Deep learning for land cover classification and environmental analysis using high-resolution remote sensing data

Vitor Souza Martins

Iowa State University

Follow this and additional works at: https://lib.dr.iastate.edu/etd

Recommended Citation
Souza Martins, Vitor, "Deep learning for land cover classification and environmental analysis using high-resolution remote sensing data" (2020). Graduate Theses and Dissertations. 18232. https://lib.dr.iastate.edu/etd/18232

This Thesis is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Graduate Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.
Deep learning for land cover classification and environmental analysis using high-resolution remote sensing data

by

Vitor Souza Martins

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Agricultural and Biosystems Engineering

Program of Study Committee:
Amy Kaleita, Major Professor
Brian K. Gelder
Joshua M. Peschel
Yuyu Zhou
Bradley A. Miller

The student author, whose presentation of the scholarship herein was approved by the program of study committee, is solely responsible for the content of this dissertation. The Graduate College will ensure this dissertation is globally accessible and will not permit alterations after a degree is conferred.

Iowa State University
Ames, Iowa
2020

Copyright © Vitor Souza Martins, 2020. All rights reserved.
DEDICATION

With love to my parents and my wife.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>LIST OF FIGURES</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIST OF TABLES</td>
<td>vii</td>
</tr>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>viii</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>ix</td>
</tr>
<tr>
<td>CHAPTER 1. GENERAL INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>Overview</td>
<td>1</td>
</tr>
<tr>
<td>Objectives</td>
<td>4</td>
</tr>
<tr>
<td>Outline</td>
<td>5</td>
</tr>
<tr>
<td>Expected Outcomes and Practical Implications</td>
<td>6</td>
</tr>
<tr>
<td>References</td>
<td>7</td>
</tr>
<tr>
<td>CHAPTER 2. EXPLORING MULTISCALE OBJECT-BASED CONVOLUTIONAL NEURAL NETWORK (MULTI-OCNN) FOR REMOTE SENSING IMAGE CLASSIFICATION AT HIGH 1-M RESOLUTION</td>
<td>10</td>
</tr>
<tr>
<td>Abstract</td>
<td>10</td>
</tr>
<tr>
<td>Introduction</td>
<td>11</td>
</tr>
<tr>
<td>Related work</td>
<td>15</td>
</tr>
<tr>
<td>Material</td>
<td>18</td>
</tr>
<tr>
<td>NAIP Aerial Imagery</td>
<td>18</td>
</tr>
<tr>
<td>Method</td>
<td>22</td>
</tr>
<tr>
<td>Convolutional Neural Network (CNN) architecture</td>
<td>22</td>
</tr>
<tr>
<td>Multi-layer perceptron network</td>
<td>27</td>
</tr>
<tr>
<td>Image segmentation</td>
<td>28</td>
</tr>
<tr>
<td>Multi-OCNN approach for scene classification</td>
<td>30</td>
</tr>
<tr>
<td>Evaluation metrics and comparison</td>
<td>33</td>
</tr>
<tr>
<td>Results</td>
<td>37</td>
</tr>
<tr>
<td>Performance of proposed CNNs</td>
<td>37</td>
</tr>
<tr>
<td>Segmentation scale effects</td>
<td>39</td>
</tr>
<tr>
<td>Performance of multi-OCNN framework</td>
<td>41</td>
</tr>
<tr>
<td>Comparison with other CNN-based methods</td>
<td>45</td>
</tr>
<tr>
<td>Discussion</td>
<td>52</td>
</tr>
<tr>
<td>Conclusions</td>
<td>56</td>
</tr>
<tr>
<td>Acknowledge</td>
<td>57</td>
</tr>
<tr>
<td>Reference</td>
<td>57</td>
</tr>
<tr>
<td>CHAPTER 3. DEEP NEURAL NETWORK FOR COMPLEX WETLAND MAPPING USING HIGH-RESOLUTION WORLDVIEW-3 AND AIRBORNE LIDAR DATA</td>
<td>66</td>
</tr>
<tr>
<td>Abstract</td>
<td>66</td>
</tr>
<tr>
<td>Introduction</td>
<td>67</td>
</tr>
<tr>
<td>Data</td>
<td>71</td>
</tr>
<tr>
<td>Study area</td>
<td>71</td>
</tr>
<tr>
<td>WorldView-3 image processing</td>
<td>73</td>
</tr>
</tbody>
</table>
CHAPTER 4. DIGITAL MAPPING OF STRUCTURAL CONSERVATION PRACTICES IN THE MIDWEST U.S. CROPLANDS: IMPLEMENTATION AND PRELIMINARY ANALYSIS

Abstract .................................................................................. 112
Introduction .............................................................................. 112
Data and methods ................................................................. 116
   Study area ........................................................................... 117
   High-resolution aerial imagery ............................................. 118
   Training data ....................................................................... 119
   Product generation ............................................................. 121
   Accuracy assessment ......................................................... 124
   Spatial analysis and ancillary products .............................. 126
Results .................................................................................... 127
   Classification performance and error sources ................. 127
   Overview of SCP distribution in the Midwest U.S. .......... 130
   Spatial analysis with other variables ............................... 133
Discussion .............................................................................. 137
   Mapping of structural conservation practices ............... 137
   Implications for conservation programs ....................... 138
   Implications for soil erosion and runoff modeling ......... 139
   Advantages and limitations of this framework ............... 140
Conclusion .............................................................................. 142
Acknowledgements ............................................................... 143
References .............................................................................. 143

CHAPTER 5. GENERAL CONCLUSION .................................................. 149
LIST OF FIGURES

Figure 2-1. Figure 2. Examples of land cover classes in IowaNet dataset (1 million samples, 10 land-cover classes) ................................................................. 21

Figure 2-2. Overview of the multiscale object-based CNN (multi-OCNN) framework. ........ 22

Figure 2-3. The architecture of CNN16 model. .......................................................... 26

Figure 2-4. Example of segmentation results. .............................................................. 29

Figure 2-5. Framework of the object analysis. ............................................................ 32

Figure 2-6. Results of object analysis for different targets. ........................................ 33

Figure 2-7. Selected counties for validation experiments .............................................. 37

Figure 2-8. Performance for (a) CNN8 and (b) CNN32 models with different training sizes .... 39

Figure 2-9. Segmentation scale effects with different spatial (hr) and spectral (hs) values in
Mean-Shift algorithm. ........................................................................................... 40

Figure 2-10. Segmentation maps with three combination of input scales \([hr, hs, Mr]\) in the
Mean-Shift algorithm ........................................................................................... 41

Figure 2-11. Classification results of proposed multi-OCNN method .............................. 44

Figure 2-12. Other examples of proposed multi-OCNN. .............................................. 45

Figure 2-13. Comparison of land cover classification with different methods ............... 49

Figure 2-14. Comparison of multi-OCNN with semantic segmentation methods .......... 50

Figure 3-1. Location of Millrace Flats Wildlife Management Area (MFWMA) in Wapello,
Iowa, USA. ............................................................................................................ 72

Figure 3-2. Relative Spectral Response (RSR) of eight VNIR WorldView-3 bands .......... 74

Figure 3-3. Field plots in the study area .................................................................... 77

Figure 3-4. Framework for wetland mapping ............................................................ 77

Figure 3-5. DNN architecture. .................................................................................. 85

Figure 3-6. Illustration of geometric comparison for over-classified and under-classified
areas ....................................................................................................................... 89
Figure 3-7. Performance of DNN for training and validation datasets............................................. 91

Figure 3-8. High-level representations from DNN layers using t-distributed Stochastic Neighbor Embedding (t-SNE)................................................................. 91

Figure 3-9. Results of DNN model for (a) classification and (b) probability map for wetland areas ............................................................................................................. 94

Figure 3-10. Classification results of (a) deep neural network, (b) random forest, (c) support vector machine, and (d) k-nearest neighbor models........................................... 96

Figure 3-11. Performance of DNN classification for different spatial resolution (1.2, 2.5, 5, 7.5, 10, 15, 20, and 30 meters) .................................................................................. 98

Figure 4-1. Framework for SCP mapping using semantic segmentation method. ......................... 117

Figure 4-2. Location of Midwest U.S. states (study area) ........................................................... 118

Figure 4-3. Development of SCP training dataset ........................................................................ 120

Figure 4-4. U-Net architecture for the SCP classification, adapted from Ronneberger et al. (2015). ............................................................................................................. 123

Figure 4-5. Location of selected counties for SCP validation .................................................... 125

Figure 4-6. Results of SCP classification ..................................................................................... 128

Figure 4-7. Example of grassed waterways and terraces ......................................................... 129

Figure 4-8. Omission and commission errors ............................................................................. 130

Figure 4-9. Spatial distribution of agricultural SCP areas in the Midwest U.S. region .......... 132

Figure 4-10. Comparison of SCP areas (km2) among the states ............................................... 132

Figure 4-11. Percentage of SCP areas in the croplands ............................................................ 134

Figure 4-12. Distribution of the soil and elevation-related properties in the high-high and low-low clusters of SCPs ................................................................. 135

Figure 4-13. Comparison of soil erosion rates in the high-high/low-low clusters of % SCP values ............................................................................................................. 136
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-1</td>
<td>Description of land cover classes.</td>
<td>21</td>
</tr>
<tr>
<td>2-2</td>
<td>Summary of CNN models. All layers have ReLu as activation function.</td>
<td>26</td>
</tr>
<tr>
<td>2-3</td>
<td>Performance of CNN models with optimum hyperparameters.</td>
<td>38</td>
</tr>
<tr>
<td>2-4</td>
<td>Confusion matrix for proposed multi-OCNN framework.</td>
<td>42</td>
</tr>
<tr>
<td>2-5</td>
<td>Comparison of proposed multi-OCNN with different methods using F-score metric (Section 4.5).</td>
<td>48</td>
</tr>
<tr>
<td>2-6</td>
<td>Comparison of multi-OCNN with semantic segmentation methods using F-score values.</td>
<td>48</td>
</tr>
<tr>
<td>2-7</td>
<td>Computational time for different methods.</td>
<td>51</td>
</tr>
<tr>
<td>3-1</td>
<td>Number of samples for training and testing in each land cover class over the study area.</td>
<td>76</td>
</tr>
<tr>
<td>3-2</td>
<td>Description of potential variables for wetland classification.</td>
<td>80</td>
</tr>
<tr>
<td>3-3</td>
<td>Feature selection using principal component analysis.</td>
<td>81</td>
</tr>
<tr>
<td>3-4</td>
<td>Performance of DNN model for different groups of input variables.</td>
<td>92</td>
</tr>
<tr>
<td>3-5</td>
<td>Confusion matrix of the classification result from DNN model.</td>
<td>95</td>
</tr>
<tr>
<td>3-6</td>
<td>Assessment of geometric-based metrics for wetland mapping from different methods.</td>
<td>97</td>
</tr>
<tr>
<td>4-1</td>
<td>Characteristics of selected U.S. states and NAIP data used in the study.</td>
<td>118</td>
</tr>
<tr>
<td>4-2</td>
<td>Confusion matrix of SCP mapping across 20 counties.</td>
<td>128</td>
</tr>
<tr>
<td>4-3</td>
<td>Top 15 Midwest counties with the largest SCP areas.</td>
<td>133</td>
</tr>
</tbody>
</table>
ACKNOWLEDGMENTS

I want to express my sincere gratitude to my advisor, Dr. Amy Kaleita, for the opportunity and for giving me support during my journey at Iowa State. Her receptivity and guidance undoubtedly helped me to go through the Ph.D. program at the Department of Agricultural and Biosystems Engineering – ABE. I also would like to thank Iowa Department of Transportation for funding this project, especially Brad Hofer and Michael Carson. In addition to them, I am grateful for my committee members: Drs. Bradley Miller, Brian Gelder, Joshua Peschel, and Yuyu Zhou.

My sincere thanks to the ABE faculty who provided valuable knowledge and experience, as well as the ABE staff (Kris Bell, Kathy Walker, Sylvia Anderson, and Susan Ackerman), who were always kind and helpful when needed. Likewise, I thank the Center for Communication Excellence for the opportunity to join their writing consultation team, particularly Dr. Sarah Huffman (great leader). It was a great chance to broaden my writing skills and support other students.

Beyond the Iowa State community, I extend my thanks to Drs. Claudio Barbosa, Demetrius Silva, Evlyn Novo, and Hugo Guedes, previous mentors who always believed in me and who keep guiding me so positively. I would particularly like to thank those whom the conclusion of the doctorate would not be possible without: family and my friends. Most importantly, my deepest gratitude to my family (Antônio, Cláudia and Clarice) for their moral support and unconditional love, and my wife Flávia Xavier, who is the light of my life that agreed to go along with me on this journey. They all made the path less arduous and will always have a special place in my heart.
Deep Learning (DL) has become a breakthrough technology in machine learning, and opportunities are emerging for applications in the Earth Observation Science. Originally, DL algorithms were developed for computer vision problems, and the feasibility of these models needs to be explored for remote sensing topics, such as land cover mapping. Most DL studies are focused on urban mapping or a single scene, and the classification framework needs to be discussed for multiple-image, large-area implementation using high spatial resolution data. In this dissertation, three studies were conducted to explore DL algorithms in different contexts: (i) development of new multi-scale object-based convolutional neural network (multi-OCNN) for large-area land cover classification at 1-m spatial resolution; (ii) evaluation of the deep neural network (DNN) and WorldView-3 image for small wetland mapping; and (iii) mapping of structural conservation practices in Midwest U.S. cropland areas using semantic segmentation method. The research gaps of these studies are further explained in each chapter. In the first study, our findings show that combination of image segmentation, object analysis, and multiscale CNNs in the new multi-OCNN method achieved higher accuracy and faster classification compared to those results from pixel-based CNN and fixed-OCNN. The object analysis allows the selection of convolutional locations and appropriate input window for CNN prediction, which reduces the number of predictions per object and improves the spatial agreement of object and input patch size. Finally, the results show that multi-OCNN approach is a practical alternative for large-area application using traditional CNNs. In the second study, a pixel-based DNN classification produced a fine-detail mapping of wetlands in the Millrace Flats Wildlife Management Area, Iowa, USA. Our results show that DNN model achieved a good classification performance (0.933) in such a complex area, and the results are quite similar to other machine learning methods (random forest,
support vector machine and k-nearest neighbor). The results illustrate the benefits of feature selection procedure in the model performance, and the combination of spectral and topographic-related metrics is recommended. Also, this study demonstrated the impact of spatial-resolution on wetland classification, where the agreement between classified and reference areas increased from 61.22% at 30-m to 90.36% at 1.2-m resolution. In the last study, the adapted U-Net architecture was implemented to classify structural conservation practices (SCP) across Midwest U.S. croplands (overall accuracy: 76.8%). In general, the states with the highest percentage of SCPs in cropland areas are Iowa (26%), Illinois (15%), and Nebraska (11%) of total area (6,642 km²). The spatial distribution of SCPs shows the largest occurrence in the southwestern Iowa and eastern Nebraska. Our findings show that occurrence of percentage of SCP in cropland area is partially associated with soil and topographic characteristics such as slope and saturated hydraulic conductivity. In addition, most regions with high soil erosion rates present the largest percentage of SCP areas in croplands as well, indicating conservation efforts by farmers. The development of this product has positive implications for conservation programs, and geospatial inventory is the easily accessible product for large-area evaluation of conservation practices across Midwest U.S. croplands. In conclusion, this dissertation explores the potential of DL algorithms for different classification problems using high spatial resolution imagery, and remote sensing users can benefit from insights of each study.
CHAPTER 1. GENERAL INTRODUCTION

Overview

In recent years, Deep Learning (DL) has become a breakthrough technology in machine learning (LeCun et al., 2015), particularly for computer vision tasks. Inspired by the human nervous system, DL is fundamentally based on artificial neural networks (ANNs). The ANN consists of multiple layers with connected units (or neurons) that learn non-linear relationships from data (Goodfellow et al., 2016). The layers between input and output are known as hidden layers, and “deep” networks are typically defined by architectures with more than two hidden layers. Since the 1940s, research efforts have been ongoing in the development of neural nets (Schmidhuber, 2015). However, back then, these models were less attractive due hardware limiting for successful implementation. More recently, significant contributions in the topic have increased the research opportunities, and several studies have shown the capabilities of these algorithms (Voulodimos et al., 2018), for applications such as object detection, image segmentation, and speech recognition.

The current rise of DL is partially explained by three main aspects: technology development, large available datasets, and open-source frameworks. First, advancements in Graphics Processing Unit (GPU) technology have speeded up significantly the training of neural networks and make possible the implementation of deeper and complex architectures. For example, Shi et al. (2017) showed that a single GPU (GTX1080) is ~34 times faster than desktop CPU (i7-3820) using TensorFlow for ResNet-50 model training. Second, a vital part of data-driven methods is the availability of large reference datasets, and the increase of public data such as MS-COCO, ImageNet, MNIST, and CIFAR-10, has contributed to more experiments with DL algorithms. Lastly, the application of deep learning was empowered by new open-source
frameworks such as Tensorflow, Torch, Theano, Keras, and Caffe, and cloud services for data processing and storage. These frameworks give certain flexibility for users to develop their neural nets with less effort, increasing engagement on the topic. Following all these aspects, new DL architectures have been created and evaluated by the computer vision community (Guo et al., 2016; Liu et al., 2017; Garcia-Garcia et al., 2017; Shrestha and Mahmood, 2019).

Notably, one of the most relevant networks for image classification is the convolutional neural networks – CNNs (Rawat and Wang, 2017). These architectures are building blocks that contain three main neural layers: convolutional, pooling, and fully connected layers. The convolutional layers have a stack of 2-D filters with pre-defined sizes that extract feature representation of data (edge, corners, and curves). The optimization of filter weights is performed by the minimization of a cost function using the backpropagation algorithm. The learned features are used in the last fully-connected layer for the output label. These models are particularly attractive for pattern recognition and are well-recognized by their performance with large training dataset. In 1998, LeCun et al. (1998) successfully introduced the LeNet-5 architecture for handwritten digit recognition, and this became a famous work for image interpretation. More recently, Krizhevsky et al. (2012) won the ImageNet 2012 contest with AlexNet architecture - five convolutional and three fully-connected layers - training on two GPU cards. These two examples were highly relevant for the current progress on CNN algorithms. In remote sensing, the application of CNNs involves model training with multiple patches of spectral bands and corresponding labels, and later, classification of the central pixel of the patch image using sliding window algorithm. This process is computationally expensive due to the redundancy of overlapped patches, which limits the large-area land cover classification using standard CNNs.
As alternative to CNN models, fully convolutional network (FCN) is a prominent deep learning architecture for semantic segmentation of images (Long et al., 2015), including satellite remote sensing data (Mboga et al., 2019). Semantic segmentation is the task of labeling every pixel of the input image with a corresponding class, commonly referred to dense prediction. The main architecture consists of encoder/decoder blocks for a full-resolution segmentation map (Long et al., 2015). Rather than fully connected layers, FCN includes up-sampling layers to generate an output classification with the same dimensions as the input image. Well-known FCNs include U-Net (Ronneberger et al., 2015), SegNet (Badrinarayanan et al., 2017), FC-DenseNet (Jégou et al., 2017), and Deeplab networks (Chen et al., 2018). Since these studies are primarily focused on computer vision tasks, there are several opportunities for model evaluation in the Earth Observation context. For instance, Henry et al. (2018) evaluated the FCNs for road segmentation using Synthetic Aperture Radar images. Likewise, Mohammadimanesh et al. (2019) applied a new FCN architecture for complex wetlands in Canada with addition of inception modules and skip connections with residual units. They illustrated that proposed FCN algorithm outperformed the random forest results and other state-of-the-art FCNs (SegNet, FCN-32s, FCN-16s and FCN-8s). The traditional machine learning methods are commonly used in remote sensing with benefits for fast implementation and generalization capabilities. In contrast, supervised classification with a machine learning algorithm requires pre-defined handcrafted features as input data (Zhou et al., 2017). Since human experience and prior knowledge are necessary for this feature selection, there is no guarantee that best set of features is selected for a given task. In contrast, DL methods are considered end-to-end solvers where “data drive the learning” of such representative features for classification. Furthermore, machine learning algorithms are fixed forms with a limited set of
adjustable parameters, while the nature of DL architectures gives the developers the freedom to adjust the model structure for a specific problem to be solved.

Regarding remote sensing applications, DL has gained attention for land cover classification using high spatial resolution imagery (Zhu et al., 2017; Ma et al., 2019). In this context, most studies are focused on urban remote-sensing mapping, covering a small area or single scene (Audebert et al., 2018; Wu et al., 2018; Huang et al., 2018; Huang et al., 2019). There are potential explanations for this limitation on spatial coverage, such as acquisition costs for large areas. Another reason is the classification challenge of heterogeneous landscapes as the fine-detail mapping imposes a high intra-class variability and requires more training samples to achieve model generalization. In addition, large-area classification with high-resolution data involves critical data management like storage and memory usage, which demands specific strategies for fast processing of this volume of data. So far, the literature using CNNs for large-area land cover classification is scarce and needs to be addressed. Furthermore, following the “no free lunch” theorem in machine learning, there is no optimal algorithm for all possible problems, and the evaluation of DL models becomes key research for remote sensing studies.

Objectives

The motivation of this research was to explore deep learning methods for land cover classification and environmental analysis using high-resolution remotely sensed data. This research includes the application of three DL algorithms, such as deep neural network (multi-layer perceptron), convolutional neural network, and fully convolutional network. They were implemented to discuss the benefits and limitations focused on different mapping applications. Due to the multidisciplinary nature of this research, the main objectives for the dissertation are clearly stated in each chapter: the development of new multiscale object-based CNN (multi-
OCNN) approach for large-area land cover classification at 1-m spatial resolution (Chapter 2); the integration of deep neural network and WorldView-3 image for small wetland mapping (Chapter 3); and lastly, the implementation of semantic segmentation approach for mapping of structural conservation practices (SCP) across Midwest U.S. croplands (Chapter 4). Based on these objectives, the following research questions were addressed in this dissertation:

1. What are the benefits of multi-OCNN approach for large-area land cover classification? How relevant is the integration of multiscale CNN models? Is a large reference dataset important for CNN training? (Chapter 2)

2. Is the deep neural network model appropriate for mapping of small wetlands using WorldView-3 imagery? What is the performance of a DNN algorithm compared to other machine learning methods? What is the impact of spatial resolution in this wetland mapping? (Chapter 3)

3. Is the semantic segmentation algorithm (U-Net) useful for SCP classification? What are the benefits and limitations of this methodology? What is the spatial distribution of structural conservation practices across Midwest U.S. croplands? (Chapter 4)

Outline

This paper-based dissertation is organized as follows:

Chapter 1 introduces the overview of deep learning progress and the opportunities for remote sensing applications such as land cover mapping and environmental analysis.

Chapter 2 presents a new object-based convolutional neural network approach for land cover classification at 1-m spatial resolution. This chapter also describes the object analysis for fast implementation of multiscale CNNs in the statewide classification.
Chapter 3 describes the classification framework using deep neural network and Worldview-3 image for small wetland mapping. It also discusses the feature selection procedure, and compare the DNN results with other machine learning methods in this wetland mapping.

Chapter 4 presents the first mapping of structural conservation practices, mainly terraces and grassed waterways, by using adapted U-Net algorithm. The study shows the classification strategies for the efficient application of this algorithm. The results illustrate the spatial distribution of BMPs across Midwest U.S. croplands and uses other variables to understand the SCP distribution.

Chapter 5 summarizes the main findings of this dissertation and makes suggestions for future research on the topic.

Expected Outcomes and Practical Implications

The experiments presented in this dissertation are expected to fulfill some of the scientific knowledge gaps in the performance of DL algorithms for remote sensing applications at high spatial resolution. More precisely, this research will (1) demonstrate a new object-based CNN framework for large-area land cover classification, (2) evaluate the feasibility of deep neural network for mapping of small wetlands using Worldview-3 image, and (3) explore the fully convolutional network capabilities for conservation mapping. The method descriptions and the findings support the recent literature focused on DL algorithms for remote sensing data and provide directions for new users toward the consolidation of such algorithms in the land cover mapping and environmental analysis.
References


CHAPTER 2. EXPLORING MULTISCALE OBJECT-BASED CONVOLUTIONAL NEURAL NETWORK (MULTI-OCNN) FOR REMOTE SENSING IMAGE CLASSIFICATION AT HIGH 1-M RESOLUTION

Vitor S. Martins a, Amy L. Kaleita a, Brian K. Gelder a, Hilton L. F. da Silveira b, Camila A. Abe c

a Agricultural and Biosystems Engineering, Iowa State University, Ames, Iowa, USA.
b Brazilian Agricultural Research Corporation, Embrapa Territorial Intelligence, Campinas, SP, Brazil.
c Civil and Environmental Engineering, University of Wisconsin-Madison, Madison, Wisconsin, USA.

