To address the problems caused by the rapid urbanization of human activities in the Halda river located in the south-eastern region of Bangladesh, Chowdhury et al. (Chowdhury et al., 2020) used remote sensing and GIS to assess land use and land cover changes in the Halda watershed. Populations in the Nile Delta, Egypt, require basic security because of the unplanned urban growth (Elagouz et al., 2020) and proposed a remote sensing technique to estimate the land use change according to the unplanned urban growth by monitoring the human activities that change in agricultural and urban areas. Consequently, remote sensing and GIS are used to classify land use in problematic regions due to the landscape (i.e., hills and lowlands) (Rujoiu-Mare & Mihai, 2016).

Nowadays, deep learning is a well-known technique proposed to address land use and land cover classification. The techniques are widely used in land use classification, for example of buildings, paved roads, vegetation density, grassland, and water bodies (Al-Najjar et al., 2019). Also, urban planning and management use deep learning models to classify land use in urban areas using high-resolution satellite imagery (VHR) (P. Zhang et al., 2018). Al-Najjar et al. (Al-Najjar et al., 2019) presented a land cover classification method to analyze an image from an unmanned aerial vehicle (UAV). C. Zhang et al. (C. Zhang et al., 2019) proposed a joint deep learning model that compounded multilayer perceptron (MLP) and convolutional neural networks (CNNs) to classify land cover and land use.

In this chapter, I propose an ensemble convolutional neural network framework, called ensemble CNN, for classification of land use in economic crop aerial images. The contributions of this paper can be summarized as follows:

- 1) I propose ensemble methods with deep convolutional neural networks, called the ensemble CNN method for land use classification. Using the ensemble method, I present the weighted average approach to find the optimal weight by applying weights to the output probabilities of each baseline CNN model. Based on the CNN architecture, I discover further

efficient models by employing pre-trained CNN models from eight CNN architectures, including InceptionResNetV2, MobileNetV2, DenseNet201, Xception, ResNet152V2, NasNetLarge, VGG16, and VGG19. My experimental results indicate that the three best CNN models on the economic crops aerial image dataset are VGG16, VGG19, and NASNET, respectively. I also realize that the simplistic data augmentation techniques, such as rotation and height shift techniques, could improve the performance of the CNN method. I have demonstrated that the data augmentation technique significantly increases the performance of the CNN model. Note that the learning process can be computationally expensive.

2) This thesis aims to enable the use of Thailand's economic crops aerial image dataset, namely the EcoCropsAID dataset, for land use classification. In this thesis, I collect the aerial image data between 2014 and 2018 by using the Google Earth program. The image quality is different depending on the different remote sensor types used by the Google Earth program; importantly, the aerial image quality is different. The EcoCropsAID dataset consists of 5,400 images that contain five categories; rice, sugarcane, cassava, rubber, and longan. Also, each category comprises approximately 1,000 images.

## **2.2 Related Work**

This research focuses on the land use classification on aerial images using deep learning algorithms. Many studies in remote sensing and geoinformatics have mainly experimented on satellite images. In this paper, however, we experiment with aerial images collected in RGB color space. I survey aerial image datasets that have been used for land use and land cover classification tasks. The datasets, such as UC Merced land use (Y. Yang & Newsam, 2010), RESISC45 (Cheng, Han, & Lu, 2017), and AID (G. S. Xia et al., 2017) datasets, were created from the Google Earth program, except the EuroSAT dataset (Helber, Bischke, Dengel, & Borth, 2019) that was collected from the Sentinel-2 satellite. I collected the aerial image data from the Google Earth program that considers only five economic crops consisting of rice, sugarcane, cassava, rubber, and longan. The information on I proposed aerial image dataset is presented in Section 2.4 The details of the aerial image datasets are described in Table 1.



**Table 1** The brief detail of the aerial image datasets that used for land use and land cover classification.

Datasets	Classes	No. of Images	Image Size	Resolution and Source	Year
UC Merced Land Use (Y. Yang & Newsam, 2010)	21	2,100 (100 per class)	256x256	0.3 m. <i>Google Earth</i>	2010
RESISC45 (Cheng et al., 2017)	45	31,500 (700 per class)	256x256	30 to 0.2 m. <i>Google Earth</i>	2017
AID (G. S. Xia et al., 2017)	30	10,000 (220 to 420 per class)	600x600	8 to about 0.5 m. <i>Google Earth</i>	2017
EuroSAT (Helber et al., 2019)	10	27,000 (2,000 to 3,000 per class)	224x224	10 m. <i>Sentinel-2 (spectral data with 13 bands)</i>	2019
Proposed dataset	5	5,400 (~1,000 per class)	600x600	30 to 0.2 m. <i>Google Earth</i>	2020

Many deep learning techniques, such as auto-encoders, stacked auto-encoders, restricted Boltzmann machine, deep belief network, and CNN architectures, are proposed to address the land use and land cover classification of the satellite images. For land use and land cover classification, however, CNN architectures comprise AlexNet, CaffeNet, GoogleNet, VGGNet, PlacesNet, Inception, and ResNet become the state-of-the-art architectures (Vaishnav, Devi, & Srinivasan, 2019).

The CNN architectures are proposed to extract the deep features from the aerial image. Xia et al. (G. S. Xia et al., 2017) proposed a benchmark aerial image dataset, namely AID, for aerial scene classification. The AID dataset contains 10,000 images and 30 categories. They then applied CaffeNet, VGG-VD-16, and GoogLeNet to extract the deep features and classify them using the Liblinear supervised classification. These methods achieved around 86% accuracy on the AID dataset. Pilipovic and Risojevic [16] evaluated three CNN architectures; GoogLeNet, ResNet, and SqueezeNet, on the high-resolution remote sensing dataset. The deep features were extracted using fine-tuned CNN models and presented to the support vector machine (SVM) method as a classifier. The results showed that the GoogLeNet fine-tuned features combined with the SVM outperformed the other methods on the UCM dataset.

Additionally, the ResNet fine-tuned features showed the best result on the AID dataset (Pilipovic & Risojevic, 2017).

Han et al. proposed a framework that combines the deep feature and discriminative evaluation methods for scene classification and annotation, called the semi-supervised generative framework (SSGF). To evaluate the performance, when considering only the deep feature method, the ResNet model combined with supervised learning outperforms all CNN models on the AID, UCM, NWPU-RESISC45, and WHU-RS19 datasets. The SSGF method that combined ResNet, VGG-S, and VGG-16 still achieved the best accuracy result on all datasets.

For aerial scene classification, In the work of (Zheng, Yuan, & Lu, 2019) propose a deep scene representation approach. In this approach, deep features are extracted by the multi-scale max-pooling method and then given to the Fisher vector method to encode the multi-scale features into a global representation. Various pre-trained CNN models were evaluated, including AlexNet, CaffeNet, GoogLeNet, and VGGNet, on different aerial scene datasets. This approach achieved an accuracy above 93% on UCM, WHU-RS19, RSSCN7, and AID datasets.

It can be seen that convolutional neural networks can be employed to address the land use classification.

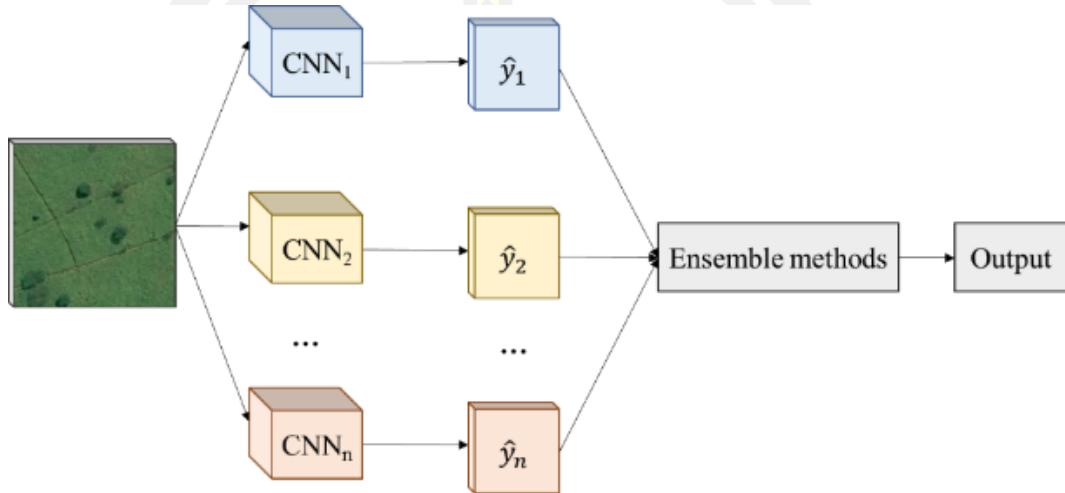
### **2.3 Proposed Ensemble Convolutional Neural Network Architecture**

The ensemble method aims to enhance the accuracy results of the classification tasks. This method combines various classifiers instead of applying only an individual classifier and provides more robust results (Yazdizadeh, Patterson, & Farooq, 2019). Figure 2 illustrates the ensemble convolutional neural network architecture. The proposed architectures consist of two schemes.

In the first scheme, I first discover the best baseline CNN model from several state-of-the-art CNN architectures, including VGGNet, Xception, ResNet, InceptionResNet, MobileNet, DenseNet, and NASNet. Second, the data augmentation techniques are employed to improve the performance of the CNN models. Finally, according to my experiments, I combine the three best CNN models. Then, the probability distribution, which is computed by

the softmax function, is assigned to classify using the ensemble methods. The detail of the first scheme is explained in Sections 2.3.1 and 2.3.2.

The second scheme aimed to compare three ensemble methods; weighted average, unweighted average, and unweight majority vote, regularly applied for neural networks. The ensemble methods are described in Section 2.3.3.

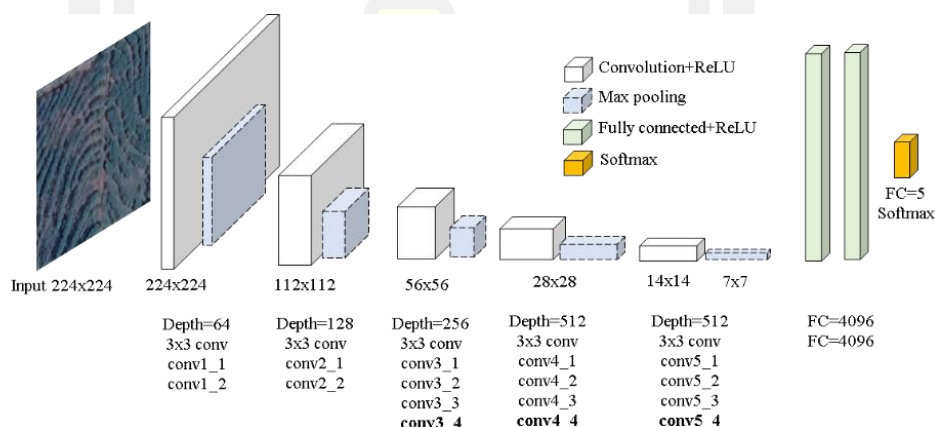


**Figure 2** The framework of ensemble convolutional neural networks for land use classification in economic crop aerial images.

### 2.3.1 Convolutional Neural Network Architectures

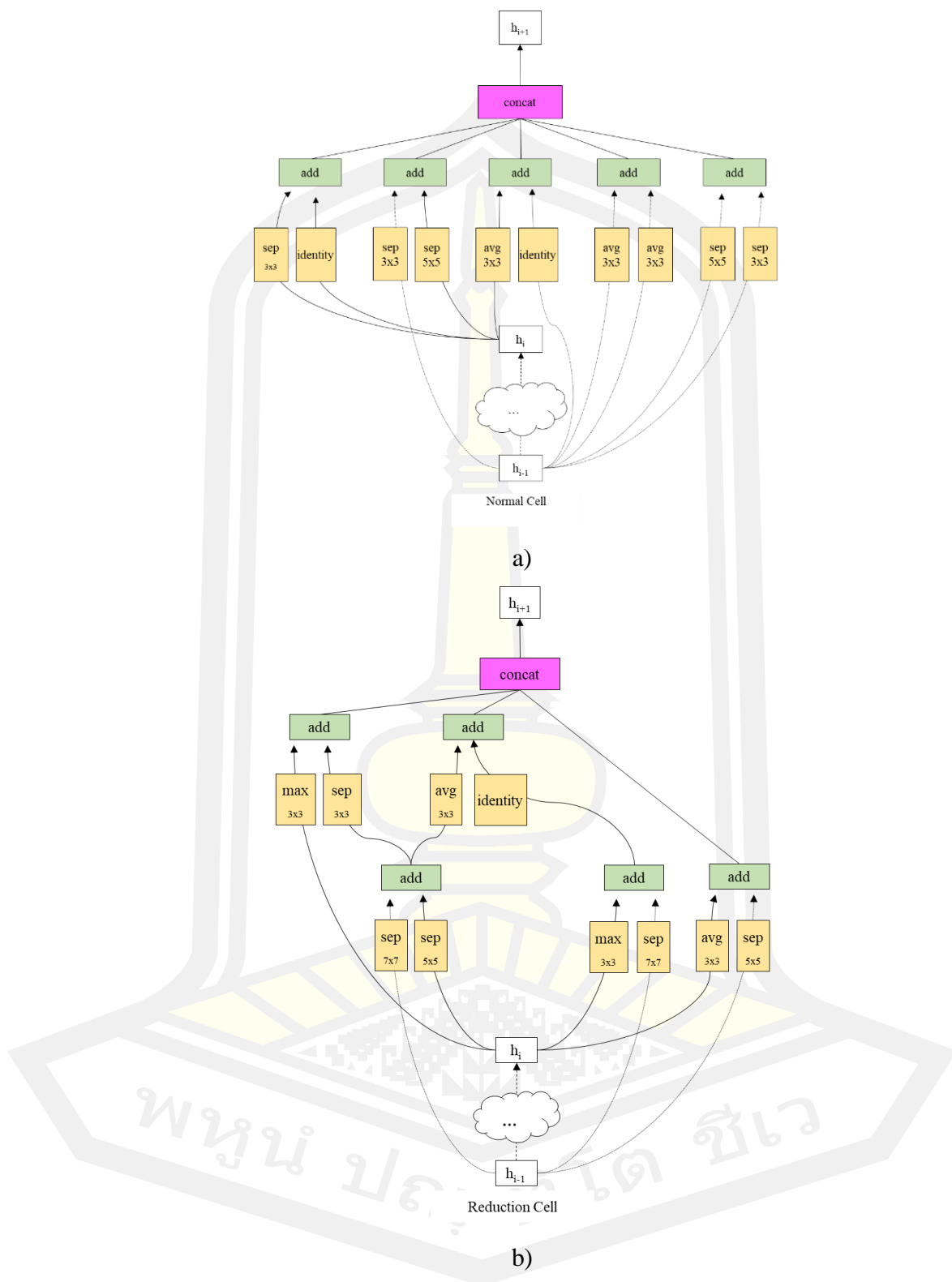
In this study, I propose convolutional neural network (CNN) architectures to address the land use classification problem on Thailand's economic crops aerial image dataset (EcoCropsAID dataset). To find the three most beneficial baseline CNN models, I compare the performance of eight CNN architectures, including Xception (Chollet, 2017), VGG16 (Simonyan & Zisserman, 2014), VGG19 (Simonyan & Zisserman, 2014), ResNet152V2 (He, Zhang, Ren, & Sun, 2016), InceptionResNetV2 (Szegedy, Ioffe, Vanhoucke, & Alemi, 2017), MobileNetV2 (Sandler, Howard, Zhu, Zhmoginov, & Chen, 2018), DenseNet201, and NASNet (Zoph, Vasudevan, Shlens, & Le, 2018). I observe that the VGG16, VGG19, and NASNet architectures achieved high performance on the EcoCropsAID dataset based on my experiments. Data augmentation techniques are also applied. I further describe the baseline CNN architectures and the data augmentation techniques as follows.

**VGGNet:** In 2015, Simonyan and Zisserman invented VGG Networks, namely VGGNets. The VGGNets are designed according to the depth of the weight layers with 16-19 layers. These networks are divided into five convolutional blocks and the max-pooling layer follows each block. In each block, to create the feature maps, the small convolution filters with the size of 3x3 are computed. Then, in each block, the feature maps are reduced by half size of the previous block. In contrast, the feature maps are increased by double layers of the last block. Moreover, the FC layer is employed as the classifier. Additionally, the two fully connected (FC) layers with 4,096 and the one final FC layer are the outputs of the network. According to the EcoCropsAID dataset, in my framework, the final FC layer is designed as five. The architecture of the VGGNets is shown in Figure 3.



**Figure 3** Network architectures of VGG16 and VGG19 (Bold).

**NASNet:** Zoph and Le proposed a neural architecture search (NAS) that generates the CNN architecture using the recurrent neural network (RNN) with reinforcement learning. In 2018, (Zoph et al., 2018) developed the learned transferable architecture by extending the NASNet architecture. In this architecture, the RNN method is employed as the search method to explore the best CNN architecture. Note that the NASNet architecture includes the normal and reduction cells searched by the RNN method (see Figure 4). For the transfer architecture, the best CNN architecture is created based on learning from the small dataset (i.e., the CIFAR-10 dataset). Consequently, the CNN architecture is transferred to learn with a large dataset (i.e., imageNet).



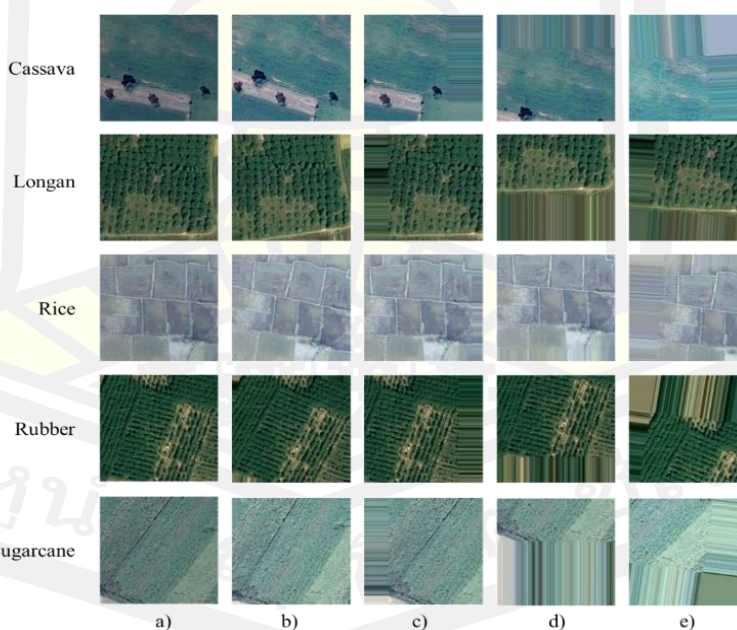
**Figure 4** Illustration of a) the normal cell and b) the reduction cell, which is generated by the RNN method (Zoph et al., 2018).

### 2.3.2 Data Augmentation for Aerial Images

The concept of data augmentation is to solve the problem of having an insufficient amount of data by increasing the number of training data (Pawara, Okafor, Schomaker, & Wiering, 2017). Therefore, the new image is a synthesis from the original images using different augmentation methods, and new diversity images are also generated. Image data augmentation techniques are classified into two main groups; basic image manipulations (i.e., flip, color space, crop, rotation, translation, etc.) and deep learning approaches (i.e., adversarial training, neural style transfer, and meta-learning) (Perez & Wang, 2017); (Shorten & Khoshgoftaar, 2019)).

Due to the aerial image data augmentation, two categories are introduced, including instance-based augmentation (such as geometric, color, deformation, enhancement, brightness, etc.) and fusion-based augmentation (i.e., the RGB channels from the satellite images are combined) (Ghaffar, McKinstry, Maul, & Vu, 2019).

I experiments perform three data augmentation techniques; rotation, width shift, and height shift. The example images of the basic manipulation techniques are shown in Figure 5.



**Figure 5** Example of data augmentation techniques. a) Original image, b) rotation, c) width shift, d) height shift, and e) combination between rotation and width shift.

### 2.3.3 Ensemble Methods

Due to an increment in the classification performance, the output probabilities of the CNN models are combined and classified using the ensemble methods. A brief explanation of the ensemble methods follows.

**Unweighted Average Method:** In this method, the CNN output probabilities of each model, which are computed by the softmax function, are averaged (Ju, Bibaut, & van der Laan, 2018). The highest probability is decided as a result. The output ( $\hat{y}_i$ ) is computed by Equation (1):

$$\hat{y}_i = \frac{1}{n} \sum_{j=1}^n y_j \quad (1)$$

where  $y_j$  is the output probabilities of the CNN model and  $n$  is the number of the CNN model .

**Weighted Average Method:** Due to the classification performance of the CNN model, the different weights are applied to the output probabilities. Hence, the higher weight is assigned to the CNN model that achieved a higher classification rate (Frazão & Alexandre, 2014). The weighted average method is given by:

$$\hat{y}_i = \frac{1}{n} \sum_{j=1}^n \alpha_j y_j \quad (2)$$

where  $\alpha$  is a weight that multiplies with the output probabilities  $y_j$  of the CNN models.

**Unweighted Majority Vote Method:** The Argmax function is applied to the output probabilities of each CNN model and determined as the predicted labels. For each class, the number of votes is counted. Then, the most maximum votes are decided as the final decision (Surinta, Schomaker, & Wiering, 2013).

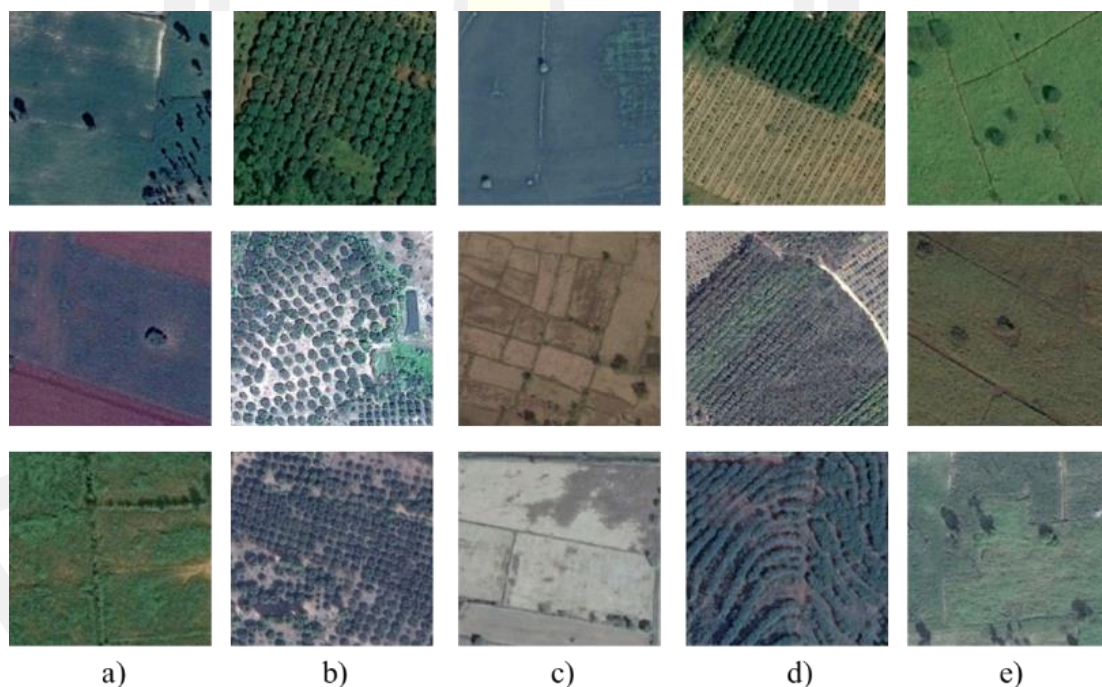
$$\hat{y}_i = \frac{1}{n} \sum_{j=1}^n \operatorname{argmax}_j y_j \quad (3)$$

where argmax is a weight that multiplies with the output probabilities  $y_j$  of the CNN models.



## 2.4 Thailand's Economic Crops Aerial Image Dataset

I introduce the novel economic crops aerial image dataset, namely the EcoCropsAID dataset. This dataset was collected in Thailand from five economic crops that were cultivated in different provinces and regions between 2014 and 2018. The aerial images of economic crops were gathered based on Agri-Map Online provided by the Ministry of Agriculture and Cooperatives and the National Electronics and Computer Technology Center (NECTEC). The Agri-Map Online is an agriculture map that all departments under the Ministry of Agriculture and Cooperatives use as an agriculture management tool. Subsequent agricultural information is accurate and up-to-date (Office of the Permanent Secretary for Ministry of Agriculture and Cooperatives, 2019). Then, the Google Earth application was employed to capture aerial images after I selected the economic crops areas in which images were to be collected. It is quite a complex dataset because the Google Earth program used several remote imaging sensors (G. S. Xia et al., 2017) to record the aerial images.



**Figure 6** Example of economic crops aerial images. a) Cassava, b) Longan, c) Rice, d) Rubber, and e) Sugarcane.



The EcoCropsAID dataset includes five categories (rice, sugarcane, cassava, rubber, and longan) and contains 5,400 images. Each class has around 1,000 images. To prepare the aerial images of the economic crops, I recorded the image with 600x600 pixels and stored it in the RGB color format. Sample aerial images of this dataset are shown in Figure 6. As seen in Figure 6a, the first row is the cassava field, that is at commencement of planting. However, the third row was a change in land use from rice field to cassava field because strange lines have appeared. Also, Figure 6e (first and second rows) shows a change in land use from rice field to sugarcane field. The pattern of the longan area is similar to the rubber area, as shown in Figure 6d in the first and second rows. As seen in Figure 6c, the first row presents a wet area. The second row of Figure 6c illustrates the not yet planting area, while the third row is the beginning of planting.

The challenges of classification on the EcoCropsAID dataset are: 1) many different image resolutions and colors are contained in the EcoCropsAID dataset due to the various remote imaging sensors, 2) the similarity of patterns amongst each class, for example, longan and rubber, 3) the difference of pattern inside the same class, for example, cassava and rice.

## 2.5 Experimental Results

In this section, I present experiments on my economic crops aerial image dataset. All experiments are tested in the same environment. I used the TensorFlow as the deep learning framework that run on Intel(R) Core-i9-9900K CPU @ 3.60GHz x16, 32GB RAM, and GPU NVIDIA GeForce GTX 1080Ti. The experiment results are explained as follows.