A manuscript submitted to ISPRS Journal of Photogrammetry and Remote Sensing

Abstract

Convolutional Neural Network (CNN) has been increasingly used for land cover mapping of remotely sensed imagery. However, large-area classification using traditional CNN is computationally expensive and produces coarse maps using a sliding window approach. To address this problem, object-based CNN (OCNN) becomes an alternative solution to improve classification performance. Previous studies were mainly focused on urban areas or small scenes, and implementation of OCNN method is still needed for large-area classification over heterogeneous landscape. Additionally, the massive labeling of segmented objects requires a practical OCNN for less computation, including object analysis and multiple CNNs. This study presents a new multiscale OCNN (multi-OCNN) framework for large-scale land cover classification at 1-m resolution over 145,740 km². Our approach consists of three main steps: i) image segmentation, ii) object analysis with skeleton algorithm, and iii) application of multiple CNNs. Also, we developed a large benchmark dataset, called IowaNet, with 1 million labeled images and C = 10 classes. In our approach, multiscale CNNs were trained to capture the best contextual information during the semantic labeling of objects. Meanwhile, skeleton algorithm
provided morphological representation (“medial axis”) of objects to support the selection of convolutional locations for CNN predictions. In general, proposed multi-OCNN presented better classification accuracy (overall accuracy ~87.2%) compared to traditional patch-based CNN (81.6%), fixed-input OCNN (82%) and semantic segmentation methods (78.9%). In addition, the results showed that this framework is 8.1 and 111.5 times faster than traditional pixel-wise CNN$_{16}$ or CNN$_{256}$, respectively. Multiple CNNs and object analysis have proved to be essential for accurate and fast classification. While multi-OCNN produced a high-level of spatial details in the land cover product, misclassification was observed for some classes, such as road versus structures and shadow versus lake. Despite these minor drawbacks, our results also demonstrated the benefits of IowaNet training dataset in the model performance; overfitting process reduces as the number of samples increases. The limitations of multi-OCNN are partially explained by segmentation quality, and limited spectral bands of aerial data. With the advance of deep learning methods, this study supports the claim of multi-OCNN benefits for operational large-scale land cover product at 1-m resolution.

**Keywords:** Deep learning; convolutional neural network; land cover; aerial imagery.

**Introduction**

Land cover classification is one of the most popular and challenging topics in remote sensing image processing (Gong et al., 2013; Zhu et al., 2014; Chen et al., 2015; Yang et al., 2018). Land cover maps have greatly advanced our knowledge about Earth’s terrestrial surface, providing critical information on natural resources and land management (Jin et al., 2013; Jia et al., 2014; Lu et al., 2016). However, despite the advances in image processing, large-area classification remains a difficult task for high-resolution satellite and aerial imagery (Yifang et al., 2015). Finer resolution data exhibit high detail (and intra-class variance), which pose a challenge for accurate
model prediction across heterogeneous landscape. Moreover, supervised classification requires i) a large annotated dataset to properly train a classifier and ii) an efficient framework to manage a massive number of pixels during the classification (Yang et al., 2018). Machine learning algorithms have been widely evaluated to provide land cover maps from remotely sensed data (Lu and Weng, 2007; Mountrakis et al., 2011; Belgou and Dragut, 2016; Mahdianpari et al., 2017; Ma et al., 2017).

Deep learning has rapidly become a key research field in machine learning. In the last decade, neural networks have achieved significant results in computer vision tasks (LeCun et al., 2010; Farabet et al., 2012; LeCun et al., 2015), such as object and facial recognition (Liu et al., 2015; Sun et al., 2015), self-driving cars (Tian et al., 2018), and audio recognition (Abdel-Hamid et al., 2014). The deep learning algorithms, such as convolutional neural networks (CNN or ConvNets), perform “end-to-end learning” to obtain hierarchical representation from input data. The multiple inter-connected layers provide the capability to fit a problem-specific model in a robust manner, without hand-crafted features or decision rules (LeCun et al., 2010). Recently, CNN architectures have been increasingly explored for remote sensing applications, including building extraction (Xu et al., 2018; Vakalopoulou et al., 2015; Alshehhi et al., 2017), vehicle or road detection (Lv et al., 2018), agriculture mapping (Kussul et al., 2017) and land cover classification (Castelluccio et al., 2015; Längkvist et al., 2016; Nogueira et al., 2017; Zhao et al., 2016; Hu et al., 2018).

While these studies have drawn significant attention to CNN architectures, there are certain limitations for practical land cover classification. Notably, patch-based CNN architecture takes advantage of contextual information from images to predict the class score at the end of the network. However, pixel-wise classification using traditional CNNs is a redundant process using
sliding window (stride = 1) across the image, and fully convolutional network (FCN) has been proposed for dense prediction output (Zhu et al., 2017). In remote sensing, patch-based CNN or FCN classification often produce coarse maps with blurred object edges (Paoletti et al., 2018). Intuitively, the input patch of CNN is often not consistent with real-world objects, leading to inaccurate classification of the edges (over-expansion or shrinkage) and geometry distortions in the final classification. To address these problems, recent studies have proposed the integration of CNN classifier and image segmentation, resulting in post-classification refinements, or the object-based CNN (OCNN) approach (Längkvist et al., 2016; Alshehhi et al., 2017; Zhao et al., 2017; Lv et al. 2018; Zhang et al., 2018; Huang et al., 2018; Liu et al., 2019; Mboga et al., 2019).

In general, the OCNN approach for land cover classification consists of two main steps: i) original image is segmented into homogeneous regions (i.e. objects) and then ii) object-based classification is performed using CNN model. As part of object-based image analysis (OBIA), segmentation tools produce relatively homogeneous groups of pixels, known as image objects, based on the spectral, geometric and spatial properties (Hay et al., 2003; Blaschke, 2010; Blaschke et al., 2014). These objects are functional representation of land targets such as buildings, roads, water bodies, and others. This OCNN integrates the advantage of edge-preserving objects and capabilities of CNN classifier to generate more consistent land cover maps. A promising OCNN approach was developed by Zhao et al. (2017). The authors classified urban objects using the most frequent CNN prediction class within the segmented area. Alshehhi et al. (2017) used the CNN probability map and segmentation results from a simple linear iterative clustering algorithm to improve the shape of classified roads and buildings. Similarly, Liu et al. (2019) proposed a CNN classification with post-classification refinement to produce object-based thematic maps using multispectral and radar data. Although successful, these studies still require a pixel-by-pixel
labeling using patch-based CNN across the entire image. This processing is computationally intensive and spatially redundant due to overlapping areas using sliding window.

In contrast, Zhang et al. (2018) developed an efficient object-based CNN (OCNN) approach for urban land use classification. The authors used object convolutional position analysis to classify the objects using less CNN predictions (instead of all pixels). The proposed method is more accurate and efficient than traditional pixel-wise CNN classification. Also, Lv et al. (2018) developed a CNN classification with region-based majority voting for high-resolution images. The authors performed CNN predictions in specific locations (point voters) within each segmented region, which reduces the number of class prediction per object. The authors also mentioned the benefits of this object-based method to preserve the boundary information. While OCNN is a promising approach for efficient land cover mapping, previous studies are mostly limited to small urban areas or single scenes (Lv et al. 2018; Zhang et al., 2018; Huang et al., 2018). Moreover, we argue that OCNN is highly dependent on multiple CNN models to solve the problem of observation scale in large context (Zhao and Du, 2016). The observation scale refers to the spatial extent under consideration and influences on how the objects appear in the images (Dabiri and Blaschke, 2019). Therefore, the spatial-related features (edge, contour, texture) are a scale-dependent information and the performance of CNNs is affected by the choice of window size (Zhang et al., 2018). With the growing interest in deep learning methods, we believe that further research is required to implement OCNN in large-area classification.

This study presents a new multiscale object-based CNN (multi-OCNN) approach for large-area land cover classification at 1-m resolution. The proposed multi-OCNN includes three main procedures: image segmentation, object analysis and scene classification using multiple CNNs. The National Agriculture Imagery Program (NAIP) aerial imagery (~6,100 tiles, 955 GB),
covering the state of Iowa in the United States, were used in this study. The performance of multi-OCNN approach was evaluated over eight regions. The main contributions are as follows: i) integration of multiscale CNNs and object-based approach for land cover classification; ii) development of a new benchmark dataset, called IowaNet (1 million images with 10 land cover types); iii) novel object analysis using skeleton-based method. Our results show that multi-OCNN provides a satisfactory framework for massive semantic segments (~1.01 billion) in a heterogeneous landscape (overall accuracy ~87.2%). As far as we know, this study is the first application of OCNN for scene classification in such a broad context. Given the challenges to turn CNN into a practical tool, this study supports the claim of capabilities of multi-OCNN for large-area classification. (Please note, the term “multiscale” is used in the context of multiple CNNs with different input sizes). The mapping results is available at NAIP Iowa 2015 DeepClass site (https://arcg.is/048PL5).

The paper is structured as follows: Section 2 describes related works using deep learning and OCNN approach. In Section 3, we describe the NAIP aerial images and IowaNet dataset. An overview of the proposed CNN architectures, image segmentation and OCNN framework are presented in Section 4. Section 5 and 6 present the results and discussion, respectively. Finally, the conclusions are drawn in the last section.

**Related work**

The promising CNN approaches were first presented for urban mapping using aerial imagery. In 2010, the successful application of patch-based CNN to road and building extraction (Mnih and Hinton, 2010) has supported other studies using these networks (e.g.: Chen et al., 2016; Maggiori et al., 2016; Saito et al., 2016; Alshehhi et al., 2017; Nogueira et al., 2017; Sharma et al., 2017; Audebert et al., 2018; Sun et al., 2019). For instance, Saito et al. (2016) proposed a building
and road extraction system using single patch-based CNN architecture. The proposed CNN predicts three classes (road, building and background) from a Massachusetts dataset. Using a multi-class method, Paoletti et al. (2018) applied 3-D CNN architecture for spatial-spectral classification of hyperspectral data in distinct areas (agricultural and urban). The authors also evaluated the performance of the deep network at different patch sizes. In another context, CNN models have been used to generate wetland inventory (Mahdianpari et al., 2018; DeLancey et al., 2020). Rezaee et al. (2018) evaluated the efficiency of patch-based CNN for wetland mapping; the classification results show a higher overall accuracy for CNN (94.82%) compared to a random forest classifier (79.11%). They mentioned a high redundancy during the classification because of overlapping input patches. Paisitkriangkrai et al. (2016) combined both CNN-derived and hand-crafted features for semantic labeling of aerial images. The post-processing step was also implemented to improve the final classification using conditional random fields. Similarly, Langkvist et al. (2016) developed a CNN classification with post-classification refinement for better representation of real-world targets (five-classes: vegetation, ground, road/parking/railroad, building and water).

In remote sensing, CNN-based classification involves the partition of original image into small patches and the trained network predicts a single label for the central pixel in the patch. As a result, land cover classification becomes a slow and expensive process for high-resolution images, especially on large-scale applications. Fully Convolutional Network (FCN) is an extension of the traditional CNN that performs semantic segmentation of images (Long et al., 2015). The FCN-based architectures consist of encoder/decoder blocks for full-resolution segmentation map, and some example are U-Net (Ronneberger et al., 2015), SegNet (Badrinarayanan et al., 2017), FC-DenseNet (Jégou et al., 2017), and Deeplab networks (Chen et al., 2017). By replacing fully-
connected layers with up-convolutional layers, FCN maintains the 2-D structure in the output classification, and predicts certain class for each pixel of original input image. This dense prediction map reduces the overlapping patches because sliding window uses the stride with the same dimension of input image. While FCNs are typically faster compared to traditional CNNs, several studies have pointed out the needed of post-processing of FCN results to refine the object boundaries (Marmanis et al., 2016; Sherrah, 2016; Fu et al., 2017; Audebert et al., 2018; Mboga et al., 2019) and efforts are needed to improve the final quality of mapping results.

As deep learning emerges as a land cover classification tool, the object-based CNN becomes a potential approach to improve the classification performance using image objects. Since segmentation provides a cluster of homogeneous pixels as objects, an effective solution should explore these objects for reducing computation. Technically, the success of OCNN approach relies on object analysis to define the convolutional locations for CNN predictions within object area. By applying the CNN in specific locations (instead of all pixels), the number of model predictions reduces substantially and speed up the overall processing. However, there are few studies on literature exploring object analysis in the OCNN framework (Lv et al. 2018; Zhang et al., 2018; Huang et al., 2018).

A relevant approach was implemented by Huang et al. (2018). The authors used a morphological operator to extract skeletons of mapping units (street blocks). The convolutional locations are selected along skeleton lines, and standard CNN is only applied in limited locations within the mapping unit. The final class label of object is defined by majority voting scheme. Although this procedure is useful for CNN classification, their proposed approach is still dependent on small urban features, such as street blocks to define mapping units. Likewise, Zhang et al. (2018) developed an object convolutional position analysis to define appropriate locations
for CNN application within the object. The method uses the shape features (linear and non-linear) of objects to select processing units for CNN prediction. The authors also emphasized that two CNN models with different input sizes were needed for complex objects (irregular shape), and future research should consider multiple models. Likewise, some studies have also highlighted the relevance of multiple CNNs for image classification (Zhao and Du, 2016; Liu et al., 2017).

The development of multiscale CNNs supports the extraction of deep spatial-related features in different observational scales (Liu et al., 2017). According to Zhao and Du (2016), multiple CNN models allow the framework to capture better contextual information during object classification. Längkvist et al. (2016) claimed the relevance of multiscale CNNs to consistently extract appropriate spatial relationships for CNN classification. Paisitkriangkrai et al. (2016) proposed a multiscale CNN that predicts an output based on the 16 × 16, 32 × 32, and 64 × 64 input patches. Following the recent progress, we proposed a new multi-OCNN approach with both object analysis and multiscale CNNs to improve the classification performance at 1-m resolution.

**Material**

**NAIP Aerial Imagery**

**Data Description**

National Agriculture Imagery Program (NAIP) is a nationwide program that provides high 0.5 to 2-meter resolution aerial ortho-imagery in the agricultural growing season across contiguous United States. This program is administered by USDA Farm Service Agency (http://www.fsa.usda.gov/). Although NAIP coverage varies across the U.S., this program typically delivers an annual dataset for most regions. Several studies have used the NAIP aerial imagery for land cover classification (Li et al., 2014; Maxwell et al., 2014, 2017; Basu et al., 2015; Nagel and Yuan, 2016). In this study, Iowa NAIP 2015 dataset was used to i) create a large training
dataset (Section 3.1.2), and ii) evaluate the application of OCNN for land cover mapping. The statewide dataset consists of ~6,100 image tiles, approximately 955 GB (140 - 170 MB per tile). The NAIP program provides clear-sky images with four spectral bands (blue, green, red and near-infrared) and 8-bit of radiometric resolution.

All NAIP imagery were obtained via USGS Earth Explorer (https://earthexplorer.usgs.gov/) and delivered in GeoTiff format with UTM 15N North American Datum 1983 (NAD 83). These images were acquired in late summer and early fall of 2015, mostly August and September. Although tiles are collected on different flight dates, the sensor quality and pre-processing assure the consistency of this publicly available dataset. The NAIP images were mosaicked into the county level, and later, all processing is implemented for each county.

**IowaNet training dataset**

The IowaNet is a new multiscale benchmark dataset that comprises 1 million images with 10 land cover classes (100,000 points per class). This dataset was developed as part of our efforts to implement this land cover classification, including intensive manual label (total: ~650 hours of work). The sample images were derived from Iowa NAIP 2015 data. The land cover classes were defined as follows (Figure 2.1 and Table 2.1): (1) structures (e.g., residential and commercial buildings); (2) roads; (3) river; (4) pond (e.g., lakes, reservoirs); (5) cultivated crops; (6) fallow; (7) shadow; (8) forest (e.g., deciduous, evergreen, and mixed forests); (9) grassland and herbaceous; and (10) barren land. These classes were defined according to other reference studies (Gong et al., 2013; Homer et al., 2015; Yang et al., 2018). Note that most studies represented river and lake classes as “water”, but they present different geometry and size, and we decided to separate them for better CNN performance. Each class has 100,000 patch images, and the dataset consists of two sets of spectral bands: natural color (blue, green, and red) and false-color bands.
(green, red, and near-infrared). The IowaNet is now available as open-access dataset (https://doi.org/10.5281/zenodo.3385318).

We performed an intensive manual labeling of 1 million points across Iowa. The label of each point was visually interpreted using 1-m NAIP 2015 image and ArcGIS online services. These points are known as “seed points” because each one is the central pixel to extract different patch sizes (growing window). By doing that, we created a multiscale dataset with six patch sizes (8 x 8, 16 x 16, 32 x 32, 64 x 64, 128 x 128, 256 x 256 pixels) with label of central pixel. For each patch size, the training dataset is a pair of labeled images \(((x_1, y_1), \ldots, (x_n, y_n))\), where \(x_n\) is the i-th input data (patch image), \(y_n\) is the i-th label data (land cover class) and \(n\) is 1 million images. Since patch images were samples in a distinct context, the dataset is spatially representative and includes the intra-class variability for large-area classification. We should also mention that the implications of atmospheric effects on classification results are considered negligible because aerial images are typically acquired at clear-sky days and low altitudes (↓ atmospheric path), statewide sampling produces high intra-class spectral variability, and CNNs explore spatial patterns besides the spectral information. Due to 1 million samples and six resolutions, the IowaNet dataset has a potential to become a benchmark remote sensing dataset for deep learning applications (see others in Zhou et al., 2018). Finally, we trained the proposed CNN models using a false-color IowaNet dataset (Section 4.1.2).
Figure 2. Examples of land cover classes in IowaNet dataset (1 million samples, 10 land-cover classes). The training dataset was derived from NAIP (National Agriculture Imagery Program) aerial imagery and there are two sets for users: natural color (blue, green, and red) and false-color (green, red, and near-infrared) datasets.

Table 2-1. Description of land cover classes.

<table>
<thead>
<tr>
<th>No.</th>
<th>Land cover</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Barren land</td>
<td>Bare soil land (80% soil background), sandy land, cattle corrals, mining areas and riverbanks</td>
</tr>
<tr>
<td>2</td>
<td>Cropland</td>
<td>Cultivated crop areas (vegetated), and irrigated farmlands</td>
</tr>
<tr>
<td>3</td>
<td>Fallow</td>
<td>Post-harvest, plant canopy senescence, and abandoned arable lands</td>
</tr>
<tr>
<td>4</td>
<td>Forest</td>
<td>Deciduous, evergreen, and mixed forest areas</td>
</tr>
<tr>
<td>5</td>
<td>Grassland</td>
<td>Herbaceous cover, pasture hay, meadow, natural pasture (open spaces)</td>
</tr>
<tr>
<td>6</td>
<td>Lake</td>
<td>Reservoirs, ponds, lakes and wetlands</td>
</tr>
<tr>
<td>7</td>
<td>River</td>
<td>Natural stream of flowing water, such as canals, streams</td>
</tr>
<tr>
<td>8</td>
<td>Road</td>
<td>Road, lane, highway, pavements, and rail tracks</td>
</tr>
<tr>
<td>9</td>
<td>Shadow</td>
<td>Tree and building shadows</td>
</tr>
<tr>
<td>10</td>
<td>Structures</td>
<td>Residential, commercial, and industrial buildings, farmhouses, barns and silos</td>
</tr>
</tbody>
</table>
Method

In this section, we describe a new multiscale OCNN framework for large-scale classification of aerial images. The proposed framework is shown in Figure 2.2. First, six CNN models were trained using the IowaNet dataset. Second, image segmentation is performed for each county image across Iowa. Third, object analysis is implemented for CNN prediction of land cover class. Finally, the labeled objects are merged in the final aerial scene classification. Further details of these steps are discussed in the following sections.

Convolutional Neural Network (CNN) architecture

Background

Convolutional Neural Network (CNN) is a specific type of feed-forward neural network used for image recognition and classification (LeCun et al., 1998; 2010). A typical CNN architecture consists of a series of layers such as i) convolutional, ii) pooling, and iii) fully-connected (FC) layers. Also, dropout and batch normalization layers can be used to avoid overfitting and improve the generalization of the model (Srivastava et al., 2014). This multi-layer architecture extracts abstract features from input data and then returns the class score at the end of the network. A convolutional layer is the central building block in the network and has a set of
trainable weights to learn spatial features in images, from low- (edge, corner, contour) to high-level (complex structures). The convolutional layer receives a three-dimensional array and computes the output feature maps. These feature maps are the result of dot product of receptive field and a set of weights in filters (or kernels). In the learning process, these weights in convolutional and FC layers are adjusted to capture the most relevant features for each class. This learning procedure is performed using a backpropagation algorithm and stochastic gradient descent (SGD). Further details of this procedure are given in LeCun et al. (1998). In each convolutional layer, the number of filters (M) is equivalent to the number of output feature maps. Therefore, a convolutional layer relates the tensor $x_j \in \mathbb{R}^{m \times n \times j}$ of j-th feature map in previous layer ($i-1$) to the output feature map $z \in \mathbb{R}^{\bar{m} \times \bar{n} \times \bar{j}}$ of current layer ($i$) as follows:

$$z_{i}^{k} = \text{pool}_{\text{max}} \left( \sigma \left( b_{k}^{i} + \sum_{j=1}^{M^{i-1}} W_{k,j}^{i} \ast x_{j}^{i-1} \right) \right)$$  \hspace{1cm} (1)

where $i$-th is the convolutional layer, $W_{k,j} \in \mathbb{R}^{l \times l \times k}$ is a k-th convolutional filter (weights), $(\ast)$ is a two-dimensional convolutional operator, $b_{k}$ is a bias term, $\sigma(\cdot)$ is an element-wise nonlinearity function (e.g. REctified Linear Unit (ReLU): $\sigma(x) = \max(0, x)$), and $\text{pool}_{\text{max}}$ is a max pooling function. A max pooling layer computes the maximum values of rectangular regions of its input. The max pooling is a typical operation after the convolutional layer and becomes an important layer to provide a translational invariant feature and to reduce the number of trainable weights. The dimensions $\bar{m}$ and $\bar{n}$ of outputs rely on pooling dimension, stride and padding parameters. In the first layer, input dimensions ($m$, $n$, and $j$) of image represent row, column, and number of spectral, respectively. After a series of convolution layers, the Flatten layer
converts the feature maps from 2-dimensional arrays to 1-D vector for FC layers. At the end of network, the last FC layer has a multi-class softmax function that computes the class probabilities. Once the architecture is defined, the next step is the optimization of network parameters using a backpropagation algorithm with gradient descent (Johnson and Zhang, 2013). In this context, the “learning” involves the minimization of error between target and predicted value calculated by loss function, such as categorical cross-entropy (multi-class problem).

**Multiscale CNNs: development and training**

In this study, we developed a multiscale approach with six CNN architectures for land cover classification (10 classes). These architectures follow the well-known AlexNet (Krizhevsky et al., 2012) and Lenet-5 networks (LeCun et al., 1998). Several studies have developed CNNs for remote sensing data based on these two architectures (Hu et al., 2015; Zhang et al., 2018; Lv et al., 2018; Li et al., 2019). These architectures were slightly modified for our purpose. For multiscale framework, six CNN architectures were developed and trained with different input resolutions: CNN8 (8 x 8), CNN16 (16 x 16), CNN32 (32 x 32), CNN64 (64 x 64), CNN128 (128 x 128) and CNN256 (256 x 256 pixels). Overall, AlexNet and LeNet architectures were modified for our purpose, and several experiments were performed to define the number of convolutional layers as well as the regularization/normalization layers. The example of CNN16 network is presented in Figure 2.3, and the summary of all models is shown in Table 2.2. For instance, the basic architecture of CNN16 consists of an input layer (16 x 16 x 3 bands), four convolution layers (+ ReLU and batch normalization), two max-pooling + dropout (2\textsuperscript{nd} and 4\textsuperscript{th} layers), flatten layer, and two fully-connected layers. The last layer contains 10 neurons and delivers the probability vector with softmax function.
While the hidden layers vary for these architectures, the hyperparameters are quite similar among them. For each model, these parameters were tuned empirically through cross-validation accuracy. For example, the number of epochs was tested from 25 to 200. The kernel size of convolutional filters was evaluated for 3 x 3, 5 x 5 and 7 x 7 with stride = 1 (Table 2.2). The pooling layer was fixed with 2 x 2 window for all models. The dropout regularization was tested with different rates (0.1 to 0.5). Following Zhong et al. (2019), Adam parameters were fixed as $\beta_1 = 0.9$, $\beta_2 = 0.999$, and learning rate = 0.001 with decay set to 0. These parameter settings are shared across all models.

The CNN architectures were fully trained from scratch using IowaNet dataset (false-color images). In the training step, the dataset was subdivided into three sets: training (90%), validation (7.5%), and test dataset (2.5%). All labeled images were normalized between 0 and 1. The training and validation datasets adapt the weights of the CNN and validate the model result in each epoch. The validation provides further evaluation of a model fit on the training dataset at the end of each epoch. Once the models are completely trained, the test dataset is used for final evaluation of predictions using “unseen” images. Six models were trained using mini-batch stochastic gradient descent (batch size was set to 64). The CNNs were trained in the High-Performance Computing (HPC) cluster at Iowa State University. The HPC cluster contains two 8-Core Intel E5 2650 with two NVIDIA Tesla K20 GPU 5GB cards. The computational performance is significantly faster with GPU card due to hundreds of cores for matrix and vector operations (instead of CPU with a few cores), but the limited memory constrains the maximum size of the CNN architecture and the number of bands (here: false-color bands). All these experiments are conducted with Keras/TensorFlow module in Python environmental and its open-source libraries, such as GDAL, PIL, NumPy, Rasterio, and Scikit-image.
Table 2-2. Summary of CNN architectures. The model name refers to input image size (e.g., CNN8 has input of 8 x 8 pixels). The acronyms are defined as follows: Conv is convolutional layer, FC is fully-connected layer, MP is maximum pooling layer, BNM is batch normalization layer, and DP is dropout. All Convs have Rectified Linear Unit as activation function. The number of filters and its size per convolutional layer are presented in the first line of each model, such CNN8 with 64 filters of 3 x 3 pixels (64@ 3 x 3) in the Conv1 layer. The dropout rate and filter dimension of max-pooling are presented in the parenthesis.

<table>
<thead>
<tr>
<th>Models</th>
<th>Conv1</th>
<th>Conv2</th>
<th>Conv3</th>
<th>Conv4</th>
<th>FC1</th>
<th>FC2</th>
<th>FC3</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN8</td>
<td>64@3x3</td>
<td>32@3x3</td>
<td>64@3x3</td>
<td>32@3x3</td>
<td>n=512</td>
<td>n=256</td>
<td>n=10</td>
</tr>
<tr>
<td></td>
<td>BNM</td>
<td>MP (2x2)</td>
<td>-</td>
<td>BNM</td>
<td>-</td>
<td>-</td>
<td>SoftMax</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DP (0.2)</td>
<td>-</td>
<td>DP (0.2)</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CNN16</td>
<td>64@5x5</td>
<td>32@5x5</td>
<td>64@3x3</td>
<td>32@3x3</td>
<td>n=1024</td>
<td>n=512</td>
<td>n=10</td>
</tr>
<tr>
<td></td>
<td>BNM</td>
<td>BNM</td>
<td>BNM</td>
<td>BNM</td>
<td>-</td>
<td>-</td>
<td>SoftMax</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MP (2x2)</td>
<td>-</td>
<td>MP (2x2)</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DP (0.2)</td>
<td>-</td>
<td>DP (0.2)</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CNN32</td>
<td>32@3x3</td>
<td>32@3x3</td>
<td>32@3x3</td>
<td>32@3x3</td>
<td>n=512</td>
<td>n=10</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>BNM</td>
<td>BNM</td>
<td>MP (2x2)</td>
<td>DP (0.5)</td>
<td>SoftMax</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MP (2x2)</td>
<td>-</td>
<td>DP (0.2)</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DP (0.2)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CNN64</td>
<td>64@5x5</td>
<td>64@5x5</td>
<td>64@3x3</td>
<td>64@3x3</td>
<td>n=512</td>
<td>n=10</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MP (2x2)</td>
<td>-</td>
<td>MP (2x2)</td>
<td>DP (0.25)</td>
<td>SoftMax</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DP (0.25)</td>
<td>-</td>
<td>DP (0.25)</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CNN128 &amp; CNN256</td>
<td>32@5x5</td>
<td>32@3x3</td>
<td>32@3x3</td>
<td>-</td>
<td>n=512</td>
<td>n=10</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>BNM</td>
<td>MP (2x2)</td>
<td>MP (2x2)</td>
<td>-</td>
<td>DP (0.5)</td>
<td>SoftMax</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>MP (2x2)</td>
<td>DP (0.25)</td>
<td>DP (0.25)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Figure 2-3. The architecture of CNN16 model.
Multi-layer perceptron network

Although this study is focused on the integration of CNN models, we also argue that observation scale plays a critical role for successful classification since the object details and spatial context are dependent on window size (Dabiri and Blaschke, 2019). In some cases, tiny objects (e.g., individual trees, small barren areas, and noise polygons) with few pixels become a challenging task for CNNs because they represent a small fraction of patch area, and contextual information can lead to misclassification. Instead of 2-D images, pixel-level information is more consistent for land cover classification. In this context, we developed and trained a Multi-Layer Perceptron (MLP) network to support the multi-OCNN framework, especially for tiny areas. In general, MLP is a feedforward artificial neural network that contains fully-connected layers (or dense layers) with arbitrary numbers of neuron units for data classification. Each neuron has a learnable weight with non-linear activation functions in the hidden layers. The network weights are adjusted in a supervised way. The training procedure minimizes the difference between outputs and correct values using back-propagation algorithm.

In this study, MLP network has pixel-level input values extracted from 1 million reference points of IowaNet (see section 3.1.2). For each point, we stored the values of four spectral bands (blue, green, red, and NIR) with corresponding label, and they were used to train the MLP network. Regarding the model architecture, five hidden layers and its neurons (1st layer: 256, 2nd layer: 512, 3rd layer: 1024, 4th layer: 512, and 5th layer: 256) were defined with some experiments through cross-validation. All hidden layers have ReLU as non-linear activation function. The dropout of 0.5 is included in 5th layer to improve the classification results. The hyperparameters were fine-tuned based on cross-validation. The learning rate of Adam was set to 0.015. The output layer gives the class probability for each land cover. The application of MLP in the multi-OCNN framework is presented in Section 4.4.
**Image segmentation**

Image segmentation is a pre-processing step for OCNN application. In this study, the image segmentation was performed by Mean-Shift algorithm (Comaniciu and Meer, 2002). This popular OBIA algorithm produces segmented images with homogeneous spatial and spectral information from input data. The mean-shift algorithm presents the advantage of simple parametrization with great edge detection in high-resolution images (Wang et al., 2012; Ming et al., 2012; Su et al., 2015; Sun et al., 2019). Additionally, this segmentation tool has proven to be useful in both natural and urban contexts (Wang et al., 2015). The region growth is defined by specific homogeneity criterion and all pixels are grouping when they are closer in both spatial and spectral domain (Hossain and Chen, 2019). The mean-shift algorithm has three scale parameters: spatial scale ($h_r$), spectral scale ($h_s$), and minimum segment size ($M_r$).

In this study, false-color bands (NIR, red, green) were used as input data in Mean-Shift algorithm, and the optimal scale values were defined by quantitative and visual assessments over eight testing counties (see counties in Section 4.5). Several combinations of scale parameters were tested by varying $h_s$ and $h_r$ values from 10 to 20 (step = 0.5). Following Zhang et al. (2018), $M_r$ value was predefined as 25, and this minimum region keeps small objects in urban area and avoids under-segmentation results. A total of manually delineated polygons ($n = 1200$) from different land cover classes were compared to segmented areas in these experiments. The Jaccard index ($J = |G \cap R| / |G \cup S|$) was calculated to measure the similarity of segmented ($R$) and reference ($G$) polygons. The Jaccard index, also known as Intersection-Over-Union, ranges from 0 (worst; no match) to 1.0 (best; perfect match) and can penalizes both over- and under-segmentation (Polak et al., 2009). Note that under-segmentation process is characterized by single segment (large polygon) representing many real-world objects, while the over-segmentation is exemplified by many segments (small polygons) representing one object. The description of J-index can be found
in Ge et al. (2007). In addition, we visually assessed the segmentation scale effects for further consideration. Based on the results (Section 5.2), the global scale parameters were selected as \([hr, hs, Ms] = [16.0, 16.5, 25]\) for segmentation of NAIP images. This segmentation step creates \(~1.01\) billion segments for entire Iowa 2015 data, ranging from 2.7 to 24.5 million segments per county. See the sub-set of image in Figure 2.4. The processing time for segmentation is 2.0 to 3.0 hours per county using Intel Xeon(R) CPU E3 -1270 @ 3.80 GHz (36 GB memory). In the next section, we explain the procedures for categorical label of these semantic-free segments using trained CNNs.

![Figure 2-4. Example of segmentation results.](image-url)
Multi-OCNN approach for scene classification

This section describes the multiscale OCNN (multi-OCNN) framework for land cover classification of high-resolution imagery. Essentially, the proposed multi-OCNN assigns a semantic label for image-objects using multiple CNNs and has two advantages. First, object-based classification reduces the number of CNN predictions compared to pixel-by-pixel approach. Second, the multiscale approach increases the agreement of object shape and CNN input patch in OCNN classification. Again, note that the term “multiscale” is associated to multiple CNNs with different input sizes. In the pre-processing step, mean-shift segmentation generates functional objects for NAIP Iowa 2015 counties (Section 4.3). The semantic-free segments are input polygons for this object analysis, where the algorithm defines a number of convolutional locations and input patch size for CNN predictions.

In general, the object analysis includes i) the application of skeleton algorithm, ii) selection of convolutional locations and iii) determination of input patch size (Figure 2.5). The object classification relies on multiple locations (or processing units) for CNN predictions, and later, the final object label is defined as the highest-class membership computed from all predictions. Each image-object is treated independently as mapping unit. The object analysis starts by generating morphological lines (“medial axis”) of object with skeletonize algorithm. This method performs a successive evaluation of border pixels and removes those pixels until that connected line is created (Zhang and Suen, 1984). This algorithm is a simple and fast technique to extract skeleton representation of any object (complex shapes with different sizes). The skeleton lines are then used to support the determination of multiple locations within the image-object. In this stage, convolutional locations are created with a 16-m interval along the skeleton line (Figure 2.5). This interval is a “user-defined constant”, but it should be close to the smallest input window to ensure the largest coverage across the object. Also, users should not expect an abrupt change in the
performance by adjusting this parameter a few meters. When the object area is lower than 256 m², 
the geometric centroid of object is defined as a single convolutional location. This location is a 
central coordinate for patch extraction from original data.

Once the object analysis produces the convolutional locations, the next step is the 
determination of optimal window size for patch extraction from original data. The algorithm 
computes the shortest distance ($S_d$) between convolutional location and object edge. The patch size 
($S$) is defined as $S = 2 \times (S_d + 0.1 \times S_d)$, and then, the algorithm chooses the CNN model with 
closest larger predefined patch to $S$ value for class prediction in the i-th location (Figure 2.5). For 
instance, when the calculated $S$ is 220 m, the closest larger predefined patch is 256 and CNN256 
model is used for this convolutional location. The factor (0.1) assures at least 10% of the contextual 
information in the input patch area. However, if the convolutional location is nearby object edge 
($S_d < 3$ meters), pixel-level MLP network is used for probability prediction in combination with 
CNNs. The reason for this that contextual information prevails in the input patch area and CNN 
application may not be consistent in some cases. Note that object analysis (skeleton → 
convolutional location → input patch size) is implemented for massive number of objects (this 
study: ~1.01 billion), and this analysis has proven to be fast and robust for our application. For 
instance, the time processing of large residence (189 m²), natural pond (850 m²), large cropland 
(259,268 m²) takes 0.002, 0.0069, and 0.49 secs, respectively. Finally, our processing system 
generates several input patches (Figure 2.6), and multiple CNNs compute per-class probability for 
each convolutional location. The object label is defined as land cover with highest class score.

In short, let us suppose that county H contains N semantic-free objects $O_r$, $r \in \{1, 2, \ldots, 
N\}$, where object $O_z$ has M convolutional locations $I_i$, $i \in \{1, 2, \ldots, M\}$. At each convolution 
location $I_i$, trained CNN predicts a n-dimensional vector $\vec{P} = [c_1, c_2, \ldots, c_n]$, where $c_n$ is the
membership probability for each class \( n \in \{1, 2, \ldots, 9, 10\} \). Since probability vectors are obtained in multiple locations across the object, the output class label (C) per object is defined as the highest value of vector \( P \) from all locations:

\[
\vec{P}_i = [c_1, c_2, \ldots, c_n] = \sum_{n=1}^{10} c_n = 1; \quad 0 \leq c_n \leq 1
\]

(2)

\[
C = \text{argmax} \left( \sum_{i=1}^{M} \vec{P}_i \right) = \text{argmax} \left( \begin{bmatrix}
    c_{1,1} + c_{1,2} + \cdots + c_{1,M} \\
    c_{2,1} + c_{2,2} + \cdots + c_{2,M} \\
    \vdots & \vdots & \ddots & \vdots \\
    c_{n,1} + c_{n,2} + \cdots + c_{n,M}
\end{bmatrix} \right)
\]

(3)

This classification procedure is applied for all image-objects. Lastly, the multi-OCNN generates a statewide land cover product merging all county-based results.

**Figure 2-5. Framework of the object analysis.** This analysis generates skeleton lines, convolutional locations and patch sizes for multiscale CNN predictions per object. \( S_d \) represents the shortest distance between convolutional location and object boundary, and \( S \) is the patch size to support the selection of CNN model.
Figure 2-6. Results of object analysis for different targets.

Evaluation metrics and comparison

The classification assessment was conducted across eight regions (Figure 2.7). A two-stage stratified random sampling was implemented for reference dataset (Stehman, 2009; Stehman and Foody, 2019). First, a simple random sampling creates 1250 points in each region. Since some classes occur in a small proportion of area, the second step is a sampling scheme to ensure at least 125 samples per class, increasing the representation of these scarce classes. A total number of 10,000 samples were labeled as reference data (1,250 samples per class). Due to spectral quality and 1-m resolution of NAIP images, the uncertainty of visual interpretation is assumed negligible because land cover classes are clearly identifiable. The confusion matrix and accuracy metrics were calculated for quantitative evaluation (Congalton and Green, 2002; Foody, 2009; Tso and
Mather, 2001), such as overall accuracy (OA), kappa coefficient (κ), producer’s (PA) and user’s accuracies (UA), and per-class F-score. In this context, the confusion matrix allows the evaluation of agreement and disagreement in the classification. The producer’s accuracy, which is related to omission errors, shows the proportion of the reference samples per class that is correctly classified in the map. The user’s accuracy, which is related to commission error, presents the proportion of classified pixels per class in the map that is actually presented on the reference samples. The F-score is the harmonic mean of PA and UA (F-score = \(2 \times (PA \times UA)/(PA+UA)\)). A larger value indicates better predictive accuracy, ranging from 0 to 100%. Additionally, other CNN-based frameworks were implemented to evaluate our proposed multi-OCNN:

**Pixel-wise CNN**: this method is the traditional pixel-wise CNN that classifies pixel-by-pixel using single CNN model and sliding window. The traditional CNN for remote sensing becomes highly redundant because the input patch overlaps to predict the class label for central pixel of the patch. **Reason**: this comparison shows the difference between object-based and pixel-wise CNN classification. In the result section, CNN\(_{16}\) states for pixel-wise CNN with input patch of 16 x 16 pixels.

**Fixed-OCNN**: this method uses the object classification with a single CNN model. The term “fixed” means a single model in this context. The object analysis is only partially implemented, because the determination of input patch size from the object was not used in this method. **Reason**: this comparison shows the benefits of multiscale CNNs to classify image objects. We evaluated this method using three CNN models (16 x 16, 64 x 64, 256 x 256 pixels). In the result section, OCNN\(_{16}\) states for object-based CNN with fixed input patch of 16 x 16 pixels.

**OCNN**: this method is similar to multi-OCNN, but it uses only geometric centroid for object classification. The object analysis is also partially implemented because the convolutional
locations were not used in this method. The OCNN uses a single prediction according to appropriate patch size. **Reason**: this comparison evaluates the benefits of skeleton algorithm with multiple convolutional locations rather than a single prediction for object label.

**OCNN all**: this ensemble method consists of application of all six models per convolutional location. The final label of object is defined by majority voting of all predictions using all locations. **Reason**: this comparison illustrates the necessity of determination of input patch size for object-based classification.

**OCNN dense**: this method was inspired in the OCNN method from Zhao et al. (2017). In general, pixel-wise classification was implemented with CNN32 model, and the final label of object is defined by majority voting within the segment. This method integrates the CNN prediction and object segments in a straightforward application. **Reason**: this method is a simple strategy for refinement of CNN results.

**U-Net**: this FCN-based model performs semantic segmentation using a symmetrical architecture with 4 convolutional blocks for encoder and decoder parts. A detailed description of U-Net can be found in Ronneberger et al. (2015). The original U-Net architecture was implemented with 32 features maps in the first layer and input/output size of 128 x 128 pixels. The output layer has class probability per pixel, and the largest probability is used to predict the final label. For this application, we developed 2000 annotated patches (250 per testing area), and then, data augmentation (e.g., transpose and rotation) was applied to generate additional 3500 patches in the final dataset (total: 5500 samples with 128 x 128 pixels). With 100 epochs (79 secs/epoch), the validation and testing accuracies were 0.966 and 0.963, respectively. **Reason**: this comparison shows the benefits of multi-OCNN compared to well-known semantic segmentation model.
SegNet: this method also performs a semantic pixel-wise labelling using deep convolutional encoder-decoder architecture (Badrinarayanan et al., 2017). The SegNet architecture has 5 convolutional blocks for each part, and we used the labeled dataset (5500 samples) for model training. The validation and testing accuracies were 0.9647 and 0.961, respectively, using 50 epochs (3117 secs/epoch). **Reason:** SegNet is a popular method for semantic segmentation and represents an additional comparison of multi-OCNN.

FPN: Feature Pyramid Network (FPN) is used for semantic segmentation using high-level semantic feature maps at different scales. This architecture implements pyramid representations of independent predictions during top-down pathway for object detection/label. A detailed description of FPN is presented in Lin et al. (2017). The original FPN model was trained with 5500 labeled samples (similar to U-Net and SegNet), and the validation and testing accuracies were 0.965 and 0.966, respectively, using 50 epochs (320 secs/epoch). **Reason:** FPN method also explores multiscale features for object classification.

The binary cross-entropy, Adam, and Softmax were used, respectively, as loss function, optimization algorithm, and output activation function for all segmentation methods (U-Net, SegNet, and FPN). Note that these models were trained with limited number of samples but it still relevant for the overall discussion.
Results

Performance of proposed CNNs

Table 2.3 shows the performance of CNN models for training, validation, and testing datasets. In general, proposed CNNs perform well in testing dataset, ranging from 83.9% (CNN8) to 93.2% (CNN256). Our findings show that increasing patch size improves model performance. In addition, we observe almost similar accuracies between training, validation and testing datasets. The agreement of training and testing accuracy is a positive measure for our study, indicating low or no overfitting in these models. In fact, the model architectures were developed to achieve a good generalization for our classification. The MLP network shows a lower accuracy (72.1%) compared to other CNNs. The pixel-level information introduces a real challenge for accurate predictions due to spectral similarity between land cover classes, such as “cropland vs. pasture” or “road vs. building roofs”. Also, the computational time for model training increases from small to large patch sizes, ranging from 241 to 6,544 seconds per epoch. The total time is dependent on model architecture (e.g., shallow or deep) and the number of epochs, such as CNN32 (~6.4 hours) versus CNN256 (~90.9 hours). It should be mentioned that total time only represents the model training, and the processing time for scene classification is presented in Section 5.3.
Table 2-3. Performance of CNN models with optimum hyperparameters.

<table>
<thead>
<tr>
<th>Model</th>
<th>Input patch</th>
<th>Time per epoch</th>
<th>Total time</th>
<th>Accuracy (%)</th>
<th>Loss</th>
<th>No of Epochs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>secs</td>
<td>hours</td>
<td>Training</td>
<td>Validation</td>
<td>Testing</td>
</tr>
<tr>
<td>MLP</td>
<td>d = 1 x 1(pixels)</td>
<td>101</td>
<td>8.41</td>
<td>71.8</td>
<td>0.832</td>
<td>71.9</td>
</tr>
<tr>
<td>CNN8</td>
<td>d = 8 x 8</td>
<td>241</td>
<td>10.0</td>
<td>83.5</td>
<td>0.450</td>
<td>83.8</td>
</tr>
<tr>
<td>CNN16</td>
<td>d = 16 x 16</td>
<td>240</td>
<td>10.0</td>
<td>88.9</td>
<td>0.305</td>
<td>88.3</td>
</tr>
<tr>
<td>CNN32</td>
<td>d = 32 x 32</td>
<td>305</td>
<td>6.4</td>
<td>90.0</td>
<td>0.295</td>
<td>90.5</td>
</tr>
<tr>
<td>CNN64</td>
<td>d = 64 x 64</td>
<td>1791</td>
<td>24.9</td>
<td>92.1</td>
<td>0.229</td>
<td>90.1</td>
</tr>
<tr>
<td>CNN128</td>
<td>d = 128 x 128</td>
<td>1450</td>
<td>28.2</td>
<td>92.7</td>
<td>0.228</td>
<td>93.3</td>
</tr>
<tr>
<td>CNN256</td>
<td>d = 256 x 256</td>
<td>6544</td>
<td>90.9</td>
<td>94.2</td>
<td>0.182</td>
<td>93.2</td>
</tr>
</tbody>
</table>

A crucial step in CNN application is the amount of training dataset required to properly learn the network parameters. The findings in Figure 2.8 demonstrate the benefits of large datasets such as IowaNet for model generalization. In this experiment, CNN8 and CNN32 models were trained with different sizes (10,000; 50,000; 100,000; 250,000; 500,000; 900,000 samples). The results show that small training dataset (10,000) leads to high difference between training and testing accuracy in both models. For example, training accuracy of CNN32 is ~95.6%, while testing is 84.3%. This disagreement should be interpreted with caution because this is strong evidence of model overfitting. Overfitting is a common problem in deep learning models where the network performs well in the training dataset (closely fitted) but it has a generalization problem to make accurate predictions for new/unseen samples. In contrast, the training accuracy decreases from small to large datasets, and the overall agreement between training and testing increases for a large number of samples. We observed that the model performance for 900,000 samples is quite similar for both training and testing datasets; this is a positive measure of model generalization.
Figure 2-8. Performance for (a) CNN8 and (b) CNN32 models with different training sizes. This experiment is performed up to 100 epochs.

Segmentation scale effects

The summary of segmentation results with different scale parameters is presented in the Figure 2.9. The Jaccard index values represent the geometric similarity of manually delineated and segmented polygons, and the largest values show more agreement between them. Overall, the best results were observed with spectral scales between 15 and 18, while the spatial values affect the segmentation, but they have less impact compared to spectral. The findings illustrate that accuracy gradually increases with increasing of spectral scale values, but it starts to decrease for spectral values higher than 18. The over-segmentation is clearly observed in the higher scales (hs > 18), where object unit is presented in multiple polygons. In contrast, segmentation errors are more evident in smaller hs scales (10 - 15). We observed that these low J-index values were caused by under-segmentation, where small objects are merged in large ones (e.g., houses and roads or grassland and cropland areas were not properly separated) and this is highly penalized in this metric. In another way, segmentation results using hs between 16 and 17.5 are near-similar and consistent when we applied hr values around 11.0 or 16.5. These scale values are potential global parameters for our study. In addition, Figure 2.10 shows the segmentation results with three
combination of scales. The visual assessment corroborates with J-index results: some objects were not delineated in small scales (see buildings and trees), while higher scales present over-segmentation in all landscapes. The results show the capabilities to represent small-sized objects in combination 2, and this highlights the applicability of predefined M_r = 25. A high quality of the image segmentation is difficult with three spectral bands at 1-m spatial resolution. However, while some results might not be fully accurate, we emphasized that the proposed multi-OCNN method manages well the merging of multiple polygons in the final product. Based on these results and further investigation of segmented outputs, the global parameters were selected as [h_r, h_s, M_r] = [16.0, 16.5, 25] for image segmentation, and the segmented objects are roughly the same size or slightly smaller than real-world target to be classified in this large-area classification.

Figure 2-9. Segmentation scale effects with different spatial (hr) and spectral (hs) values in Mean-Shift algorithm. The Jaccard index values (Section 4.3) were calculated using 1200 reference and segmented polygons over eight counties. The Jaccard index ranges from 0 (no match) to 1 (perfect match). Smaller scales tend to smooth objects, while higher scales tend to over-segment objects in image. These experiments were conducted with minimum region value of 25 pixels.
Figure 2-10. Segmentation maps with three combination of input scales \([hr, hs, Mr]\) in the Mean-Shift algorithm. The parameters are spatial \((hr)\) and spectral \((hs)\) and minimum region \((Ms)\). (a) combination 1 = [10, 10, 25], (b) combination 2 = [15, 15, 25], (c) combination 3 = [20, 20, 25].