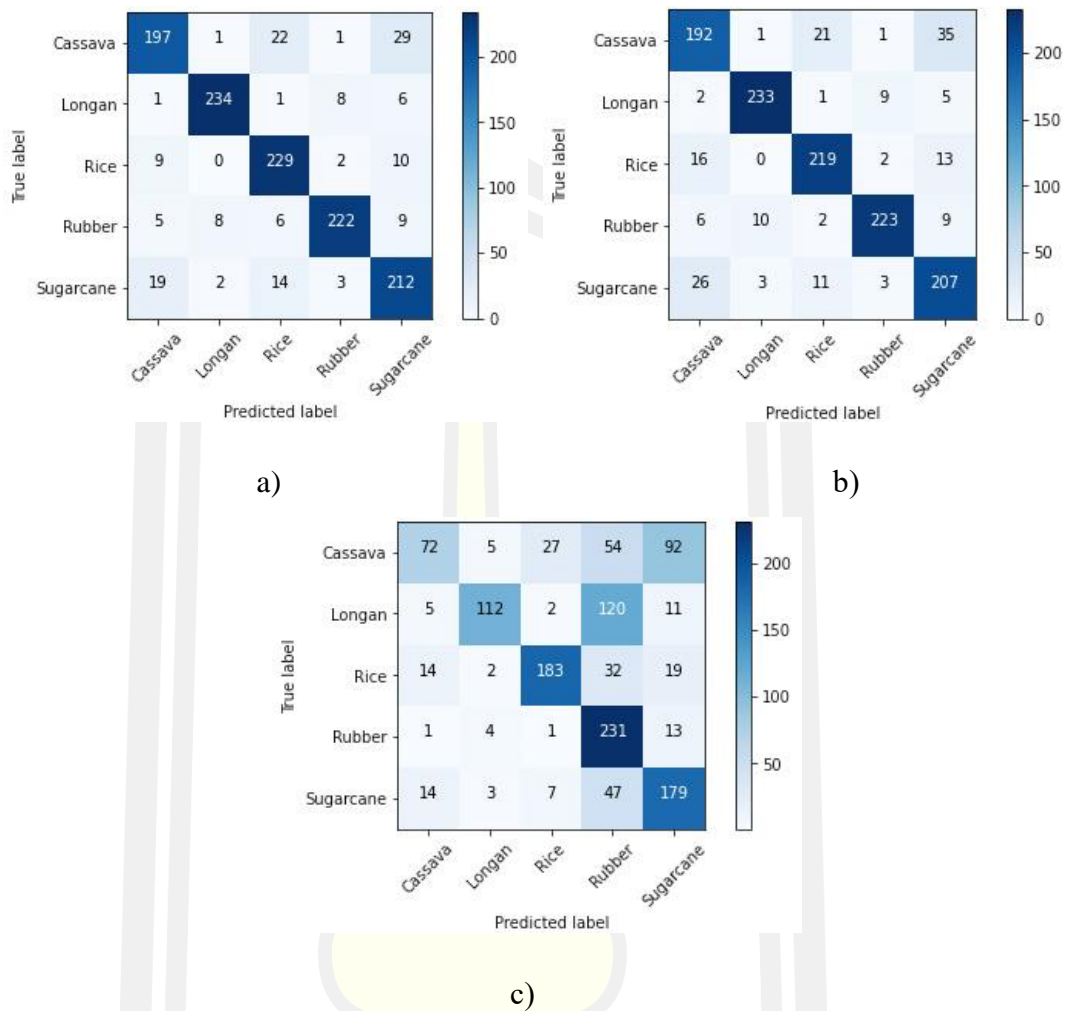
### 2.5.1 Experiments with Convolutional Neural Network Architectures and Data Augmentation Techniques

For the CNN experiments, I will present the best setting parameters of the CNN models on the EcoCropsAID dataset. However, to find the best setting parameters of the CNN models, I randomly select 25% (1,350 images) for the training data and test data. Eight CNN models are selected; Xception, VGG16, VGG19, ResNetV2, InceptionResNetV2, MobileNetV2, DenseNet, and NASNet. I use the transfer learning technique to train eight pre-trained CNN models. I focus on performing two optimization algorithms, including stochastic gradient descent (SGD) and Adam optimizers. The batch size experiments used sizes of 4, 8, 16, 32, and 64. The learning rate and the number of training epochs are determined as 0.001 and 100.

**Table 2** The best performances of the convolutional neural network architectures on the EcoCropsAID dataset.

Models	Optimizers	Batch Size	Accuracy	Training Time	Number of Parameters
InceptionResNetV2	SGD	16	48.00	29min 21s	54,828,261
MobileNetV2	SGD	32	48.40	5min 16s	2,571,589
DenseNet201	SGD	64	50.75	14min 31s	18,792,389
Xception	Adam	16	52.99	21min 8s	21,885,485
ResNet152V2	SGD	64	59.87	19min 25s	58,833,413
NASNetLarge	Adam	8	62.29	1h 21min	87,356,183
VGG19	SGD	64	85.92	12min 21s	20,149,829
VGG16	SGD	16	<b>87.57</b>	11min 38s	14,840,133

Table 2 shows the best convolutional neural network parameters and classification performances achieved from eight CNN models. The experiment results show that the VGGNet performs much better than other CNN models. The VGG16 significantly outperforms the VGG19. Also, VGGNet requires less computation time (it took around 12 minutes). However, based on my experiments, the worst performance with approximately 48% accuracy is the InceptionResNetV2 and MobileNetV2 models. Subsequently, the NASNet model requires more computation time and spends around 1 hour and 20 minutes. When comparing the accuracy between different optimizers and different batch sizes, I found that SGD obtained better results than Adam, except for Xception and NASNetLarge. The Adam optimizer performed better when using a small batch size, while the SGD optimizer gave better experiments when using a large batch size. Considering the results, as shown in Table 3, I selected and performed other experiments based on three CNN models; VGG16, VGG19, and NASNet.

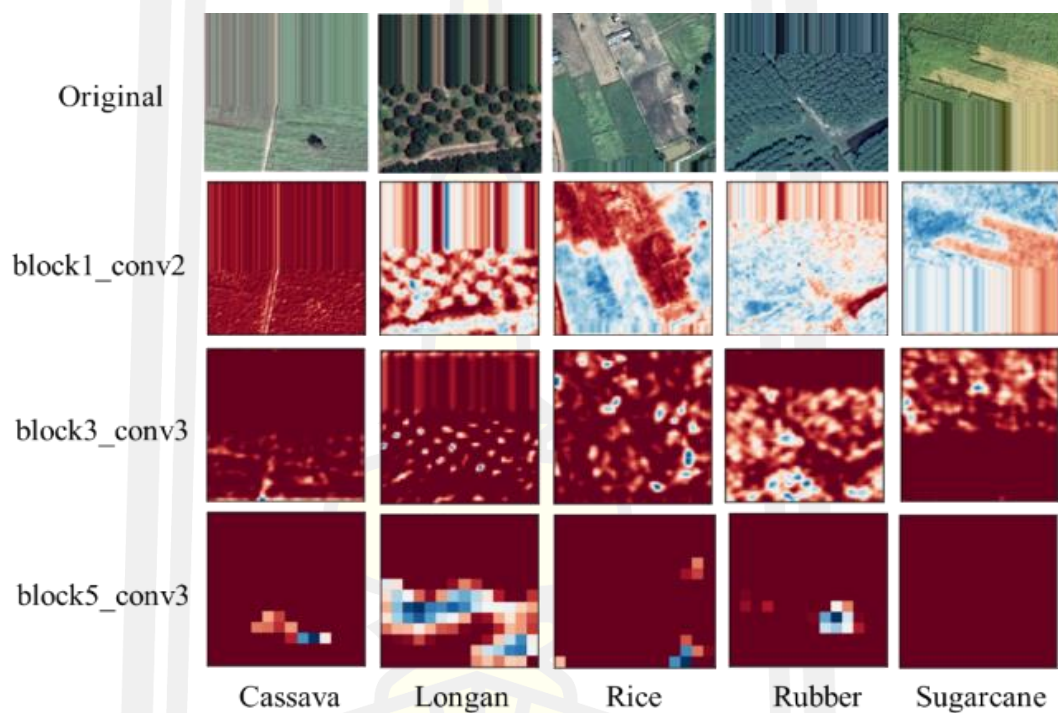


**Figure 7** The of confusion matrix of a) VGG16, b) VGG19, and c) NASNet models.

Figure 7 shows the confusion matrix of VGG16, VGG19, and NASNet models. The confusion matrix shows that the cassava class was misclassified as belonging to the sugarcane class. The VGG16, VGG19, and NASNet misclassified 29, 35, and 92 images, respectively and this is because the pattern of the cassava and sugarcane classes are quite similar.

For data augmentation experiments, the EcoCropsAID dataset is divided into a training set 80% (4,320 images) and a test set 20% (1,080 images). As seen from the results in Table 3, I experimented with three CNN models, VGG16, VGG19, and NASNet, as pre-trained models. I also considered three data augmentation techniques; rotation, width shift, and height shift to increase the performance of the CNN models. These data augmentation techniques do not destroy the aerial image spectral information (X. Yu, Wu, Luo, & Ren, 2017).

I explore the optimal values by configuring the parameters of the data augmentation techniques as follows. The rotation technique parameter is 10, 20, and 30 degrees; width and height shift parameters are 0.1, 0.2, and 0.3 ratios. As a result, the optimal parameter of the rotation is 10-degree, width shift (Wshift) is 0.3 ratio, and height shift (Hshift) is 0.6 ratio. It should be noted that I examine data augmentation techniques by combining two more data augmentation techniques (Pawara, Okafor, Schomaker, & Wiering, 2017); rotation+Wshift, rotation+Hshift, Wshift+ Hshift, and Wshift+Hshift.



**Figure 8** Visualization of three different feature maps, taken from the VGG16 (block1\_conv2, block3\_conv3, block5\_conv3). Blue pixels activate a unit, red pixels decrease the activation.

I also visualize different feature maps of VGG16, which is the best model on the EcoCropsAID dataset, as shown in Figure 8. Three feature map layers are illustrated, including block1\_conv2, block3\_conv3, and block5\_conv3 and the size of the feature maps is 14x14, 14x14, and 512x512 pixels, respectively.

**Table 3** Test accuracy (%) of the CNN architectures and data augmentation techniques on the EcoCropsAID dataset.

Models (Data Augmentation)	Optimizers	Batch Size	Accuracy	Training Time	Number of Parameters
NASNet (Hshift)	Adam	8	59.40	1d 1h 12min 14s	87,356,183
NASNet (Rotation+Hshift)	Adam	8	79.60	1d 1h 29min 53s	87,356,183
VGG19 (Rotation+Hshift)	SGD	64	86.50	6h 23min 47s	20,149,829
VGG19 (Hshift)	SGD	64	88.30	6h 20min 10s	20,149,829
VGG16 (Rotation+Hshift)	SGD	16	<b>91.50</b>	6h 39min 52s	14,840,133
VGG16 (Hshift)	SGD	16	<b>91.50</b>	6h 39min 43s	14,840,133

Table 3 shows the experimental results of the CNN models and data augmentation techniques. The experiments show that the VGG16 (Rotation+Hshift) and VGG19 (Hshift) are the best models with an accuracy of 91.50%. These VGGNets spend around 6 hours and 30 minutes when training. However, The NASNet (Rotation+Hshift) obtains 79.60% accuracy, which is 11.90% lower than the VGGNet models. When training the model, the NASNet requires more than 24 hours. The results also showed that the VGG16 model performed better than VGG19 and NASNet models. It is concluded that the deep layers and number of parameters did not affect the land use classification accuracy.

### 2.5.2 Experiments with Ensemble Methods

In these experiments, I have evaluated the ensemble methods consisting of average, unweight majority vote (UMV), and weight methods. For the weight method, we optimize the weight parameters using the grid-search method. I perform four ensemble CNN methods as follows. 1) NASNet (Rotation+Hshift) + VGG19 (Hshift) + VGG16 (Hshift), called E1 model. 2) VGG16 (Rotation+Hshift) + VGG16 (Rotation+Wshift) + VGG16 (Rotation+Wshift+

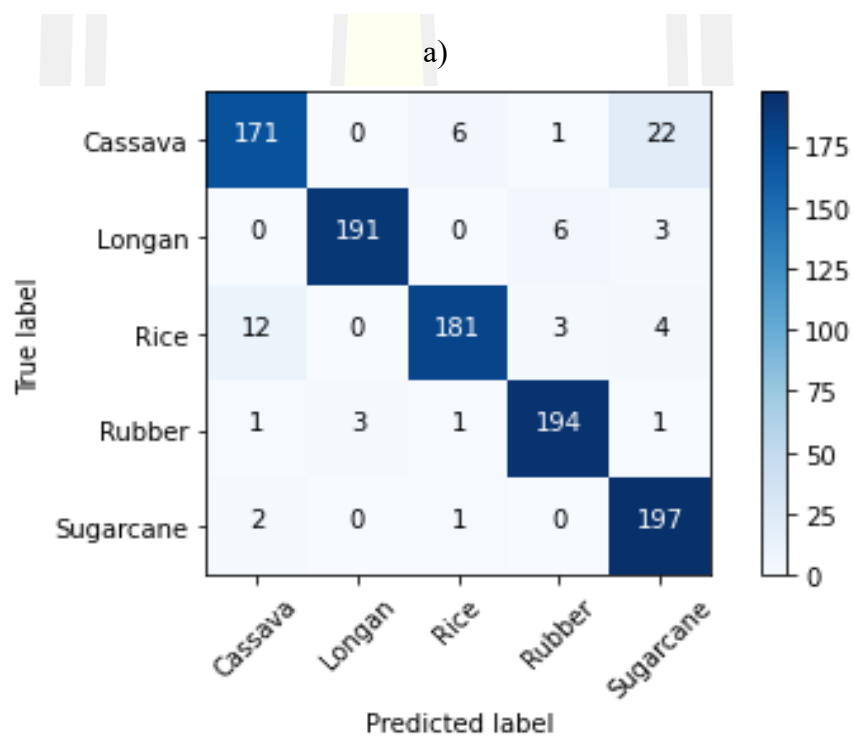
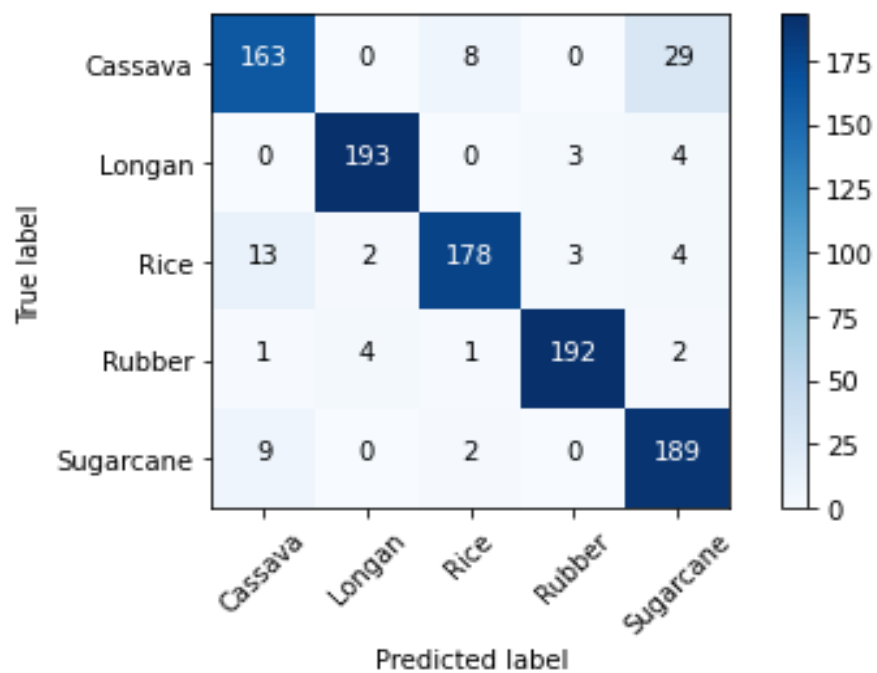
Hshift), called E2 model. 3) VGG16 (Rotation+Wshift) + VGG16 (Rotation+Wshift+Hshift) + VGG16 (Wshift+Hshift), called E3 model. and 4) VGG16 (Rotation+Hshift) + VGG16 (Rotation+Wshift) + VGG16 (Wshift+Hshift), called E4 model.

**Table 4** Performances of the ensemble CNN methods on the EcoCropsAID dataset.

Models	Ensemble Method		
	Unweighted Average	Weighted Average	Unweighted Majority Vote
E1	92.60	92.60	92.00
E2	92.40	92.70	91.90
E3	92.30	92.70	92.30
E4	92.60	<b>92.80</b>	92.50

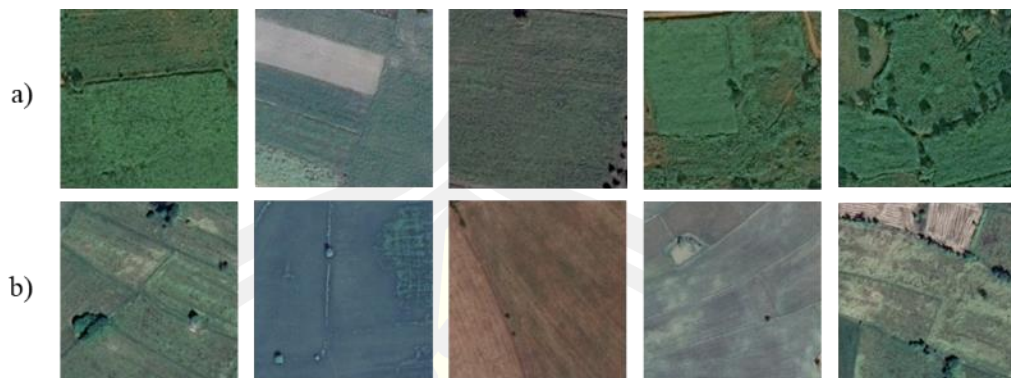
Table 4 provides the accuracy results of three ensemble methods on the EcoCropsAID dataset. I observed that the weighted average ensemble method insignificantly outperforms the average ensemble method on model E2-E4, but with model E1 it was equally accurate.

In the grid search experiments, I defined the range of the weighted parameters of 0-0.995. I explored the weighted parameters on the training set. The total amount of the weighted parameters is equal to one. It took about 18 minutes to search. The best-weighted parameters for each ensemble CNN model (E1-E4) were as follows; E1 = [0.08, 0.27, 0.65], E2 = [0.07, 0.86, 0.07], E3 = [0.86, 0.09, 0.05], E4 = [0.07, 0.86, 0.07].



**Figure 9** Illustration the confusion matrix of (a) the VGG16 architecture when using the data augmentation technique with the height shift technique and (b) the ensemble CNN with the weight average method on the EcoCropsAID dataset.





**Figure 10** The results indicate that (a) several samples of (~29) cassava class are misclassified as sugarcane class and (b) (~13) rice class are misclassified as cassava class.

Additionally, as can be seen from Figure 9, the confusion matrix explains that the misclassified from cassava class to sugarcane class is reduced from 29 to 22 images and misclassified from sugarcane class to cassava class is reduced from 9 to 2 images. Consequently, it can be seen that the ensemble CNN method achieves higher efficiency than training with only the individual CNN architecture.

The confusion matrix of the VGG architecture is illustrated in Figure 9a. The results show that 29 cassava images are misclassified as belonging to the sugarcane class. Also, 8 cassava images are misclassified as rice class. Subsequently, 13 rice and 9 sugarcane images are misclassified as cassava. When capturing the aerial images at the beginning of the cultivation period, the cassava and sugarcane pattern always appear to be similar and it is challenging to distinguish between these two classes. The misclassification between the cassava and rice classes is shown in Figure 10. Furthermore, the confusion matrix of the ensemble CNN method is shown in Figure 9b. The incidence cassava images being misclassified as the sugarcane class is decreased from 29 to 22 images. Also, the misclassification of sugarcane as cassava is reduced from 9 to 2 images.

## 2.6 Conclusions

In this paper, I have compared several convolutional neural network (CNN) architectures to discover the baseline CNN models for land use classification. For land use classification, I propose a novel dataset of Thailand's economic crops aerial image called the EcoCropsAID. This dataset includes five classes (rice, sugarcane, cassava, rubber, and longan)



and contains 5,400 economic crops aerial images collected from the Google Earth program. I chose eight CNN models comprising InceptionResNetV2, MobileNetV2, DenseNet201, Xception, ResNet152V2, NasNetLarge, VGG16, and VGG19, in order to discover the best CNN model. From the experimental results, I conclude that the performance of the VGG16 and VGG19 models is unequivocally better than other CNN models. The experiments show that the VGG architecture significantly outperforms NASNet with an accuracy of approximately 20% and InceptionResNetV2 with 37% accuracy. I have also demonstrated the impact of the data augmentation technique. Surprisingly, even the uncomplicated data augmentation techniques, such as rotation and height shift, improve the performance of the CNN architectures. The accuracy increased by 17.31% when training the NASNet architecture using data augmentation techniques.

I propose the ensemble CNN framework to achieve higher performance of land use classification. According to the ensemble method, three ensemble methods are then compared, including majority vote, averaging, and weighted average. The experiments showed that the three baseline CNN architectures combined with the weighted average ensemble method outperform when combined with the other ensemble methods. In order to evaluate the proposed framework, my ensemble CNN method achieved the recognition accuracy of 92.80% on the EcoCropsAID dataset. Although the ensemble method is an outstanding way to enhance the recognition performance, it does not improve the accuracy results if there is a high variance in the accuracy between each model.

In future work, to enhance the performance of land use classification, I will experiment on various ensemble methods, such as the snapshot ensemble (Wolpert, 1992); (G. Huang et al., 2017) and stacked ensemble (Pari, Sandhya, & Sankar, 2020). I will also consider self-supervised feature learning to extract the spatial features (Zheng et al., 2019) from the economic crops aerial image and classify the feature vector with other deep learning such as long short-term memory (LSTM) network (Z. Sun, Di, & Fang, 2019). I am interested in using generative adversarial networks (GANs) (Cap, Uga, Kagiwada, & Iyatomi, 2020); (Q. Wu, Chen, & Meng, 2020) as the data augmentation technique.

## Chapter 3

### Snapshot Ensemble CNN and New Learning Rate Schedule

The ensemble learning method is a necessary process that provides robustness and is more accurate than the single model. The snapshot ensemble convolutional neural network (CNN) has been successful and widely used in many domains, such as image classification, fault diagnosis, and plant image classification. The advantage of the snapshot ensemble CNN is that it combines the cyclic learning rate schedule in the algorithm to snap the best model in each cycle. In this research, I proposed the dropCyclic learning rate schedule, which is a step decay to decrease the learning rate value in every learning epoch. The dropCyclic can reduce the learning rate and find the new local minimum in the subsequent cycle. I evaluated the snapshot ensemble CNN method based on three learning rate schedules: cyclic cosine annealing, max-min cyclic cosine learning rate scheduler, and dropCyclic then using three backbone CNN architectures: MobileNetV2, VGG16, and VGG19. The snapshot ensemble CNN methods were tested on three aerial image datasets: UCM, AID, and EcoCropsAID. The proposed dropCyclic learning rate schedule outperformed the other learning rate schedules on the UCM dataset and obtained high accuracy on the AID and EcoCropsAID datasets. I also compared the proposed dropCyclic learning rate schedule with other existing methods. The results show that the dropCyclic method achieved higher classification accuracy compared with other existing methods.

#### 3.1 Introduction

Most remote sensing applications have been proposed for scene understanding, including scene classification, scene retrieval, and scene-driven object detection (Li, Zhang, & Zhu, 2021), and computed based on 11 bands of satellite imagery. However, many published datasets have been collected from aerial images that include only three bands (red, green, and blue) and proposed for classification, segmentation, and retrieval tasks (Gu, Wang, & Li, 2019). In the last decade, traditional methods, including image processing and machine learning, have been studied extensively (Prakash, Manconi, & Loew, 2020; Zhang et al., 2020).

The appearance of deep learning algorithms has brought about an interest in by researchers neural networks. Deep learning algorithms have been applied to solve many problems, such as image and video classification, speech recognition, and natural language processing. Additionally, the remote sensing community has shifted its attention to deep learning algorithms and succeeded in image analysis, land use and land cover evaluation, scene classification, segmentation, object detection, and many other tasks (Ma et al., 2019). The deep learning performance depends on the resolution level of remote sensing images. Although, many researchers have proposed methods to classify land use and land cover from the moderate resolution remote sensing images (Yu et al., 2018).

The theory of the ensemble learning method is to combine outputs of various machine learning and even deep learning models and then decide from many strategies a final prediction resulting in better performance (Ganaie, Hu, Tanveer, & Suganthan, 2021). It uses many different decision strategies to find the final prediction, such as unweighted average, weighted average, majority voting, Bayesian optimal, and stacked generalization. Kulkarni and Kelkar used ensemble learning methods to classify multispectral satellite images. Three ensemble learning methods were compared (bagging, boosting, and AdaBoosting) and it was found that the ensemble learning method achieved better classification results than the single model. Cao et al. segmented the building areas from the remote sensing images using a stacking ensemble deep learning model. In their method, images were first segmented using three models: FCN-9s, U-Net, and SegNet followed by, optimizing prediction results using a fully connected conditional random field (CRF). Finally, the multilayer features were extracted using a sparse autoencoder. Then, the final prediction results were computed using the Euclidean distance weighting method.

Additionally, several researchers have proposed ensemble learning methods for classifying satellite images. Minetto et al. proposed an ensemble of convolutional neural networks (CNNs) for geospatial land classification. In their method the geospatial images were first sent to CNNs to predict the output. Hence, the predicted outputs from CNNs were determined as the final output using a majority voting method. Diengdoh et al. used weighted and unweighted ensemble learning for land cover classification from the predicted output of various machine learning methods. Huang et al. proposed an ensemble learning method for urban land use mapping tasks based on satellite images, street-view images, building footprints,

points-of-interest, and social sensing data to explain the associations of land cover, socioeconomic activities, and land use categories.

Furthermore, new kinds of ensemble learning require expensive computation with no additional training cost while training neural network models, called snapshot ensemble learning (G. Huang et al., 2017). The snapshot ensemble learning method aims to discover several local minimum values in one training. While training the model, we defined the number of cycles that we desired to snap the best model. For example, defining three cycles will return the three best models from each cycle, called snapshot. Further, the best model was snapped at the minimum loss value. Additionally, the learning rate schedule was used to quickly reduce the training loss value using the cyclic cosine annealing function. In addition, Wen et al. proposed a new max-min cosine cyclic learning rate scheduler invented to find the acceptable ranges of maximum and minimum learning rates used in training.

**Contribution.** In this research, we focus on proposing the new cosine cyclic learning rate schedule by adding a step decay function to reduce the learning rate that directly decreases the training loss to converge local minimum in each cycle, called dropCyclic. For the dropCyclic learning rate schedule, the learning rate starts at the maximum learning rate. Further, the training loss decreases to converge on a local minimum while training in the first cycle. In the next cycle, the new maximum learning rate, which is a smaller value than the previous learning rate, is defined using the dropCyclic method. Consequently, the dropCyclic method narrows the learning rate range from the start until the last cycle. The snapshot ensemble CNN based on the dropCyclic learning rate schedule is proposed for aerial image classification. The proposed method is evaluated on three aerial image datasets: UCM, AID, and EcoCropsAID, and achieved good performance.

**Outline of the paper.** This paper is organized into five sections, as follows. Surveys of the related works are presented in topic 3.2. Topic 3.3 presents the snapshot ensemble CNN for aerial image classification and the new learning rate schedule. In topic 3.4, three aerial image datasets are briefly described. Topic 3.5 presents the experimental results and discussions. The conclusion and future work are presented in Section 6.

## 3.2 Related Work

In this section, I briefly explain the research related to the ensemble learning and snapshot ensemble CNNs, including ensemble learning, snapshot ensemble CNN, and learning rate schedules for snapshot ensemble CNN.