Performance of multi-OCNN framework

This section presents the classification performance of proposed multi-OCNN framework. Table 2.4 shows the confusion matrix and overall classification accuracy (OA) using 10,000 reference samples. In general, scene classification using multi-OCNN achieves satisfactory accuracy of 87.2% and \(\kappa = 85.8\%\). The producer’s accuracy (PA) varies among the land cover classes, from 79.0 to 94.8%. The higher PA values were observed for cropland (91.3%), grassland (91.7%), and river (94.8%), while lower values were found for structure (79%), fallow (79.8%), and barren (83.2%). Regarding the user’s accuracies (UA), the values range from 76.9 to 94.1% (Table 2.4). User’s accuracies were greater than 90% for five land cover classes: cropland (94.1), lake (93.7), river (91.9), shadow (90.1), and structure (90.8%). Note that user’s and producer’s
accuracies are quite similar; they mostly differ between 1 and 6%. In particular, both PA and UA values were higher than 90% for cropland, which is a positive measure for crop-dominated landscape as Iowa. In contrast, some classes present persistent confusion. For instance, road class is often classified as structure and barren (and vice-versa). Also, lower UA was observed for grassland class (76.9%), where these grassland pixels are incorrectly classified as cropland, fallow and forest areas. Not surprisingly, misclassification occurs for classes with most similar spatial-spectral features. Likewise, labeled pixels as “lake” are typically mixed with river and shadow classes. The definition of river and lake classes makes sense for CNN classifier (spatial features are different), but these pixels are quite similar in spectral domain. We observed that shadow pixels might be confused in dense forest areas, while tree shadow is often confused with dark lake pixels. The shadow pixels are typically observed close to forest areas (tree shadow). The spectral mixing leads to some confusion between these classes.

Table 2-4. Confusion matrix for proposed multi-OCNN framework.

<table>
<thead>
<tr>
<th>No of pixels</th>
<th>Classified data</th>
<th>Reference data</th>
<th>User acc. (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Barren</td>
<td>Cropl.</td>
<td>Fallow</td>
</tr>
<tr>
<td>Barren (0)</td>
<td>832</td>
<td>3</td>
<td>77</td>
</tr>
<tr>
<td>Cropland (1)</td>
<td>0</td>
<td>913</td>
<td>20</td>
</tr>
<tr>
<td>Fallow (2)</td>
<td>57</td>
<td>9</td>
<td>798</td>
</tr>
<tr>
<td>Forest (3)</td>
<td>5</td>
<td>16</td>
<td>4</td>
</tr>
<tr>
<td>Grassland (4)</td>
<td>5</td>
<td>57</td>
<td>95</td>
</tr>
<tr>
<td>Lake (5)</td>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>River (6)</td>
<td>8</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Road (7)</td>
<td>64</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>Shadow (8)</td>
<td>9</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Structure (9)</td>
<td>17</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Total</td>
<td>1000</td>
<td>1000</td>
<td>1000</td>
</tr>
<tr>
<td>Prod. acc. (%)</td>
<td>83.2</td>
<td>91.3</td>
<td>79.8</td>
</tr>
</tbody>
</table>

Overall accuracy (OA): 87.2 %; Overall kappa (κ): 85.8 %

Figure 2.11 presents the land cover mapping derived from multi-OCNN framework. These examples show the benefits of multi-OCNN to preserve the high-level of spatial details. For example, panels (11-b) and (11-e) present a dense residential area at fine mapping. Small houses,
individual trees, and grassland areas are clearly observed in these areas. Natural targets such as rivers and lake were also well-represented with the correct label (panels 11-c and 11-h). Also, conservation practices in cropland areas were correctly captured in the map, such as grassed waterways (panels 11-d, 11-f, and 11-g). Despite the spectral similarities between cropland and grassland classes, the pasture pixels were well distinguished (panels 11-f and 11-g). In panel (11-e), we observe a reasonable classification for small urban areas, but the road/pavement class was misclassified as a barren surface. This problem is observed in other scene areas. In contrast, we observed the capability for cropland delineation, providing a good distinction between grassland and cropland areas. This CNN-based cropland boundaries can support the refinement of cropland mapping, such as Crop Data Layer product. Note that NAIP imagery are collected on different dates across the state and the land cover results are accurate for a specific moment but they are limited for comparison to the broader area. This temporal effect of image acquisition is easily observed in the land cover results; see the fallow banding with adjacent cropland areas. Figure 2.12 presents other mapping results for our framework. In general, these examples show a consistent map in these panels, but some confusion is also observed for structures and roads (panel 2.12-b). Note that object-based classification reduces the salt-and-pepper effects in the final map. Moreover, the riparian forest is well-classified around the river. Bright areas (roads, structures, and sand) are often misclassified due to intra-class spectral similarity. For instance, panel (a) in Fig. 2.12 shows sand near to river is misclassified as structure. This error appears in some areas. In contrast, the shadow in forest area is clearly observed in panel 2.12-d. The lake area is well-classified in panel 2.12-e, but we also observe the river pixels in wrong area. Also, detailed mapping is observed in small city (panel 2.12-f).
Figure 2-11. Classification results of proposed multi-OCNN method. The panels are subset images with land cover result. Note that large banding of the “fallow” class is caused by different flight dates during NAIP 2015 acquisition.
Comparison with other CNN-based methods

Table 2.5 presents the comparison of proposed multi-OCNN with other methods. This summary shows per-class F-score accuracy, overall accuracy, and kappa coefficient for each method. As mentioned, pixel-wise CNN is the traditional method for land cover classification using patch-based input, while the fixed-OCNN is the object-based classification using a single model (instead of multiscale). In general, overall accuracy ranges from 68.7 to 87.2%. The proposed multi-OCNN outperforms all experimental methods (OA = 87.2%), improving 5.2, 18.5, and 5.6% regarding to fixed-OCNN$_{16}$, OCNN$_{c}$, and pixel-wise CNN$_{16}$. Also, OCNN$_{all}$ results did not outperform the multi-OCNN, suggesting that input size is relevant for object-based classification. In this comparison, lower input size produces slightly higher classification accuracy. In contrast, the overall accuracies of fixed-OCNN and pixel-wise CNN are almost similar; there is
no significant improvement by implementing object-based classification without multiscale CNNs. In contrast, OCNN\textsubscript{c} method achieves the lowest classification accuracy (68.7\%). This result suggests the importance of skeleton-based analysis and multiple convolutional locations for better prediction.

While some methods have higher F-score per class (bold text), multi-OCNN performs the best for five classes (barren, cropland, grassland, shadow, and structure). In contrast, natural targets of large areal extent are typically well-classified by large input patches (OCNN\textsubscript{256}). We observe that both fixed-OCNN and pixel-wise CNN perform more poorly in the structure class for urban areas. Also, note that CNN\textsubscript{256} presents a higher F-score for road class (91.1\%), but we should evaluate this result with caution. It was observed that these elongated targets are over-expanded, causing shape distortion (Fig. 2.13). This finding suggests the importance of correct observation scale for classification in urban areas, and consequently, the application of multiple CNNs (and pixel-based MLP). Table 2.6 shows the performance of semantic segmentation methods in comparison with multi-OCNN. Among the methods, U-Net architecture presented a higher accuracy (78.9\%), but the F-score values per class were quite similar between these methods. For instance, cropland areas presented a higher F-score values (~89 - 92), while barren class had the lowest values (54 – 62\%). In addition, the overall accuracies of these semantic segmentation methods were lower than that of multi-OCNN, reinforcing the capability of our framework in such context.

The visual assessment of these methods is shown in Fig. 2.13. In these experimental areas, we illustrate the benefits of multi-OCNN method compared to pixel-wise CNN and fixed-OCNN, especially in urban area. The traditional CNN smooths small targets and introduces uncertainties in the target edges. For instance, pixel-wise CNN\textsubscript{64} has salt-and-pepper effects and some mistakes
are observed by misclassifying the roads as structures. Small targets, such as residences, were difficult to achieve good results with pixel-wise CNN\textsubscript{64} (64 x 64 pixels). However, a similar problem is also observed for fixed-OCNN\textsubscript{64}. The object-based classification should preserve the geometry fidelity, but fixed-OCNN\textsubscript{64} expands the small objects and smooths linear features such as grassed waterways. The results demonstrate that urban areas are not easy to classify with input patch of 64 x 64 pixels. Also, Fallow pixels are incorrectly labeled as barren by both fixed-OCNN and pixel-wise CNN, but these classes are commonly mixed ones. At the end, these results suggest that multiscale CNNs improve the quality of the map, and object-based classification is only relevant when multiple CNNs are implemented. Additionally, Figure 2.14 shows U-Net, FPN and multi-OCNN results in two contexts. While the performances of U-Net and multi-OCNN are quite similar in the urban area, FPN results show smooth edge of residences and misclassified of road areas. Also, both methods were partially limited for large targets, such as cropland and fallow areas. For example, the misclassification of cropland and pasture areas was observed in U-Net results (see squared cropland areas in grassland). The output map (128 x 128) of U-Net and FPN does not consider the spatial relation between neighbor patches, which produces a spatial discontinuity for large objects. The SegNet results present rounded corners and detection problems for small targets, such as shadow and houses (not shown). Also, it is important to mention that semantic segmentation methods were trained with 5500 labeled samples, and a large reference dataset might improve the model generalization.
Table 2-5. Comparison of proposed multi-OCNN with different methods using F-score metric (Section 4.5). The F-score is the harmonic mean of producer’s and user’s accuracies ($F$-score = $2 \times (PA \times UA) / (PA + UA)$). The best result for each class is marked in bold.

<table>
<thead>
<tr>
<th>Land cover classes</th>
<th>pixel-wise classification</th>
<th>object classification</th>
<th>object + multiscale</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CNN$_{16}$</td>
<td>CNN$_{64}$</td>
<td>CNN$_{256}$</td>
</tr>
<tr>
<td>Barren</td>
<td>73.6</td>
<td>71.5</td>
<td>60.5</td>
</tr>
<tr>
<td>Cropland</td>
<td>89.3</td>
<td>85.8</td>
<td>84.7</td>
</tr>
<tr>
<td>Fallow</td>
<td>81.0</td>
<td>79.6</td>
<td><strong>84.7</strong></td>
</tr>
<tr>
<td>Forest</td>
<td><strong>87.1</strong></td>
<td>82.7</td>
<td>79.8</td>
</tr>
<tr>
<td>Grassland</td>
<td>68.5</td>
<td>61.3</td>
<td>59.9</td>
</tr>
<tr>
<td>Lake</td>
<td>84.5</td>
<td>93.0</td>
<td>95.6</td>
</tr>
<tr>
<td>River</td>
<td>83.9</td>
<td>91.0</td>
<td>95.1</td>
</tr>
<tr>
<td>Road</td>
<td>86.8</td>
<td>87.7</td>
<td><strong>91.1</strong></td>
</tr>
<tr>
<td>Shadow</td>
<td>72.5</td>
<td>70.9</td>
<td>68.3</td>
</tr>
<tr>
<td>Structure</td>
<td>83.5</td>
<td>81.1</td>
<td>77.7</td>
</tr>
<tr>
<td>Kappa ($\kappa$)</td>
<td>79.5</td>
<td>78.8</td>
<td>78.7</td>
</tr>
<tr>
<td>OA:</td>
<td>81.6</td>
<td>80.9</td>
<td>80.8</td>
</tr>
</tbody>
</table>

Table 2-6. Comparison of multi-OCNN with semantic segmentation methods using F-score values.

<table>
<thead>
<tr>
<th>Land cover classes</th>
<th>Semantic Segmentation</th>
<th>Multi-OCNN</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>U-Net</td>
<td>SegNet</td>
</tr>
<tr>
<td>Barren</td>
<td>62.0</td>
<td>54.7</td>
</tr>
<tr>
<td>Cropland</td>
<td>89.2</td>
<td>92.1</td>
</tr>
<tr>
<td>Fallow</td>
<td>64.7</td>
<td>78.6</td>
</tr>
<tr>
<td>Forest</td>
<td><strong>88.5</strong></td>
<td>79.8</td>
</tr>
<tr>
<td>Grassland</td>
<td>74.5</td>
<td>72.7</td>
</tr>
<tr>
<td>Lake</td>
<td>82.0</td>
<td>82.5</td>
</tr>
<tr>
<td>River</td>
<td>73.9</td>
<td>77.9</td>
</tr>
<tr>
<td>Road</td>
<td>78.6</td>
<td>77.3</td>
</tr>
<tr>
<td>Shadow</td>
<td><strong>89.9</strong></td>
<td>67.4</td>
</tr>
<tr>
<td>Structure</td>
<td>81.2</td>
<td>71.0</td>
</tr>
<tr>
<td>Kappa ($\kappa$)</td>
<td>76.5</td>
<td>73.2</td>
</tr>
<tr>
<td>OA:</td>
<td>78.9</td>
<td>75.9</td>
</tr>
</tbody>
</table>
Figure 2-13. Comparison of land cover classification with different methods. (a) true-color image, (b) pixel-wise CNN64, (c) fixed-OCNN64, and (d) proposed multi-OCNN.
Figure 2-14. Comparison of multi-OCNN with semantic segmentation methods: (a) true-color image, (b) U-Net, (c) Feature Pyramid Network, and (d) proposed multi-OCNN results.

This study claims the applicability of multi-OCNN framework for large-area classification. The results of computational time are summarized in Table 2.7. These experimental results were developed for 100,000 m² area. Although the computational time depends on number of objects and convolutional positions, the proposed multi-OCNN presents reasonable time (15.9 secs) for 100,000 m². There is clear evidence of time reduction for object-based compared to pixel-wise classification, varying from a few seconds to hours. For instance, the total time of multi-OCNN framework is 8.1 and 111.5 times faster than traditional pixel-wise CNN₁₆ or CNN₂₅₆, respectively. In comparison, OCNNₑ and FCN-based methods present the faster classifications (total time = 1.0
– 3.0 secs) due to a single convolutional location per object or dense output map. However, these methods also show the lowest overall accuracy/kappa among the approaches. We observed that lower input patch has faster processing time among pixel-wise CNNs. For the same model, object-based classification (e.g., OCNN<sub>16</sub>) is 10 times faster than pixel-wise classification (e.g., CNN<sub>16</sub>). With this time of performance and a single computer, the statewide classification (e.g.: Iowa) could take 268 days for multi-OCNN compared to ~48 years for CNN<sub>64</sub>. Thus, both results would be limited for operational mapping. However, note that multi-OCNN classification was implemented in HPC clusters with several computer nodes. As a result, classification result is achieved in less than 12 days (up to 25 nodes). Therefore, this suggests that our framework is faster than traditional CNN, but it still requires significant computational resources in large-scale applications.

Table 2-7. Computational time for different methods. The experimental results were performed for a scene area of 100,000 m<sup>2</sup>.

<table>
<thead>
<tr>
<th>Method</th>
<th>Computation time (secs)</th>
<th>Speed (pixels/sec)</th>
<th>Total time (secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Segmentation*</td>
<td>Object analysis*</td>
<td>Scene classification**</td>
</tr>
<tr>
<td>Pixel-wise CNN&lt;sub&gt;16&lt;/sub&gt;</td>
<td>-</td>
<td>-</td>
<td>128.6</td>
</tr>
<tr>
<td>Pixel-wise CNN&lt;sub&gt;64&lt;/sub&gt;</td>
<td>-</td>
<td>-</td>
<td>1040.0</td>
</tr>
<tr>
<td>Pixel-wise CNN&lt;sub&gt;256&lt;/sub&gt;</td>
<td>-</td>
<td>-</td>
<td>1772.2</td>
</tr>
<tr>
<td>Fixed-OCNN&lt;sub&gt;16&lt;/sub&gt;</td>
<td>1.4</td>
<td>4.7</td>
<td>6.2</td>
</tr>
<tr>
<td>Fixed-OCNN&lt;sub&gt;64&lt;/sub&gt;</td>
<td>1.4</td>
<td>4.7</td>
<td>30.4</td>
</tr>
<tr>
<td>Fixed-OCNN&lt;sub&gt;256&lt;/sub&gt;</td>
<td>1.4</td>
<td>4.7</td>
<td>90.4</td>
</tr>
<tr>
<td>OCNN&lt;sub&gt;c&lt;/sub&gt;</td>
<td>1.4</td>
<td>1.1</td>
<td>0.5</td>
</tr>
<tr>
<td>OCNN&lt;sub&gt;all&lt;/sub&gt;</td>
<td>1.4</td>
<td>4.7</td>
<td>218.4</td>
</tr>
<tr>
<td>OCNN&lt;sub&gt;dense&lt;/sub&gt;</td>
<td>1.4</td>
<td>-</td>
<td>269</td>
</tr>
<tr>
<td>U-Net</td>
<td>-</td>
<td>-</td>
<td>1.2</td>
</tr>
<tr>
<td>SegNet</td>
<td>-</td>
<td>-</td>
<td>1.9</td>
</tr>
<tr>
<td>FPN</td>
<td>-</td>
<td>-</td>
<td>1.93</td>
</tr>
<tr>
<td>Multi-OCNN</td>
<td>1.4</td>
<td>4.7</td>
<td>9.8</td>
</tr>
</tbody>
</table>

*Computer Resource: Intel Xeon(R) E3-1270 (3.80 GHz) processor

**Computer Resource: two 8-Core Intel Xeon (R) E5 2650 (2.0 GHz) processor
Discussion

This study describes a new multiscale object-based CNN framework for large-area land cover classification at 1-m resolution. The proposed framework integrates (i) image segmentation, (ii) object analysis, and (iii) multiple CNNs. While previous studies have implemented OCNN classification, this study is the first large-area classification exploring multiscale CNNs and skeleton algorithm. This multi-OCNN was successfully applied to NAIP imagery across Iowa, United States. The overall result of multi-OCNN is shown in Table 2.4. In general, our findings show that multi-OCNN provides a consistent land cover product with a high-level of spatial detail (Figs. 2.11 and 2.12), achieving a satisfactory accuracy (OA ~ 87.2% and kappa ~ 85.8%). The assessment in Fig. 2.13 showed that multi-OCNN helped to improve the classification results compared to other algorithms, such as fixed-OCNN and pixel-wise CNN. Similar benefits of OCNN were also presented in other studies (Langkvist et al., 2016; Zhao et al., 2017; Lv et al., 2018; Zhang et al., 2018; Jin et al., 2019; Liu et al., 2019). For instance, Zhao et al. (2017) presented an improvement of 0.87 to 5.46% for OCNN compared to pixel-wise CNN in different classification datasets (Worldview-2, Pavia center, Vaihingen). Similarly, Liu et al. (2019) achieved the overall accuracy of 95.33% using object-based post-classification refinement of CNN maps in urban context. While these studies highlight the advantage of OCNN, we observed that these benefits are highly dependent on multiple CNNs for correct classification (Fig. 2.13).

Typically, standard CNN extracts deep spatial-related features for fixed input patch size. However, each object has its distinctive geometric characteristics, and spatial-related features are scale-dependent (Zhao and Du, 2016). Therefore, the classification using a single CNN (with fixed input size) will not solve the problem of observation scale across heterogeneous landscape (ranging from small tree to large cropland). The results in Table 2.5 confirm the relevance of multiple CNNs for object classification, where fixed-OCNN achieved lower accuracy than proposed multi-OCNN.
In this context, Zhang et al. (2018) combined two CNN models with large (128 x 128) and small (48 x 48) window size for object-based classification. However, they recognized the limitation of two CNN models to represent complex geometries, recommending the integration of multiple models. The importance of multiple models was demonstrated in other studies. Langkvist et al. (2016) developed four CNNs with different input sizes (15, 25, 35, 45) and found higher accuracy for multiple CNN architecture (94.49%) compared to single CNN (90.02%). Sun et al. (2019) proposed a fusion of deep multiscale features (three CNNs) for building extraction. Our findings agree with these previous studies, and the fine-detail mapping in Fig. 2.12 corroborates to prove the value of multi-OCNN using six CNN models. Note that MLP network is also used to support the classification of small objects. However, pixel-level MLP network is only used when the convolutional location is close to object edge (S_d < 3 m), and some developers might prefer to simplify our approach without MLP.

While the multiscale CNNs represent a solution for the problem of observational scale, the application of multi-OCNN is highly dependent on object analysis with decision-rules. Admittedly, this processing stage involves further efforts to create a robust and fast algorithm for heterogeneous landscape. In this analysis, skeleton algorithm was successfully used to generate the morphological representation of segmented objects (Section 4.4). This method is essential for definition of convolutional locations, and subsequently, the determination of input patch size. The results in Table 2.5 also confirm the importance of multiple convolutional locations, where poor result of OCNNc illustrates the limitation of single prediction for final object label. Previously, Huang et al. (2018) proposed a skeleton-based decomposition with street blocks for land use mapping. However, the application was limited to regular urban blocks instead of any functional object (e.g. road, lake, trees). In contrast, Zhang et al. (2018) used object analysis in the OCNN
approach for urban land use classification, but this method has some limitations for heterogeneous landscape. For geometrically complex objects, the object analysis does not have a specific criterion to define the appropriate window size or implementation of multiple CNN models. Therefore, our object analysis contributes to OCNN application in broader context, giving the flexibility to work for any type of land target.

Despite this recent progress, the generation of large-area land cover product using CNNs is still a challenge for high-resolution data. Some studies have used post-classification refinement of CNN results (Langkvist et al., 2016; Zhao et al., 2017; Liu et al., 2019) or FCN application for semantic segmentation (Audebert et al. 2018; Sun and Wang, 2018). The results shown in Table 2.6 demonstrate that multi-OCNN outperformed the semantic segmentation methods. Notably, semantic segmentation methods reduce the redundant computations (non-overlapping patch processing) and processing time (Table 2.7), but they presented problems for large objects, such as croplands (Fig. 2.14). The data handling of huge number of scenes requires computational resources and integrated algorithm. Thus, object analysis presents an advantage to reduce the number of CNN predictions and speed up the overall processing. The results in Table 2.7 demonstrate that multi-OCNN is 8.1 and 111.5 times faster than traditional pixel-wise CNN16 or CNN256, respectively. These findings are similar to other studies. For instance, Zhang et al. (2018) shows that pixel-wise CNN is ~100 times longer than OCNN with limited computational resources. At the end, this time improvement is only possible because object-based CNN gives the flexibility to develop a different scheme for CNN application instead of pixel-by-pixel approach.

The segmentation step is a relevant part of object-based classification, and the effects of scale parameters are shown in Figure 2.9. For mean-shift algorithm, the application of smaller spatial/spectral scales creates smoother objects, and higher scales are useful when the targets are
small and spectrally similar such as streets and building roofs (Figure 2.10). The optimal selection of segmentation parameters is a well-known challenge in heterogeneous landscape at high spatial resolution (Drăguț et al., 2010; Zhang et al., 2014), and the automation of scale parameter selection is under investigation (Anders et al., 2011; Drăguț et al. 2014; Yang et al., 2015). Ideally, the global scale parameters need to preserve the geometric representation of small objects but also avoid the over-segmentation of large ones. The applied segmentation optimization with multiple experiments has proven to be an effective way for this study. The global scale parameters produced a slight over-segmentation results in some cases but proposed multi-OCNN manages well the fragmented objects to produce meaningful unit in the final product. While our efforts were concentrated on a practical strategy for statewide application, some users/developers might adopt locally defined scales in their studies, especially when the segmentation is applied to single scene or small area.

Exploring the development of multi-OCNN, many lessons were learned to turn CNN into a practical tool. The challenges for multi-OCNN application are identified as computational resources, programming skills, development of training samples and multiple networks, and data manipulation (~1 TB). The development of multiple CNNs involves an intensive training in GPUs using different configurations. In addition, multi-OCNN for large-area classification is highly dependent on computational resources. For instance, the statewide classification required a massive object label (this study: ~1.01 billion objects), and each object will have multiple convolutional locations for CNN application. For this research HPC cluster allowed access to multiple computer nodes, which make the overall classification more reasonable in terms of time. Regarding the input bands, this study uses three false-color bands but other developers might include additional information such as normalized difference vegetation index. The optimization
of object analysis could allow the same process in a single computer, including the selection of faster segmentation algorithm and fine-tuning of parameters (e.g., interval).

**Conclusion**

Convolutional neural network offers the ability to explore spatial-related deep features for land cover classification. However, CNN classification introduces certain challenges for operational mapping, especially in a broader context. In this research, we presented a new multiscale object-based CNN for large-area land cover classification. This framework includes a series of stages, including image segmentation, object analysis with skeleton algorithm and multiple CNNs. We also developed a new benchmark dataset (“IowaNet”) with 1 million images (10 classes) for training of multi-resolution CNNs. In general, we demonstrated that multi-OCNN is a feasible method for operational land cover mapping at 1-m resolution, achieving overall accuracy of 87.2%. We found that multi-OCNN presents faster and accurate classification compared to other OCNN frameworks. The results have shown that traditional pixel-wise CNN64 and FCN-based mapping produce blurred boundaries, while multi-OCNN preserves the target edges in different contexts. Moreover, the results indicated relevance of multiscale CNNs (six models) in object-based classification: multi-OCNN shows higher accuracy than fixed-OCNN method. The results suggest the benefits of object analysis for appropriate selection of CNN model, preserving the geometry fidelity in this object-based classification. Also, we found that OCNN improves the speed in overall processing; multi-OCNN is 8.1 and 111.5 times faster than traditional pixel-wise CNN16 or CNN256, respectively. The object analysis allows the reduction of CNN predictions (and hence, less computation) compared to pixel-wise CNN, but it still slower than FCN-based methods such U-Net and SegNet. While multi-OCNN offers a practical framework for large-area mapping, this application is highly dependent on computational
resources for data storage and handling (this study classifies ~1.01 billion of objects). With increased availability of EO data, further research is recommended to understand the potential of multi-OCNN in satellite datasets, such as GeoEye-1 and WorldView-4 imagery.

**Acknowledge**

We gratefully acknowledge the USDA's Farm Service Agency (FSA) for providing the NAIP 2015 dataset. The multi-resolution IowaNet dataset is available at https://doi.org/10.5281/zenodo.3385318. We thank the HPC-ISU and ResearchIT team for support in the Condo and Pronto clusters. The processed data in this study are available from the corresponding author upon reasonable request.

**Reference**


CHAPTER 3. DEEP NEURAL NETWORK FOR COMPLEX WETLAND MAPPING USING HIGH-RESOLUTION WORLDVIEW-3 AND AIRBORNE LIDAR DATA

Vitor S. Martins a, Amy L. Kaleita a, Brian K. Gelder a, Gustavo W. Nagel b, Daniel A. Maciel c

a Agricultural and Biosystems Engineering, Iowa State University, Ames, Iowa, USA.
b Remote Sensing Division, National Institute for Space Research (INPE), São José dos Campos, SP, Brazil.
c Image Processing Division, National Institute for Space Research (INPE), São José dos Campos, SP, Brazil.

A manuscript submitted to International Journal of Applied Earth Observation and Geoinformation

Abstract

Wetland inventory maps are essential information for the conservation and management of natural wetland areas. The classification framework is crucial for successful mapping of complex wetlands, including the model selection, input variables and training procedures. In this context, deep neural network (DNN) is a powerful technique for remote sensing image classification, but this model application for wetland mapping has not been discussed in the previous literature, especially using commercial WorldView-3 data. This study developed a new framework for wetland mapping using DNN algorithm and WorldView-3 image in the Millrace Flats Wildlife Management Area, Iowa, USA. This study area has several wetlands with a variety of shapes and sizes, and the minimum mapping unit was defined as 20 m2 (0.002 ha). A set of potential variables was derived from WorldView-3 and supplementary LiDAR data, and a feature selection procedure using principal components analysis (PCA) was performed to identify the most important variables for wetland classification. Furthermore, traditional machine learning methods (support vector machine, random forest and k-nearest neighbor) were also implemented for the comparison of results. In general, the results show that DNN achieved satisfactory results in the study area
(overall accuracy = 93.33%), and we observed a high spatial overlap between reference and classified wetland polygons (Jaccard index ~ 0.8). Our results confirm that PCA-based feature selection was effective in the optimization of DNN performance, and vegetation and textural indices were the most informative variables. In addition, the comparison of results indicated that DNN classification achieved relatively similar accuracies to other methods. The total classification errors vary from 0.104 to 0.111 among the methods, and the overlapped areas between reference and classified polygons range between 87.93 and 93.33%. Finally, the findings of this study have three main implications. First, the integration of DNN model and WorldView-3 image is useful for wetland mapping at 1.2-m, but DNN results did not outperform other methods in this study area. Second, the feature selection was important for model performance, and the combination of most relevant input parameters contributes to the success of all tested models. Third, the spatial resolution of WorldView-3 is appropriate to preserve the shape and extent of small wetlands, while the application of medium resolution imagery (30-m) has a negative impact on detection of these areas. Since commercial satellite data are becoming more affordable for remote sensing users, this study provides a framework that can be utilized to integrate very high-resolution imagery and deep learning in the classification of complex wetland areas.

**Keywords:** Deep learning, small wetlands, machine learning, optical and LiDAR data, PCA.

### Introduction

Wetlands are ecological units that sustain biodiversity and environmental services for aquatic plants and terrestrial wildlife (Halls, 1997; Dahl, 2000; Maltby et al., 2011). Wetlands are typically areas of permanently or seasonally saturated soil and/or with shallow standing water depth and produce a transition zone between purely aquatic and terrestrial habitats. These natural areas can provide ecosystem services, including sediment control, carbon sequestration and
nutrient mitigation (Mitsch et al., 2013). Despite the ecological significance of wetlands, these habitats are commonly threatened by land change for multiple uses, especially for agricultural practices (Zedler & Kercher, 2005; Bartzen et al., 2010). In 1971, the Ramsar Convention proposed a framework for conservation and rational use of wetlands and their resources (DeGroot et al., 2006). Recently, United Nations’ 2030 Agenda for Sustainable Development emphasizes the importance of protection and restoration of inland freshwater ecosystems, including wetlands (Long, 2019). In this context, accurate maps are essential for management and preservation efforts, and policy-makers and natural resource managers have been interested in the comprehensive assessment of wetland location and extent (Rebelo et al., 2009). However, wetland areas are difficult to access and require intensive fieldwork to collect in-situ data, which is only practical in small-scale studies. Given these limitations for field surveys, the application of remote sensing allows a cost-effective wetland inventory (Ozesmi and Bauer, 2002; Guo et al., 2017).

Earth Observation (EO) data have proven to be useful for mapping wetland systems (Mackay et al., 2009; Adam et al., 2010; Wang & Yésou, 2018). Optical data with medium- and coarse-resolution (10 m – 1000 m) have been widely used for these applications, such as Landsat sensors (Nielsen et al., 2008; MacAlister et al., 2009; Kayastha et al., 2012), EOS MODIS (Landmann et al., 2010; Tana et al., 2013), SPOT (Davranche et al., 2010), Sentinel-2 MSI (Hird et al., 2015; Whyte et al., 2018), and NOAA AVHRR (Zoffoli et al., 2008). Nielsen et al. (2008) proposed a wetland change probability map for the U.S. mid-Atlantic region with 30-meter Landsat. Also, Jin et al. (2017) developed a framework for long-term monitoring of wetland inundation (1985 - 2011) on the East Coast of the United States using Landsat time-series imagery. These examples illustrate the capabilities of medium-resolution sensors (e.g., Landsat) at regional mapping. Additionally, several countries have developed mapping systems for National Wetland
Inventories (NWIs) using remote sensing datasets. For example, the Canadian Wetland Inventory has consolidated an initiative to detect wetlands combining Landsat TM and Radarsat data (Fournier et al., 2007). In 1979, the U.S. Fish and Wildlife Service proposed a NWI for the United States (Cowardin et al. 1979), and since then, this program has produced digital wetland data and historical map information (https://www.fws.gov/wetlands/Data/Mapper.html) (Tiner, R. W., 2009). Kloiber et al. (2015) emphasized that the NWIs can be the basis for guiding wetland management, but they need significant ongoing development. Manual delineation using aerial photo-interpretation is one of the main technical limitations for continuous updating of the inventory records, and the minimum (or target) mapping unit is often not suitable for the detection of tiny and narrow wetlands (< 0.2 ha). While the benefits of these small-area wetlands are typically overlooked (Blackwell and Pilgrim, 2011), the investigation of their occurrence and extent is difficult since they are largely neglected in the national inventories. Therefore, there is still a significant gap in fine-scale mapping of wetlands to support the continuous update of inventories, especially for small-area wetlands (Ozesmi and Bauer, 2002; Leonard et al., 2012; Mwita et al., 2013; Gallant, 2015).

Commercial satellite missions can provide very high-resolution imagery (0.5 – 5 m) that may improve EO capabilities in many scientific topics (Belward and Skøien, 2015). Notably, DigitalGlobe's constellation represents a successful private initiative of Earth imaging, offering a suite of satellite products such as WorldView, GeoEye-1, Ikonos, and QuickBird. In particular, WorldView satellites (2009 - present) have been successfully used for studying wetland habitats. For instance, McCarthy et al. (2015) assessed wetland extent in Tampa Bay estuary, Florida (USA) using WorldView-2 images. Araya-López et al. (2018) evaluated the applicability of Worldview-2 imagery to identify and map Andean wetlands in central Chile, achieving an overall accuracy of
~87%. Although commercial imagery involves acquisition costs, these costs have significantly decreased in recent years, and long-term contracts could satisfy the requirement for a larger set of commercial imagery for wetland inventory. As part of next-generation of high-resolution satellites, WorldView-3 (WV-3) was launched in mid-August 2014 and enhanced land monitoring with multiple spectral bands. This multi-payload satellite acquires 0.31 m panchromatic, 1.24 m visible and near-infrared (VNIR) (8 bands), and 3.7 m short-wave infrared bands (8 bands). While multispectral WV-3 dataset provides high-quality remote sensing data for fine mapping (Mahdavi et al., 2018), few studies have evaluated the application of WV-3 for wetland applications (Vanderhoof et al., 2017). In addition to the dataset, the classification framework (input variables, model selection, training procedure) is a crucial part of successful mapping because wetland areas impose certain challenges for accurate remote sensing classification, such as mixed spectral information, boundary uncertainties, and complex spatial structure (Corcoran et al., 2011; Rapinel et al., 2015).

Recent studies have implemented machine learning methods for wetland classification (Mui et al., 2015; Zhu et al., 2017; Berhane et al., 2018), and supervised classifiers are typically used for these habitats, such as decision tree (Wright and Gallant, 2007; Bwangoy et al., 2010), support vector machine (Huang et al., 2009), and random forest (Fu et al., 2017; Mahdianpari et al., 2017; DeVries et al., 2017). In the same context, deep learning methods have gained significant relevance with the progress of computer technology (Zhang et al., 2016; Rezaee et al., 2018). The feed-forward neural networks (or multi-layer perceptron) provide the ability to learn complex non-linear functions from a multivariable dataset, and produce more flexible models for heterogeneous landscapes. So far, the successful applications of deep neural networks were presented in many research fields, but the literature is very limited for wetland mapping (Ghedira et al., 2000; Li et
For this reason, a classification framework using deep neural network and multispectral WorldView-3 is still an opportunity for wetland applications, especially complex wetland areas.

This study explores the integration of WorldView-3 image and feed-forward deep neural network algorithm for wetland mapping at a 1.2-m resolution. The study was conducted on wetland-dominated landscape in Iowa, USA. This paper contributes to the understanding of (1) the development of classification framework for complex wetlands using WorldView-3 image; (2) the application of feature selection to identify relevant input variables derived from optical and LiDAR data; (3) the performance of DNN algorithm and inter-comparison with other machine learning techniques; and (4) the importance of spatial resolution in fine-scale wetland mapping. Our framework integrates a set of multispectral and LiDAR variables to train a deep neural network architecture. These variables were identified by the feature selection approach using principal component analysis (PCA). The reasons to use deep neural network are that: they are good function approximators; complex architectures can be designed for better control of learning and generalization; and there is unexplored potential of this type of deep learning algorithm to classify wetland areas. Lastly, this study is not only concerned about the final accuracy, but we are also interested in the classification framework (model design, selection of input variables, and training procedures) for improving our mapping capabilities in these habitats. As far as we know, this study is the first assessment of WorldView-3 and deep neural network for wetland mapping, especially for small wetland areas.

Data

Study area

This study was conducted on the Millrace Flats Wildlife Management Area (MFWMA) (41°12′16″ N, 91°12′36″ W), located north of Wapello, Iowa, USA (Figure 3.1). The area of interest covers over 5.64 km² along the Iowa River, and the landscape presents prairie, shrubs,
wetland, and mature forest. There are several wetland areas (ephemeral and depressional wetlands, temporary shallow ponds) with clear water and emergent algae. The climate condition is characterized by distinct seasons. The rainy season occurs from April to August, while other months are relatively dry. The average annual precipitation is approximately 1019 mm, and monthly precipitation ranges from 45 to 150 mm. The terrain is characterized by a flat relief; elevation variation is up to 4 meters across the area. The selection of this site was based on four criteria: (i) high occurrence of wetlands; (ii) landscape features are common in the state of Iowa; (iii) variety of wetland sizes; (iv) site accessibility and data availability (WorldView-3 and LiDAR).

Figure 3-1. Location of Millrace Flats Wildlife Management Area (MFWMA) in Wapello, Iowa, USA.
WorldView-3 image processing

WorldView-3 is a commercial satellite that is part of DigitalGlobe’s constellation (https://www.digitalglobe.com/). This satellite was launched on August 13, 2014 (18:30:30 UTC), and operates in a sun-synchronous orbit (altitude: 617 km) with a swath width of 13.1 km. This orbital/sensor design allows a daily revisit time, with a global capacity of 680,000 km² per day. The WV-3 satellite provides the following spatial resolutions: 0.31 m for the panchromatic band (450 – 800 nm); 1.24 m visible and near-infrared (VNIR) bands (400 – 1040 nm); 3.7 m for the Short-Wave Infrared bands (1080 – 2375 nm); and 30 m for the CAVIS (Clouds, Aerosols, Vapors, Ice, and Snow) bands. Figure 3.2 shows Relative Spectral Responses (RSR) of multispectral WV-3 bands. DigitalGlobe allows customers to acquire images either from historical archive or tasking order. The spectral bands have different radiometric resolutions, such as 11 bits in VNIR and 14 bits in CAVIS. The WV-3 provides numerous opportunities for high spatial resolution studies, such as agricultural (Sidike et al., 2019), tree species classification (Hartling et al., 2019), geology (Sun et al., 2017), and inundation extent (Vanderhoof et al., 2017).

In this study, a WorldView-3 image was acquired on August 2, 2015 (5.4 × 4.5 km footprint, sun elevation of 63.02°, 16° off-nadir view angle). The image was delivered via eMap International, commercial partner of DigitalGlobe. The eight VNIR bands were delivered at 1.24-m spatial resolution using WGS84 UTM zone 15N projection. Note that SWIR bands were not used because it was not available in this data archive. The metadata describes image attributes, and a series of procedures were implemented in this study, such as radiometric conversion from digital number to top-of-atmosphere radiance (see documentation: http://www.digitalglobe.com/resources/technical-information), and TOA radiance to surface reflectance via atmospheric correction. The latter was performed with Second Simulation of the Satellite Signal in the Solar Spectrum-Vector version (6SV) model (Vermote et al., 1997) and
input variables from MODIS MCD19A2 (aerosol and water vapor) and MOD08D3 (ozone) products on the same day of WV-3 image. This procedure was described in a previous work (Martins et al., 2017; 2018). After the processing, all data were resampled from 1.24 to 1.2 m using nearest neighbor for the same geographic extent, and stacked in a single file. These processed bands were used for calculation of potential variables in Section 3.1.

Figure 3-2. Relative Spectral Response (RSR) of eight VNIR WorldView-3 bands.

LiDAR data processing

Several studies have shown the value of auxiliary topographic data in wetland classification (Hird et al., 2015; Kloiber et al., 2015; Wu et al., 2019). In this context, the Iowa Light Detection and Ranging (LiDAR) Mapping Project is the main source of topographic data derived from light-emitting scanning laser. The Airborne Laser System data were collected from 2007 – 2010, and the flights were planned as parallel flight lines across state (https://geodata.iowa.gov/dataset/iowa-lidar-project-2007-2010). This LiDAR dataset is a high-quality product assured by U.S. Geological Survey. The vertical accuracy of the Iowa LiDAR is 18.5 cm for bare earth and 37 cm
for vegetated areas, and the point density is ~ 0.5 pts/m². The dataset is publicly distributed by the GeoTREE data portal from University of Northern Iowa (http://www.geotree.uni.edu/lidar/). For our study, we processed LiDAR point cloud data for Millrace Flats Wildlife Management Area. The 3D LiDAR points were collected in late spring 2010 and delivered in LAS binary files with all first and ground returns. The bare-earth digital elevation model (DEM) and digital surface model (DSM) were produced using ArcMap 3D toolbox from the last (class 2) and first returns at 1.2-m, respectively. The last returns of the LiDAR pulses are likely to represent the surface level and first returns are typically the canopy tops. The canopy height model (CHM) was computed by subtracting the DEM from DSM. Although there is a time gap between multispectral and LiDAR data, the application of this dataset is reasonable: this quality-assured dataset is publicly available by USGS, allowing research reproducibility; topographic features are generally more stable in natural areas, where the human-driven activities are not intense; and this study is focused on wetland areas which make the uncertainties on canopy less relevant for our application. The DEM and CHM metrics are used in Section 3.1 to generate LiDAR-derivative variables, such as terrain hillshade.

**Training and testing samples**

The study area has seven land cover classes, including forest, grassland, shrubland, fallow, road, wetland-clear and wetland-algae. We also included an eighth class for land that was in shadow, regardless of its use. Although this application is focused on wetland mapping, these additional land cover categories are required for “upland” classification. Stratified training samples were then generated for each class (Table 3.1), and a total of 2260 points were labeled based on visual interpretation on WorldView-3 image and auxiliary data (ESRI service layers and 1-m NAIP 2017 aerial photo). The sample size was greater than 100 per class. These points were categorized
as fallow (~11.8%), forest (~13.4%), grassland (~14.4%), road (~7.7%), shadow (~12.1%), shrubland (~11.2%), wetland-clear (~16.6%), and wetland-algae (~12.8%) classes. These samples were randomly split into training (1470) and testing (790) samples, where the training set is used to train the models (Section 3.2.2 and 3.3) and testing samples were used for further evaluation of DNN results (Section 4.2.). In addition, a reference wetland map (polygons) was produced for geometric-based assessment (Section 3.4). The reference polygons were manually digitalized from WorldView-3 at 1:1,000 scale (fixed), and wetland area represents approximately 5.37% of the total study area (5.64 km²). The individual wetlands range from ~29 to 73,200 m². Forested wetlands were omitted in this study. The minimum mapping unit (MMU) in this study was defined as 20 m² (0.005 acres); note that U.S. NWI MMU is ~2023 m² (0.5 acres). Other consolidated studies have also used manual interpretation for reference mapping of wetlands (Dahl et al., 2009; Rokus et al., 2015; Kloiber et al., 2015). In addition to reference map, a field survey was conducted on August 2018 and some wetland locations were verified in the study area (Figure 3.3).

Table 3-1. Number of samples for training and testing in each land cover class over the study area.

<table>
<thead>
<tr>
<th>Land cover</th>
<th>Train</th>
<th>Test</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forest</td>
<td>195</td>
<td>108</td>
<td>Deciduous broadleaf forest</td>
</tr>
<tr>
<td>Grassland</td>
<td>213</td>
<td>112</td>
<td>Herbaceous vegetation (&gt; 50% cover)</td>
</tr>
<tr>
<td>Shrubland</td>
<td>180</td>
<td>73</td>
<td>Open and closed shrublands</td>
</tr>
<tr>
<td>Barren\Fallow</td>
<td>186</td>
<td>80</td>
<td>Exposed surface soil or harvested area</td>
</tr>
<tr>
<td>Road</td>
<td>123</td>
<td>51</td>
<td>Road with impervious surface</td>
</tr>
<tr>
<td>Shadow</td>
<td>184</td>
<td>89</td>
<td>Tree shadow</td>
</tr>
<tr>
<td>Wetland-clear</td>
<td>220</td>
<td>156</td>
<td>Wetland with freshwater</td>
</tr>
<tr>
<td>Wetland-algae</td>
<td>169</td>
<td>121</td>
<td>Wetland with algae dominance</td>
</tr>
<tr>
<td>Total</td>
<td>1470</td>
<td>790</td>
<td></td>
</tr>
</tbody>
</table>
Methods

This classification framework includes the development of a deep neural network, calculation of potential variables (multispectral and LiDAR), PCA-based feature selection and comparison with other machine learning methods (Figure 3.4).
Potential variables and feature selection

The optimal set of input variables is crucial for most efficient use of variables in a supervised classification (Berhane et al., 2018). In this study, several spectral and topographic indices were calculated as potential explanatory variables, and later, feature selection was performed to define the better set of input data. First, we computed spectral metrics from multispectral WV-3 bands: 10 normalized indices and 5 textural metrics. The normalized indices can enhance the classification performance by exploring the contrast of optical properties between land and water targets. Following Fu et al. (2017), Grey-Level Co-occurrence Matrix (GLCM) metrics were developed for textural information, such as contrast, correlation, dissimilarity, energy, and homogeneity (Haralick et al., 1973). These metrics represent the structural variation of reflectance (and its spatial dependence) for a given local window. The window size was defined by 5 x 5 pixels using NIR-1 band (B7). In addition, 5 LiDAR-derived metrics were proposed as auxiliary data for this classification. The topographic metrics support the understanding of the terrain context and shape of the relief in the study area. For instance, terrain ruggedness index supports the evaluation of local variability of elevation between adjacent cells (window: 5 x 5 pixels). All these potential variables were derived on a pixel-basis (1.2 x 1.2 m), and later, they were normalized from 0 to 1 based on the range values found in the study area. A complete list of these variables is given in Table 3.2. Note that other metrics were also tested (e.g.: OSAVI, band ratios, Ratio Vegetation Index, Water Ratio Index), but they were highly correlated (R > 0.95) with others.

Given the number of potential variables for our classification (8 spectral bands + LiDAR CHM/DEM + 20 metrics), a robust feature selection procedure is necessary to create a parsimonious set of input variables, minimize the information redundancy and input model complexity (Dash and Liu, 1997; Corcoran et al., 2013). In this way, PCA is a non-parametric
method that reduces data dimensionality and can be used to select the most explanatory variables from the original dataset (Jolliffe and Cadima, 2016; Silva et al., 2016). Following Li et al. (2008), PCA-based feature selection was performed on the potential variables (total: 30) to identify a relevant subset of these variables. The first eight principal components (PCs) contain 91.07% of the total variation in the original variables, and eigenvector values (or loadings) of these PCs were used for feature selection. The eigenvector values represent the contribution of each metric for the PC, and we selected the variables with loadings higher than |0.3|. A total of 13 variables were selected as input variables for wetland classification (Table 3.3): GI, MCARI, MSR, NDWI, GLCM-Correlation, GLCM-Energy, GLCM-Homogeneity, TPI, TRI, TWI, So, CHM, B8. These PCA-based selected variables were used to train the DNN algorithm (Section 3.2.2) and other machine learning algorithms (Section 3.3). Further evaluation was implemented using different groups of input variables to compare the DNN performance.
Table 3-2. Description of potential variables for wetland classification.

<table>
<thead>
<tr>
<th>Category</th>
<th>Index</th>
<th>Acronym</th>
<th>Formula</th>
<th>Reference</th>
</tr>
</thead>
</table>
| Optical  | Chlorophyll Green index | CGI | \[
\frac{\rho_{NIR1}}{\rho_{Green} + \rho_{Rededge}}
\] | Datt (1999) |
|          | Enhanced Vegetation Index | EVI | \[
\frac{\rho_{NIR1} - \rho_{Red}}{\rho_{NIR1} + 6 \times \rho_{Red} - 7.5 \times \rho_{Coastal} + 1}
\] | Huete et al. (2002) |
|          | Green Chlorophyll Index | GCI | \[
\frac{\rho_{NIR1}}{\rho_{Green}} - 1
\] | Gitelson et al. (2003) |
|          | Global Environmental Monitoring Index | GEMI | \[
GEMI = \eta(1 - 0.25 \times \eta) \times \frac{\rho_{Red} - 0.125}{1 - \rho_{Red}}
\] | Pinty and Verstraete (1992) |
|          | Greenness Index | GI | \[
\frac{\rho_{Green}}{\rho_{Red}}
\] | Le Maire et al. (2004) |
|          | Modified Chlorophyll absorption reflectance index | MCARI | \[
\left(\frac{\rho_{Rededge}}{R_{Red}}\right) \left(\frac{\rho_{Rededge}}{\rho_{Red}} - 0.2 \times \left(\rho_{Rededge} - \rho_{Green}\right)\right)
\] | Daughtry et al. (2000) |
|          | Modified Soil Adjusted Vegetation Index | MSAVI | \[
2\rho_{NIR1} + 1 - \sqrt{\left(2 \times \rho_{NIR1} + 1\right)^2 - 8 \times \left(\rho_{NIR1} - \rho_{Red}\right)}
\] | Qi et al. (1994) |
|          | Modified Simple Ratio | MSR | \[
\left(\frac{\rho_{NIR1} \rho_{Red} - 1}{\rho_{NIR1} \rho_{Red} + 1}\right)
\] | Chen (1996) |
|          | Normalized Difference Vegetation Index | NDVI | \[
\frac{\rho_{NIR1} - \rho_{Red}}{\rho_{NIR1} + \rho_{Red}}
\] | Tucker (1979) |
|          | Normalized Difference Water Index | NDWI | \[
\frac{\rho_{Green} - \rho_{NIR2}}{\rho_{Green} + \rho_{NIR2}}
\] | McFeeters (1996) |
|          | Gray-Level Co-occurrence Matrix: Contrast, Homogeneity, Dissimilarity, Energy, Correlation | GLCM | See equations in reference | Haralick et al. (1973) |
| LiDAR    | Topographic Position Index | TPI | \[
TPI = z - \bar{z}(r); z \text{ is the elevation of central pixel, } \bar{z} \text{ is the average within radius } r
\] | Weiss (2001) |
|          | Terrain Ruggedness Index | TRI | See equations in reference | Riley et al. (1999) |
|          | Topographic Wetness Index | TWI | \[
TWI = \ln \left(\frac{As}{\tan(B)}\right), \text{As is the specific catchment area, } \tan(B) \text{ is the local slope}
\] | Beven and Kirkby (1979) |
|          | Hillshade | HS | \[
255 \times \left(\frac{\cos(\text{zenith}) \times \cos(\text{slope}) + \cos(\text{azimuth} - \text{aspect})}{\sin(\text{zenith}) \times \sin(\text{slope}) \times \cos(\text{azimuth} - \text{aspect})}\right)
\] | - |
|          | Slope | So | - | - |
Table 3-3. Feature selection using principal component analysis. The selected variables (bold) present eigenvector values higher than \( |0.3| \). PC is the given principal component. For simplicity, eigenvector values of non-selected features were omitted.