### 3.2.1 Ensemble learning

Ensemble learning methods have been a growing research area in recent years. In this study, I surveyed ensemble learning methods with only two strategies: decision and ensemble.

#### 1) The decision strategy

The outputs of other classifiers are combined and classified to the final output with various strategies, such as unweighted average, weighted average, majority vote, Bayes optimal, and stacked generalization (Ganaie, Hu, Tanveer, & Suganthan, 2021). Kim and Lim proposed the ensemble CNNs method to learn on a large vehicle type dataset. The dataset contained more than 500,000 images and had 11 classes. The bagging method was used to randomly select the training data because the image distribution in each class was imbalanced. In the ensemble CNNs, the training images selected using the bagging method were transferred to the three CNNs. While training the CNN, the data augmentation techniques (flip, rotation, AR-fixed, AR-fixed rotation) were applied. The weighted average method was applied for the final prediction and achieved high performance. Minetto et al. used state-of-the-art CNNs (ResNet50 and DenseNet161) and a majority voting method for geospatial land classification on multispectral images. In the first step, 12 CNN models were created using various settings, such as data augmentation, image crop style, and class weighting. The output of this step was the probabilities obtained from 12 CNN models. In the second step, the output probabilities were classified using the majority voting method. However, the correct prediction was accepted when the outputs from the CNNs were correct in more than five models. Their proposed method achieved an accuracy of 94.51% on the FMOW dataset.

Moreover, Diengdoh et al. classified land cover using the ensemble learning method based on satellite imagery. Their study classified the land cover images into six classes using the unweighted ensemble prediction method. First, four machine learning techniques: K-nearest neighbor (KNN), naive Bayes (NB), random forest (RF), and support vector machine (SVM), were proposed to predict probability outputs. Second, the probability outputs were

classified using the unweighted ensemble learning method for the final output. Seifin et al. used three voting methods: unison vote, absolute majority, and no majority, to detect a land cover change from time-sequence Sentinel-2 images. The main architecture was the combination between the fully convolutional neural network (FCN) and long short-term memory (LSTM), called FCN+LSTM architecture. The time-sequence images were first classified by the FCN+LSTM model and output as six classification maps and the final output classified using the voting method. The results showed that the final predicted class using unison or the absolute majority method achieved high accuracy.

## 2) The ensemble strategy

The ensemble strategy uses the weak learner to create a stronger learner and minimize errors while training. It also has various ensemble strategies. For instance, the bagging ensemble randomly selects subsets of the independent data of the same size. Then, the first, second, and  $N$  subsets are trained using the first, second, and  $N$  classifiers, respectively. Finally, fusing the output of the base classifiers with the majority voting method for predicting the final output (Ganaie, Hu, Tanveer, & Suganthan, 2021; Kim & Lim, 2017). The boosting strategy, the original data is given to classify using a weak classifier. The original data that was misclassified from the weak classifier is weighted, and called weighted data, due to a decrease in bias obtained while training the weak classifier. Further, the weighted data is sent to the second weak classifier and again weighted to the misclassified data. It could repeat training with a weak classifier many times until it obtained the best weak classifier (Kulkarni & Kelkar, 2014).

The idea of a combination between ensemble strategy (bagging and boosting learning) and the CNN-based method was proposed to short-term load forecasting (Dong, Qian, & Huang, 2018). In Dong et al., the CNN model was firstly trained on the existing dataset to create the pre-trained CNN model. Then, the fine-tuned model was created by training the pre-trained CNN model from the first phase with the new dataset. Finally, the weak CNN models from phases one and two were constructed to create a robust model. Consequently, the average weighted method was used to compute the prediction result.

Korzh et al. proposed the bagging ensemble and the stacking of CNN to classify remote sensing imagery. In their method, image processing techniques were first applied to the original images to reduce noise and increase sharpness. Then, the set of the original images was sent to CNN models (AlexNet, GoogLeNet, and VGG19) to extract the



first feature. Also, the set of processing images was sent to CNN models to extract the second feature. Hence, the first and second features of more than 16,000 features were concatenated before sending to the machine learning technique. In their experiments, many machine learning techniques were compared, including SVM with different kernels (linear, radial basis function, polynomial), random forest, logistic regression. As a result, the SVM with a linear kernel obtained the highest performance on the Brazilian coffee scenes dataset with an accuracy of 96.11%.

### 3.2.2 Snapshot ensemble CNN

The snapshot ensemble method was first proposed by Huang et al. The main purpose of the snapshot ensembles was to train CNN one time and obtain more CNN models. Therefore, while training the original CNN model, the model converged the minimum training loss value at the end. Hence, only one CNN model was obtained from the original CNN model. On the other hand, cyclic cosine annealing was used to converge multiple training loss values. The best CNN model in each cycle was used, called snapshot. Consequently, the output probability of each CNN model was calculated using the softmax function. Additionally, the unweighted average method was used as the final prediction. The output probabilities were averaged and the maximum probability was selected. The snapshot ensemble method was evaluated on various image classification datasets and achieved the best performance compared with a single CNN model.

In 2019, Wen et al. proposed a new snapshot ensemble CNN for fault diagnosis. The max-min cosine cyclic learning rate scheduler (MMCCLR) was proposed instead of cyclic cosine annealing. The log-linear learning rate testing (LogLR) method was invented to search the fitting range of the max-min learning rate when encountering new datasets. The MMCCLR method was evaluated on three datasets (bearing dataset of Case Western Reserve University, self-priming centrifugal pump dataset, and bearing dataset) and achieved very high accuracy on three datasets with 99.9%.

Moreover, Babu and Annavarapu modified the snapshot ensemble method to classify COVID-19 from chest X-ray images. For training the CNN model, the pre-trained model ResNet50 was used and trained on the chest X-ray images. The data augmentation techniques (rotation, zoom, flip, and shift) were also applied while training. Subsequently, the weighted average method was used for ensemble learning instead of the unweighted average method. Hence, the weighted parameter was updated until it did not improve accuracy



performance. The modified snapshot ensemble method achieved 95.18% accuracy on the COVID-19 XCR dataset and outperformed existing methods. Puangsuwan and Surinta used the snapshot ensemble method to classify plant leaf diseases. Three CNN architectures (VGG16, MobileNetV2, InceptionResNetV2, and DenseNet201) were used as the backbone architecture of the snapshot ensemble method. The rotation method was used as the data augmentation technique while training the CNN models. In the snapshot ensemble method, training the DenseNet201 model using four cosine annealing cycles achieved the highest accuracy of 69.51% on the PlantDoc dataset compared to other ensemble methods (unweighted ensemble and weighted ensemble).

For the aerial images, Dede et al. studied various ensemble strategies (including homogeneous, heterogeneous, and snapshot ensemble) to classify aerial scene images. In their experiments, two pre-trained CNN models were used: Inception and DenseNet. The snapshot ensemble method with Inception as a backbone architecture achieved an accuracy of 96.01% on the RESISC45 dataset. However, the snapshot ensemble method did not attain the best accuracy on the AID dataset. The best algorithm on the AID dataset was the heterogeneous strategy combining Inception and DenseNet and classified using the multi-layer perceptron (MLP). It achieved an accuracy of 97.15% on the AID dataset.

### **3.2.3 The cyclical learning rate for snapshot ensemble CNN**

The popular optimization algorithm used while training the CNN model is stochastic gradient descent (SGD). SGD is used to update parameters of the CNN model until it converges to the local minimum value. In the original snapshot ensemble method, the SGD optimizer and the cyclic cosine annealing were computed to quickly decrease the training loss to converge the local minimum (Huang et al., 2017). The training loss decreased very fast compared to the original CNN model. Wen et al. proposed a new snapshot ensemble method that used the MMCCLR method to find the range of learning rates. Petrovska et al. used an adaptive learning rate schedule with a triangular policy to train the snapshot ensemble method. Furthermore, Hung et al. proposed a two-stage cyclical learning rate method using triangular methods. The triangular and triangular2 methods were used in the first and second states to find the best stable model and required few iterations while training.

### 3.3 Proposed snapshot ensemble CNN for aerial image classification

The snapshot ensemble CNN was first proposed by Huang et al. with the simple concept of finding many local minima values and then snapping the best CNN model at the local minimum in each cycle. Subsequently, the outputs of CNN models were combined and computed using the ensemble method. The essence of the snapshot ensemble CNN method is the cyclic learning rate schedule, which is the cyclic cosine annealing (CCA) schedule. The CCA schedule allowed the learning rate to decrease quickly, stimulating the CNN model to reach local minimum after a few epochs. The snapshot ensemble CNN produces lower errors than a single CNN model. This section briefly describes 1) The cosine cyclic learning rate schedule, including cyclic cosine annealing, max-min cosine cyclic, and the proposed dropCyclic. 2) The snapshot ensemble method.

#### 3.3.1 Cosine cyclic learning rate schedule

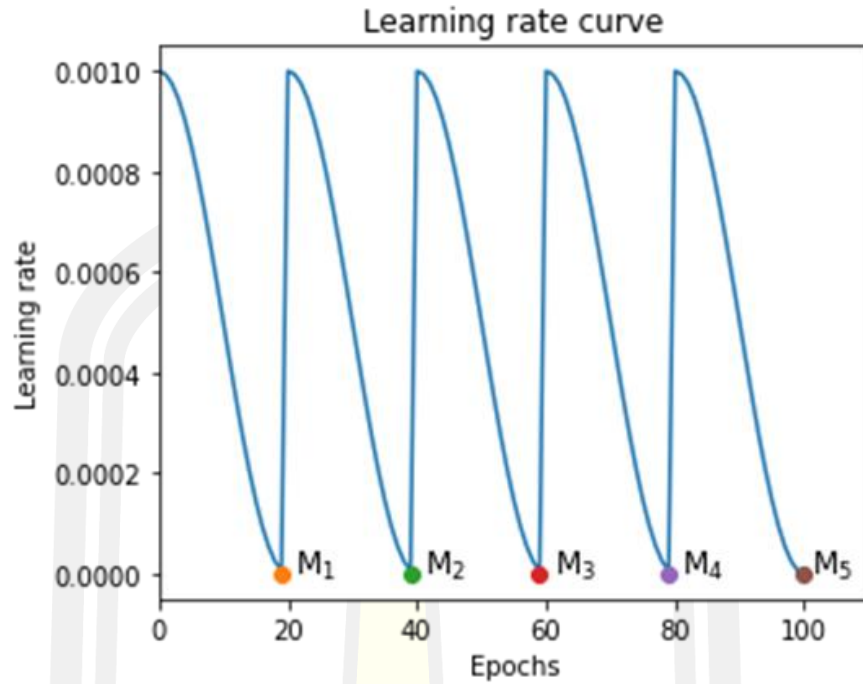
##### 1) Cyclic cosine annealing

The cyclic cosine annealing (CCA) is the primary learning rate schedule of the snapshot ensemble CNN method used while training the CNN model. CCA allows the CNN to lower the learning rate faster than the traditional CNN model and converge to diverse local minimums (Huang et al., 2017). The CCA curve training with 100 epochs and using five cycles ( $M$ ), when  $M_1, M_2, \dots, M_5$  denote the CNN model for each local minimum, as shown in Figure 11. The CCA is computed as Equation (1).

$$\eta = \frac{\eta_{init}}{2} \left( \cos \left( \frac{\pi \text{mod}(t-1, [T/M])}{[T/M]} \right) + 1 \right) \quad (4)$$

where  $\eta$  stands for the learning rate of current iteration,  $\eta_{init}$  is the initial learning rate,  $t$  is for the current iteration number,  $T$  is the total iterations, and  $M$  is the number of cycles.

In the CCA, only the initial learning rate ( $\eta_{init}$ ) is required to be adjusted. As a result, the wrong learning rate will cause the training process to not converge with the local minimum at the end of each cycle.



**Figure 11** Illustration of the cyclic cosine annealing curve training with 100 epochs and using five cycles.

## 2) Max-min cosine cyclic learning rate scheduler

Wen et al. proposed the max-min cosine cyclic learning rate scheduler (MMCCLR). The upper and lower boundaries of the learning rate were proposed to adjust the boundary of the learning rate. However, in the MMCCLR, the log linear learning rate test (LogLR Test) method was proposed to find the learning rate range, which is the max and min learning rates. The LogLR Test method and the MMCCLR are calculated as Equation (5) and Equation (6).

$$\log_{10}\eta = \log_{10}\eta_{min}^{LR} + (\log_{10}\eta_{max}^{LR} - \log_{10}\eta_{min}^{LR}) \times \frac{[t,b]}{[T,b]} \quad (5)$$

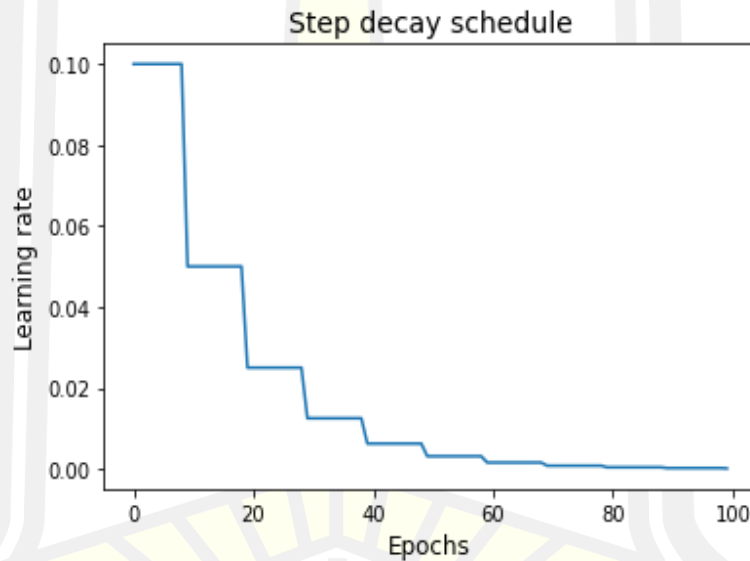
$$\eta = \eta_{min} + \frac{(\eta_{max} - \eta_{min})}{2} \times \left(1 + \cos\left(\frac{\pi \text{mod}(t-1, [T/Mb])}{[T/Mb]}\right)\right) \quad (6)$$

where  $\eta_{min}$  is the minimum learning rate and  $\eta_{max}$  is the maximum learning rate that is tested using Equation (5).

### 3) Proposed drop cyclic cosine learning rate schedule

In this study, I proposed a drop cyclic cosine learning rate schedule, called dropCyclic. The dropCyclic is the systematic reduction of the learning rate over a specific time during training. This research aims to decrease the learning rate that cuts by a constant factor called the drop parameter, every constant number of epochs (see Figure 12), as same as the step decay schedule (Ge, Kakade, Kidambi, & Netrapalli, 2019). The dropCyclic is also efficient in discovering the diversity of local minimums in each cycle using the  $c$  parameter.

Moreover, in dropCyclic, the maximum learning rate in each cycle is changed according to the drop parameter. While the learning rate range is limited, the CNN model can faster converge to the local minimum. The equation of the dropCyclic is computed as Equation (7).



**Figure 12** Illustration of the step decay schedule.

$$\eta = \frac{\eta_{init}}{2} \times drop^{\lfloor (1+t)/c \rfloor \times (1 + \cos(\frac{\pi \text{mod}([t,b],[T/Mb])}{[T/Mb]}))} \quad (7)$$

where  $\eta_{init}$  is the initial learning rate,  $drop$  is the step decay parameter that drops the learning rate in every  $n$  epoch,  $c$  is a constant number that lets the model change to the new local minimum in the next cycle.

### 3.3.2 Snapshot ensemble methods

The ensemble method is the final step of the snapshot ensemble CNN to enhance accuracy based on diverse CNN models from single training. I trained the CNN model using the proposed dropCyclic method and snapped the best CNN model from each cycle in the previous step. The output probabilities of each CNN computed using the softmax function were combined and classified using the unweighted average ensemble method. Indeed, the last  $N$  models are significant to have the lowest error. I then normally ensemble the last  $N$  models. The ensemble method is calculated as Equation (8).

$$\hat{y} = \frac{1}{M} \sum_{j=1}^M f_j \quad (8)$$

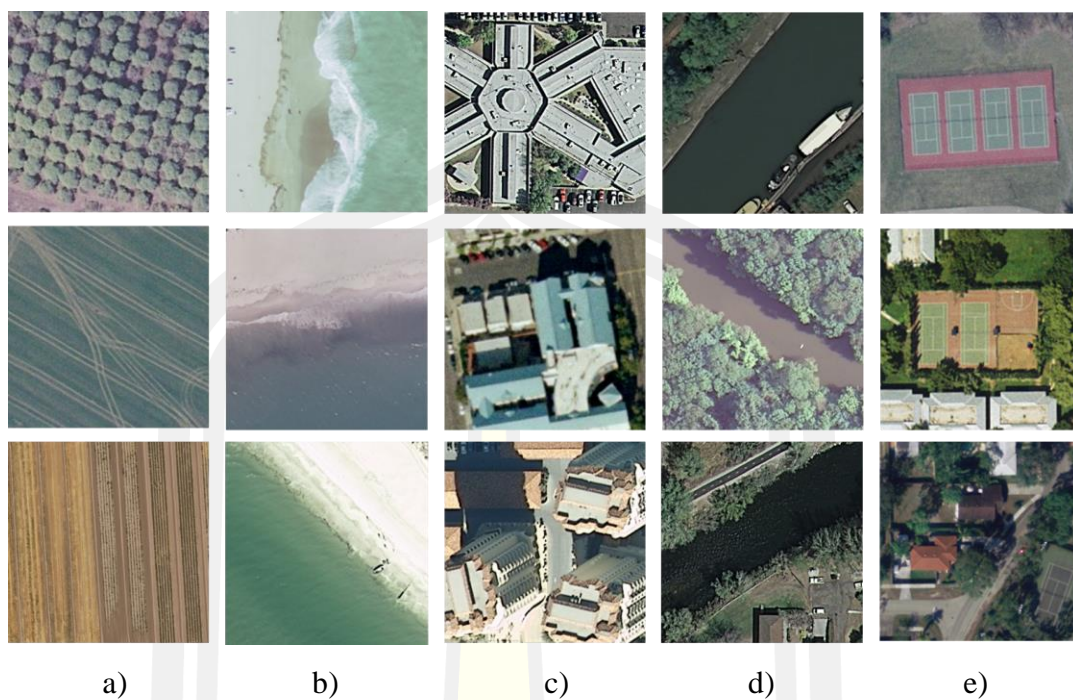
where  $M$  is the number of CNN models and  $f_j$  is the output probabilities of CNN model  $j$  that is computed using the softmax function.

## 3.4 Aerial image datasets

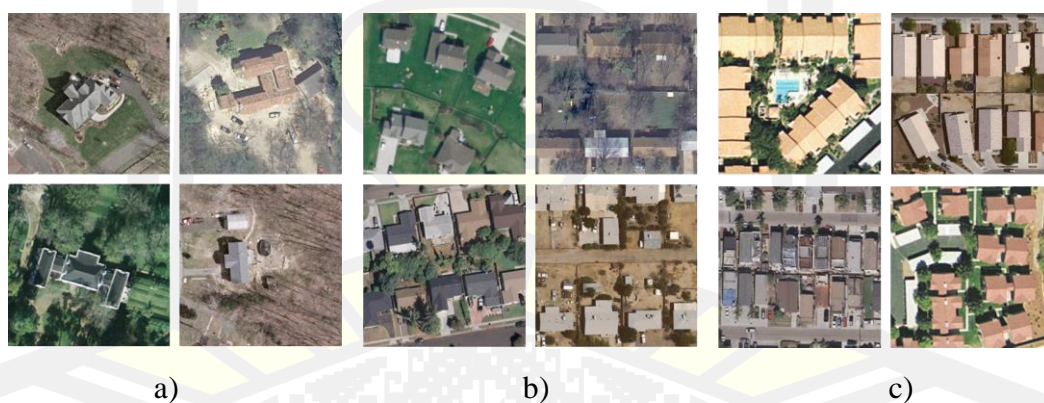
### 3.4.1 UC merced land use (UCM) dataset

Yang and Newsam first proposed the UCM dataset for land use classification tasks collected from the USGS national map urban area imagery. The aerial images were extracted from large images and divided into 21 classes, such as agricultural, forest, golf course, beach, harbor, buildings, medium residual, sparse residential, and dense residential. It is stored in the RGB color space image with 256x256x3 pixels. The UCM dataset contains 2,100 images and some examples of the UCM dataset are shown in Figure 13. However, the challenge of the UCM dataset is that classes of medium residual, sparse residential, and dense residential, are similar and difficult to classify, as shown in Figure 14.





**Figure 13** Examples of the UCM dataset: a) agricultural, b) beach, c) buildings, d) river, and e) tennis court.

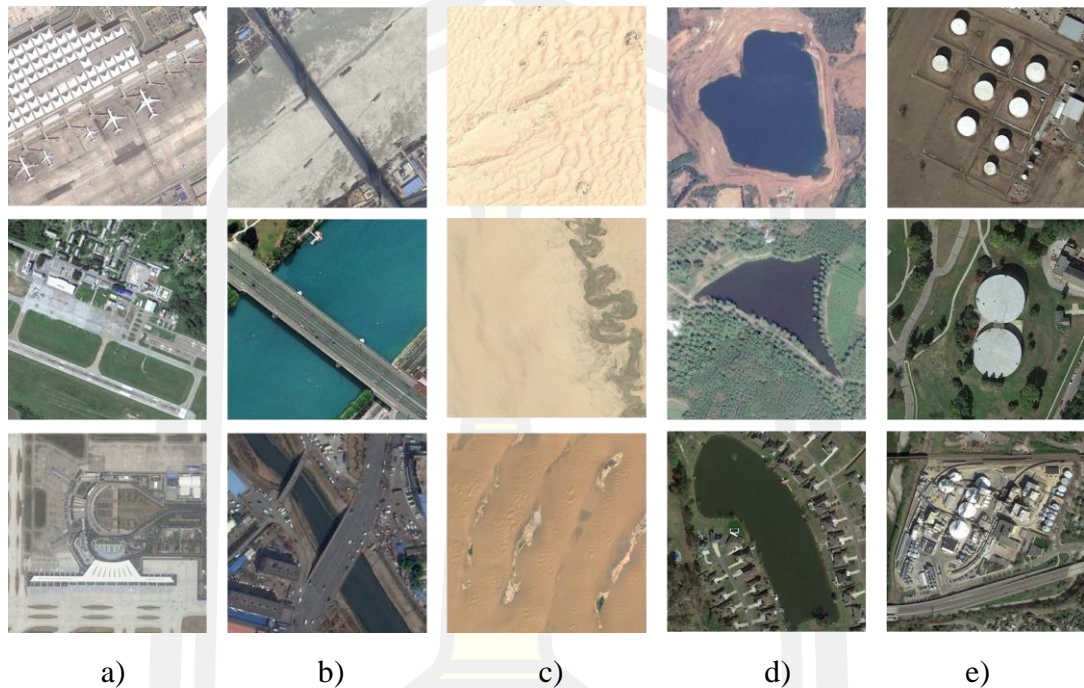


**Figure 14** Illustration of the (a) sparse residential, (b) medium residential, and (c) dense residential.

### 3.4.2 Aerial image dataset (AID)

The AID (Xia et al., 2017) was proposed for aerial scene classification tasks. It has 10,000 images and contains 30 different aerial scene classes, for example, dense residential, medium residential, sparse residential, stadium, industrial, bridge, and baseball field. Each class has approximately 200 to 400 images of 600x600 pixels. The AID was collected from the

Google Earth application at resolution of 8 to about 0.5 meters. Examples of the AID are shown in Figure 15.

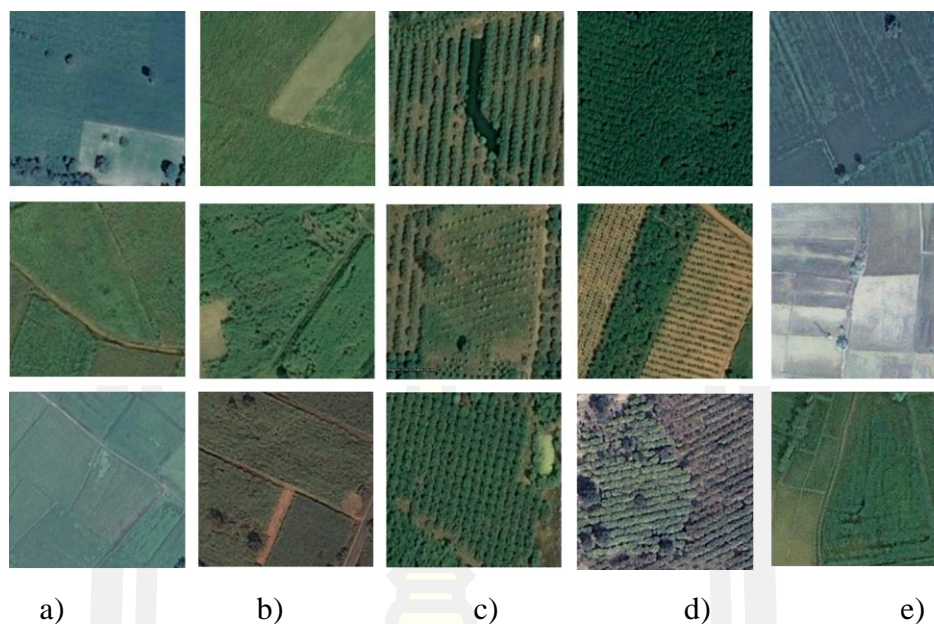


**Figure 15** Example aerial images of the AID dataset: a) airport, b) bridge, c) desert, d) pond, and e) storage tanks.

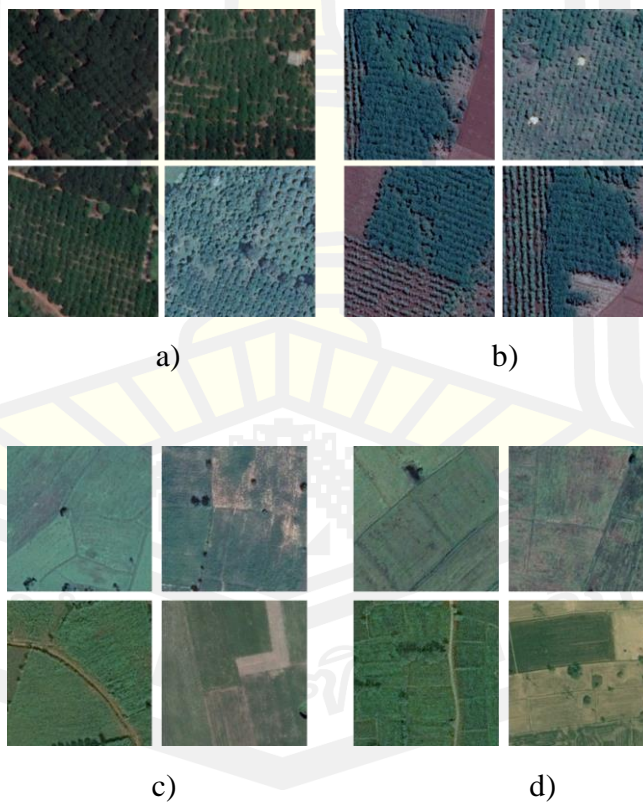
### 3.4.3 EcocropsAID dataset

Thailand's economic crops aerial image dataset (EcoCropsAID) was proposed by Noppitak and Surinta for land use classification. The EcoCropsAID dataset was collected according to the information on the cultivation of economic crops in different regions between 2014 and 2018 obtained from Agri-Map Online. The economic crops aerial images were collected from the Google Earth application at resolution of 30 to 0.2 meters. The images were stored in the RGB format with 600x600 pixels. It has 5,400 aerial images of five classes: rice, sugarcane, cassava, rubber, and longan. Example images of the EcoCropAID dataset are shown in Figure 16. The challenges of the EcoCropsAID dataset are that the pattern of each class is quite similar (see Figure 17), and various patterns occur in the same class (see Figure 18).