<table>
<thead>
<tr>
<th>Variables</th>
<th>PC1</th>
<th>PC2</th>
<th>PC3</th>
<th>PC4</th>
<th>PC5</th>
<th>PC6</th>
<th>PC7</th>
<th>PC8</th>
</tr>
</thead>
<tbody>
<tr>
<td>CGI</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EVI</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GCI</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GEMI</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GI</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-0.477</td>
</tr>
<tr>
<td>MCARI</td>
<td>0.320</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MSAVI</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MSR</td>
<td>0.322</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NDVI</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NDWI</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.314</td>
</tr>
<tr>
<td>GLCM-Contrast</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GLCM-Correlation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-0.367</td>
</tr>
<tr>
<td>GLCM-Dissimilarity</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GLCM-Energy</td>
<td>0.309</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GLCM-Homogeneity</td>
<td>0.413</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TPI</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.685</td>
</tr>
<tr>
<td>TRI</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.383</td>
<td></td>
</tr>
<tr>
<td>TWI</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-0.325</td>
</tr>
<tr>
<td>HS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>So</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.474</td>
<td></td>
</tr>
<tr>
<td>CHM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-0.375</td>
</tr>
<tr>
<td>DEM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>B8</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.316</td>
</tr>
</tbody>
</table>

Deep learning: feed-forward neural network

Background

This study evaluates the application of feed-forward deep neural network algorithm (simple DNN also known as multi-layer perceptron) for wetland mapping at fine 1.2-m resolution. A feed-forward neural network consists of multiple layers that learn non-linear relationships between input variables and outputs (e.g., categories) (Schmidhuber, 2015). The network architecture has
input, hidden, and output layers with multiple neurons (or computing units). Essentially, the neurons are inter-connected to create the network, and each neuron computes a weighted sum of inputs (+ bias) and passes it through the activation function. The basic computation in neuron $j$ is defined as (Equation 1):

$$y_j^h = \sigma_j \left( \sum_{i=1}^{n} (W_{i,j}^h \cdot x_i^{h-1}) + b_j^h \right)$$

Where $y_j$ is the output of the $j$-th neuron using inputs $X = \{x_1, x_2, \ldots, x_n\}$ ($i \in [1,n]$) from the previous layer ($h - 1$), $W_{i,j}$ is the weight in the layer ($h$), $b_j$ is the bias term, $n$ is the number of input parameter (or number of neurons from previous layer). A typical activation function ($\sigma_j$) is ReLU (Rectified Linear Unit) and allows the network to learn non-linear functions for input data.

The number of hidden layers defines the network complexity, where models with multiple hidden layers ($> 3$) is typically recognized as “deep” networks. Each hidden layer has different number of neurons for the abstraction of input data and requires empirical design for parallel computations in the graphic processing unit (GPU). Once the architecture is defined, the next step is the supervised learning of neural network parameters (basically, weights and biases) using a backpropagation algorithm with gradient descent (Johnson and Zhang, 2013). In this context, the “learning” involves the minimization of error (known as cost function) between the desired and actual values of the output neurons. An example of cost function is the least mean square error (Equation 2).

$$E(W) = \frac{1}{2} \sum_{i=1}^{n} (y_j^h - \bar{y}_j)^2$$

Where $y_j^h$ is the derived value for output node $j$ in last layer, and $\bar{y}_j$ is the target output of that neuron. The supervised training comprises the feed-forward propagation of input variables in
the network to obtain an abstract representation, and then, the backward way uses the propagation of error at output units. In this way, this interactive process can minimize the error $E$ by gradually updating a set of weights and bias repeatedly. Each weight $W_{i,j}$ changes with the increment $\Delta w_{j,i}^h$ to reduce the $E$ until satisfactory results. The same idea is applied for bias term.

$$\Delta w_{j,i}^h(t) = -\eta \frac{\partial E(W)}{\partial w_{j,i}^h}$$

(3)

$$w_{j,i}^h(t + 1) = w_{j,i}^h(t) + \Delta w_{j,i}^h(t)$$

(4)

Where the $\eta$ is learning-rate parameter, and $t$ is the interaction step. Each iteration will update the weights and biases with gradient descent. In fact, backpropagation is a set of calculus, and more details are available in Haykin (2009). Once trained, the network is then used to predict labels from a new set of variables for each pixel in the image. The softmax function is used to create a $K$-dimensional vector with a categorical probability distribution (Equation 5) and the final label is defined by class-label $j$ with the highest probability (Equation 6).

$$S(X) = P(y_i \mid X; \theta) = \frac{e^{y_i}}{\sum_j e^{y_j}}$$

(5)

$$C = \text{argmax}(S(X))$$

(6)

Given 1-D input vector $X$, $S(X)$ is the probability distribution from $y_i$ in the output layer using parameter $\theta$ (weights and biases), and $C$ is the number of output classes. For image classification, the network calculates the probability for each class for a given pixel and the final label is assigned for the class with the highest probability.
Deep Neural Network (DNN) model

In this study, we developed a deep neural network (5 hidden layers) using 13 input variables for wetland mapping. The first layer has 13 nodes \( n \) for input variables \( X = \{ x_1, x_2, \ldots, x_n \} \) \( i \in [1,n] \) (Section 3.1), and the output layer has eight outputs (classes: wetland-clear, wetland-algae, grassland, shrubland, forest, barren, road and shadow). The deep neural network has five hidden layers with different number of neurons: 512 (1\(^{\text{st}}\)), 1024 (2\(^{\text{nd}}\)), 4096 (3\(^{\text{rd}}\)), 1024 (4\(^{\text{th}}\)), and 512 (5\(^{\text{th}}\) hidden layer). All layers have ReLU activation function. The batch normalization (1\(^{\text{st}}, 3^{\text{rd}} \) and 4\(^{\text{th}}\) layers) and dropout (2\(^{\text{nd}} - 5^{\text{th}}\) layers) were implemented to minimize overfitting during training steps (Ioffe and Szegedy, 2015). All hyperparameters were fine-tuned using a cross-validation procedure. For instance, the number of epochs was selected as 500 (tests: 50, 100, 250, 500, 750, 1000). The batch size was set to 16 (tests: 8, 16, 32, 64). The standard Adam algorithm is used in our application (learning rate = 10\(^{-3}\), beta\(_1\)=0.9, beta\(_2\)=0.999, epsilon=10\(^{-8}\), decay=0). In the last layer the softmax function is used to define multi-class output with per-class probability for each class. The model architecture is shown in Figure 3.5. With this basic architecture, training procedure was implemented with partition of 1470 samples (Section 2.4) into training (80\%) and validation (20\%) sets. All these experiments were performed on Intel E5 2650 CPU with NVIDIA Tesla K20 GPU with 4 GB of memory at the High-Performance Computing cluster - Iowa State University.
Evaluation methods

An inter-comparison of DNN results with traditional machine learning methods was performed. This evaluation supports our understanding of DNN performance compared to other well-known methods. These machine learning models were implemented using Python language and the scikit-learn package (Pedregosa et al. 2011). A brief description of these methods is presented as follows:

Random Forest (RF): The RF is non-parametric machine learning algorithm commonly used in remote sensing applications (Breiman, 2001; Belgiu and Dragut, 2014). Recent studies have shown the capabilities of RF for the remote sensing classification tasks, including wetland classification (Corcoran et al., 2013; Mahdianpari et al., 2017; Berhane et al., 2018). This ensemble learning method develops multiple decision trees using different subset (with replacement) from entire training data. This procedure is called bootstrap aggregating or bagging. During the training process, each tree grows in number of nodes (splits) with specified number of randomly selected
features according to complexity of classification. The cross-validation is then performed with testing samples (out-of-bag). With all decision trees, the final classification is defined by highest voted class from “forest” (all trees). The primary advantage of RF is the interpretation of outputs and its ability to handle irrelevant features. We developed a RF model with 13 parameters for eight land cover classes. The reference samples (train: 1470, see Table 3.1) were also split into 80% for training and 20% for testing. The RF requires the selection of two parameters: number of trees ($n_{trees}$) and the number of candidate variables for each split ($m_{try}$). We selected $m_{try} = \text{all variables (13)}$, and model structure was created with 200 trees. This later parameter was selected using interactive evaluation ($n_{trees}$: 50, 100, 200, 500, 1000). After training the RF model was applied to study area for wetland mapping.

**Support vector machine (SVM):** SVM is a statistical learning method that is commonly used for supervised classification of multi-spectral satellite images (Mountrakis et al., 2010). Essentially, this technique explores high-dimensional feature space to maximize the separation of classes, and then, linear (or non-linear) model classification is constructed with hyperplanes. The mapping of input data to higher dimensional feature space is performed by kernel functions, and this transformation supports the linear (or non-linear) model classification. In the classification process, the algorithm uses support vectors to find the hyperplane. The support vectors are the subset of training samples (“critical elements”) that lie nearest to the decision boundary between classes. The optimal hyperplane is defined as a hyperplane with maximum distance (margin) between hyperplane and support vectors. Originally, SVM theory was developed for binary classification problems, but different approaches emerged for multi-class SVM, such as one-versus-all (one classifier per class), or one-versus-one (pairwise). The positive attributes of SVM are its application with multiple features, effectiveness in high-dimensional spaces, and no
assumption about data distribution. We developed a multiclass SVM for wetland mapping, and SVM has three parameters for model development: kernel function, gamma and C parameter. The linear kernel function was selected in this study (tests: linear, polynomial, radial and sigmoid). Also, gamma and C (or regularization) parameters were defined as $1/\text{variables}$ and $1.0$, respectively. The PCA-based combination of $13$ variables was used to train the model.

*K-Nearest neighbor (KNN):* KNN is a relatively simple machine learning algorithm that uses distances in feature space for classification. The algorithm computes the distance ($D$) between feature vectors, and then, the categorical label for unknown samples is assigned to the most frequent label class in $k$ closest samples and their labels. The main parameters are the number of $k$ samples and distance function. The selection of appropriate $k$ value is important because the $k$ value defines the number of nearest neighbors that algorithm uses for classification of unknown samples. Higher $k$ values include too many neighbors (computationally expensive) and lower $k$ values might lead to uncertainties caused by outliers. The distance function is a key component for this classifier, and standard Euclidean distance is typically used. In this study, we used the grid search for $k$ selection ($k$ range: $1$ to $50$), and $k$ was set to $8$. The KNN has a certain flexibility in the decision boundary and this can lead to overfitting problems. Note that an attributes’ scale has an impact on the distance measure, and each feature should be normalized for better performance.

In addition to the above tests, we evaluated the influence of spatial resolution on wetland mapping using DNNs. Multi-resolution classification was performed using the resampled WorldView-3 image for $2.5$, $5$, $7.5$, $10$, $15$, $20$ and $30$ m. This analysis is relevant for users implementing wetland mapping for other sensors, such as RapidEye ($5$-m) or Landsat OLI ($30$-m). The nearest neighbor interpolation was applied to rescale the pixel size and later the trained DNN was used to classify the resampled images.
**Accuracy assessment**

The wetland assessment was performed using geometric-based metrics. Specifically, the wetland reference map (R) is compared with classified wetland (C) areas to measure the geometry similarity and discrepancies. The wetland reference map was derived in Section 2.4, and the evaluation is performed for polygons with spatial overlap higher than 50% (Figure 3.6). Features smaller than minimum mapping unit (20 m²) were deleted or merged. Following Belgiu and Dragut (2014), the over-classification (OC) and under-classification (UC) were calculated using overlapped $R_i$ and $C_j$ polygons as follows (Clinton et al., 2010):

\[
OC = 1 - \frac{A_{R_i} \cap A_{C_j}}{A_{C_j}}
\]

\[
UC = 1 - \frac{A_{R_i} \cap A_{C_j}}{A_{R_i}}
\]

Where reference wetland polygon is $R_i$, $i \in \{1, 2, \ldots, m\}$; $m =$ number of wetlands, and the classified wetland is $C_j$, $j \in \{1, 2, \ldots, m\}$. The letter A stands for polygon area. Both OC and UC are close to zero for perfect classification. Note that Clinton et al. (2010) proposed these indices for segmentation, but we are using adapted equations changing the denominator for classification evaluation (some previous articles are mixing these terms: under-segmentation ~ over-classification). The total classification error (TCE, or root mean square) combines the OC and UC to evaluate the closeness of the reference and classified polygons (Clinton et al., 2010):

\[
TCE = \sqrt{\frac{(OC)^2 + (UC)^2}{2}}
\]

The TCE ranges from 0 to 1, and a lower value of TCE represents a higher overall accuracy of wetland mapping according to OC and UC metrics. Other evaluation metrics are Area Fit Index
(AFI) (Lucieer and Stein, 2002) and Jaccard index (J-index). The J-index measures the similarity for two polygons, ranging from 0 (worst) to 100% (best).

\[
AFI = \frac{A_{R_i} - A_{C_j}}{A_{R_i}}
\]

(10)

\[
J\text{-index} = 100 \times \frac{A_{R_i} \cap A_{C_j}}{A_{R_i} \cup A_{C_j}}
\]

(11)

In addition, point-based evaluation was also performed for overall classification using a reference testing dataset for DNN classification (see testing dataset in Table 3.1). This assessment was performed using confusion matrix and three metrics: overall accuracy (OA), producer’s and user’s accuracy.

Figure 3-6. Illustration of geometric comparison for over-classified and under-classified areas.
Results

DNN performance

Figure 3.7 shows the DNN performance (accuracy and loss) for training and validation datasets. The DNN model achieved an average of last 100 validation accuracies of 0.941 and presented a low difference between training and validation losses at the end of training procedure. The first epochs present large offset between training and validation accuracies, and the “learning” process gradually improves the accuracies until the end (500 epochs). In the last 100 epochs, the agreement of training and testing accuracy shows a low or no overfitting in this model, which is a positive measure for our application. Since neural networks are typically called “black box” models, the visualization of DNN outputs helps the interpretation of model representations in each hidden layer. Figure 3.8 reveals the high-level representations from multiple layers (five hidden and output) using t-Distributed Stochastic Neighbor Embedding (t-SNE). This method allows the dimensionality reduction for visualization of features learned by DNN. In the first hidden layer, we observed that some land cover classes are clustered (e.g.: forest and grassland), while other classes are well dispersed such as road and barren pixels. The results show that data structure changes towards the output layer, increasing the class separation. Given that each layer explores the data structure for better classification, this change on representation is desired (and expected) between layers. With this visualization, remote sensing users can understand the learning representation in the DNN algorithm.
Figure 3-7. Performance of DNN for training and validation datasets.

Average of last 100 scores
validation accuracy = 0.941±0.01

Figure 3-8. High-level representations from DNN layers using t-distributed Stochastic Neighbor Embedding (t-SNE). The output layer represents the data structure before the softmax function.
Table 3.4 presents the summary of DNN performance with different groups of input variables. The performance varies for each combination of variables. For instance, the training performance achieves up to 87.7% using G2 group (only spectral bands), while the accuracy increases up to 91 % for G3 group by including DEM and CHM metrics. Our results show the highest accuracy (0.941) for G9 group, which is the combination defined by PCA-based feature selection. This performance highlights the benefits of feature selection procedure compared to the application of all variables (G8). In contrast, the worst performance was presented for G6 group (only LiDAR-elevation-based metrics) which suggests the importance of spectral and textural variables rather than only topographic indices. Further, our findings show that only original bands (G1 and G2) limited the model accuracy (< 0.880). Given the large number of possibilities for combinations, PCA-based feature selection is a simple and effective strategy to reduce the number of input variables and achieve better model performance.

Table 3-4. Performance of DNN model for different groups of input variables.

<table>
<thead>
<tr>
<th>Group name</th>
<th>Summary</th>
<th>Variables</th>
<th>Validation accuracy of last 100 epochs</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>VNIR bands</td>
<td>B2, B3, B5, B7</td>
<td>0.875 (±0.016)</td>
</tr>
<tr>
<td>G2</td>
<td>All bands</td>
<td>B1 - B8</td>
<td>0.877 (±0.020)</td>
</tr>
<tr>
<td>G3</td>
<td>All bands and 2 LiDAR</td>
<td>B1 - B8, DEM, CHM</td>
<td>0.910 (±0.011)</td>
</tr>
<tr>
<td>G4</td>
<td>All bands, 2 LiDAR, 2 indices</td>
<td>B1 - B8, DEM, CHM, NDVI, NDWI</td>
<td>0.912 (±0.020)</td>
</tr>
<tr>
<td>G5</td>
<td>All bands, 2 LiDAR, 10 indices</td>
<td>B1-B8, DEM, CHM, 10 spectral indices</td>
<td>0.934 (±0.009)</td>
</tr>
<tr>
<td>G6</td>
<td>All Elevation metrics</td>
<td>DEM, CHM, hillshade, slope, TWI, TPI, TRI</td>
<td>0.522 (±0.027)</td>
</tr>
<tr>
<td>G7</td>
<td>All textural and spectral indices</td>
<td>5 GLCMs, 10 spectral metrics</td>
<td>0.906 (±0.025)</td>
</tr>
<tr>
<td>G8</td>
<td>All variables</td>
<td>B1 - B8, 22 metrics</td>
<td>0.900 (±0.014)</td>
</tr>
<tr>
<td>G9</td>
<td>PCA-based feature selection</td>
<td>GI, MCARI, MSR, NDWI, GLCM (Correlation, Energy, Homogeneity), TRI, HS, So, CHM, B8.</td>
<td>0.941 (±0.010)</td>
</tr>
</tbody>
</table>
DNN model for wetland mapping

Figure 3.9 presents the wetland classification and probability across the study area. The visual assessment of wetland map illustrates the high quality of fine-scale mapping at 1.2 m resolution (Fig. 3.9 a). In the circles, we observed that wetland boundaries were well-represented in this classification. The transition from clear water to algae dominated area in the edges were also mapped. The wetland areas are spatially distributed in the study area, where long and narrow wetlands were observed in the northwestern region while the “dispersed” wet areas were more frequent over southeastern region. Also, DNN classification was effective in capturing the edges and shapes of complex wetlands (shallowness and blurry edge). Fig. 3.9b shows the probability map with high values in the wetland areas, as expected. The probability values between 0.3 and 0.7 were typically observed in the wetland edges. Note that some wetland areas have glint effects (Fig. 3.9, circles iv and vi), but it did not affect our classification because wetland-clear samples also included some representative samples for these glint areas as well. Although this classification is a pixel-based approach, the results show no or small salt-and-pepper effects across the study area. In contrast, the probability map shows isolated pixels with high values over forest area. These pixels are potentially forested wetland areas, but they were not assessed in this study due to limitations for wetland detection in leaf-on period. Table 3.4 shows the confusion matrix for the point-based evaluation of DNN classification. The overall accuracy (93.3%) of mapping is reasonable for environmental studies. In particular, the confusion matrix demonstrates that wetland-clear pixels were mixed with shadow and barren pixels in few cases. While the wetland-algae areas are visually similar to grassland or shrubland pixels, the classification achieved a high producer (98.3%) and user (95.2%) accuracy. This result suggests the capability to interpret the differences wetland conditions such as fresh and algae dominated waters.
Figure 3-9. Results of DNN model for (a) classification and (b) probability map for wetland areas.
Table 3-5. Confusion matrix of the classification result from DNN model.

<table>
<thead>
<tr>
<th>No of pixels Classified data</th>
<th>Reference data</th>
<th>User acc. (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Barren</td>
<td>77 0 1 0 0 2 0 80</td>
<td>96.3</td>
</tr>
<tr>
<td>Forest</td>
<td>98 0 2 3 0 0 103</td>
<td>95.1</td>
</tr>
<tr>
<td>Grassland</td>
<td>108 0 0 6 0 0 116</td>
<td>93.1</td>
</tr>
<tr>
<td>Road</td>
<td>51 4 0 4 1 0 52 98.1</td>
<td></td>
</tr>
<tr>
<td>Shadow</td>
<td>84 0 4 0 1 0 92 91.3</td>
<td></td>
</tr>
<tr>
<td>Shrubland</td>
<td>6 3 1 64 0 1 75 85.3</td>
<td></td>
</tr>
<tr>
<td>Wetland-clear</td>
<td>1 0 0 2 0 143 1 147 97.3</td>
<td></td>
</tr>
<tr>
<td>Wetland-algae</td>
<td>0 0 0 0 0 6 119 125 95.2</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>80 108 112 51 89 73 156 121 790</td>
<td></td>
</tr>
<tr>
<td>Prod. acc. (%)</td>
<td>96.3 90.7 96.4 100.0 94.4 87.7 91.7 98.3</td>
<td></td>
</tr>
</tbody>
</table>

Overall accuracy (OA): 93.3%

Inter-comparison of methods

This section presents the inter-comparison of machine learning methods for wetland mapping (Figure 3.10). The geometry-based metrics were calculated for each method by comparing the reference and classified wetland polygons (Table 3.6). These metrics indicate the agreement of shape and location achieved for all classifications. In general, our findings show that DNN classification has similar performance to other traditional machine learning methods. This is clearly observed in the Figure 3.10 as the total classification errors range from 0.104 to 0.111, and the overlapped area between reference and classified is within 87.93 - 93.33%. The AFI values are negative for RF and DNN, which indicates that these methods tend to over-classified wetland areas. Also, RF and DNN presents the highest percentage of overlapped area (93.33 and 90.36%, respectively). In contrast, SVM and KNN have the lowest over-classification results, largest AFI values, and the shortest processing time. This is relevant for users with interested in large-scale application, because they can prioritize the faster model when the results across methods are similar in the study area. Our results in Table 3.6 also show similar Jaccard index (~0.81) between
methods. This positive result (also known as closeness index) indicates that the delineation of wetland polygons matches the corresponding reference in terms of spatial location and shape. Therefore, by analysis of all these metrics, this study shows the benefits of very high-resolution image and DNN algorithm (as well as other methods) to preserve the high-level of spatial details.

Figure 3-10. Classification results of (a) deep neural network, (b) random forest, (c) support vector machine, and (d) k-nearest neighbor models. The bar graphics present the overlapped, under- and over-classified areas (%) for each method. Note that the sum of overlapped and under-classified areas is the total area of reference wetland mapping.
Table 3-6. Assessment of geometric-based metrics for wetland mapping from different methods.

<table>
<thead>
<tr>
<th></th>
<th>OA</th>
<th>TCE</th>
<th>OC</th>
<th>UC</th>
<th>AFI</th>
<th>J</th>
<th>Overlap (%)</th>
<th>Processing (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNN</td>
<td>0.942</td>
<td>0.111</td>
<td>0.124</td>
<td>0.096</td>
<td>-0.032</td>
<td>0.801</td>
<td>90.36</td>
<td>31047</td>
</tr>
<tr>
<td>RF</td>
<td>0.947</td>
<td>0.105</td>
<td>0.133</td>
<td>0.067</td>
<td>-0.076</td>
<td>0.817</td>
<td>93.33</td>
<td>34824</td>
</tr>
<tr>
<td>SVM</td>
<td>0.942</td>
<td>0.104</td>
<td>0.084</td>
<td>0.121</td>
<td>0.041</td>
<td>0.814</td>
<td>87.93</td>
<td>791</td>
</tr>
<tr>
<td>KNN</td>
<td>0.941</td>
<td>0.111</td>
<td>0.105</td>
<td>0.116</td>
<td>0.012</td>
<td>0.800</td>
<td>88.37</td>
<td>1981</td>
</tr>
</tbody>
</table>

Note: OA, TCE, OC, UC, AFI, and J refer to overall accuracy, total classification error, over-classification, under-classification, area fit index, and Jaccard index, respectively. * OA is calculated from validation points. Computing resource: Intel Xeon(R) E3-1270 (3.80 GHz) processor.

Impact of spatial resolution on the wetland mapping

Figure 3.11 summarizes the DNN classification developed for different spatial resolutions. As described in Section 3.3, we resampled the WorldView-3 image to generate seven spatial resolutions, and then, DNN model is implemented on these resampled images to classify the wetland areas. Not surprisingly, the mapping quality decreases from very high-resolution (1.2 m) to medium resolution (30-m), and the overlapped area between reference and classified wetland polygons decreases from 90.36% (1.2-m) to 61.22% (30-m). Also, OC and UC values increased as the spatial resolution decreases. The under-classification is almost ~40% of wetland areas using 30-m resolution. This result highlights the importance of appropriate satellite dataset resolution when applied to complex wetland with small and narrow areas. In fact, the spatial resolution influences the minimum mapping unit, and comprehensive inventories face this limitation when using medium spatial resolution datasets. In contrast, remote sensing users should also consider the balance of mapping quality and processing time, especially for large-scale applications. For instance, the wetland classification of 2.5-m resolution image reduces the processing time in ~5 times compared to 1.2 m resolution data, while the Jaccard similarity index remains quite similar.
between these two resolutions (J-index (1.2-m): 0.800, and J-index (2.5-m): 0.779). Therefore, some studies can take the advantage of resampled the original data to a smaller spatial resolution (e.g.: 1.2 to 2.5 m) without affect the final mapping quality.

Figure 3-11. Performance of DNN classification for different spatial resolution (1.2, 2.5, 5, 7.5, 10, 15, 20, and 30 meters). (a) percentage of overlapped area between reference and classified wetland polygons, (b) percentage of under- and over-classified areas, (c) processing time for wetland classification.

Discussion

This paper presents a new framework for wetland mapping using a deep neural network algorithm and a WorldView-3 image. Our methodology includes the development of DNN architecture, calculation of multiple variables from multi-spectral and LiDAR data, application of PCA-based feature selection, and inter-comparison of methods (RF, SVM, KNN). In general, the DNN model effectively delineated wetland areas (Figure 3.9), and classified wetlands were
spatially consistent with reference mapping (Table 3.6). Our results corroborate with previous findings demonstrating the high-quality of wetland mapping using WorldView imagery (McCarthy et al., 2015; Whiteside et al., 2015; Vanderhoof et al., 2017). The DNN architecture showed a great capability for target distinction in the study area (Figure 3.8), and the final classification illustrated the method's ability to classify small and narrow wetlands (minimum mapping unit of 20 m²). The results in Figure 3.10 indicate that DNN classification achieved similar performance to other traditional methods (RF, SVM, and KNN). The appropriate selection of input parameters explains, at least in part, the success of all models. Other studies have also reported high classification accuracies using these algorithms (Corcoran et al., 2013; Rezaee et al., 2018; Whyte et al., 2018; Mahdianpari et al., 2019). For instance, Jiang et al. (2018) found similar accuracies for water extraction in China using SVM and multi-layer perceptron models. Although the performances were similar, we emphasize that neural networks offer a great flexibility in the model architecture (number of layers, neurons, activation functions, and regularization layers), which can be useful to explore data pattern when the traditional classifiers fail to increase the accuracy. With the advance of machine learning packages (TensorFlow) and computer power (GPU resources), the application of deep learning methods has become a potential alternative for image classification focused on wetland areas.

In addition, this study highlights the selection of varied input variables (spectral, textural and topographic) to improve the DNN performance (Table 3.4). In this context, the WV-3 sensor offered a sufficient set of spectral bands for the calculation of different indices (Table 3.2). This study also used the Iowa LiDAR data for elevation metrics (e.g.: hillshade, terrain ruggedness index), and the classification was thus not merely dependent on the spectral information. Although the calculation of potential variables required further efforts for implementation, our results
showed that only original bands or either topographic metrics were not able to increase the model accuracy over 88% (Table 3.4). So far, some studies have applied multiple input variables without feature selection (Lane et al., 2014; Huang et al., 2014; Fu et al., 2017; DeLancey et al., 2019). However, we emphasize the fact that this procedure is relevant and necessary to the identification of parsimonious set of input variables, improving inter-class separability and model accuracy. These findings support the previous studies that used feature selection for modelling and classification (Corcoran et al., 2013; Waser et al., 2014; Silva et al., 2016). For example, Berhane et al. (2018) proposed 37 potential variables from QuickBird and LiDAR data for wetland classification, but they found the highest accuracy (87.9%) of RF using only three variables. Such example illustrates the importance of further analysis of potential variables to explore efficient modeling.

According to our analysis (e.g.: Figure 3.11), very high-resolution images are important for accurate mapping of complex wetland areas. Previous studies have developed applications with available medium resolution data, such as Landsat (Bwangoy et al., 2010; Huang et al., 2014) and Sentinel-2 (Araya-López et al., 2018; Whyte et al., 2018), but there is a relevant discussion about limitations for small wetlands (Ozesmi and Bauer, 2002; Leonard et al., 2012; Mwita et al., 2013; Gallant, 2015). Notably, our analysis shows high over- and under-classification of wetlands using medium spatial resolution (20 or 30-m). These findings reinforce the idea that the satellite data must be selected according to the wetland characteristics under investigation. In this study area the final 1.2 m classification preserved the wetland shape and size, and shows the potential of WorldView-3 image. Recent literature has also claimed the importance of commercial satellites for detection and delineation of complex wetland areas, such as QuickBird, GeoEye, and Worldview (White et al., 2011; Whiteside et al., 2015; Rapinel et al., 2015; Mui et al., 2015;
Vanderhoof et al., 2017). In fact, spatial resolution is a crucial component for definition of minimum mapping unit in the classification, which will have a great impact on the total distribution and location for wetland inventories. For instance, Morrissey and Sweeney (2006) revealed that 82% of wetlands ≤ 3 acres (number and areal extent) were omitted from the U.S. NWI maps. While these national inventories are an official resource for wetland management, our study suggest that WorldView-3 imagery are potential dataset to support the wetland inventories, offering appropriate resolutions (spatial, radiometric and spectral) for fine-mapping of these areas.

Although this study shows the benefits of WorldView-3 image and DNN algorithm for wetland studies, we recognize some limitations for large-scale implementation. First, the acquisition of commercial satellite images involves significant cost, and this is a critical factor for continuous application of WorldView imagery. This experiment was conducted using a single image (this was the only cloud-free image in the archive), but comprehensive inventories typically require a large volume of data. Alternatively, the national wetland programs can target some sites with large abundance of small wetlands to reduce the volume of purchased data. Second, forested wetlands were not considered in this study because the classification of these areas is typically dependent on images collected in leaf-off period (Vanderhoof et al., 2017). So, future studies should include images from different seasons for forested wetland areas. Third, areas under natural management allow the application of historical LiDAR elevation datasets, but other regions may not have appropriate data available. In this case, our framework requires the adaptation to perform using only spectral and textural variables, which had a reasonable performance as well. Additional research is needed to expand this framework for other wetland-dominated sites, including the classification of wetland types (bog, fen, marsh and swamp).
Conclusion

In this study, we investigated a new classification framework using DNN algorithm and WorldView-3 image for complex wetland mapping. To the best of our knowledge, this paper is the first assessment of this framework (data + method) for fine-scale mapping of wetlands. Our results showed a high classification accuracy (0.933) of DNN results and mutual spatial overlap between reference and classified wetlands polygons (J-index ~ 0.801). Our research has three contributions to the recent discussion on complex wetland mapping: (1) this study shows the value of DNN algorithm and WorldView-3 image for fine-mapping of complex wetland areas (minimum mapping unit ~ 20m²). The implication of this is that NWI programs can potentially consider the acquisition of commercial satellite imagery to maintain and refine the continuous monitoring of wetlands and reduce the mapping unit to accommodate a greater variety of shape and size. (2) The combination of spectral, textural and elevation variables influences the DNN performance, and the identification of most predictive variables should be implemented using feature selection such as PCA. (3) In addition, our results demonstrated that very high-resolution image is relevant for complex and small wetlands in low-relief terrain. Given the fact that commercial satellites with high spatial resolution are becoming more affordable for remote sensing users, this study has potential to support ongoing operational wetland programs which can benefit of these strategies/scheme for implementation of high-resolution satellite images in NWI wetlands.

Acknowledgements

This project was undertaken with the financial support of the Iowa Department of Transportation. The authors thank Iowa-DOT staff for their support and insights, especially Brad Hofer and Michael Carlson. We also would like to thank Iowa LiDAR project for data availability.
References


CHAPTER 4. DIGITAL MAPPING OF STRUCTURAL CONSERVATION PRACTICES IN THE MIDWEST U.S. CROPLANDS: IMPLEMENTATION AND PRELIMINARY ANALYSIS

Vitor S. Martins a, Amy L. Kaleita a, and others

a Agricultural and Biosystems Engineering, Iowa State University, Ames, Iowa, USA.

A manuscript in preparation to be submitted to Science of the Total Environment

Abstract

Best management practices are long-term conservation efforts in the Midwest U.S. croplands, and many farmers have adopted structural conservation practices (SCPs) to reduce soil erosion and runoff, such as terraces and grassed waterways. Despite that, the geographic distribution of these practices is barely known in the region, and mapping initiatives are required to develop timely and spatially explicit inventories of SCP areas to support conservation programs. This study presents the first automated mapping of SCPs in the agricultural areas over 12 Midwest U.S. states. The National Agriculture Imagery Program data (2018 – 2019) were used to map the SCP areas at 2-m spatial resolution (490.2 billion pixels). The adapted U-Net architecture was trained with 500,000 labeled patches, and parallel computing was implemented for this large-area semantic segmentation. The mapping results achieved 76.8% overall accuracy across 20 counties. Overall, the spatial distribution of SCPs is highly distinct among the states, and our results indicate that 52% of SCP areas are distributed over Iowa (26%), Illinois (15%) and Nebraska (11%). In contrast, the states with lowest SCP areas are Michigan and North Dakota, with less than 4% of SCP areas. Since the SCP extent is also dependent on the number of cropland areas per state, the percentage of SCP per cropland area was calculated. Specifically, the average percentage of SCP area per cropland is 1.19%, ranging from ~0.8 (e.g., North Dakota and south Minnesota) to 5.5% (e.g., northeast Kansas and southwest Iowa). We observed magnitude agreement between
classified SPC areas and NRCS funded practice counts for grassed waterways/terraces areas among the states. Our findings show that distribution of SCP areas is partially associated to the soil and topographic features in the landscape. For instance, moderate slope (> 2%) and low saturated hydraulic conductivity (< 7 µm/s) values were observed in the high-high cluster of SCP areas. Interestingly, our results also illustrate that regions with high soil erosion rates present the largest percentage of SCP areas in croplands as well, indicating conservation efforts by farmers. While this preliminary analysis shows some limitations in the mapping quality (mislabel, non-accurate location or discontinuity of SCP areas), the framework has the potential to be operationally used for conservation monitoring. The development of such products has positive implications for conservation programs, and this geospatial inventory is an easily accessible product for large-area evaluation of conservation practices across Midwest U.S. croplands.

**Keywords:** Conservation, terraces, grassed waterways, semantic segmentation, U-net model.

**Introduction**

Over the last several decades, U.S. farmers have implemented conservation practices that benefit both agricultural production and environmental protection (Hobbs et al., 2007; Knowler and Bradshaw, 2007; Kassam et al., 2009; Floress et al., 2018). Despite efforts to minimize nutrient loss and soil erosion (Carpenter et al., 1998; Schoumans et al., 2014), agricultural non-point source pollution remains a major concern for water quality in the United States (Stoddard et al., 2016). Sediment transport and nutrient export from crop fields lead to degradation of water quality in freshwater systems by increasing algal growth and turbidity levels (Kröger et al., 2013). The Gulf of Mexico “Dead Zone” is one example of the ecological impact caused by nutrient-laden water from the Mississippi River reaching the coastal waters (Rabalais et al., 2002; Diaz and Rosenberg, 2008; Dale et al., 2010). As a result, U.S. Environmental Protection Agency established a goal of
45% reduction of nutrient loads (nitrogen and phosphorus) to surface waters along the Mississippi basin (Dale et al., 2010). In these efforts, USDA Natural Resources Conservation Service has provided financial and technical assistance for the adoption of best management practices (BMPs) at the farm level.

Agricultural BMPs are a set of guidelines, practices, and structural controls designed to preserve soil and water resources and maintain them within agricultural fields. Some examples of structural conservation practices (SCPs) are i) grassed waterways, ii) contour buffer strips, iii) terraces, iv) filter strips, v) riparian buffer, and others. These practices are typically implemented in the most sensitive areas (e.g., highly erodible land fields), and each of them has a specific role in the agricultural landscape. For instance, terraces are earthen ridges around a hillside that prevent soil erosion on steep slopes (Tarolli et al., 2014), while grassed waterways are natural or constructed vegetated channels that control surface runoff, erosion, and nutrient loss in the drainage pathways (Fiener and Auerswald, 2003). Recent studies have quantified the benefits of these BMPs in cropland areas (Zhang and Zhang, 2011; Reimer et al., 2012; Liu et al., 2013; Liu et al., 2017). Kröger et al. (2012) revised the BMP effectiveness in row-crop agriculture over the Lower Mississippi Alluvial Valley and they demonstrated that nine BMPs provide a significant reduction of nutrient loss (range: 15 - 100%), such as total nitrogen and phosphorus. Similarly, Panagopoulos et al. (2011) showed that filter strips reduce the delivery of total phosphorus (up to 50%) in the surface water of a Western Greece catchment. In a modeling approach, Haas et al. (2017) showed that buffer strips reduce nitrate loads reduction (up to 10%) in the catchment of the river Treene. Given the relevance of conservation techniques (Liu et al., 2017; Xiong et al., 2018), conservation agencies and research organizations are promoting local networks and access to information to increase the engagement of farmers towards sustainable practices (Baumgart-Getz
et al., 2012), such as North Central Region Water Network (https://northcentralwater.org/) and Iowa Learning Farms (https://www.iowalearningfarms.org/).

In this perspective, the accurate mapping of structural conservation practices becomes crucial information to understand the geographic extent of current practices and its function in the agricultural landscape. Recently, the Agricultural Conservation Planning Framework (ACPF) was implemented to provide meaningful conservation plans (“menu-driven approach”) at the watershed level (Tomer et al., 2015; Lewandowski et al., 2020). The framework incorporates geospatial data to identify vulnerable areas and recommend conservation options. However, existing practices are not considered in this framework, and the watershed plan often targets areas with already implemented BMPs. Rundhaug et al. (2018) compared the existing and potential ACPF practices in three Iowa watersheds, and they emphasized the importance of BMP mapping to support the development of conservation scenarios. Regarding the soil erosion modeling, Panagos et al. (2015) highlighted that conservation practices are typically neglected in the soil erosion risk modeling because they are difficult to assess and quantify for large areas. Conceptually, assuming no conservation or uniform values, significant uncertainties may be introduced in the erosion estimates, especially in agricultural areas. These examples show the benefits of geospatial information of conservation practices to support environmental analysis, but mapping initiatives are barely presented in the literature. In this context, Iowa BMP Mapping Project (IBMP) is a unique initiative that offers a detailed spatial database about vegetative/structural practices across Iowa watersheds (ISU, 2016). The mapping framework includes the visual interpretation of National Agriculture Imagery Program (NAIP) aerial imagery (2007-2010) and LiDAR-derived products.
While a comprehensive inventory is a valuable data resource (Lam et al., 2011; Rundhaug et al., 2018), the product generation is dependent on manual classification performed by multiple GIS specialists/interns and takes multiple years for completion. This limitation reinforces the demand for a timely and reliable framework for SCP mapping which will support the evaluation of other U.S. states. This study provides the SCP mapping in Midwest U.S. croplands, mainly corn and soybean areas. The methodology applies NAIP aerial imagery (2018 – 2019) and semantic segmentation method (adapted U-Net algorithm) to classify SCP/non-SCP areas in the croplands. This product generation involves the classification of ~490.2 billion pixels at 2-m resolution, and the computational strategies for efficient implementation were detailed in this study. We also investigated the spatial distribution of SCP areas and other landscape characteristics, such as soil properties and topographic-related variables. Note that agricultural BMPs represent a variety of conservation practices (management of pests and nutrients, crop rotations and tillage practices), and this study is only focused on vegetative/structural conservation practices, mostly terraces and grassed waterways. There is no study that provides a spatially explicit mapping of these practices in the Midwest U.S. region, and the potential implications are further discussed in the Section 4.

Data and methods

The proposed framework applies publicly available imagery (NAIP) and semantic segmentation method (adapted U-Net architecture) to generate a SCP/non-SCP mask in agricultural areas. The generalized procedure is shown in Fig. 4.1 and broadly described as: 1) NAIP pre-processing; 2) model training; 3) SCP classification and 4) validation and spatial analysis. These steps are detailed in the following sub-sections.
Study area

The study area includes one of the major cropland areas in the world: Midwest United States. A total of 12 states were selected for this analysis, and they represent the most grain productive region of the United States (Table 4.1). For example, Iowa is a leader of U.S. corn production (2.7 billion of those bushels) with approximately 87,500 farms (Iowa Corn, 2019). The region presents mostly a hot-summer humid climate and cold winters, and commercially and environmentally relevant U.S. rivers are crossing the region such as Mississippi and Missouri rivers. While cropland areas are the most predominant land cover, the natural landscape consists of prairie and savanna, forests, and wetland areas. The terrain landscape is typically flat or moderately rolling hills such as in southwest Wisconsin or western Iowa. The frequency of corn/soybean areas illustrates the importance of selected states for national crop production (Fig. 4.2), and consequently, the conservation efforts are highly expected in this region.
Figure 4-2. Location of Midwest U.S. states (study area).

Table 4-1. Characteristics of selected U.S. states and NAIP data used in the study.

<table>
<thead>
<tr>
<th>States</th>
<th>State area (km²)</th>
<th>Cultivated area* (km²)</th>
<th>Annual rainfall (mm)</th>
<th>Elevation (m)</th>
<th>Slope (%)</th>
<th>NAIP imagery Year</th>
<th>Size (GB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Illinois (IL)</td>
<td>145,968</td>
<td>85,984</td>
<td>1071</td>
<td>191</td>
<td>1.10</td>
<td>2019</td>
<td>175.5</td>
</tr>
<tr>
<td>Indiana (IN)</td>
<td>93,789</td>
<td>45,498</td>
<td>1171</td>
<td>229</td>
<td>1.50</td>
<td>2018</td>
<td>108.0</td>
</tr>
<tr>
<td>Iowa (IA)</td>
<td>145,667</td>
<td>92,176</td>
<td>945</td>
<td>324</td>
<td>1.74</td>
<td>2019</td>
<td>134.4</td>
</tr>
<tr>
<td>Kansas (KS)</td>
<td>213,184</td>
<td>25,293</td>
<td>746</td>
<td>585</td>
<td>1.30</td>
<td>2019</td>
<td>188.6</td>
</tr>
<tr>
<td>Michigan (MI)</td>
<td>153,620</td>
<td>19,077</td>
<td>925</td>
<td>276</td>
<td>1.39</td>
<td>2018</td>
<td>172.0</td>
</tr>
<tr>
<td>Minnesota (MN)</td>
<td>218,781</td>
<td>62,952</td>
<td>740</td>
<td>371</td>
<td>1.20</td>
<td>2019</td>
<td>240.1</td>
</tr>
<tr>
<td>Missouri (MO)</td>
<td>180,431</td>
<td>32,733</td>
<td>1125</td>
<td>262</td>
<td>2.60</td>
<td>2018</td>
<td>190.1</td>
</tr>
<tr>
<td>Nebraska (NE)</td>
<td>200,365</td>
<td>58,572</td>
<td>623</td>
<td>795</td>
<td>1.75</td>
<td>2018</td>
<td>181.0</td>
</tr>
<tr>
<td>North Dakota (ND)</td>
<td>183,135</td>
<td>27,697</td>
<td>500</td>
<td>556</td>
<td>1.31</td>
<td>2019</td>
<td>175.0</td>
</tr>
<tr>
<td>Ohio (OH)</td>
<td>108,978</td>
<td>35,247</td>
<td>1096</td>
<td>283</td>
<td>2.60</td>
<td>2019</td>
<td>115.6</td>
</tr>
<tr>
<td>South Dakota (SD)</td>
<td>199,718</td>
<td>42,139</td>
<td>534</td>
<td>665</td>
<td>1.84</td>
<td>2018</td>
<td>196.2</td>
</tr>
<tr>
<td>Wisconsin (WI)</td>
<td>145,562</td>
<td>22,447</td>
<td>971</td>
<td>332</td>
<td>2.13</td>
<td>2018</td>
<td>107.2</td>
</tr>
</tbody>
</table>

* Corn and soybean areas were observed during more than five years between 2010-2019 in the NASS CDL program.

High-resolution aerial imagery

The National Agriculture Imagery Program (NAIP) is a comprehensive program administered by the USDA’s Farm Service Agency (FSA) that acquires high-resolution aerial imagery across the United States. The NAIP imagery are mostly collected during the agricultural growing season (leaf-on) and have the appropriate resolution for a variety of environmental studies (Basu et al., 2015; Peter et al., 2018). Beginning in 2003, NAIP projects were developed on a 5-
year cycle to collect images at natural color (RGB) using film cameras. Recently, NAIP imagery are acquired by digital sensors with 4-band (RGB + NIR), and the imagery dataset is publicly available by USDA Natural Resources Conservation Service. In this study, false-color NAIP imagery were obtained for entire Midwest U.S in 2018 and 2019. The data are available in Natural Resources Conservation Service (NRCS) (https://nrcs.app.box.com/v/naip). A total of 1054 county-based image were delivered in MrSID compressed format, resulting in ~1.98 TB of data. For the pre-processing step (Fig. 4.1), these images were decoded to GeoTiff format, and then, it was resampled from full-resolution to 2-m resolution using nearest neighbor method. This resampling reduces the storage needs and keeps sufficient resolution for our analysis. After that, the entire dataset was re-projected to USA Contiguous Albers Conic Equal Area (ESRI:102003). Note that this study does not perform a temporal analysis because older NAIP images present different number of bands and image quality compared to recent data.

**Training data**

The development of large training dataset is one of the major challenges for deep learning applications. The training of semantic segmentation methods, such as U-Net model (Section 2.4.1), requires fully labeled image patches, and this study exploits the Iowa BMP dataset to generate these training samples (Figure 4.1). As described, the Iowa BMP project has been funded by multiple institutions and produced a BMP inventory in geodatabases for the 2007-2010 timeframe. The Iowa-BMP is focused on baseline conservation practices, such as terraces, WASCOB, grassed waterways, strip cropping and contour buffer strips. The complete set of BMPs is available at watershed-level through the project’s website (https://www.gis.iastate.edu/gisf/projects/conservation-practices). Among these BMPs, this study uses two relevant structural conservation practices (SCPs): grassed waterway and terrace. These
practices are the most abundant in the Iowa inventory (> 90% of mapped areas), and they are typically observed within cropland boundaries. Here, it should be emphasized that these practices are visually identified in the images, but they are difficult to separate using optical data due to similar spectral response and shape in some cases (even by human interpretation). So, binary classification (SCP/non-SCP) is appropriate for this study, and this limitation is further discussed in Section 4.4. For this training dataset, SCP polygons were firstly converted to raster files (256 x 256 pixels), creating a binary label (SCP: 1, non-SCP: 0). All these labeled patches are stored with corresponding false-color images from NAIP 2010 data, as used in the Iowa BMP project. The next step was the random selection of 500,000 pair samples (image + label), where 90% of these samples have SCP areas and the other 10% of samples have non-SCP areas, such as crop, grassland, building, road and forest targets. This combination of SCP and non-SCP patches gives a comprehensive dataset (size: 122 GB) for model training, and this dataset is publicly available at https://doi.org/10.5281/zenodo.3762370.

Figure 4-3. Development of SCP training dataset. The samples contain SCP (90%) and non-SCP (10%) patches obtained from Iowa BMP project.
Product generation

Semantic segmentation: U-Net model

Recently, fully convolutional network became a prominent deep learning architecture for semantic segmentation tasks (Long et al., 2015; Ronneberger et al., 2015; Badrinarayanan et al., 2017; Jegou et al., 2017). These architectures are well-established in remote sensing applications and have been used to map land cover (Stoian et al., 2019), forest degradation (Wagner et al., 2020), building (Xu et al., 2017) and roads (Zhang et al., 2017). A detailed review of different studies is described in Ma et al. (2019). In this research, an adapted U-Net network was used to perform a binary segmentation of a vegetative/structural SCP areas. Briefly, U-Net is a deep fully convolutional network that was originally designed for semantic segmentation of biomedical images (Ronneberger et al., 2015). This model presents the advantage of 2D feature extraction to explore spatial-contextual information compared to standard pixel-based methods (e.g., random forest and support vector machine). The U-Net has an encoder-decoder architecture where (i) the encoder part extracts spatial features from the input image and (ii) decoder constructs the segmentation map from the encoded features. We adapted the U-Net architecture (Fig. 4.5) with three-band input image and additional convolutional block to explore deeper features of our large training dataset. Furthermore, we added batch normalization and dropout layers to prevent overfitting in each block. In general, our network contains five blocks with two convolutional (3 x 3) layers followed by ReLU activation function and max pooling (2 x 2) in the encoder part. The number of filters duplicates for each stage of convolution operations. In the transition stage, two convolutional layers connect the both encoder and decoder parts. In the decoder part, the transposed convolutional layer is responsible for upsampling the feature map before the block of two convolutional layers (+ ReLU and max pooling). Also, the feature maps from encoder part are transferred and concatenated with the output of the up-sampling step. These “skip connections”
are introduced to recover the fine-spatial information from down-sampling to up-sampling layers. A sequence of upsampling and convolutional layers reconstructs the dimension of input patch until the end of network. Finally, the last layer has convolutional layer (1 x 1) with sigmoid activation function and generates the probability map with the same dimension of input. The SCP label is attributed to pixels with probability values higher than 0.5.