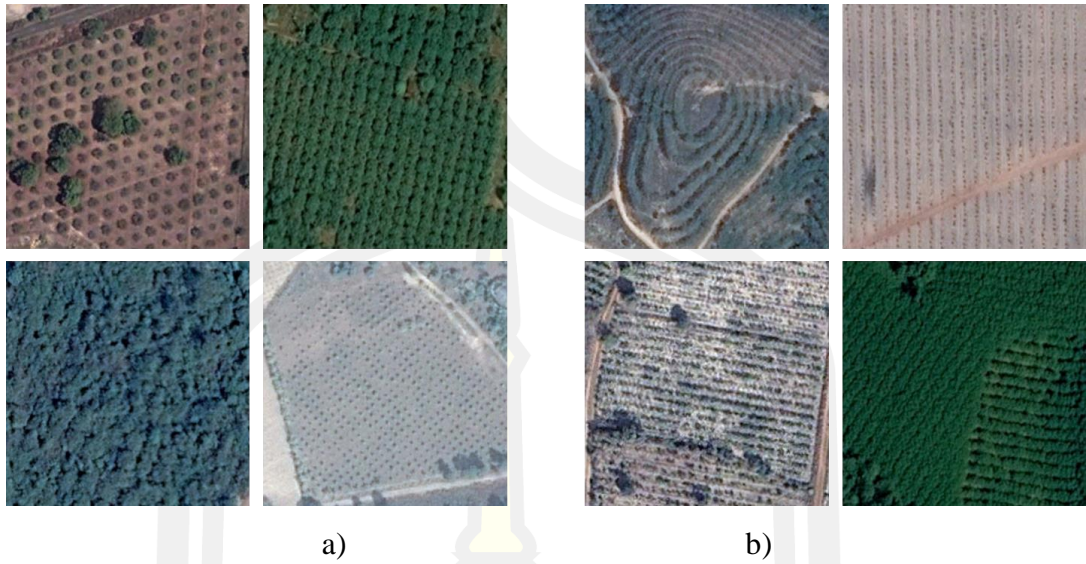




**Figure 16** Examples of the EcoCropsAID dataset: a) cassava, b) sugarcane c) longan d) rubber, and e) rice.



**Figure 17** Illustration of the similarity patterns between two classes: (a) longan and (b) rubber and (c) cassava and (d) rice, of the EcoCropsAID dataset.



**Figure 18** Illustration of the diversity in the patterns of the EcoCropsAID dataset. Examples of (a) longan and (b) rubber images.

The details of the three aerial image datasets used in my experiments are summarized in Table 5.

**Table 5** The summary of three aerial image datasets: UCM, AID, ECOcropsAID.

Aerial Image Datasets	Classes	Images per Class	Total images	Image size (pixels)	Resolution (meters)	Year	Training images	Test image
UCM (Y. Yang & Newsam, 2010)	21	100	2,100	256 x 256	0.3	2010	1,680	420
AID (G.-S. Xia et al., 2017)	30	220 to 420	10,000	600 x 600	8 to 0.5	2017	5,000	5,000
EcoCropsAID (Sangdaow Noppitak & Olarik Surinta, 2021)	5	~1,000	5,400	600 x 600	30 to 0.2	2021	4,320	1,080

### 3.5 Experimental results and discussion

I demonstrated the effectiveness of the proposed cyclic learning rate (dropCyclic) and compared it with two existing cosine cyclic learning rate methods: cyclic cosine annealing (CCA) and the max-min cyclic cosine learning rate scheduler (MMCCLR) on three aerial

image datasets. For the backbone of the snapshot ensemble, I compared three CNN architectures: MobileNetV2, VGG16, and VGG19.

All the experiments were trained and evaluated on a Linux operating system using Intel(R)Core-i9-9900K CPU @ 3.60GHz x 16, RAM 32GB, and GPU GeForce GTX 1080Ti with RAM 11GB GDDR5x. I implemented all snapshot ensemble methods based on the TensorFlow deep learning framework with the Keras library.

### 3.5.1 Evaluation metrics

In this experiment, I used K-fold cross-validation (cv) with K=5 over the training set to prevent overfitting problems. Hence, the overall accuracy (%) and standard deviation evaluated the training set. Further, test accuracy was used to evaluate the classification performance, and the results were compared with existing snapshot ensemble methods. The accuracy performance was computed as shown in Equation (9).

$$ACC = \frac{TP+TN}{TP+FN+FP+TN} \times 100 \quad (9)$$

where  $TP$  is a true positive (indicates the positive samples that are correctly classified),  $TN$  is a true negative (indicates the negative samples that are correctly classified),  $FP$  is a false positive (indicates the negative samples that are misclassified), and  $FN$  is a false negative (indicates the positive samples that are misclassified).

In order to prevent overfitting problems and compare different learning rate methods, I used the loss difference (LD) metric to evaluate snapshot ensemble methods when the difference learning rate policy was performed. The LD is the evaluation metric that indicates the robustness of the model against the overfitting problems (Wu et al., 2019). Overfitting problems appear when the low loss value is on the training set, but the high loss value is on the test set. Hence, it results in low accuracy. The smallest LD value shows the robustness of the model, which is computed as Equation (10).

$$LD = valid_{loss} - training_{loss} \quad (10)$$

where  $training_{loss}$  and  $valid_{loss}$  are the loss values obtained while training on the training set and validation set, respectively.

### 3.5.2 Training setting

#### 1) Data ratio and number of experiments

I reported the data ratio and the number of experiments of each dataset. I divided the dataset into training, validation, and test sets. I split with the ratio of 4:1:5 for the AID dataset and 7:1:2 for the UCM and EcocropsAID datasets. Due to the randomness of the training and validation sets, I computed experiments three times and reported the mean accuracy and standard deviation on the validation set. Further, I trained the model again on the training and validation sets with the best setting and evaluated the test set.

#### 2) Backbone CNN architectures

In my previous study (Noppitak & Surinta, 2021), several CNN architectures, including InceptionResNetV2, DenseNet201, Xception, ResNet152V2, NASNetLarge, MobileNetV2, VGG16, and VGG19 were experimented with. I found that the VGG16 and VGG19 achieved the highest accuracy. Subsequently, MobileNetV2 showed worse accuracy compared to VGG16 and VGG19. Hence, in this study, I mainly experimented with the snapshot ensemble CNN using three state-of-the-art architectures as a backbone CNN: VGG16, VGG19 (Simonyan & Zisserman, 2014), and MobileNetV2 (Sandler, Howard, Zhu, Zhmoginov, & Chen, 2018); to prove that the snapshot ensemble could manage both the best and the worst CNN architectures and also enhance the classification performance on the land use images.

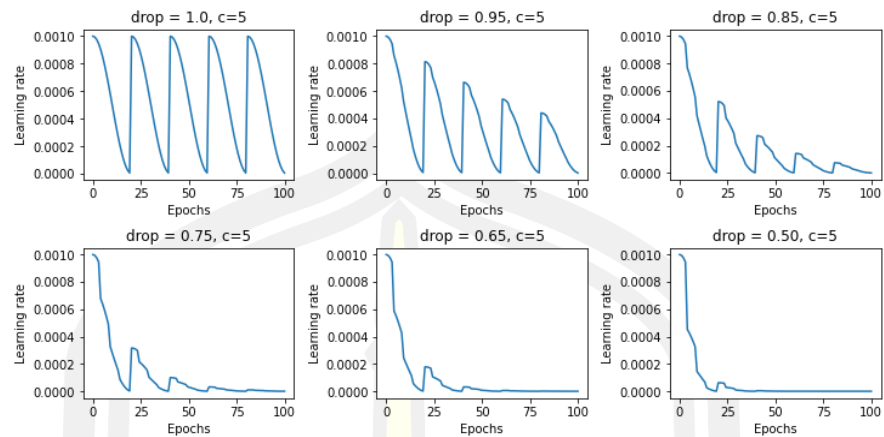
#### 3) Snapshot ensemble methods

I compared the proposed drop cyclic cosine learning rate schedule (dropCyclic) with two existing learning rate schedules: CCA and MMCCLR. I trained the snapshot ensemble with 100 epochs and the snapshot parameter with  $M = 5$  cycles.

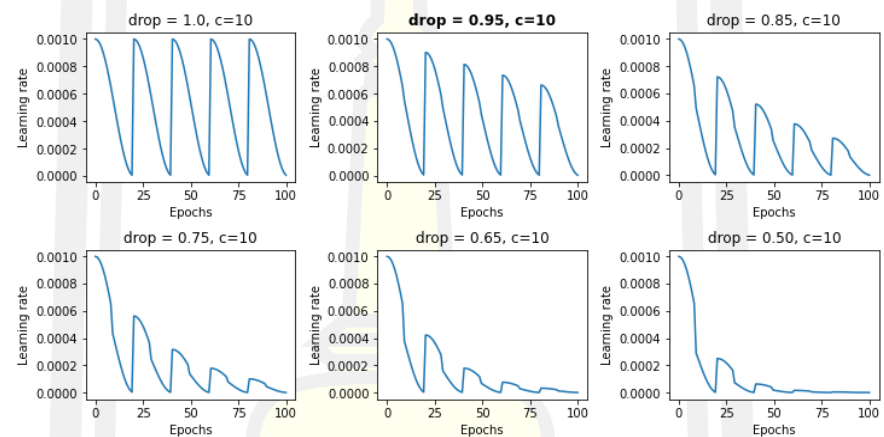
#### 4) dropCyclic learning rate schedule method

As shown in Figure 19, I illustrated the learning rate curve of the dropCyclic learning rate schedule. The learning rate was computed using Equation 7.

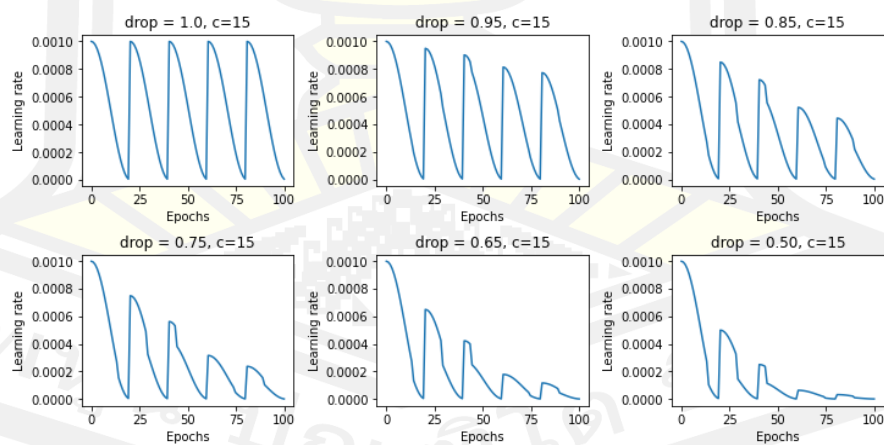




a)



b)



c)

**Figure 19** Illustration of the dropCyclic learning rate curve when the parameters were set as;  $M=5$  cycles (20 epochs per cycle),  $\text{drop}=1.0, 0.95, 0.85, 0.75, 0.65, 0.50$  and , (a)  $c=5$ , (b)  $c=10$ , (c)  $c=15$ .

The small  $c$  value made the model capable of escaping the local minimum to find a new local minimum, as shown in Figure 19(a). With the large  $c$  value, the model could have the energy to discover a new local minimum, as shown in Figures 19(b) and 19(c). In the case of  $drop = 1.0$ , at the first cycle, the maximum learning rate was 0.001 and the learning rate decreased in each epoch until zero. In the second cycle, the learning rate started again at 0.001. Further, in the case of  $drop = 0.95$ , the maximum learning rate in the next cycle was slightly dropped until the last cycle according to the  $drop$  parameter. In my dropCyclic experiments, the  $drop$  and  $c$  values were 0.95 and 10, respectively.

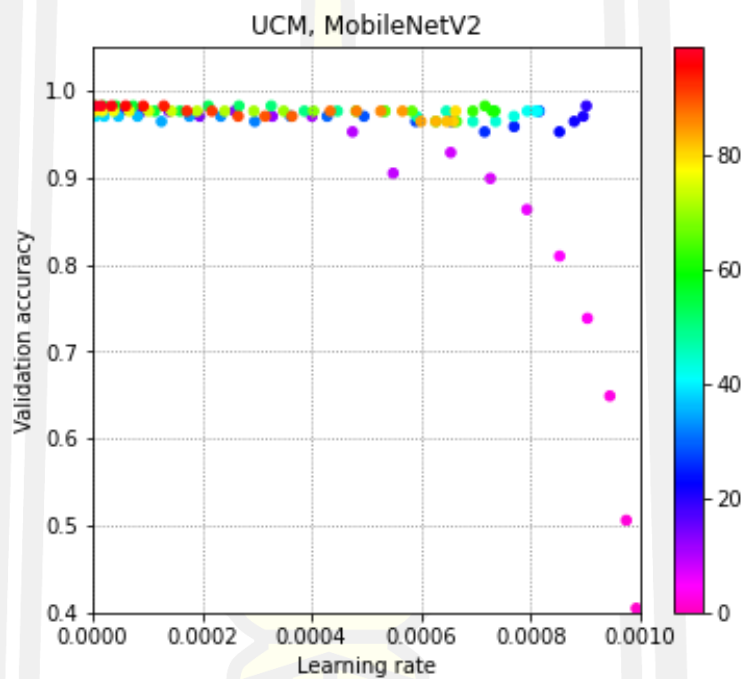
The details of the hyperparameter settings on CNN architectures and the learning rate schedule are summarized in Table 6.

**Table 6** Summarize the hyperparameter settings of the CNN model and snapshot ensemble methods.

Parameter Settings		Learning Rate Schedules		
		CCA	MMCCLR	dropCyclic
CNN Parameters	Image normalization	0-1	0-1	0-1
	Mini batch size	16	16	16
	Data augmentation	No	No	No
	Optimizer	SGD	SGD	SGD
	Momentum	0.9	0.9	0.9
	Learning rate	0.001	Min = 0 Max = 0.001	0.001
Snapshot Ensemble Parameters	Cycle (M)	5	5	5
	$drop$	-	-	0.95
	$c$	-	-	10

### 3.5.3 Classification results on the UCM dataset

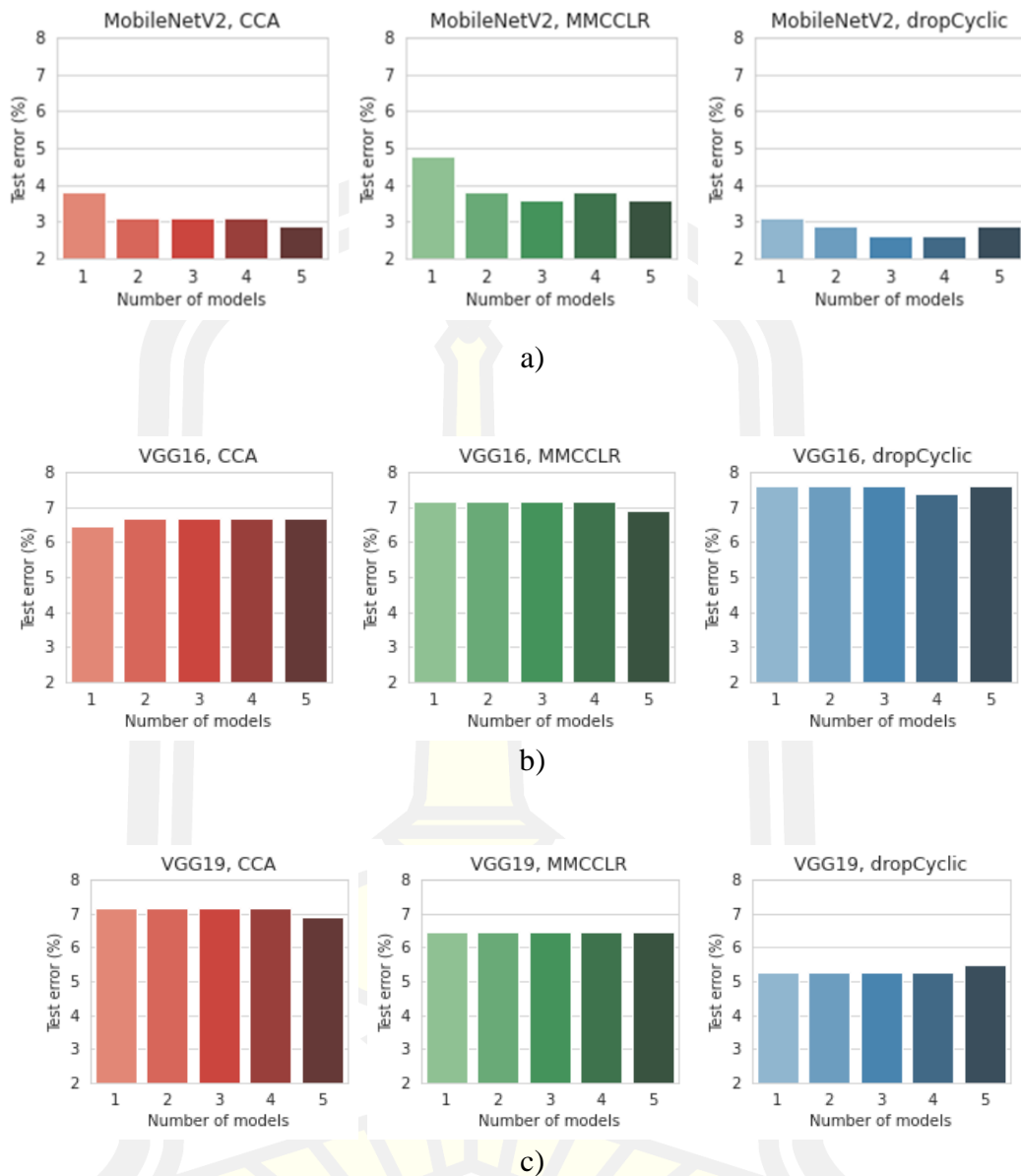
I first observed the optimal learning rate values to confirm that the proposed dropCyclic learning rate schedule performs well when using the optimal learning rate. In this experiment, I trained the snapshot ensemble learning using the MobileNetV2 as a backbone CNN and the proposed dropCyclic learning rate schedule on the UCM dataset. In this study, I illustrated the accuracies of various learning rates, including 0.1, 0.01, 0.001, and 0.0001. The validation accuracies are shown in Figure 20.



**Figure 20** The validation accuracy of the snapshot ensemble CNN using MobileNetV2 and the dropCyclic method as a learning rate schedule with different learning rate values (0~0.001) on the UCM dataset.

The scatter plot Figure 20, show that the validation accuracies closed to approximately 98% were achieved when the learning rate value reached zero. However, I observed that the validation dropped when the learning rate was high. In this study, the validation accuracy dropped from approximately 80% to 40% when the learning rates were in the range of 0.0008 to 0.0010.





**Figure 21** Loss error (%) of snapshot ensemble CNN with different learning rate schedule methods: CCA (first column), MMCCLR (second column), dropCyclic (third column) and CNN architectures: (a) MobileNetV2 (b) VGG16, (c) VGG19, on the UCM dataset. Each snapshot ensemble CNN was trained with  $M=5$  cycles.

Figure 21 presents the test error (%) of snapshot ensemble CNN with different learning rate schedule methods. I trained the snapshot ensemble CNN with five cycles, and then the test error of the ensemble method when combined one, two, three, four, and five models, respectively, were reported. As seen in Figure 21(a) in the third column, the snapshot ensemble CNN using MobileNetV2 with dropCyclic learning rate schedule obtained the lowest test error

when ensemble with only three models. In comparison, CCA methods achieved the lowest test error when ensemble with five models (see Figure 21(a) in the first column).

Furthermore, when training the snapshot ensemble CNN using VGG19 with MMCCLR and dropCyclic learning rate schedule methods, as shown in Figure 21(c) in the second and third columns, the lowest error was achieved by using only one model. Therefore, the VGG19 discovered the optimal local minimum at the first cycle according to the number of aerial images in the UCM dataset with only 2,100 images. The results are presented in Table 7.

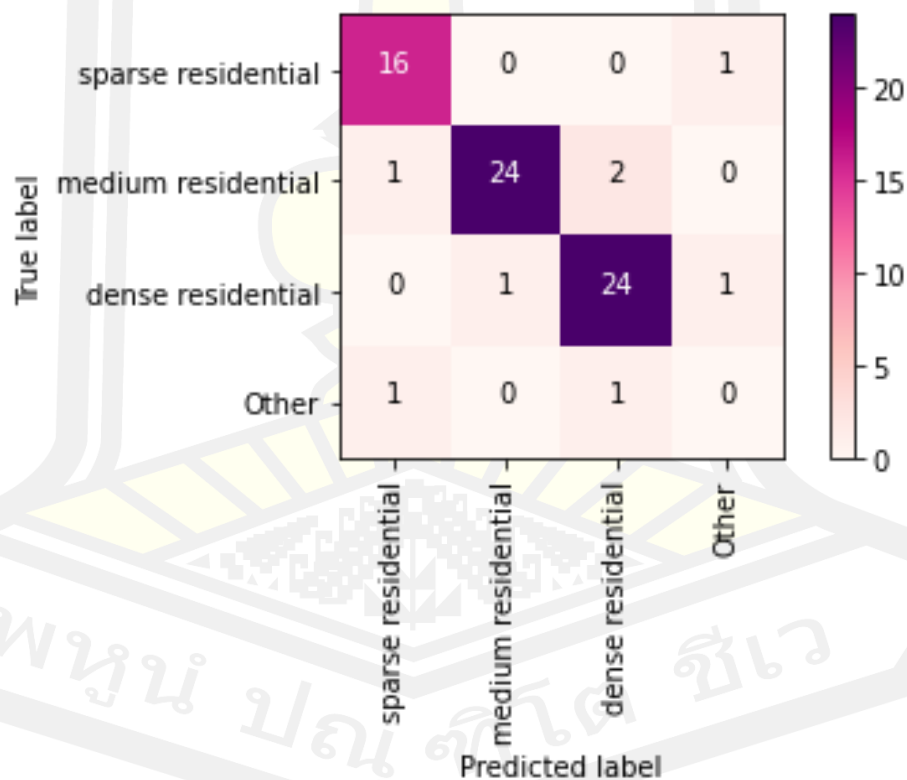
**Table 7** Classification performances (LD, mean validation accuracy, standard deviation and test accuracy) of the snapshot ensemble CNN using different learning rate schedules: CCA, MMCCLR, dropCyclic and training with different state-of-the-art CNNs: MobileNetV2, VGG16, VGG19, on the UCM dataset.

CNNs	LR methods	LD	Validation	Test
MobileNetV2	CCA	<b>0.0422</b>	97.30±0.0085	97.14
	MMCCLR	0.0626	97.10±0.0081	96.43
	dropCyclic	0.0560	<b>97.38±0.0050</b>	<b>97.38</b>
VGG16	CCA	0.1650	95.47±0.0050	<b>93.57</b>
	MMCCLR	<b>0.1371</b>	95.83±0.0042	93.10
	dropCyclic	0.1965	<b>96.51±0.0054</b>	92.62
VGG19	CCA	0.2285	<b>96.98±0.0097</b>	93.10
	MMCCLR	0.1475	96.03±0.0056	93.57
	dropCyclic	<b>0.0896</b>	96.63±0.0020	<b>94.76</b>

I evaluated the snapshot ensemble CNN using three evaluation metrics: LD, validation (mean accuracy and standard deviation), and test accuracy. Note that the LD value presented the best method for preventing the overfitting problem. On examining Table 7, I discovered that all learning rate schedule methods: CCA, MMCCLR, and proposed dropCyclic, can address the problem with overfitting because all learning rate schedule methods achieved low LD values. Consequently, the accuracies of the validation and test did not show an

enormous difference value. As a result, the proposed dropCyclic learning rate schedule outperformed the existing learning rate schedules: CCA and MMCCLR when training the CNN model with MobileNetV2 and VGG19 in terms of test accuracy. The proposed dropCyclic method also outperformed when training with MobileNetV2 and VGG16 on the validation set. In conclusion, the snapshot ensemble CNN using the MobileNetV2 as a backbone CNN and the proposed dropCyclic learning rate schedule ( $drop = 0.95$  and  $c = 10$ ) achieved the highest test accuracy of 97.38% on the UCM dataset.

I illustrated the confusion matrix to show that the snapshot ensemble CNN method can be proposed to learn from many aerial image patterns and even similar patterns between two or more classes, such as classes of residential, including sparse, medium, and dense. The medium residential was misclassified as sparse residential (2 misclassified images) and dense residential (one misclassified image), as shown in Figure 22.



**Figure 22** The confusion matrix of residential classes: sparse, medium, dense that has a similar pattern.

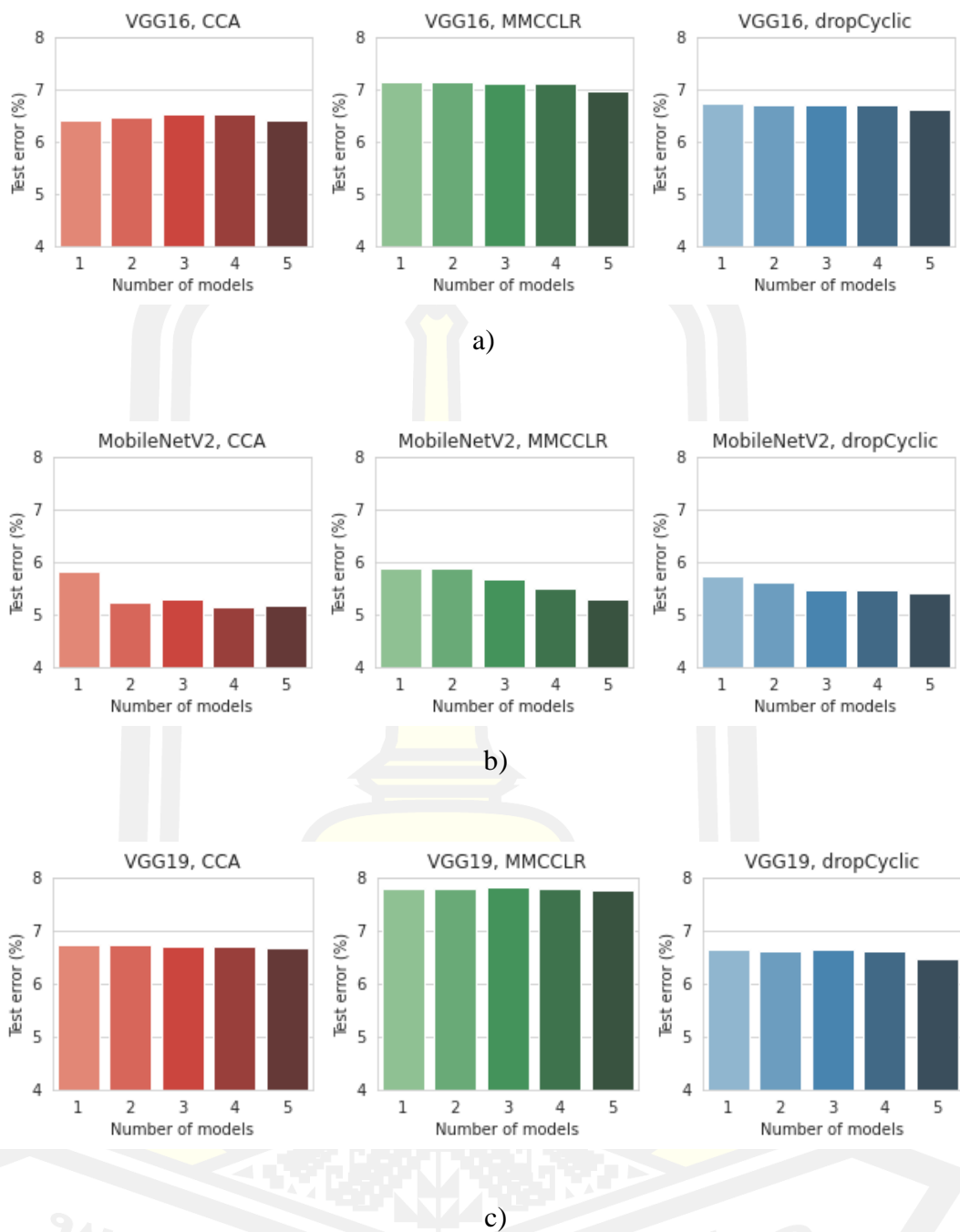
**Table 8** The performance comparison of the snapshot ensemble CNN using the proposed dropCyclic learning rate schedule with existing techniques on the UCM dataset.

Methods	Accuracy	References
GoogLeNet	94.31±0.89	G.-S. Xia et al. (2017)
CaffeNet	95.02±0.81	G.-S. Xia et al. (2017)
VGG-VD-16	95.21±1.20	G.-S. Xia et al. (2017)
AlexNet-SPP-SS	96.67±0.94	X. Han, Zhong, Cao, & Zhang (2017)
GC-DSDM	97.05	Yuan, Li, & Jiang (2017)
IRELBP+SDSAE	<b>97.61±0.36</b>	Zhao, Mu, Yi, & Yang (2018)
dropCyclic	97.38	Proposed

I compared the snapshot ensemble CNN using the proposed dropCyclic learning rate schedule with existing methods. The experimental results in Table 8 show that my method achieved 97.38% on the UCM dataset and outperformed other methods, except only the IRELBP+SDSAE method that slightly obtained better accuracy with 97.61%.

### 3.4.4 Classification results on the AID dataset

I experimented on a snapshot ensemble CNN using three CNNs: MobileNetV2, VGG16, and VGG19 and three learning rate schedules: CCA, MMCCLR, and dropCyclic. The test errors of each experiment are illustrated in Figure 23. The graphs show that combining more models obtained better performance than using a single model. I obtained the lowest test error when using MobileNetV2 as a backbone CNN, as shown in Figure 23(a). Furthermore, using the learning rate schedule with the CCA method outperformed other learning rate schedules on both validation and test sets. The overall performance is shown in Table 9.

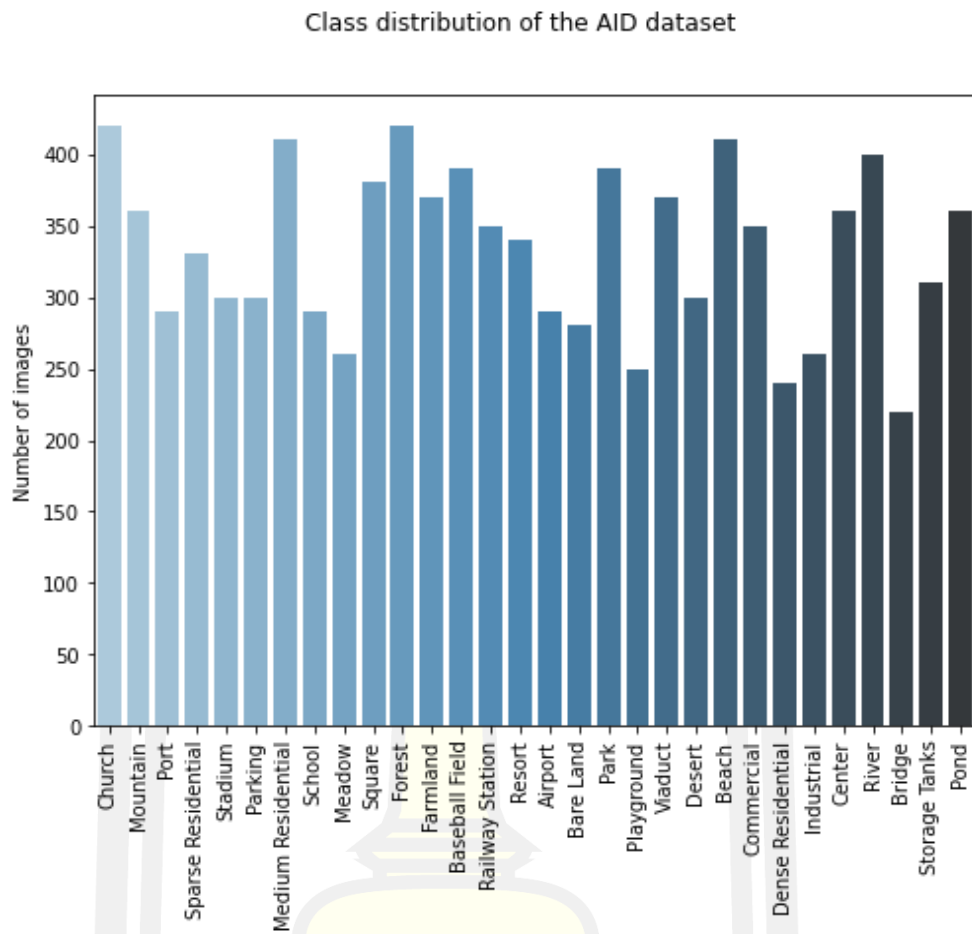


**Figure 23** Loss error (%) of snapshot ensemble CNN with different learning rate schedule methods: CCA (first column), MMCCLR (second column), dropCyclic (third column) and CNN architectures: (a) MobileNetV2 (b) VGG16, (c) VGG19, on the AID dataset. Each snapshot ensemble CNN was trained with  $M=5$  cycles.

**Table 9** Classification performances of the snapshot ensemble CNN using different learning rate schedules and training with different state-of-the-art CNNs on the AID dataset.

CNNs	LR methods	LD	Validation	Test
MobileNetV2	CCA	<b>0.1498</b>	<b>93.84±0.0121</b>	<b>94.86</b>
	MMCCLR	0.2018	93.28±0.0036	94.70
	dropCyclic	0.2052	93.57±0.0065	94.58
VGG16	CCA	0.3217	91.61±0.0084	<b>93.60</b>
	MMCCLR	0.3070	91.27±0.0052	93.02
	dropCyclic	<b>0.2189</b>	<b>91.79±0.0072</b>	93.38
VGG19	CCA	0.3695	<b>92.25±0.0022</b>	93.32
	MMCCLR	0.3243	91.47±0.0051	92.24
	dropCyclic	<b>0.3208</b>	91.93±0.0070	<b>93.54</b>

Table 9 shows the performance of the snapshot ensemble CNN methods on the AID dataset, which is an unbalanced dataset because each class has between 220 to 420 aerial images (see Figure 24). I used the LD value to measure the overfitting problems that can be found when training the CNN model. I found LD values between 0.1-0.3 with all experiments. Hence, the test accuracies of all experiments were higher than the validation accuracies. The snapshot ensemble CNN can address the problem with overfitting problems. As a result, using MobileNetV2 and the CCA learning rate schedule achieved 94.86% accuracy and outperformed other methods on the AID dataset.

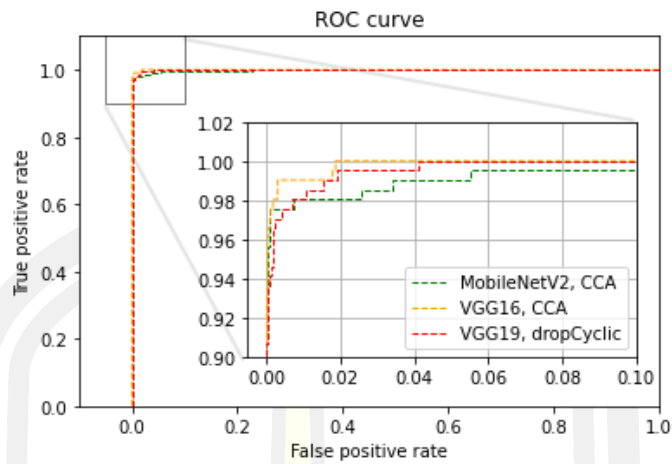


**Figure 24** Class distribution of the AID dataset.

Three models of snapshot ensemble CNN with various learning rate schedules were selected for receiver operating characteristic (ROC) comparison, as shown in Figure 25. The snapshot ensemble CNN using MobileNetV2 and CCA learning rate schedule attained an AUC value of 0.9982. Using VGG16 and the CCA learning rate schedule and VGG19 and the dropCyclic learning rate schedule achieved AUC values of 0.9982 and 0.9981, respectively.

Consequently, the dropCyclic learning rate schedule outperformed other learning rate schedules when training with the VGG19. Moreover, I concluded that the snapshot ensemble CNN using MobileNetV2 could address the unbalanced data better than VGG16 and VGG19 architectures.





**Figure 25** Illustration of the ROC curve for snapshot ensemble CNN models. The highlighted area is zoomed in at the upper left area of the curve.

**Table 10** The performance comparison of the snapshot ensemble CNN using the proposed dropCyclic learning rate schedule with existing techniques on the AID dataset.

Methods	Accuracy	References
GoogLeNet	86.39±0.55	G.-S. Xia et al. (2017)
CaffeNet	89.53±0.31	G.-S. Xia et al. (2017)
VGG-VD-16	89.64±0.36	G.-S. Xia et al. (2017)
IRELBP+ SDSAE	94.11±0.55	Zhao et al. (2018)
dropCyclic	94.58	Proposed
BiMobileNet	<b>96.87±0.23</b>	D. Yu et al. (2020)

Table 10 compares the performance of the proposed dropCyclic method with other models for aerial image classification. The proposed method outperformed other methods, except for the BiMobileNet method. The proposed method achieved an accuracy of 94.58%, while the BiMobileNet method obtained an accuracy of 96.87% on the AID dataset. The snapshot ensemble CNN method based on the proposed dropCyclic method did not solve the unbalanced dataset problem.

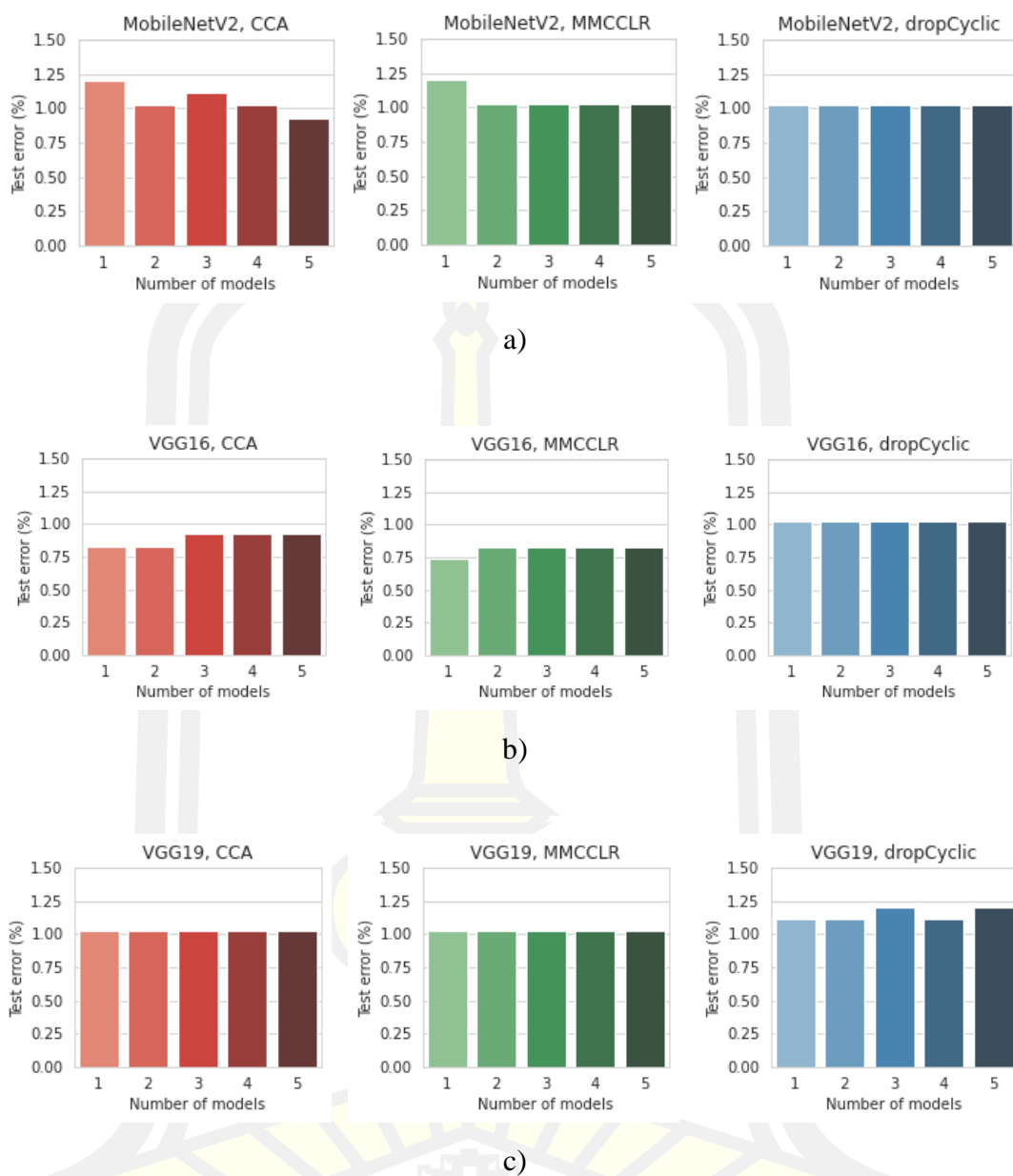
### 3.4.5 Classification results on the EcocropsAID dataset

This experiment discovered that combining only one or two models could achieve the lowest test error, as shown in Figure 26. Because the EcoCropsAID dataset has only five classes and each class contains ~1,000 aerial images, the CNN models can cope well with many patterns in each class and are well classified. The overall performance is shown in Table 11.

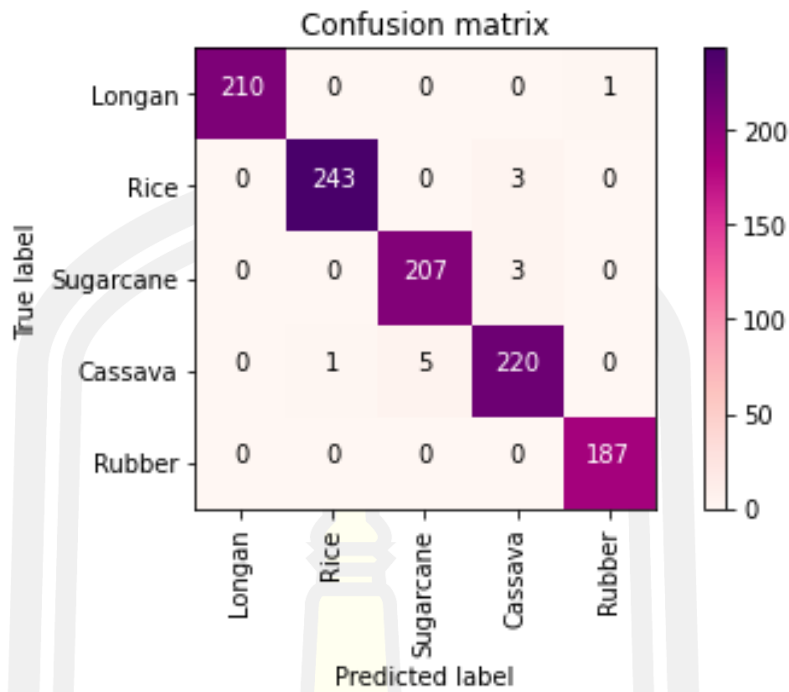
Table 11 compares the performance of the snapshot ensemble CNN methods on the EcoCropsAID dataset. All the snapshot ensemble CNN models were trained using five cycles. I found that all the experiments prevented the overfitting problems with the small LD values between approximately 0.006 - 0.01. In this experiment, training the VGG16 using the dropCyclic and the CCA learning rate schedules achieved the highest validation accuracy of 99.60%. However, the MMCCLR learning rate schedule slightly outperformed with an accuracy of 99.26%. Figure 27 shows the confusion matrix of the snapshot ensemble CNN based on dropCyclic and MobileNetV2 on the EcoCropsAID dataset and illustrates only a few misclassifications.

**Table 11** Classification performances of the snapshot ensemble CNN using different learning rate schedules and training with different state-of-the-art CNNs on the EcoCropsAID dataset.

CNNs	LR methods	LD	Validation	Test
MobileNetV2	CCA	0.0010	99.37±0.0025	<b>99.07</b>
	MMCCLR	<b>0.0061</b>	99.32±0.0012	98.98
	dropCyclic	0.0041	<b>99.43±0.0038</b>	98.98
VGG16	CCA	0.0100	<b>99.60±0.0012</b>	99.17
	MMCCLR	<b>0.0098</b>	99.43±0.0017	<b>99.26</b>
	dropCyclic	0.0343	<b>99.60±0.0012</b>	98.98
VGG19	CCA	<b>0.0177</b>	<b>99.29±0.0019</b>	<b>98.98</b>
	MMCCLR	0.0280	99.13±0.0027	<b>98.98</b>
	dropCyclic	0.0492	99.12±0.0023	98.89



**Figure 26** Illustrated loss error (%) of snapshot ensemble CNN with different learning rate schedule methods: CCA (first column), MMCCLR (second column), dropCyclic (third column) and CNN architectures: (a) MobileNetV2 (b) VGG16, (c) VGG19, on the EcoCropsAID dataset. Each snapshot ensemble CNN was trained with  $M=5$  cycles.



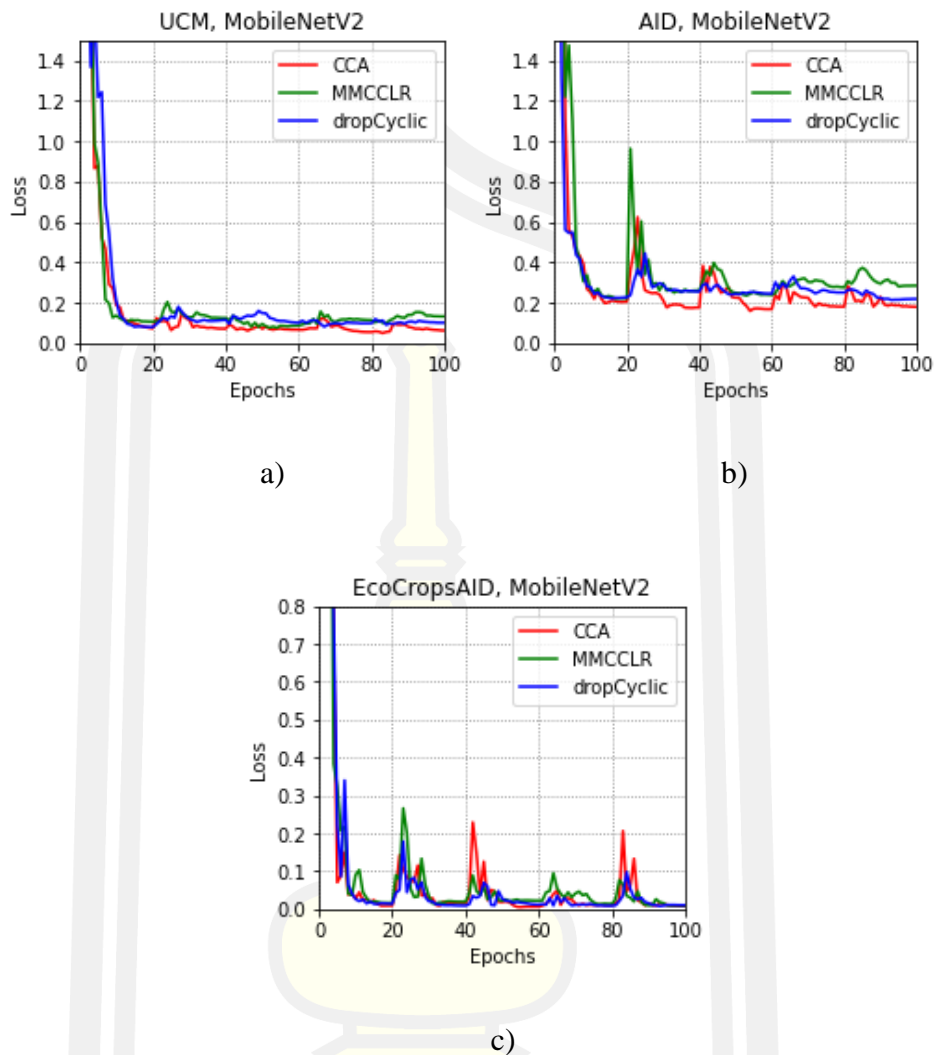
**Figure 27** The confusion matrix of the snapshot ensemble CNN based on dropCyclic and MobileNetV2 on the EcoCropsAID dataset.

In addition, I compared the snapshot ensemble CNN based dropCyclic learning rate schedule and MobileNetV2 with the existing method (Noppitak & Surinta, 2021). The result showed that I proposed method achieved an accuracy that was 6% higher than the existing method, which achieved only 92.80%.

## 3.6 Discussion

### 3.6.1 Loss error curve of the different learning rate schedules

I discovered that the learning rate parameter directly affects the accuracy of the CNN model. Hence, I have seen existing research focused on tuning the learning rate parameter (X. Wu, Ward, & Bottou, 2018; Y. Wu et al., 2019; Xue, Li, & Luo, 2022). However, the learning rate parameter was not the primary priority in my experiment because the dropCyclic method was proposed to change the maximum of the learning rate in each cycle. Further, the maximum learning rate was decreased in each cycle according to the drop parameter, as presented in Figure 19.



**Figure 28** Illustration of the training loss values when training with the snapshot ensemble CNN using the MobileNetV2 and evaluated with various learning rate schedules: CCA, MMCCLR, dropCyclic on (a) UCM, (b) AID, and (c) EcoCropsAID datasets.

Figure 28 shows the training loss values of each learning rate schedule: CCA, MMCCLR, and dropCyclic when training with the snapshot ensemble CNN using the MobileNetV2 on UCM (see Figure 28(a)), AID (see Figure 28(b)), and EcocropsAID (see Figure 28(c)) datasets. In this experiment, the hyperparameters were adjusted, as shown in Table II. I adjusted the cycle of the snapshot ensemble CNN to 5 cycles. The loss values started with a high value and quickly decreased to the lowest, called the local minimum. Subsequently, the loss values increased and then decreased again to the lowest value in the next cycle to find other local minimum values. I then snapped the best CNN model at the local minimum value of

each cycle and used it in the ensemble method. However, only training on the UCM dataset showed that the loss value did not increase too much because the aerial images in the UCM dataset contained only 2,100 images. Hence, the CNN model can learn and create the model that is appropriate with a small number of aerial images.

### 3.6.2 Cosine Cyclic Learning Rate Schedule with Max and Min Values

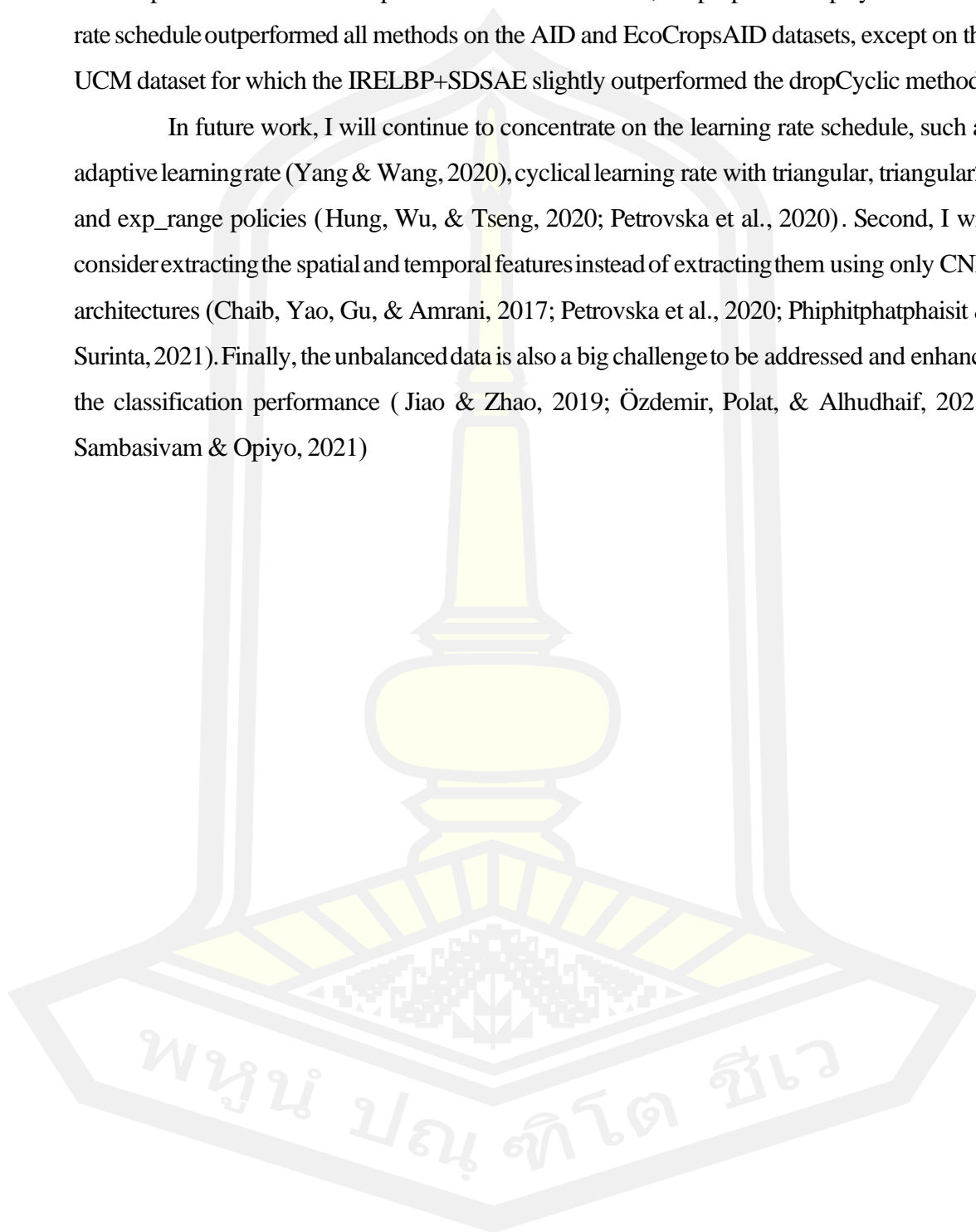
In my proposed dropCyclic learning rate schedule, I utilized the idea of a step decay schedule to drop the learning rate in every epoch by adding the step decay process into the cosine cyclic learning rate schedule (see Equation (7)). In my dropCyclic method, only two parameters were required. The *drop* and *c* parameters. The *drop* parameter was proposed as a step decay that drops the learning rate by half every *n* epoch. The *c* parameter allows the model to shift to the new local minimum in the next cycle, as shown in Figure 19(b). As a result, the best dropCyclic parameters were  $drop = 0.95$  and  $c = 10$ . Consequently, the proposed dropCyclic learning rate schedule outperformed other learning rate schedules on the UCM dataset. Further, the proposed dropCyclic learning rate schedule achieved high accuracy on the AID and EcoCropsAID aerial image datasets. In conclusion, the proposed dropCyclic learning rate schedule has the advantage that it restricts the maximum learning rate in each cycle by using step decay parameters: *drop* and *c*. Hence, the maximum learning rate is not the priority parameter required to adjust.

## 3.7 Conclusion and future work

This research proposed a new learning rate schedule called the dropCyclic. I developed the concept of the step decay schedule that decreases half of the learning rate value in every *c* epoch, call drop. The *drop* parameter was contained in the cosine cyclic learning rate schedule. It contained two parameters, the *drop* and *c* parameters. The benefit of the dropCyclic learning rate schedule is that the learning rate was dropped in the next cycle according to the drop parameter. The method allows the convolutional neural network (CNN) model to discover the new local minimum in the subsequent cycle using the *c* parameter. I evaluated the proposed dropCyclic learning rate schedules and the existing methods: cyclic cosine annealing (CCA) and max-min cyclic cosine learning rate scheduler (MMCCLR) on three aerial image datasets, including UCM, AID, and EcoCropsAID datasets. Three CNN architectures were compared for the backbone CNN architectures, consisting of MobileNetV2, VGG16, and VGG19. The proposed dropCyclic learning rate schedule achieved the best results

on the UCM dataset. The dropCyclic method obtained very high results on the AID and EcoCropsAID datasets. In comparison with other methods, the proposed dropCyclic learning rate schedule outperformed all methods on the AID and EcoCropsAID datasets, except on the UCM dataset for which the IRELBP+SDSAE slightly outperformed the dropCyclic method.

In future work, I will continue to concentrate on the learning rate schedule, such as adaptive learning rate (Yang & Wang, 2020), cyclical learning rate with triangular, triangular2, and exp\_range policies (Hung, Wu, & Tseng, 2020; Petrovska et al., 2020). Second, I will consider extracting the spatial and temporal features instead of extracting them using only CNN architectures (Chaib, Yao, Gu, & Amrani, 2017; Petrovska et al., 2020; Phiphitphatphaisit & Surinta, 2021). Finally, the unbalanced data is also a big challenge to be addressed and enhance the classification performance (Jiao & Zhao, 2019; Özdemir, Polat, & Alhudhaif, 2021; Sambasivam & Opiyo, 2021)





# Chapter 4

## Instance Segmentation

Land use is constantly changing, and water plays a critical role in the change process. If changes are noticed quickly or are predictable, land use planning and policies can be devised to mitigate almost any problem. Accordingly, I present a mask region-based convolutional neural network (Mask R-CNN) for water body segmentation from aerial images. The system's Aerial image water resources dataset (AIWR) was tested. The AIWR areas were agricultural and lowland areas that require rainwater for farming. Many wells were located throughout the agricultural areas. The AIWR dataset presents two types of data: natural water bodies and artificial water bodies. The two different areas appear as aerial area images that are different in colour, shape, size, and similarity. A pre-trained model of Mask R-CNN was used to reduce network learning time. ResNet-101 was used as backbone architecture. The aerial images gathered in the learning process was limited, and only 720 images were produced. I used data augmentation to increase the amount of information for training using affine image transformation, including scale, translation, rotation, and shear. The experiment found that mask R-CNN architecture can specify the position of the water surface. The evaluation method in this case is the mAP value.

### 4.1 Introduction

The two terms “land cover” and “land use” are typically used together (Storie & Henry, 2018). Over the past ten years the difference between land cover and land use has attracted many researchers (Caldas et al., 2015) prompted by a change in land cover to accommodate changes in land use. As such, if land use data are accurate and up-to-date, that information can be applied to many objectives, such as city planning, environmental audit or evaluation, and national policy (Treitz & Rogan, 2004).

Elagouz et al. tracked land use in the Nile River, Egypt with RS technology to determine the impact of land changes in urban areas during and after the year 2011. The land changed because of the unplanned expansion of a nearby city. (Jazouli, Barakat, Khellouk, Rais, & Baghdadi, 2019) said that soil erosion was the most important cause of land degradation throughout the world. Jazouli et al. predicted the impact of land use changes, which

affect soil erosion, in the Oum Er Rbia basin, Morocco. They studied mountainous areas with steep, slopes, and clay soil and such places face a higher risk of soil erosion. Soil erosion is sometimes caused by human activities and local weather. Further research would be beneficial for generating land use prediction maps, detecting land use changes, and creating yearly mapping for soil erosion.

Nowadays, deep learning research is very popular, For example, land cover analysis research (Storie & Henry, 2018). Such studies used deep neural networks for analysis of Landsat 5/7 satellite images to show land cover maps for agriculture, including agriculture areas, water, grass, mixed woodland, and border. (Kussul et al., 2017), used convolutional neural network (CNN), which is the method for classification of recorded images, in remote sensing work. The CNN classified recorded images in categories of optical and synthetic aperture radar (SAR) derived from Landsat-8 and Sentinel-1A using CNN type one-dimensional (1-D) and 2-D. The results from CNN were compared with the random forest method and the ensemble neural networks technique. 2D-CNN achieved the highest score with a 94.6% accuracy rate. However, 2D-CNN still has some problems distinguishing small objects. Spatial resolution of the satellite images is 30 meters, which is low resolution.

Miao et al. presented water body segmentation using restricted receptive field deconvolution network (RRF DeconvNet) for extraction of water body from high-resolution spectrum images. This method did not require infrared spectrum images, and also decreased the blurring boundaries problem by using a new loss function called edges weighting loss (EWLoss). I tested this method with a dataset collected from Google Earth. The images from Google earth were in the visible spectrum at 50 meters spatial resolution of rural areas at Suzhou and Wuhan, China. The experiments showed that RRF DeconvNet method using EWLoss obtained an accuracy rate of 96.9%..

Wen et al. used Mask R-CNN to segment the building area and the background from Google Earth images. They created a new dataset with 2,000 aerial images in Fujian province, China. The sizes of images used was in the experiment range from 1,000x1,000 to 10,000x10,000 pixels. All aerial images were tagged with a label. In the experiment, the researcher used a pre-trained model with ResNet architecture. All images were resized to 500x500 pixels. The result showed that Instance Segmentation using Mask R-CNN resulted in a mean Average Precision (mAP) value at 0.9063.

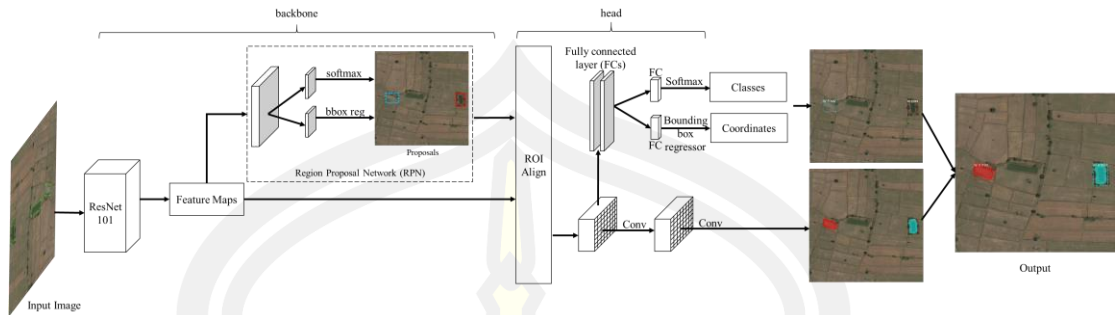
**Contribution:** This article presents a mask region-based convolutional neural network (Mask R-CNN) for water body segmentation from aerial images. This method has been called instance segmentation. ResNet-101 was used as backbone architecture. Mask R-CNN architecture was tested with aerial image water resources dataset (AIWR). The AIWR comprises images of agricultural areas in the northeast region of Thailand; these are fertile agricultural areas where people grow rice. The areas require rainwater for farming and many wells can be identified throughout the agricultural area. Data were collected for two types of water body; natural water bodies (W1) and artificial water bodies (W2). The aerial images of water bodies were different in color, shape, size, and similarity. This dataset included 800 images, so the AIWR dataset challenges the instance segmentation process.

This research also attempted to add data augmentation in the category of affine image composed of 4 different methods: scale, translation, rotation, and shear. Augmentation processes were used only in the training process. Data augmentation would be a random parameter value. The images, which trained in each epoch using mask in the R-CNN process, were different. The experiments found that data augmentation had improved the performance of Mask R-CNN in the instance of the segmentation process when used with AIWR Dataset. The result showed better performance for specifying water bodies. The mAP value increases from 0.30 to 0.59 when the researcher used data augmentation.

## 4.2 Mask R-CNN Architecture

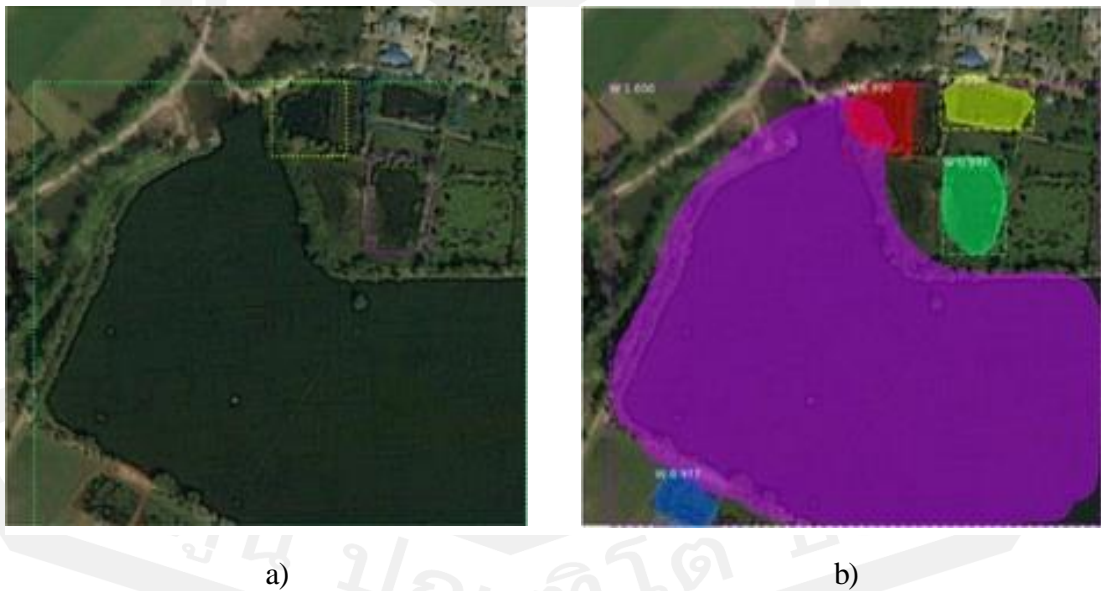
Mask R-CNN was presented by (He, Gkioxari, Dollár, & Girshick, 2020) in 2017 for improving instance segmentation performance. Mask R-CNN was developed from Faster R-CNN, which was presented by (Ren, He, Girshick, & Sun, 2017).

Faster R-CNN was designed to use a convolutional network (ConvNet) for feature map extraction of Images. ConvNet can use VGGNet and ZFNet architectures. After that, the region proposal network (RPN) was used for inspecting the object areas. RPN operates location inspection for each object. The reason for running RPN was to create a bounding box of each object, which is called a ROI pooling layer. In the ROI pooling layer, the ROI in each section were sent to fully connected layers (FCs) for ROI feature vector calculation before sending the value to the softmax function for consideration of ROI as an object. After that, the function will predict the object type in ROI as shown in Figure 29.



**Figure 29** Mask R-CNN Framework

According to the introduction, Faster R-CNN (Ren et al., 2017) is an object detector, so this function cannot specify an object in pixel-to-pixel, or it would be referenced to as instance segmentation as shown in Figure 30 a). Mask R-CNN has been designed to help instance segmentation by using the capabilities of RPN to specify ROI. The next step is to segment the ROI areas to specify the edge of an object as shown in Figure 30 b).



**Figure 30** Result from a) Faster R-CNN method and b) Mask R-CNN.

#### 4.2.1 Backbone Architecture

The backbone architecture consists of 2 main networks, ConvNet and RPN. ConvNet used in this research is ResNet-101 architecture, using a pre-trained model derived

from learning of the COCO dataset. This architecture can reduce network learning time. The main function of ResNet architecture is to extract feature maps from aerial images, then use the region proposal network (RPN) to find the location of an object using ResNet-101 architecture.

#### **4.2.2 Head Architecture**

An advantage of Mask R-CNN is that it can perform instance segmentation by using the location of any object, derived from RPN, which is another name for the region of interest (ROI). The ROI will be considered whether it is an object or not. If the ROI area is an object, then types of an object would be considered in the next step. This step is similar to Faster R-CNN. After that, areas will be calculated for intersection over union (IoU). As shown in Equation (11), the IoU values are assigned to be greater or equal to 0.5.

Any area with an IoU value greater or equal to 0.5 is required to find the perimeter of an image. Sometimes, this method is also known as a segmentation mask. This process is an additional process from Faster R-CNN. In each ROI area, there is only one class. Then, the semantic segmentation model is created. It is the same as using binary classification to distinguish an object from background.

### **4.3 Aerial Image Water Resources Dataset**

According to the standard of land use code by fundamental geographic data set (FGDS), Thailand (GISTDA, 2013) land use classification requires the analysis and transformation of satellite images data together with field survey data. In this article, I studied only land use in water bodies. The water bodies in this research can be divided into 2 levels: natural body of water (W1) artificial body of (W2) water.

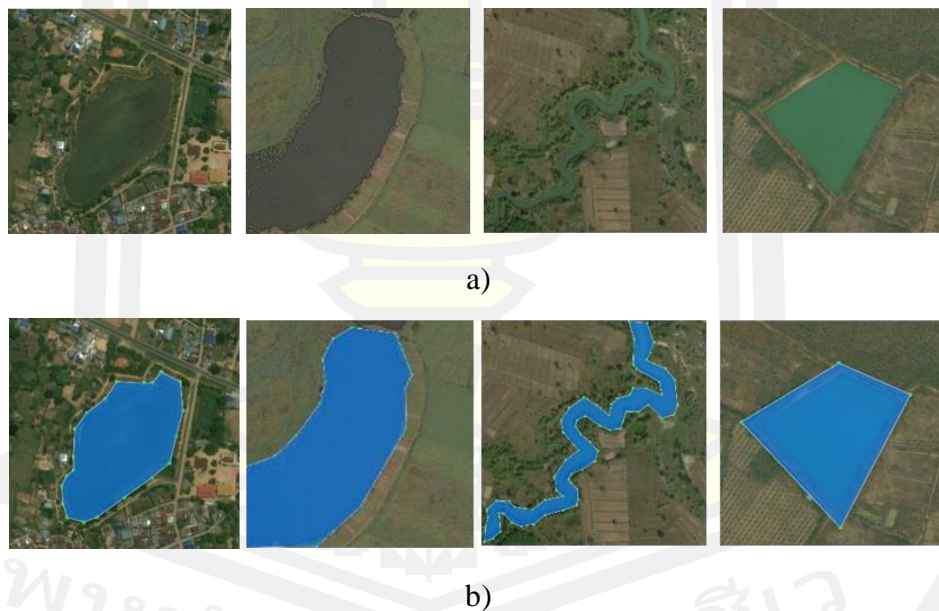
The deep learning method was used for aerial image data analysis. The aerial images were derived from Bing map by collecting only data in the northeastern region of Thailand. The northeast of Thailand is a lowland area mainly used for growing rice, there are also agricultural areas that rely on rainwater for agriculture and as such, there are many ponds in and around the agricultural areas.

The experiments in the study used the Mask R-CNN algorithm which is a suitable method for performing instance segmentation. The model in this experiment can be further developed and applied to water management tasks. Farmers in the northeastern region of Thailand can also create water management plans.



The aerial image data used in this research was 1:50 meters. Every aerial image had 650 x 650 pixels. Those images included water bodies type W1 and W2 as shown in Figure 31 a). Ground truth of all aerial images was set for before sending it to be analyzed and interpreted by remote sensing experts. This assured that the water bodies groupings were correct. An example of ground truth, which has been checked by experts as shown in Figure 31 b). Ground truth has been used in learning the algorithm in deep learning mode and was also used in further evaluation.

The aerial images used in this experiment consisted of water body: types W1 (see, Figure 31, Column 1, 2, and 3) and W2 (see, Figure 31 Column 4). The Aerial Image Water Resources Dataset, AIWR has 800 images. Data were chosen at random and divided into 3 sections: training, validation, and test set with ratio 8:1:1. Therefore, 640 aerial images were used for learning and creating the model, 80 images were used for validation, and the remaining 80 images were used for test.



**Figure 31** Example of aerial images. a) Water bodies W1 and W2 b) ground truth of water resources.

This dataset presents a challenge to the instance segmentation process because the water bodies include W1 and W2 types. There are 4 challenging objectives: color, shape, size, and similarity as follows.

- Color. Figure 32 a) shows that water bodies have different color, for example white, blue, gray and black. Some areas are covered by unwanted flora, so the images are seen as dark green and black.
- Shape. The shapes of the areas have different characteristics such as triangles, squares, curves, U-shaped and zigzag as shown Figure 32 b).
- Size. The water body sizes are different. Size measurement in Bing maps found that the water bodies sizes range from 10, 20, 30, 60 and 120 meters as shown in Figure 32 c). When researchers observe 10 meter wide water sources, only a small point can be seen.
- Similarity. Aerial images of some water bodies are similar to other types of land use, for example flooded areas, water areas that are obscured by trees, or buildings on water areas etc. Figure 32 d) uses dotted lines to show areas that have the characteristics as mentioned above

#### 4.4 Experiment and discussion

A deep learning algorithm was used in this research for instance segmentation. This method can identify the areas in pixel-to-pixel by using Mask R-CNN architecture. The method is suitable for water body segmentation because it can analyze both natural water bodies and artificial water bodies. The data were collected from aerial image data from agricultural areas in the northeastern region of Thailand, there was a total of 800 aerial images. Those images were divided by the 10-fold cross-validation method. There were 720 images for training, and 80 images were used for test. All aerial images were resized to 512x512 pixels.

In this research, the TensorFlow platform was used for training and testing the Mask R-CNN algorithm which runs on GPU GeForce GTX 1070 Ti, Intel(R) Core-i5, 7400CPU @ 3.00GHz, 8GB RAM, Linux operating system. ResNet architecture is backbone architecture for learning aerial imagery learning. This research used transfer learning (Bunrit et al., 2019) to reduce learning time of ResNet architecture. A pre-trained model of ResNet-101 architecture, which was derived from the learning process of COCO dataset, was also used. I then used the mentioned model to perform Fine-Tune for adjusting the parameters in order to make it become suitable for the AIWR dataset.

The parameters used for Fine-tune consist of NUM\_CLASSES=2, BATCH\_SIZE=4, FPN\_CLASSIF\_FC\_LAYERS\_SIZE=512, IMAGES\_PER\_GPU = 1, IMAGE\_



MIN\_DIM = 512, IMAGE\_MAX\_DIM = 512, IMAGE\_SHAPE=[512, 512, 3], RPN\_ ANCHOR\_SCALES= (8, 16, 32, 64, 128), STEPS\_PER\_EPOCH=100, TRAIN\_ROIS\_ PER\_IMAGE=32, VALIDATION\_STEPS=5, and LEARNING\_RATE=0.0001.

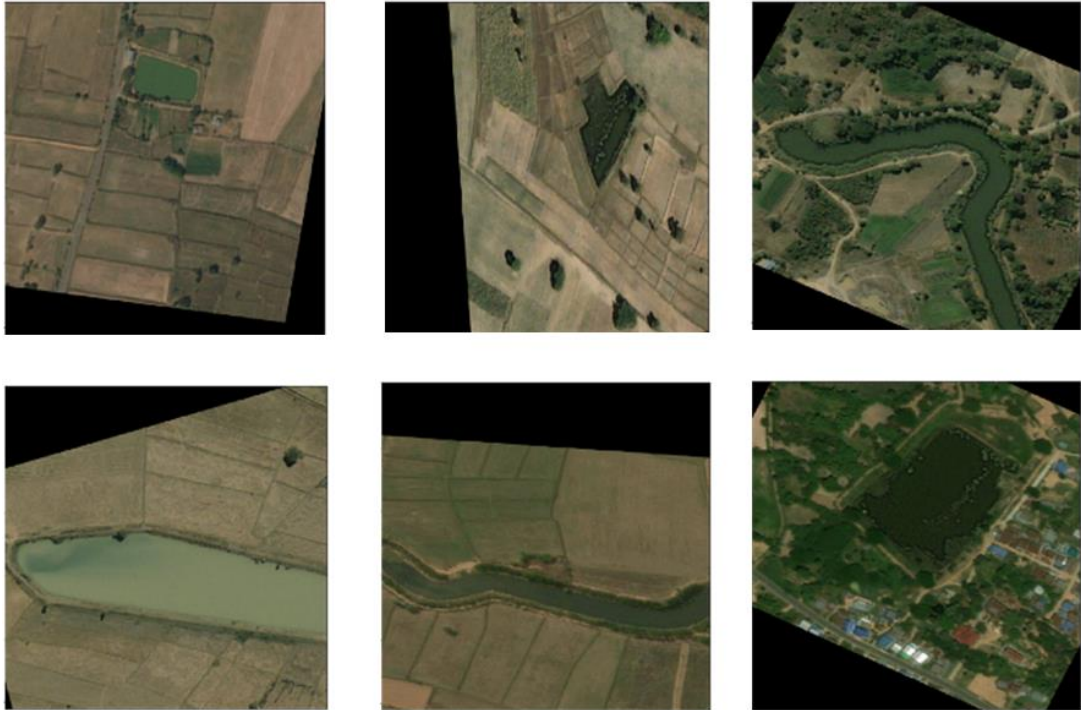


**Figure 32** Challenges of instance segmentation of collected data are a) color, b) shape, c) size, and d) similarity.

One of the deep learning problems is that the amount of training data is too small. A common way to solve the problem is to perform data augmentation, which can be divided into 2 groups including the traditional, white-box method or black-box method. Two common

methods for image augmentation in traditional transformations are affine image transformations and color modification (Mikołajczyk & Grochowski, 2018).

This research used data augmentation, affine image transformations series, which includes  $\text{scale}=\{\text{"x"}: (0.8, 1.2), \text{"y"}: (0.8, 1.2)\}$ ,  $\text{translate\_percent}=\{\text{"x"}: (-0.2, 0.2), \text{"y"}: (-0.2, 0.2)\}$ ,  $\text{rotate}=( -25, 25)$ , and  $\text{shear}=( -8, 8)$ . Examples of aerial images obtained after data augmentation are shown in Figure 33.



**Figure 33** Examples of data augmentation.

#### 4.4.1 Model Evaluation

To Evaluate the Mask R-CNN algorithm, I used mean average precision (*mAP*) (Everingham, Van Gool, Williams, Winn, & Zisserman, 2010), which is a method for evaluating the effectiveness of image retrieval by an intersection over union (*IoU*) calculation from the following Equation (11).

$$IoU = \frac{\text{area}(B_p \cap B_{gt})}{\text{area}(B_p \cup B_{gt})} \quad (11)$$

where  $B_p \cap B_{gt}$  are the areas of intersection between the predicted area. Ground truth ( $gt$ ) is bounding boxes and  $B_p \cup B_{gt}$  is the area of union, determined by the value of  $IoU \geq 0.5$

After that, true positive ( $TP$ ), a correct detection, and false positive ( $FP$ ) (A wrong detection) were calculated. The detection was performed with  $values \geq 0.5$ , false negative ( $FN$ ) (A ground truth not detected) and true negative ( $TN$ ) (corrected misdetection). The  $TP, FP, FN, TN$  value are taken to calculate precision ( $P$ ) and Recall ( $R$ ) value.

The  $AP$  value was considered as an average of maximum precision at a set of 11-spaced recall levels. Equation (12) is as follows:

$$AP = \frac{1}{11} \sum_{r \in \{0,0.1,\dots,1\}} P_{inter p}(r) \quad (12)$$

$$\text{with } P_{inter p}(r) = \max_{\tilde{r}, \tilde{r} \geq r} p(\tilde{r})$$

where  $p(\tilde{r})$  is the measured precision at recall  $\tilde{r}$ .

After that,  $mAP$  value is calculated as the following Equation (13).

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (13)$$

where  $N$  is number of queries.

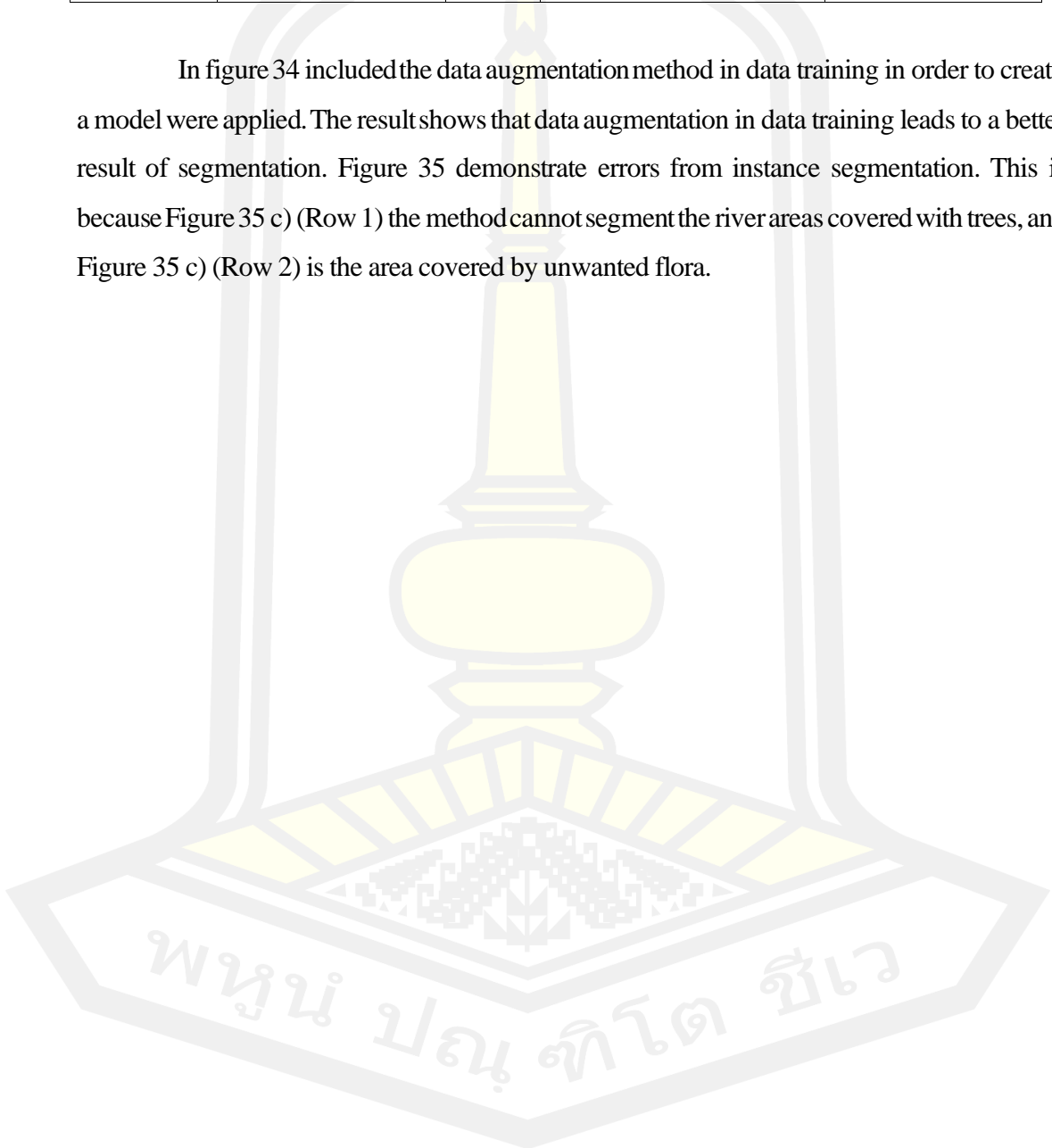
#### 4.4.2 Result of Instance Segmentation of Water Bodies

Table 12 shows the results of the experiment of mask R-CNN architecture to segment water bodies from the AIWR dataset. Augmentation data experiments of AIWR dataset were performed by affine image transformations method, including scale, translation, rotation, and shear. The result shows that the loss error value from training processes was up to 1.08. That resulted in the  $mAP$  being as low as 0.30, but when I tested again using data augmentation, the loss errors were reduced to only 0.41 and the  $mAP$  increased to 0.59, which is almost 2 times higher. However, the data augmentation process takes 12 days and 9 hours to learn.

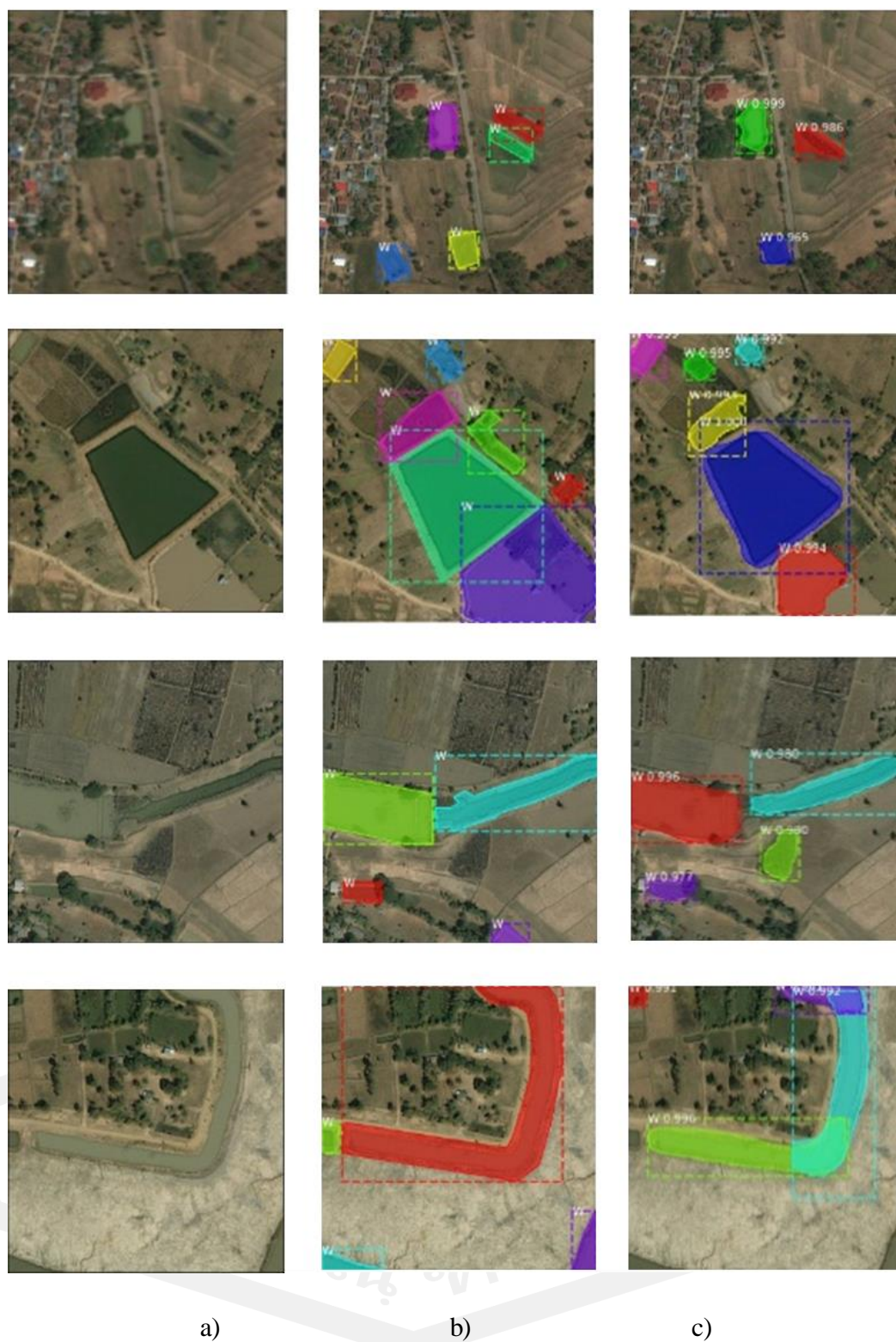
**Table 12** The result of the experiment using mask R-CNN with the AIWR Dataset.

Augment	Validation loss	mAP	Training time	Test time/image
False	1.08	0.30	11d 15h 16min 27s	3 $\mu$ s
True	0.41	0.59	12d 9h 48min 25s	4 $\mu$ s

In figure 34 included the data augmentation method in data training in order to create a model were applied. The result shows that data augmentation in data training leads to a better result of segmentation. Figure 35 demonstrate errors from instance segmentation. This is because Figure 35 c) (Row 1) the method cannot segment the river areas covered with trees, and Figure 35 c) (Row 2) is the area covered by unwanted flora.

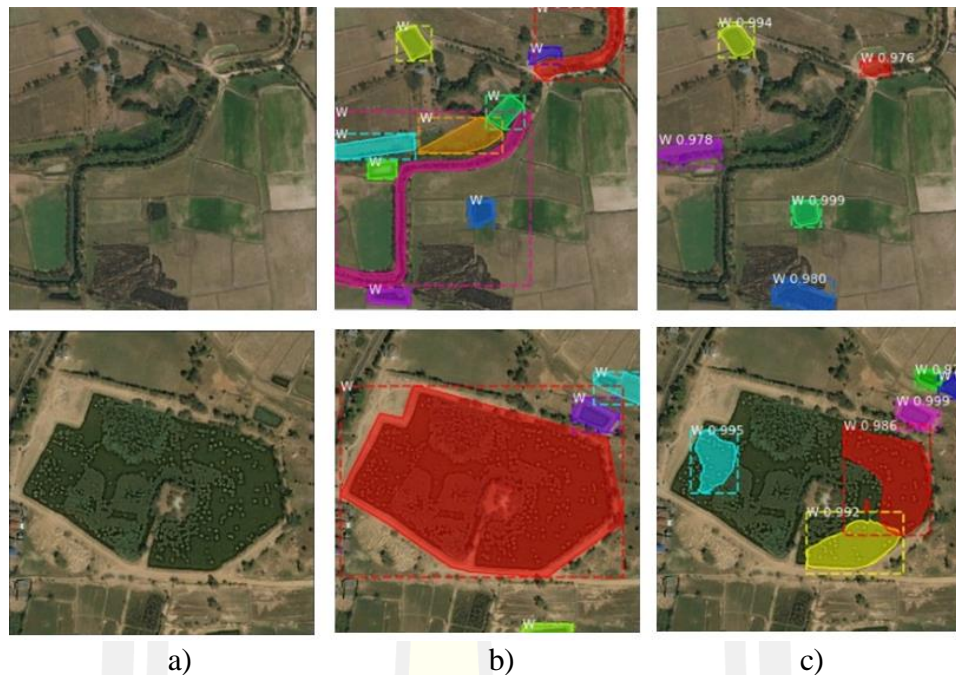






**Figure 34** Result of instance segmentation using mask R-CNN with data augmentation.

a) Aerial images b) images with ground truth, and c) instance segmentation.



**Figure 35** Error results in segmentation. a) Aerial images b) images with ground truth and c) error of instance segmentation.

## 4.5 Conclusion

This paper evaluated the accuracy of instance segmentation by Mask R-CNN together with data augmentation using aerial images of water resources dataset (AIWR). The mAP values were used as the measuring method. The areas are the lowlands which require rainwater for farming. The challenges of the AIWR dataset arise because it is a collection of images of 2 types of water bodies; natural water bodies and artificial water bodies. The two types of data are different in color, shape, size, and similarity. This paper used a pre-trained model to reduce learning time of the Mask R-CNN. This research has shown that the mask R-CNN architecture combined with data augmentation can identify the water surface using the mAP value for measurement. The value was high as 0.59 and this was almost two times greater than not using the data augmentation method.

Because the data tested is aerial photography obtained from Bing map, only RGB colors can be evaluated. If future research can use data from satellites, such as Landsat, which has a band specifically for water analysis, the result of an analysis of water bodies with different color might give higher accuracy. Any new architecture suitable for water body analysis might be used to expect an even higher accuracy rate.



## Chapter 5

### Discussion

In this thesis, I have demonstrated that the proposed algorithms are very efficient for improving land use and land cover recognition on aerial images. The findings of this research, I contribute two main types of research; classification and segmentation, to address the challenge of aerial images using deep learning techniques. I will now briefly discuss the challenges of the land use and land cover classification systems and my findings.

**For the classification task,** I have demonstrated that the convolutional neural networks (CNNs) had many hyperparameters to optimize, such as activation function, learning rate, and batch size. There are many techniques to enhance the accuracy of the CNN model, including applying data augmentation techniques while training the CNN model and employing ensemble CNNs. In this research, I introduce the novel economic crops aerial image dataset, called the EcoCropsAID dataset. It is collected in Thailand from five economic crops (rice, sugarcane, cassava, rubber, and longan). The aerial images of the economic crops are selected based on Agri-Map Online given by the Ministry of Agriculture and Cooperatives and the National Electronics and Computer Technology Center (NECTEC). First, I proposed to use various CNN models, such as InceptionResNetV2, MobileNetV2, DenseNet201, Xception, ResNet12V2, NASNetLarge, VGG16, and VGG19, to classify the economic crops aerial images. The experimental results did not guarantee high accuracy. Second, I applied the data augmentation with only basic techniques; height shift, width shift, rotation. This technique provided better accuracy results when compared with without using data augmentation techniques. Third, I proposed to use a snapshot ensemble CNN method to classify the economic crops aerial images. The experimental results showed that the snapshot ensemble CNN method outperformed the CNNs. Finally, I proposed a novel learning rate schedule that aimed to increase the performance of the snapshot ensemble CNN method, called the drop cosine cyclic (dropCyclicLR) learning rate schedule. In the dropCyclicLR, only one parameter was required to adjust, which was the drop rate. The performance of the snapshot ensemble CNN method was improved when using the dropCyclicLR learning rate schedule.

**For the segmentation task,** I applied the mask region-based CNN, called mask R-CNN, to segment the water body from aerial images. In this research, I collected many water

body images from Google Earth, namely aerial image water resources dataset (AIWR). It contained two types of water bodies: natural and artificial water bodies, which are different in color, shape, size, and similarity. For the mask R-CNN, it contained two architectures; backbone and head. In the backbone architecture, I combined the ResNet-101 and region proposal network (RPN) to find the location of water bodies. In the head architecture, the regions detected from the backbone architecture were transferred to segment the exact area of the water bodies. In the experiments, I applied data augmentation techniques; affine image transformations and color modification. The experimental result showed that using the data augmentation techniques obtained two times higher than not using the data augmentation techniques.

### 5.1 Answers to the Research Questions

**Objective 1:** I aim to classify land use and land cover of five economic crops from the aerial images using CNN methods. I also enhance the performance of the land use classification method using ensemble learning methods.

**Research Question 1:** In Thailand, the cultivation of economic crops, including rice, sugarcane, cassava, rubber, and longan, throughout the country. It is not simple to analyze the area of the economic crops, because I need the expert in geographic information systems (GIS) and remote sensing (RS) areas to analyze the economic crops area. It requires information from other government sections to compute and evaluate. It requires more time to collect the relevant information from all sections and incident delays in data analysis by experts will occur. From this issue, how can I reduce the waiting time by collecting all information from government sections? Could convolutional neural networks (CNNs), a type of deep learning method, classify the areas of economic crops from the aerial images? If possible, applying the economic crops classification will help the planner, the policy-maker monitor the land use and land cover of the economic crops faster. In addition, can the ensemble learning method improve the land use and land cover classification performance?

In answering the RQ1, I proposed to use the deep learning method for land use and land cover. The proposed method used the aerial images for training to create a robust model and classifying the output of the aerial images. It could help us to collect the aerial images ourselves by collecting data from the google earth program. Consequently, it reduced the waiting time that I have to wait for the high-resolution satellite images from government

sections. I proposed a new benchmark dataset, namely the EcoCropsAID dataset. This data set contained five classes (rice, sugarcane, cassava, rubber, and longan) and has 5,400 aerial images that collected the data between 2014 and 2018.

This thesis used eight convolutional neural network (CNN) architectures to create the model and classify the aerial images. I decided to choose only three CNN models (NASNetLarge, VGG19, and VGG16) that obtained high performance on the test set of the EcoCropsAID dataset. Further, I combined three data augmentation techniques (rotation, height shift, and width shift) while training the CNN models. The experimental results showed that the VGG16 models combined with data augmentation techniques obtained accuracy above 90%. It proved that I could create a system to classify the economic crops from the aerial images using CNN architectures.

Moreover, the ensemble CNN model was proposed to enhance the classification in economic crops aerial images. The ensemble CNN model combined two parts; ensemble CNNs and ensemble learning method. I created an ensemble CNNs consisting of 3 CNNs based on the robust CNN models found from the previous step. For the ensemble learning method, I proposed to use the weighted average method. Further, the grid-search method was proposed to find the optimal weighted parameters of each CNN. As a result, the ensemble CNN model obtained better results than using only the single state-of-the-art CNN.

**Objective 2:** I proposed to use the snapshot ensemble CNN technique to improve the performance of the land use classification from the aerial images.

**Research Question 2:** Many types of ensemble learning methods, such as bagging ensemble (Duin & Tax, 2005), boosting ensemble (Choi et al., 2019), stacked ensemble (R. Sun, 2019), and snapshot ensemble (Bunrit et al., 2019), are proposed to solve the image classification problem. Furthermore, the snapshot ensemble CNN method was designed to find the optimal CNN models. The cyclic cosine annealing method was proposed to decrease the learning rate value while training the CNN model. It made the learning loss of the CNN drop very fast compared to the other CNN methods. Consequently, the outputs of optimal models were then combined and given to the weighted ensemble learning method to predict the output. Then, can the snapshot ensemble CNN method improve the performance of the economic crops classification on the aerial images? I am also concerned with the learning algorithm that

proposes to decrease the learning loss and also increase the performance of the snapshot ensemble CNN.

To answer the RQ2, I propose to use the snapshot ensemble CNN method to classify aerial images on three aerial image datasets; UCM, AID, and EcoCropsAID. The idea of the snapshot is to divide the training iterations into cycles and snap the CNN model at the optimal minimum value in each cycle. For example, if the number of cycles was defined at four cycles and training the CNN with 100 iterations. Then the CNN models with the lowest loss value in every 25 iterations are snapped. Five CNN models in total are obtained. Further, the 5 CNN models were classified using an unweighted average ensemble learning. When classifying the aerial images, the snapshot ensemble CNN method showed a higher performance than classifying with using only one CNN.

Furthermore, I proposed a new learning rate schedule method called the dropCyclic. I trained the snapshot ensemble CNN method based on the proposed dropCyclic learning rate schedule and evaluated it on three aerial image datasets; UCM, AID, and EcoCropsAID. The experimental results showed that, when training the snapshot ensemble CNN method with the proposed dropCyclic learning rate schedule, it outperformed the original snapshot ensemble CNN (G. Huang et al., 2017) and the new snapshot ensemble CNN (L. Wen et al., 2019) on the UCM datasets.

**Objective 3:** I proposed to use the instance segmentation technique to segment the water body from the aerial images.

**Research Question 3:** When using the LCM model to calculate land use, I found that the water resources were reduced each year (Suwanlertcharoen et al., 2013). It may affect agriculture because water resources are essential for cultivation. To find the water resources areas, the expert always uses geographic information systems (GIS) and remote sensing systems to analyze the areas from the high-resolution satellite image. It spent much computation time and more money as well. Importantly, is it possible to employ the deep learning method to find the water resources from the aerial images? If possible, it will reduce the cost of computation. Also, everyone can download aerial images from the general application, such as Google maps and analyze the water resources using the deep learning method. Furthermore, if it is accurate, I can also analyze the amount of water and plan to manage water usage.

To answer the RQ3, the computing on the high-resolution satellite image was ignored because I plan to download the aerial image of water resources from the google earth application. Then, I applied the mask region-based CNN (mask R-CNN), which is the instance segmentation technique, to find the region of interest (ROI) of the water resource areas. In this experiment, the mask R-CNN method could segment two water bodies (natural and artificial). The natural and artificial water bodies were varied in color, shape, size, and similarity. Furthermore, the data augmentation techniques were included in the training process to increase the amount of training data, consisting of affine image transformations and color modification. The experimental results showed that the mask R-CNN combined with data augmentation techniques obtained two times better performance than without using data augmentation techniques. Consequently, the mask R-CNN method spent more computation time, but it performed fast when segmentation.

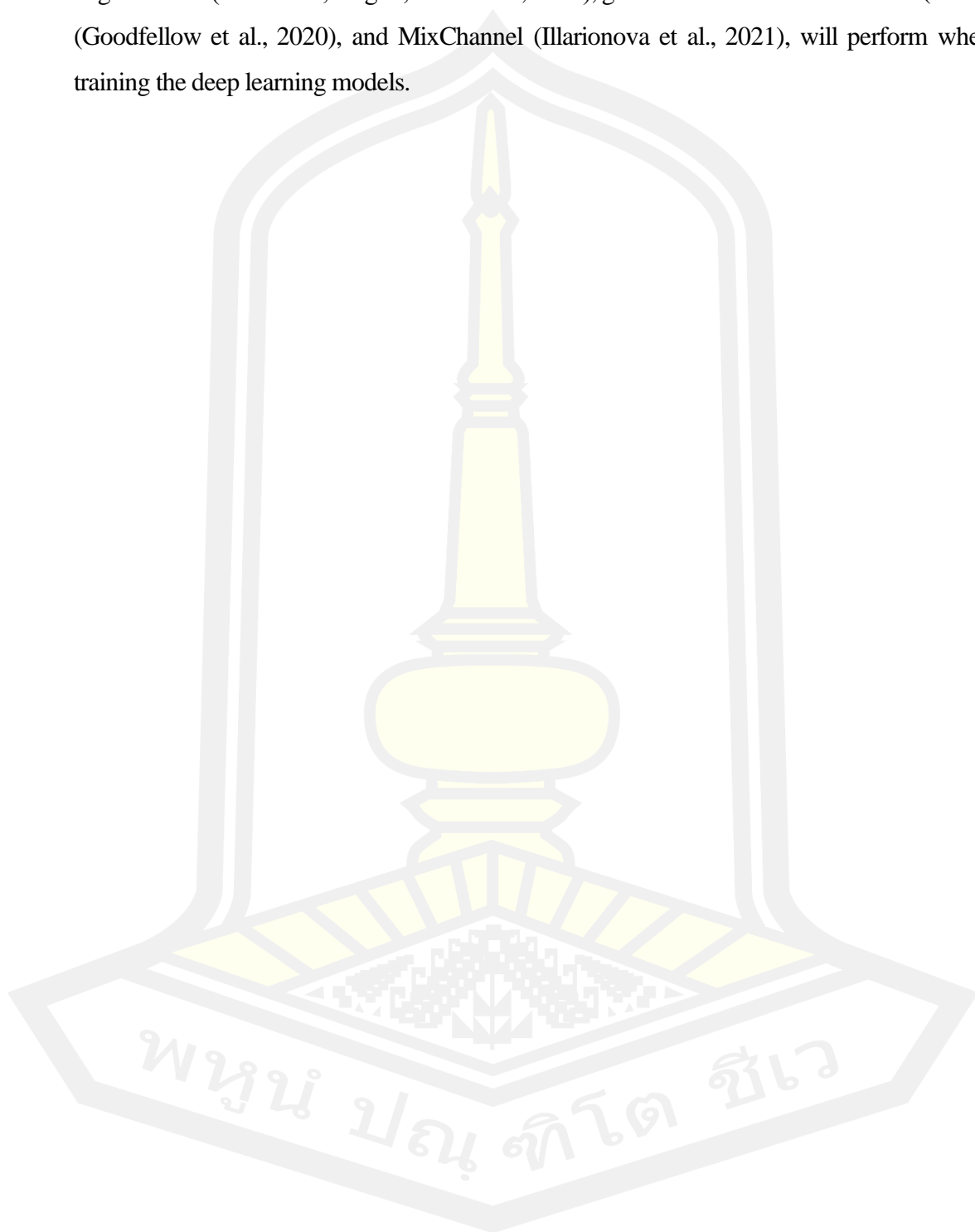
## 5.2 Future work

Several future works present in the following could be used as a direction for researchers who are interested in inventing better land use and land cover recognition systems in aerial images using deep learning techniques. I divide the future directions toward two tasks; classification and segmentation.

For classification techniques, in the case of finding the ensemble learning method, I will increase the performance of the snapshot ensemble convolutional neural network (CNN) by designing a new learning rate schedule based on the max-min cosine cyclic learning rate scheduler (L. Wen et al., 2019) that can decrease the training loss and avoid overfitting while training. In addition, the unweighted average method was included in the snapshot ensemble CNN. I plan to replace the unweighted average method with the cost-sensitive probability method (Rojarath & Songpan, 2021).

For segmentation techniques, in recent years, various deep learning architectures for aerial images are popular and have been applied to various applications, such as segmentation of buildings (X. Li, Jiang, Peng, & Yin, 2019) and environments (urban, suburban, and rural) (Morocho-Cayamcela, 2020), to detect the changes caused by a natural disaster (Gupta, Watson, & Yin, 2021). I will consider applying the new deep learning methods to enhance the performance of the water resource segmentation. In this thesis, the experimental results showed that data augmentation could help the deep learning method increase

segmentation performance. The new data augmentation techniques, such as generative data augmentation (Kumdakci, Öngün, & Temizel, 2021), generative adversarial networks (GAN) (Goodfellow et al., 2020), and MixChannel (Illarionova et al., 2021), will perform when training the deep learning models.





## REFERENCES



## REFERENCES

- Almeida, C. A. de, Coutinho, A. C., Esquerdo, J. C. dalla M., Adami, M., Venturieri, A., Diniz, C. G., Dessay, N., Durieux, L., & Gomes, A. R. (2016). High spatial resolution land use and land cover mapping of the Brazilian legal Amazon in 2008 using Landsat-5/TM and MODIS data. *Acta Amazonica*, 46(3), 291–302. <https://doi.org/10.1590/1809-4392201505504>
- Al-Najjar, H. A. H., Kalantar, B., Pradhan, B., Saeidi, V., Halin, A. A., Ueda, N., & Mansor, S. (2019). Land cover classification from fused DSM and UAV images using convolutional neural networks. *Remote Sensing*, 11(12), 1–18. doi: 10.3390/rs11121461
- Bunrit, S., Kerdprasop, N., & Kerdprasop, K. (2019). Evaluating on the transfer learning of CNN architectures to a construction material image classification tasks. *International Journal of Machine Learning and Computing*, 9(2), 201–207. doi: 10.18178/ijmlc.2019.9.2.787
- Caldas, M. M., Goodin, D., Sherwood, S., Campos Krauer, J. M., & Wisely, S. M. (2015). Land-cover change in the Paraguayan Chaco: 2000–2011. *Journal of Land Use Science*, 10(1), 1–18. doi: 10.1080/1747423X.2013.807314
- Cao, D., Xing, H., Wong, M. S., Kwan, M. P., Xing, H., & Meng, Y. (2021). A stacking ensemble deep learning model for building extraction from remote sensing images. *Remote Sensing*, 13(19), 1–22. doi: 10.3390/rs13193898
- Cap, Q. H., Uga, H., Kagiwada, S., & Iyatomi, H. (2020). LeafGAN: An effective data augmentation method for practical plant disease diagnosis. *IEEE Transactions on Automation Science and Engineering*, 1–10. <https://doi.org/10.1109/TASE.2020.3041499>
- Chaib, S., Yao, H., Gu, Y., & Amrani, M. (2017). Deep feature extraction and combination for remote sensing image classification based on pre-trained CNN models. In C. M. Falco & X. Jiang (Eds.), *Proceedings of the 9th International Conference on Digital Image Processing (ICDIP), Hong Kong, China*. doi: 10.1117/12.2281755

- Cheng, G., Han, J., & Lu, X. (2017). Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 105(10), 1865–1883. doi: 10.1109/JPROC.2017.2675998
- Choi, D., Shallue, C. J., Nado, Z., Lee, J., Maddison, C. J., & Dahl, G. E. (2019). On empirical comparisons of optimizers for deep learning. *ArXiv*, 1–27. <https://arxiv.org/pdf/1910.05446.pdf>
- Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. *Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1800–1807. doi: 10.1109/CVPR.2017.195
- Chowdhury, M., Hasan, M. E., & Abdullah-Al-Mamun, M. M. (2020). Land use/land cover change assessment of Halda watershed using remote sensing and GIS. *Egyptian Journal of Remote Sensing and Space Science*, 23(1), 63–75. doi: 10.1016/j.ejrs.2018.11.003
- Dede, M. A., Aptoula, E., & Genc, Y. (2018). Deep network ensembles for aerial scene classification. *IEEE Geoscience and Remote Sensing Letters*, 16(5), 732–735. doi: 10.1109/LGRS.2018.2880136
- Deepan, P. (2019). Fusion of deep learning models for improving classification accuracy of remote sensing images. *Journal of Mechanics of Continua and Mathematical Sciences*, 14(5), 189–201. doi: 10.26782/jmcms.2019.10.00015
- Diengdoh, V. L., Onde, S., Hunt, M., & Brook, B. W. (2020). A validated ensemble method for multinomial land-cover classification. *Ecological Informatics*, 56(2), 1–10. doi: 10.1016/j.ecoinf.2020.101065
- Dong, X., Qian, L., & Huang, L. (2018). A CNN based bagging learning approach to short-term load forecasting in smart grid. *Proceedings of the IEEE SmartWorld : Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation, San Francisco Bay Area, California, USA*, 1–6. IEEE. doi: 10.1109/UIC-ATC.2017.8397649

- Duin, R. P. W., & Tax, D. M. J. (2005). Statistical pattern recognition. In *Handbook of Pattern Recognition and Computer Vision, 3rd Edition*, 3–24. doi: 10.1142/9789812775320\_0001
- Elagouz, M. H., Abou-Shleel, S. M., Belal, A. A., & El-Mohandes, M. A. O. (2020). Detection of land use/cover change in Egyptian Nile Delta using remote sensing. *Egyptian Journal of Remote Sensing and Space Science*, 23(1), 57–62. doi: 10.1016/j.ejrs.2018.10.004
- Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (2010). The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2), 303–338. doi: 10.1007/s11263-009-0275-4
- Frazão, X., & Alexandre, L. A. (2014). Weighted convolutional neural network ensemble. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8827, 674–681. doi: 10.1007/978-3-319-12568-8\_82
- Ganaie, M. A., Hu, M., Tanveer, M., & Suganthan, P. N. (2021). Ensemble deep learning: A review. *ArXiv*, 1–48. <https://arxiv.org/pdf/2104.02395.pdf>
- Ge, R., Kakade, S. M., Kidambi, R., & Netrapalli, P. (2019). The step decay schedule: A near optimal, geometrically decaying learning rate procedure for least squares. *Proceedings of the 33rd Neural Information Processing Systems (NeurIPS), Vancouver, Canada*, 32, 14977–14988. Vancouver, Canada.
- Ghaffar, M. A. A., McKinstry, A., Maul, T., & Vu, T. T. (2019). Data augmentation approaches for satellite image super-resolution. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4, 47–54. doi: 10.5194/isprs-annals-IV-2-W7-47-2019
- GISTDA. (2013). *Fundamental geographic data set (FGDS)*. Bangkok.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, 63(11), 139–144. doi: 10.1145/3422622

- Gu, Y., Wang, Y., & Li, Y. (2019). A survey on deep learning-driven remote sensing image scene understanding: Scene classification, scene retrieval and scene-guided object detection. *Applied Sciences*, 9(10), 2110. doi: 10.3390/APP9102110
- Gupta, A., Watson, S., & Yin, H. (2021). Deep learning-based aerial image segmentation with open data for disaster impact assessment. *Neurocomputing*, 439, 22–33. doi: 10.1016/j.neucom.2020.02.139
- Han, W., Feng, R., Wang, L., & Cheng, Y. (2018). A semi-supervised generative framework with deep learning features for high-resolution remote sensing image scene classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 145(November), 23–43. doi: 10.1016/j.isprsjprs.2017.11.004
- Han, X., Zhong, Y., Cao, L., & Zhang, L. (2017). Pre-trained AlexNet architecture with pyramid pooling and supervision for high spatial resolution remote sensing image scene classification. *Remote Sensing*, 9(8), 848. doi: 10.3390/rs9080848
- He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2020). Mask R-CNN. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 42(2), 386–397. doi: 10.1109/TPAMI.2018.2844175
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Identity mappings in deep residual networks. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9908 LNCS, 630–645. Springer Verlag. doi: 10.1007/978-3-319-46493-0\_38
- Helber, P., Bischke, B., Dengel, A., & Borth, D. (2019). Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7), 2217–2226. doi: 10.1109/JSTARS.2019.2918242
- Huang, G., Li, Y., Pleiss, G., Liu, Z., Hopcroft, J. E., & Weinberger, K. Q. (2017). Snapshot ensembles: Train 1, get M for free. *Proceedings of the 5th International Conference on Learning Representations (ICLR), Toulon, France*, 1–14. Toulon, France. Retrieved from <https://arxiv.org/abs/1704.00109>

- Huang, Z., Qi, H., Kang, C., Su, Y., & Liu, Y. (2020). An ensemble learning approach for urban land use mapping based on remote sensing imagery and social sensing data. *Remote Sensing*, *12*(19), 1–18. doi: 10.3390/rs12193254
- Hung, S.-C., Wu, H.-C., & Tseng, M.-H. (2020). Remote sensing scene classification and explanation using RSSCNet and LIME. *Applied Sciences*, *10*(18), 6151. doi: 10.3390/app10186151
- Illarionova, S., Nesteruk, S., Shadrin, D., Ignatiev, V., Pukalchik, M., & Oseledets, I. (2021). MixChannel: Advanced augmentation for multispectral satellite images. *Remote Sensing*, *13*(11). doi: 10.3390/rs13112181
- Jazouli, A. El, Barakat, A., Khellouk, R., Rais, J., & Baghdadi, M. El. (2019). Remote sensing and GIS techniques for prediction of land use land cover change effects on soil erosion in the high basin of the Oum Er Rbia River (Morocco). *Remote Sensing Applications: Society and Environment*, *13*, 361–374. doi: 10.1016/j.rsase.2018.12.004
- Jiao, L., & Zhao, J. (2019). A survey on the new generation of deep learning in image processing. *IEEE Access*, *7*, 172231–172263. doi: 10.1109/ACCESS.2019.2956508
- Ju, C., Bibaut, A., & van der Laan, M. (2018). The relative performance of ensemble methods with deep convolutional neural networks for image classification. *Journal of Applied Statistics*, *45*(15), 2800–2818. doi: 10.1080/02664763.2018.1441383
- Kim, P. K., & Lim, K. T. (2017). Vehicle type classification using bagging and convolutional neural network on multi view surveillance image. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, Hawaii, USA*, 41–46. Honolulu, Hawaii, USA. doi: 10.1109/CVPRW.2017.126
- Korzh, O., Cook, G., Andersen, T., & Serra, E. (2017). Stacking approach for CNN transfer learning ensemble for remote sensing imagery. *Proceedings of the Intelligent Systems Conference (IntelliSys), London, United Kingdom*, 599–608. IEEE. doi: 10.1109/IntelliSys.2017.8324356



- Kulkarni, S., & Kelkar, V. (2014). Classification of multispectral satellite images using ensemble techniques of bagging, boosting and adaboost. *Proceedings of the International Conference on Circuits, Systems, Communication and Information Technology Applications (CSCITA), Mumbai, India*, 253–258. IEEE. doi: 10.1109/CSCITA.2014.6839268
- Kumdakı, H., Öngün, C., & Temizel, A. (2021). Generative data augmentation for vehicle detection in aerial images. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12668 LNCS, 19–31. doi: 10.1007/978-3-030-68793-9\_2
- Kussul, N., Lavreniuk, M., Skakun, S., & Shelestov, A. (2017). Deep learning classification of land cover and crop types using remote sensing data. *IEEE Geoscience and Remote Sensing Letters*, 14(5), 778–782. doi: 10.1109/LGRS.2017.2681128
- Li, X., Jiang, Y., Peng, H., & Yin, S. (2019). An aerial image segmentation approach based on enhanced multi-scale convolutional neural network. *Proceedings of the IEEE International Conference on Industrial Cyber Physical Systems (ICPS)*, 47–52. IEEE. doi: 10.1109/ICPHYS.2019.8780187
- Li, Y., Zhang, Y., & Zhu, Z. (2021). Error-tolerant deep learning for remote sensing image scene classification. *IEEE Transactions on Cybernetics*, 51(4), 1756–1768. doi: 10.1109/TCYB.2020.2989241
- Ma, L., Liu, Y., Zhang, X., Ye, Y., Yin, G., & Johnson, B. A. (2019). Deep learning in remote sensing applications: A meta-analysis and review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 152, 166–177. doi: 10.1016/j.isprsjprs.2019.04.015
- Miao, Z., Fu, K., Sun, H., Sun, X., & Yan, M. (2018). Automatic water-body segmentation from high-resolution satellite images via deep networks. *IEEE Geoscience and Remote Sensing Letters*, 15(4), 602–606. doi: 10.1109/LGRS.2018.2794545
- Mikołajczyk, A., & Grochowski, M. (2018). Data augmentation for improving deep learning in image classification problem. *Proceeding of the International*

- Interdisciplinary PhD Workshop (IIPhDW)*, 17–122. IEEE. doi: 10.1109/IIPHDW.2018.8388338
- Minetto, R., Pamplona Segundo, M., & Sarkar, S. (2019). Hydra: An ensemble of convolutional neural networks for geospatial land classification. *IEEE Transactions on Geoscience and Remote Sensing*, 57(9), 6530–6541. doi: 10.1109/TGRS.2019.2906883
- Mishra, P. K., Rai, A., & Rai, S. C. (2020). Land use and land cover change detection using geospatial techniques in the Sikkim Himalaya, India. *Egyptian Journal of Remote Sensing and Space Science*, 23(2), 133–143. doi: 10.1016/j.ejrs.2019.02.001
- Morocho-Cayamcela, M. E. (2020). Increasing the segmentation accuracy of aerial images with dilated spatial pyramid pooling. *Electronic Letters on Computer Vision and Image Analysis*, 19(2), 17–21. doi: 10.5565/rev/elcvia.1337
- Noppitak, S., Gonwirat, S., & Surinta, O. (2020). Instance segmentation of water body from aerial image using mask region-based convolutional neural network. *Proceedings of the 3rd International Conference on Information Science and System (ICISS)*, 61–66. doi: 10.1145/3388176.3388184.
- Noppitak, S., & Surinta, O. (2021). Ensemble convolutional neural network architectures for land use classification in economic crops aerial images. *ICIC Express Letters*, 15(6), 531–543. doi: 10.24507/icicel.15.06.531
- Office of Agricultural Economics. (2019a). *Agricultural economic outlook*. Bangkok.
- Office of Agricultural Economics. (2019b). *Agricultural economic report: 3rd quarter 2019 and outlook for 2019*. Bangkok.
- Office of the Permanent Secretary for Ministry of Agriculture and Cooperatives. (2019). *Agri-map online*. Bangkok: Ministry of Agriculture and Cooperatives.
- Özdemir, A., Polat, K., & Alhudhaif, A. (2021). Classification of imbalanced hyperspectral images using SMOTE-based deep learning methods. *Expert Systems with Applications*, 178, 114986. doi: 10.1016/j.eswa.2021.114986

- P, S. A. B., & Annavarapu, C. S. R. (2021). Deep learning-based improved snapshot ensemble technique for COVID-19 chest X-ray classification. *Applied Intelligence*, 51(5), 3104–3120. doi: 10.1007/s10489-021-02199-4
- Pari, R., Sandhya, M., & Sankar, S. (2020). A multitier stacked ensemble algorithm for improving classification accuracy. *Computing in Science and Engineering*, 22(4), 74–85. doi: 10.1109/MCSE.2018.2873940
- Pawara, P., Okafor, E., Schomaker, L., & Wiering, M. (2017). Data augmentation for plant classification. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10617 LNCS, 615–626. doi: 10.1007/978-3-319-70353-4\_52
- Perez, L., & Wang, J. (2017). The effectiveness of data augmentation in image classification using deep learning. *ArXiv*, 1–8. <https://arxiv.org/pdf/1712.04621.pdf>
- Petrovska, B., Atanasova-Pacemska, T., Corizzo, R., Mignone, P., Lameski, P., & Zdravevski, E. (2020). Aerial scene classification through fine-tuning with adaptive learning rates and label smoothing. *Applied Sciences*, 10(17), 5792. doi: 10.3390/app10175792
- Petrovska, B., Zdravevski, E., Lameski, P., Corizzo, R., Štajduhar, I., & Lerga, J. (2020). Deep learning for feature extraction in remote sensing: A case-study of aerial scene classification. *Sensors*, 20(14), 3906. doi: 10.3390/s20143906
- Phiphitphatphaisit, S., & Surinta, O. (2021). Deep feature extraction technique based on Conv1D and LSTM network for food image recognition. *Engineering and Applied Science Research*, 48(5), 581–592.
- Pilipovic, R., & Risojevic, V. (2017). Evaluation of convnets for large-scale scene classification from high-resolution remote sensing images. *Proceedings of the 17th IEEE International Conference on Smart Technologies (EUROCON)*, 932–937. doi: 10.1109/EUROCON.2017.8011248
- Prakash, N., Manconi, A., & Loew, S. (2020). Mapping landslides on EO data: Performance of deep learning models vs. traditional machine learning models. *Remote Sensing*, 12(3), 346. doi: 10.3390/rs12030346

- Puangsuwan, T., & Surinta, O. (2021). Enhancement of plant leaf disease classification based on snapshot ensemble convolutional neural network. *ICIC Express Letters*, 15(6). doi: 10.24507/icicel.15.06.669
- Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6), 1137–1149. doi: 10.1109/TPAMI.2016.2577031
- Rojarath, A., & Songpan, W. (2021). Cost-sensitive probability for weighted voting in an ensemble model for multi-class classification problems. *Applied Intelligence*, 51(7), 4908–4932. doi: 10.1007/s10489-020-02106-3
- Rujoiu-Mare, M.-R., & Mihai, B.-A. (2016). Mapping land cover using remote sensing data and gis techniques: a case study of Prahova subcarpathians. *Procedia Environmental Sciences*, 32, 244–255. doi: 10.1016/j.proenv.2016.03.029
- Sambasivam, G., & Opiyo, G. D. (2021). A predictive machine learning application in agriculture: Cassava disease detection and classification with imbalanced dataset using convolutional neural networks. *Egyptian Informatics Journal*, 22(1), 27–34. doi: 10.1016/j.eij.2020.02.007
- Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, (CVPR), Salt Lake City, Utah, USA*, 4510–4520. doi: 10.1109/CVPR.2018.00474
- Schaefer, M., & Thinh, N. X. (2019). Evaluation of land cover change and agricultural protection sites: A GIS and remote sensing approach for Ho Chi Minh city, Vietnam. *Heliyon*, 5(5), e01773. doi: 10.1016/j.heliyon.2019.e01773
- Sefrin, O., Riese, F. M., & Keller, S. (2021). Deep learning for land cover change detection. *Remote Sensing*, 13(1), 1–27. doi: 10.3390/rs13010078
- Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1), 1–48. doi: 10.1186/s40537-019-0197-0

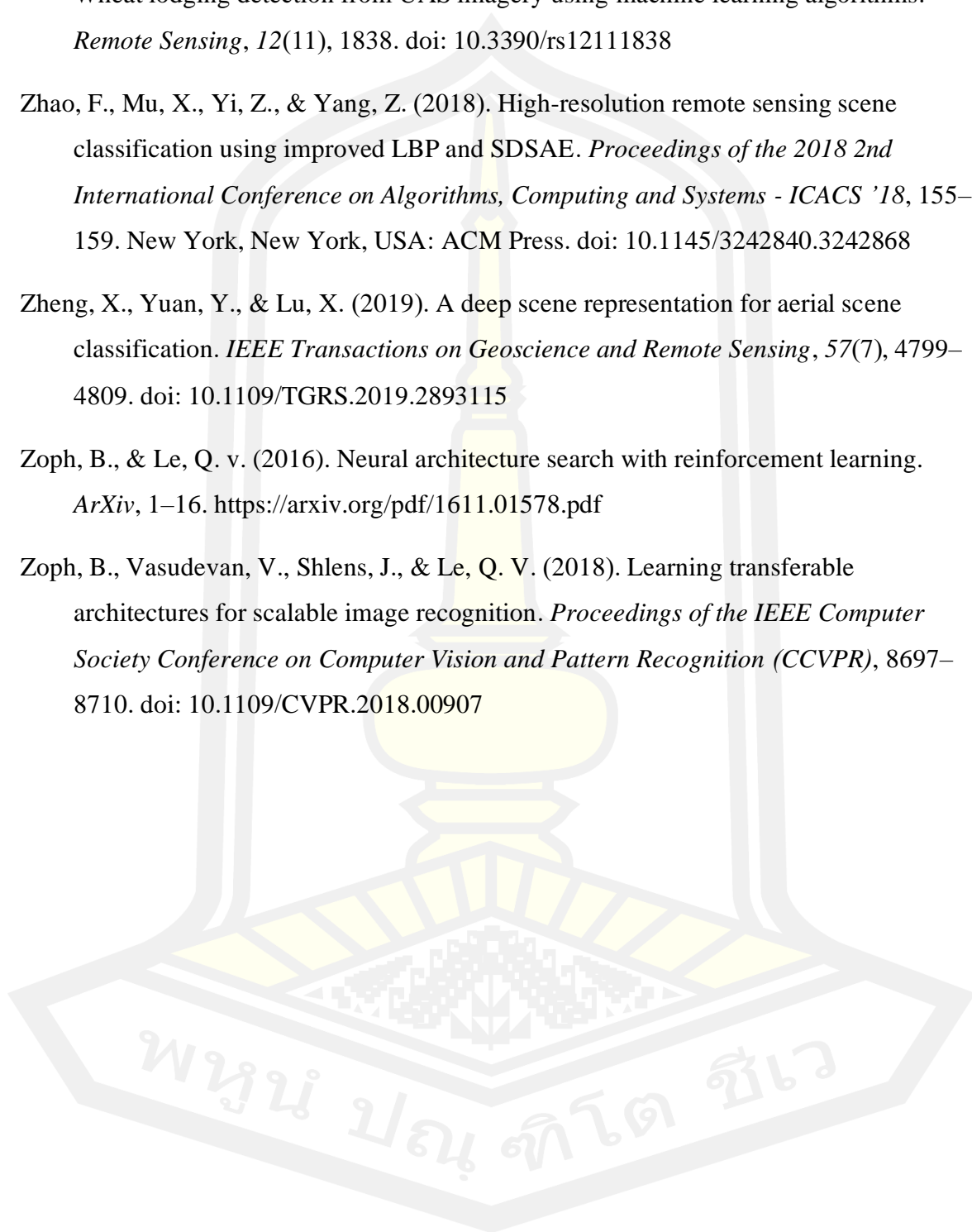
- Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *Proceedings of the 3rd International Conference on Learning Representations (ICLR), San Diego, California, USA*. San Diego, California, USA.
- Storie, C. D., & Henry, C. J. (2018). Deep learning neural networks for land use land cover mapping. *International Geoscience and Remote Sensing Symposium (IGARSS)*, 3445–3448. doi: 10.1109/IGARSS.2018.8518619
- Sun, R. (2019). Optimization for deep learning: theory and algorithms. *ArXiv*, 1–60. <https://arxiv.org/pdf/1912.08957.pdf>
- Sun, Z., Di, L., & Fang, H. (2019). Using long short-term memory recurrent neural network in land cover classification on landsat and cropland data layer time series. *International Journal of Remote Sensing*, 40(2), 593–614. doi: 10.1080/01431161.2018.1516313
- Surinta, O., Schomaker, L., & Wiering, M. (2013). A comparison of feature and pixel-based methods for recognizing handwritten bangla digits. *Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR)*, 165, 165–169. doi: 10.1109/ICDAR.2013.40
- Suwanlertcharoen, T., Prukpitikul, S., Buakaew, V., & Kaewpoo, N. (2013). Land use and land cover prediction and change detection using Geo-Infomatics in Khlong Kui Watershed, Prachuap Khiri Khan Province. *Proceedings of the Geoinfotech 2013*, 1–15. Retrieved from <http://research.gistda.or.th/assets/uploads/pdfs/eb43f-12.-.pdf>
- Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. A. (2017). Inception-v4, inception-ResNet and the impact of residual connections on learning. *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI)*, 4278–4284. AAAI.
- Treitz, P., & Rogan, J. (2004). Remote sensing for mapping and monitoring land-cover and land-use change-an introduction. *Progress in Planning*, 61, 269–279. doi: 10.1016/S0305-9006(03)00064-3
- Vaishnave, M. P., Devi, K. S., & Srinivasan, P. (2019). A study on deep learning models for satellite imagery. *International Journal of Applied Engineering Research*, 14(4), 881–887.

- Wen, L., Gao, L., & Li, X. (2019). A new snapshot ensemble convolutional neural network for fault diagnosis. *IEEE Access*, 7, 32037–32047. doi: 10.1109/ACCESS.2019.2903295
- Wen, Q., Jiang, K., Wang, W., Liu, Q., Guo, Q., Li, L., & Wang, P. (2019). Automatic building extraction from google earth images under complex backgrounds based on deep instance segmentation network. *Sensors*, 19(2), 333. doi: 10.3390/s19020333
- Wolpert, D. (1992). Stacked generalization. *Neural Networks*, 5(2), 241–259. doi: 10.1016/S0893-6080(05)80023-1
- Wu, Q., Chen, Y., & Meng, J. (2020). Dcgan-based data augmentation for tomato leaf disease identification. *IEEE Access*, 8, 98716–98728. doi: 10.1109/ACCESS.2020.2997001
- Wu, X., Ward, R., & Bottou, L. (2018). WNGrad: Learn the learning rate in gradient descent. *ArXiv*, 1–16. <https://arxiv.org/pdf/1803.02865.pdf>
- Wu, Y., Liu, L., Bae, J., Chow, K. H., Iyengar, A., Pu, C., ... Zhang, Q. (2019). Demystifying learning rate policies for high accuracy training of deep neural networks. *Proceedings of the IEEE International Conference on Big Data, Los Angeles, California, USA*, 1971–1980. IEEE. doi: 10.1109/BigData47090.2019.9006104
- Xia, G. S., Hu, J., Hu, F., Shi, B., Bai, X., Zhong, Y., Zhang, L., & Lu, X. (2017). AID: A benchmark data set for performance evaluation of aerial scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(7), 3965–3981. <https://doi.org/10.1109/TGRS.2017.2685945>
- Xue, M., Li, J., & Luo, Q. (2022). Toward optimal learning rate schedule in scene classification network. *IEEE Geoscience and Remote Sensing Letters*, 19, 1–5. doi: 10.1109/LGRS.2020.3040359
- Yang, J., & Wang, F. (2020). Auto-ensemble: An adaptive learning rate scheduling based deep learning model ensembling. *IEEE Access*, 8, 217499–217509. doi: 10.1109/ACCESS.2020.3041525



- Yang, Y., & Newsam, S. (2010). Bag-of-visual-words and spatial extensions for land-use classification. *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems (ACM SIGSPATIAL GIS), San Jose, California, USA*, 270–279. doi: 10.1145/1869790.1869829
- Yazdizadeh, A., Patterson, Z., & Farooq, B. (2019). Ensemble convolutional neural networks for mode inference in smartphone travel survey. *IEEE Transactions on Intelligent Transportation Systems*, 1–8. doi: 10.1109/tits.2019.2918923
- Yu, D., Xu, Q., Guo, H., Zhao, C., Lin, Y., & Li, D. (2020). An efficient and lightweight convolutional neural network for remote sensing image scene classification. *Sensors*, 20(7), 1999. doi: 10.3390/s20071999
- Yu, L., Su, J., Li, C., Wang, L., Luo, Z., & Yan, B. (2018). Improvement of moderate resolution land use and land cover classification by introducing adjacent region features. *Remote Sensing*, 10(3), 1–16. doi: 10.3390/rs10030414
- Yu, X., Wu, X., Luo, C., & Ren, P. (2017). Deep learning in remote sensing scene classification: a data augmentation enhanced convolutional neural network framework. *GIScience & Remote Sensing*, 54(5), 741–758. doi: 10.1080/15481603.2017.1323377
- Yuan, L.-X., Li, S.-J., & Jiang, Y.-P. (2017). Remote sensing scene classification using a preclassification strategy and an improved structural feature. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(9), 4094–4103. doi: 10.1109/JSTARS.2017.2707519
- Zhang, C., Sargent, I., Pan, X., Li, H., Gardiner, A., Hare, J., & Atkinson, P. M. (2019). Joint deep learning for land cover and land use classification. *Remote Sensing of Environment*, 221, 173–187. doi: 10.1016/j.rse.2018.11.014
- Zhang, P., Ke, Y., Zhang, Z., Wang, M., Li, P., & Zhang, S. (2018). Urban land use and land cover classification using novel deep learning models based on high spatial resolution satellite imagery. *Sensors*, 18(11). doi: 10.3390/s18113717

- Zhang, Z., Flores, P., Igathinathane, C., L. Naik, D., Kiran, R., & Ransom, J. K. (2020). Wheat lodging detection from UAS imagery using machine learning algorithms. *Remote Sensing*, 12(11), 1838. doi: 10.3390/rs12111838
- Zhao, F., Mu, X., Yi, Z., & Yang, Z. (2018). High-resolution remote sensing scene classification using improved LBP and SDSAE. *Proceedings of the 2018 2nd International Conference on Algorithms, Computing and Systems - ICACS '18*, 155–159. New York, New York, USA: ACM Press. doi: 10.1145/3242840.3242868
- Zheng, X., Yuan, Y., & Lu, X. (2019). A deep scene representation for aerial scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 57(7), 4799–4809. doi: 10.1109/TGRS.2019.2893115
- Zoph, B., & Le, Q. v. (2016). Neural architecture search with reinforcement learning. *ArXiv*, 1–16. <https://arxiv.org/pdf/1611.01578.pdf>
- Zoph, B., Vasudevan, V., Shlens, J., & Le, Q. V. (2018). Learning transferable architectures for scalable image recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 8697–8710. doi: 10.1109/CVPR.2018.00907



## BIOGRAPHY

<b>NAME</b>	Sangdaow Noppitak
<b>DATE OF BIRTH</b>	19 May 1981
<b>PLACE OF BIRTH</b>	Nakhon Ratchasima
<b>ADDRESS</b>	109/9 Moo.1, Tambon Khok Sung, Amphur Nong Ki, Buri Ram, 31210
<b>POSITION</b>	Lecturer
<b>PLACE OF WORK</b>	Buriram Rajabhat University
<b>EDUCATION</b>	2004 Bachelor of Science (B.Sc.) Computer Science, Khon Kaen University. 2010 Master of Science (M.Sc.) Remote Sensing and Geographic Information Systems, Khon Kaen University. 2022 Doctor of Philosophy (Ph.D.) Information Technology, Mahasarakham University.
<b>Research output</b>	Noppitak, S., Gonwirat, S., & Surinta, O. (2020). Instance segmentation of water body from aerial image using mask region-based convolutional neural network. Proceedings of the 3rd International Conference on Information Science and System (ICISS), 61–66. doi: 10.1145/3388176.3388184.  Noppitak, S., & Surinta, O. (2021). Ensemble convolutional neural network architectures for land use classification in economic crops aerial images. ICIC Express Letters, 15(6), 531–543. doi: 10.24507/icicel.15.06.531.

พหุบัณฑิตวิทยา