The adapter U-Net model was trained with 500,000 image patches. The first 90% of the dataset is used for training (450,000 patches), while the last 10% is used for validation (37,500 patches) and testing (12,500 patches). For network training, a custom data generator was designed to feed the model (batch-by-batch) because there is not enough memory to load all the 500,000 samples at the same time. We used the Adam optimizer (lr = 0.001) and the binary cross-entropy as loss function, which is most appropriate for this segmentation task. The number of epochs was set as 100 (batch size of 32) since the improvement did not exceed 0.002 of accuracy after 80 epochs. This model was trained in the ISU HPC Pronto cluster with Intel Xeon Silver 4114 CPU @ 2.20GHz, and GPU NVIDIA Tesla v100 32GB. Note that this adapted U-Net architecture has 31,099,429 trainable parameters, and large GPU memory is needed for this application. The total training takes up to 13 days (~10,850 secs per epoch). The training, validation, and testing accuracies were similar (0.983, 0.974, 0.975, respectively), which indicates small or no overfitting during the training. All these experiments are conducted with Python's high-level package Keras (TensorFlow backend) in Python environmental and its open-source libraries, such as GDAL, NumPy, and Scikit-image.
SCP classification and post-processing

This product generation demands computational resources for fast processing of large volume of high-resolution data across the Midwest region; more specifically, this study classified ~490.2 billion pixels. In this research, we performed all the classification process in the High-Performance Computing (HPC) Condo cluster at Iowa State University. There are relevant aspects for the efficient processing of these high-resolution images. The county-based images are large files (range: 300 MB to 17 GB) which imposes a high load on the nodes with intense read/write of files and memory limitations for multi-processing jobs. To solve that, the first strategy was the subdivision of original image in 18 tiles, and then, perform the parallel processing of these small files. This step takes advantage of multiple cores in HPC nodes and we run dozens of tiles simultaneously using 15 nodes. Another part of patch-based classification is the sliding window process. One of the main advantages of FCNs is the “dense” prediction where all pixels are labeled in the output patch, and this reduces the redundancy in the classification. However, SCPs are...
typically long and continuous areas, and some overlap is needed to maintain the spatial-contextual features on the border and between adjacent patches. We adopted a stride of 192 pixels between input patches (in other words, overlap area equal to 64 pixels) to balance run time and quality of results. Finally, the entire Midwest U.S. is classified in five days (processing time: ~14 min per 100 km\(^2\) of image), generating 1.39 TB of mapping results. After that, a post-processing step is implemented for the final product, including the filter of non-cropland areas and noise areas. The cropland data layer from USDA was used to filter the non-cropland areas. The cropland mask was created with corn or soybean areas that occur at least 5 times between 2010 and 2019 (see areas in Fig. 4.2 with frequency \(\geq 5\)). Since noise and isolated pixels are likely expected at 2-m classification, and we filtered out small areas (up to 20 pixels). Finally, we aggregated all tiles to produce a binary mask (SCP/non-SCP) per county.

Accuracy assessment

The assessment of SCP/non-SCP mask was performed using error matrices between reference samples and classified results. Briefly, a total of 5,000 reference samples were labeled in 20 Midwest counties (Fig. 4.5). For each county, 250 sample pixels were randomly distributed in the first step. Given that there are more non-SCP pixels in the landscape, a second step was performed to increase the representativeness of SCP areas in the validation dataset. The random samples are relocated to the nearest SCP pixel (true positive sample). If there are no valid SCP pixels in the search area, this sample is relocated for any non-SCP pixel within cropland area (false positive sample). Each sample was labeled as either SCP or non-SCP pixels by visual interpretation of high-resolution imagery (NAIP image and ArcGIS image online service). The error matrices were generated with F1-score, precision, recall and overall accuracies.
F1-score = \( 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \)  

\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2} 

\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3} 

\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{4} 

Where, true positive (TP) and true negative (TN) represent the samples that were correctly classified as SCP and non-SCP, respectively. In turn, the false positive (FP) samples are those pixels mistakenly classified as SCP, while the false negative (FN) samples are those pixels mistakenly classified as non-SCP areas. The F1-score uses a harmonic average of the precision and recall metrics, ranging from 0 to 100. Precision is the measure of the fraction of classified pixels is true for each category (TP and FP), and recall is the ratio of the number of samples correctly classified as SCP and the total number of samples belonging to SCP class (TP and FN).

Figure 4-5. Location of selected counties for SCP validation.
Spatial analysis and ancillary products

The spatial analysis includes the discussion of SCP distribution, spatial autocorrelation (Global Moran’s I test), and Local Indicators of Spatial Association (LISA) analysis. The SCP areas were calculated per grid with regular size (10 x 10 km), giving a common spatial area. Furthermore, the adoption of conservation practices is potentially affected by a variety of factors such as soil and terrain properties (Knowler and Bradshaw, 2007), and the association between SCP and some variables was evaluated in the high and low clusters of SCP extent. The used variables are described as follows:

Topographic-related features: The 30-m elevation product from Shuttle Radar Topography Mission (SRTM) was used to calculate the topographic-related features, such as slope and topographic wetness index (TWI). The SRTM 30-m images were acquired in the web interface (SRTM Tile Downloader, https://dwtkns.com/srtm30m/) for Global Change Master Directory dataset (https://gcmd.nasa.gov)

Soil-related features: Saturated hydraulic conductivity (µm/s), bulk density (g/cm³), and soil organic matter (kg/m²) were used in this study. These products were distributed by SoilWeb portal (https://casoilresource.lawr.ucdavis.edu/) from California Soil Resource Lab at UC Davis, and they were derived by aggregating USDA-NCSS soil survey data (SSURGO back-filled with STATSGO where SSURGO is not available) within 800m² grid cells.

Soil erosion rate: Global Soil Erosion 2012 (tons/ha/yr) and K-factor maps from Borrelli et al. (2017) were used in this study. This 25 km resolution product is a re-sampled version of the original soil erosion map derived from RUSLE-based Global Soil Erosion Modelling platform. The product is available at European Soil Data Centre (https://esdac.jrc.ec.europa.eu/content/global-soil-erosion).
All these variables were re-projected to Albers Conic Equal Area projection and filtered by cropland mask, and the average values of these variables were calculated for each 10 x 10 km grid.

**Results**

**Classification performance and error sources**

This section presents the SCP mapping results obtained from adapted U-Net architecture. The confusion matrix is presented in Table 4.1, and the overall accuracy and F1-score of the SCP mapping are 76.8% and 80%, respectively. This validation shows large number of false negatives (1119) and precision of 67.5% which suggest the underestimation of areas in this product. In turn, the satisfactory recall (98.2%) and small false positives samples suggest a small overestimation. The visual inspection shows some positive aspects of 2-m product (Figure 4.6). In the Figs. 4.1 and 4.2, we observed the successful classification of terraces and grassed waterways in different context. For instance, we show the performance in both vegetated and fallow areas with correct label (Fig. 4.6.2 versus 4.6.5) and even in the transition areas. It is worthwhile to emphasize that urban and forest pixels were filtered out by cropland mask, but we did not observe a systematic error in the raw product, and they are typically classified as non-SCP pixels as expected. Figure 4.7 is another example of SCP mapping, and the visual comparison illustrates the performance of our classification. For example, the terraces are consistent with hillshade map and different shapes and dimension of terraces are well-represented.
Table 4-2. Confusion matrix of SCP mapping across 20 counties.

<table>
<thead>
<tr>
<th>Classified</th>
<th>Reference</th>
<th>Overall accuracy: 76.8 %; F1-score: 80.0%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Non-SCP</td>
<td>SCP</td>
</tr>
<tr>
<td>Non-SCP</td>
<td>1514 (TN)</td>
<td>1119 (FN)</td>
</tr>
<tr>
<td>SCP</td>
<td>43 (FP)</td>
<td>2324 (TP)</td>
</tr>
<tr>
<td>Total</td>
<td>1557</td>
<td>3443</td>
</tr>
</tbody>
</table>

Figure 4-6. Results of SCP classification.
While this overall accuracy can be claimed as satisfactory for this general analysis, the farm-level analysis is potentially affected by omission/commission errors. Figure 4.8 shows some limitations of this mapping. For example, the discontinuity of SCP area is one of the most common omission errors, where the grassed waterway is partially mapped, or terraces are missed (Fig 8a). In fact, as suggested in the confusion matrix, the omission of SCP area is the main error source in this mapping and can lead to certain underestimation of SCP areas. In another way, the commission error is also observed in the regions where the crop surfaces are visually complex and quite heterogeneous. We observed that wetter areas present noise mislabels of SCPs in the crop field. Such errors are expected as the model performance is dependent on clear distinction of spatial pattern between SCP and non-SCP targets. In the same context, it should be emphasized that irrigated areas are not common practices across Iowa, which make it poorly represented in the
training samples. As consequence, we found a critical confusion of SCP areas in the central pivot irrigation areas where the pivot wheel path or edges of irrigated areas are confused as conservation practice (Fig. 4.8b). Lastly, the flooding farmlands present complex water paths, and some errors were observed in North Dakota and Wisconsin. Despite these aspects, this preliminary analysis shows sufficient quality for overview analysis of SCP distribution across the U.S. Midwest.

![Figure 4-8. Omission and commission errors.](image)

**Overview of SCP distribution in the Midwest U.S.**

The spatial distribution of grid-based SCP areas is shown in Figure 4.9, and the percentage of total area is summarized per state in the chart. In general, a total of 6,642 km² was classified as SCP areas in the cropland across the entire region. From this product, we observe that 52% of SCP areas are distributed over Iowa (26%), Illinois (15%) and Nebraska (11%). The states with lowest SCP areas are the North Dakota and Michigan, with 4% of total. Regarding the spatial distribution, the variability of SCP areas indicates different conservation needs and efforts across the states. For example, the largest extent is clearly observed in the western and eastern Iowa, while the north Iowa (Des Moines Lobe region) remain with lower SCP areas. The state of Illinois shows quite
similar distribution of SCP areas, with mean of 0.67 km$^2$ (± 0.44) per 100 km$^2$. By visual inspection, we observed that terraces are the large number of classified SCPs, which highlights the conservation needs in this state. Figure 4.10 shows the absolute SCP areas per state and the number of funded terraces and grassed waterways by NRCS conservation programs between 2005 and 2019. In agreement, the largest SCP areas and practice counts are both observed in Iowa, which might indicate the importance of these programs towards conservation. Interestingly, we observed several terraces across Kansas but there are no records of NRCS funding for this practice. The small SCP areas in North Dakota and Michigan also agrees with low number of NRCS records for terrace/grassed waterways projects. This overview of classified SCP and NRCS records is relevant to discuss the effectiveness of cost-share programs. Table 4.3 shows the counties with highest SCP areas in this mapping, and Iowa, Kansas and Nebraska present the most counties. This information is useful to understand the regions with consolidated SCPs, and the conservation studies can be conducted in successful cases to understand the farmer’s motivation. Lastly, note that some areas were not considered because they do not present consolidated corn and soybean areas (see cropland mask in Fig. 4.9). In next section, these results were normalized by crop areas to give more insights about SCP distribution and its percentage in the croplands.
Figure 4-9. Spatial distribution of agricultural SCP areas in the Midwest U.S. region. The proportion of SCP areas per state is represented in the top-right chart. Note that SCP areas are calculated within grid area (10 x 10 km). The cropland mask is presented in the bottom-right map.

Figure 4-10. Comparison of SCP areas (km2) among the states. The yellow (grassed waterways) and red (terraces) bars represent the number of practices implemented with support of conservation programs. NRCS Source: [https://www.nrcs.usda.gov/Internet/NRCS_RCA/reports/](https://www.nrcs.usda.gov/Internet/NRCS_RCA/reports/).
Table 4-3. Top 15 Midwest counties with the largest SCP areas.

<table>
<thead>
<tr>
<th>State</th>
<th>County</th>
<th>County area (km²)</th>
<th>SCP area (km²)</th>
<th>Cropland area (km²)</th>
<th>Coordinates (Lat, Lon)</th>
</tr>
</thead>
<tbody>
<tr>
<td>IA</td>
<td>Pottawattamie</td>
<td>2468.8</td>
<td>55.1</td>
<td>1725.9</td>
<td>41.34 N, 95.54 W</td>
</tr>
<tr>
<td>KS</td>
<td>Brown</td>
<td>1489.6</td>
<td>48.6</td>
<td>978.3</td>
<td>39.82 N, 96.52 W</td>
</tr>
<tr>
<td>KS</td>
<td>Nemaha</td>
<td>1882.2</td>
<td>42.1</td>
<td>890.6</td>
<td>39.78 N, 96.01 W</td>
</tr>
<tr>
<td>NE</td>
<td>Gage</td>
<td>2272.4</td>
<td>40.5</td>
<td>1396.5</td>
<td>40.26 N, 96.68 W</td>
</tr>
<tr>
<td>KS</td>
<td>Marshall</td>
<td>2326.5</td>
<td>40.4</td>
<td>1084.2</td>
<td>39.78 N, 96.52 W</td>
</tr>
<tr>
<td>IA</td>
<td>Shelby</td>
<td>1499.4</td>
<td>38.6</td>
<td>1250.2</td>
<td>41.68 N, 95.32 W</td>
</tr>
<tr>
<td>IA</td>
<td>Benton</td>
<td>1877.8</td>
<td>38.0</td>
<td>1368.4</td>
<td>42.07 N, 92.07 W</td>
</tr>
<tr>
<td>IA</td>
<td>Plymouth</td>
<td>2230.0</td>
<td>37.2</td>
<td>1659.9</td>
<td>42.74 N, 96.22 W</td>
</tr>
<tr>
<td>IA</td>
<td>Tama</td>
<td>1872.3</td>
<td>36.1</td>
<td>1221.9</td>
<td>42.08 N, 92.54 W</td>
</tr>
<tr>
<td>IA</td>
<td>Crawford</td>
<td>1858.9</td>
<td>34.2</td>
<td>1433.3</td>
<td>42.03 N, 95.39 W</td>
</tr>
<tr>
<td>IL</td>
<td>McLean</td>
<td>3041.2</td>
<td>33.5</td>
<td>2487.1</td>
<td>40.49 N, 88.85 W</td>
</tr>
<tr>
<td>MO</td>
<td>Saline</td>
<td>1982.2</td>
<td>32.6</td>
<td>1133.9</td>
<td>39.14 N, 95.56 W</td>
</tr>
<tr>
<td>IA</td>
<td>Adair</td>
<td>1465.0</td>
<td>32.3</td>
<td>950.9</td>
<td>41.33 N, 94.48 W</td>
</tr>
<tr>
<td>IA</td>
<td>Jasper</td>
<td>1886.3</td>
<td>30.9</td>
<td>1257.8</td>
<td>41.69 N, 93.06 W</td>
</tr>
<tr>
<td>IA</td>
<td>Poweshiek</td>
<td>1516.6</td>
<td>30.8</td>
<td>1011.3</td>
<td>41.69 N, 92.53 W</td>
</tr>
</tbody>
</table>

Spatial analysis with other variables

While absolute areas give the overview of SCP extent across the Midwest states, the number of cropland areas influences on the magnitude of these SCP areas. For this reason, we calculate the percentage of SCPs in cropland areas (Figure 4.11). Overall, the values range from 0 to 8% of SCP in croplands, and it highlights different areas compared to Figure 4.9. For instance, north Iowa or central Indiana show less than 1% of SCP areas in the croplands while north-east Kansas achieves the highest percentage in the entire region. Ohio shows more than 1% in the middle of state, but the rest of state has lower than 0.8% of SCP in croplands. Note that this result only shows grids with more than 10% of cropland areas. In addition, the cluster and outlier analysis is illustrated in the same figure. The Global Moran’s I shows a spatial autocorrelation of % SCP values (clustered, p-value < 0.05) and LISA results demonstrate that high-high clusters occur in large extent over Iowa, Kansas and Nebraska. As a hotspot region, the gridded SCP areas present high similarity with their neighbors, and average SCP percentage of 2.65% (± 0.73) is observed in the high-high cluster. As observed, North Dakota, Minnesota and Michigan present most areas of
low-low clusters, which indicate less efforts in the SCP implementation. By exploring other soil and topographic variables with this cluster map, we can understand the farmer’s motivations to adopt these practices. Figure 4.12 shows the histogram of values extracted from high-high and low-low clusters from Anselin Local Moran’s I analysis. In general, high-high clusters of % SCP are observed in moderate slope (1.58%), low TWI (4.45) and Ksat (6.68 µm/s) compared to low-low clusters with slope of 0.75%, TWI of 5.07 and Ksat of 19.8 µm/s. Also, there are slight differences in the k-factor, bulk density and SOM values between high-high and low-low clusters. While these results suggest some relationship of these soil/topographic variables with SCP area, it should be highlighted that spatial correlation analysis of these variables and % SCP shows no statistical significance using all grids. As expected, these variables are only part of the motivation for SCP implementation and other factors are also important on the SCP adoption, such as farm profitability, knowledge, tools and management. Also, this mapping is only focused on vegetative/structural practices, but there are many practices that are able to be implemented in regions with potential erosion problems as well.

Figure 4-11. Percentage of SCP areas in the croplands. The result of Anselin Local Moran’s I is presented in the top-right map, where the cluster types are defined at 5% significance level. Only grid values with more than 10% of cropland areas are used in this cluster analysis.
Figure 4-12. Distribution of the soil and elevation-related properties in the high-high and low-low clusters of SCPs. Distribution of the soil and elevation-related properties in the high-high and low-low clusters of SCPs. (a) topographic wetness index, (b) slope, (c) saturated hydraulic conductivity, (d) soil erodibility factor, (e) bulk density, and (f) soil organic matter. The brief description of these variables is described in the Section 2.5. The cluster map is presented in the Figure 4.11.
When one compares the spatial pattern of potential soil erosion and % SCP values in Fig. 4.13, the conservation efforts by farmers in high soil risk areas are evident. The histogram also shows the higher erosion values (average: 8.82 tons/ha/yr) in the high-high cluster of SCP areas compared to low-low cluster (average: 3.34 tons/ha/yr). Since this result highlights certain spatial overlap of high erosion and SCP areas, an interesting reflection emerges in the context. The soil erosion modeling indicates the potential erosion according to climate-topographic conditions of the region. In contrast, the classified SCP areas illustrates the action of farmers by adopting vegetative/structural practices to minimize the erosion impacts. Since this erosion modeling product does not incorporate the spatially explicit information of support practice factor (P-factor), we can assume that this erosion map shows the “potential” erosion and the real erosion rate might be different when we incorporate the conservation practices. Another insight on this result is the assessment of counties with conservation needs (↑ soil erosion, ↓ SCP areas), such as Cedar/Nebraska, Crawford/ Iowa, Shelby/Iowa, and Seward/Nebraska.

Figure 4.13. Comparison of soil erosion rates in the high-high/low-low clusters of % SCP values. Soil Erosion 2012 map is from Borrelli et al. (2017).
Discussion

Mapping of structural conservation practices

This study performed the first automated mapping of SCP areas in Midwest U.S. croplands using NAIP aerial imagery (2018-2019). So far, there is no study that provides a spatially explicit mapping of these practices for the entire region, and the results give insights about conservation efforts from farmers and landowners. The results showed the suitability of adapted U-Net algorithm (Fig. 4.6 and 4.7), and the mapping achieved the overall accuracy of 76.8% across 20 counties (Table 4.2). In general, the spatial distribution reveals some patterns in the SCP extent among Midwest states (Figure 4.9). For instance, our findings show that southwest Iowa, northeast Kansas and east Nebraska present the highest percentage of SCPs per cropland areas, while central Iowa (Des Moines Lobe) presents an intense agricultural activity with low percentage of SCP areas (< 1% of SCP in cropland). Further, we observed that Illinois presents near-similar magnitude of SCP areas in absolute terms. In the visual inspection, we observed that some regions have a high number of terraces in the croplands such as Kansas. In contrast, the grassed waterways were vastly classified in the Iowa and Illinois. Historically, terraces and grasslands are well-recognized for reducing runoff and sediment delivery from agricultural areas (Fiener and Auerswald, 2003; Tarolli et al., 2014), and this final product allowed the quantification of these two practices across these states.

Regarding the spatial pattern (Fig. 4.11), the results show slightly difference between soil and topographic-related variables between high-high and low-low cluster of SCP areas, such as slope and saturated hydraulic conductivity (Fig. 4.12). These results suggest that regional landscape characteristics and risk perceptions can potentially influence on the farmer actions. Several studies have performed meta-analysis to understand the motivations and barriers for farmer’s adoption of BMPs (Prokopy et al., 2008; Jackson-Smith et al., 2010; Baumgart-Getz et
al., 2012). Among many, the local network, conservation adoption by neighbors, and geophysical characteristics of the land (soil properties, slope) are listed as potential factors (see review in Liu et al. (2018)). Likewise, Baumgart-Getz et al. (2012) showed that access to information, financial capacity, and being connected to agency or watershed groups have impact on farmer motivation. In this topic, the evaluation of existing practices and social surveys might support the understanding of farmer needs and attitudes in support to conservation practices.

**Implications for conservation programs**

As parts of the U.S. Farm Bills, conservation programs have broadly encouraged farmers to adopt BMPs (Reimer, 2015) through directives as Environmental Quality Incentives Program, and Conservation Stewardship Program. These financial and technical assistance programs have become an essential mechanism for promoting conservation among farmers and rural landowners (Reimer and Prokopy, 2014). According to Resources Conservation Act report (RCA, 2020), Iowa NRCS programs have supported the adoption of 32,674 terraces and 38,877 grassed waterways between 2005 and 2019. Until now, this information is primarily available on historical records (tabular) or local\regional projects such as Iowa BMP project. Although formal conservation program records are useful for general insights (Figure 4.10), they are typically limited in the monitoring of successful degree of SCP implementation. Jackson-Smith et al. (2010) mentioned the need of efficient tracking system for monitoring the funded contracts after implementation of SCPs, which may help a better understanding of long-term consequences of these cost-share programs. In this context, the quantitative assessment of SCP areas can have positive implications for these programs. For example, conservation program managers can use this geospatial inventory to follow-up the farmer’s actions and the consolidation of sponsored projects. In addition, these results are useful to advertise the positive environmental outcomes achieved by these programs.
Another benefit is the evaluation of SCP distribution to support the decision of future projects (Figure 4.11). By understanding the location of current practices and conservation needs, new contracts and priority areas can be determined. In these efforts, the application of ACPF tool becomes essential to identify the preferential locations for certain practices (Tomer et al., 2013). However, Rundhaug (2018) showed at least 78% of potential grassed waterways from ACPF results were already implemented in three Iowa watersheds. This is a positive validation for ACPF method, but it also showed the importance of integration of existing practices to improve the watershed conservation plan prior to implementation of project. With long-term environmental goals, the combination of all these factors (conservation plan, financial incentives, information) can influence on farmers’ actions (Carlisle, 2016).

**Implications for soil erosion and runoff modeling**

The structural conservation practices are directly related to efforts on soil erosion control and reduction of nutrient loss from agricultural lands (Xiong et al., 2018), and this mapping allows further discussion of soil-related issues. In general, our results show that most counties with high erosion-prone areas have large extent of SCP areas (Fig 13), suggesting the conservation efforts have paid off with implementation by farmers. Previous studies that have shown high soil erosion rates in the Midwest U.S. region (Doetterl et al., 2012; Borrelli et al., 2017; Tan et al., 2020) have poorly considered conservation practice P factor into soil erosion estimates (Xiong et al., 2018). As discussed by Panagos et al. (2015) and Sartori et al. (2019), P-factor is one of the most uncertain and difficult pieces of information to access. Naipal et al. (2015) recognized the importance of P-factor in local variation of soil erosion but they did not consider the P-factor due to data limitation on a global scale. More recently, Xiong et al. (2019) stated that P-factor datasets are relevant to improve soil erosion modeling and advanced image processing techniques should be considered
to fill the knowledge gap for large-area projects. While additional processing is required to convert this mapping product into P-factor values (Wang et al., 2016), the proposed mapping of SCPs brings a new opportunity for further improvements of soil erosion maps and risk scenarios promoted by agricultural activities in the Midwest region. In a similar context, accurate modeling of surface management practices as well as surface runoff volumes and peak runoff rates influences erosion and entrainment of pollutants, which enables modeling of the load reduction goals of a pollutant to meet water quality standards. Thus sub-field-level analysis of SCPs can potentially improve the runoff estimates as they consider the practices affecting this process (Lee et al., 2010).

This study has no intention to explore these modeling aspects, but this discussion illustrates potential benefits and applications of the SCP product.

**Advantages and limitations of this framework**

As discussed in other studies (Garcia-Garcia et al., 2018; Guo et al., 2018), the successful application of semantic segmentation methods requires computational resources. The implemented U-Net model has a deeper architecture and large input size (256 x 256 pixels), and the training process took 13 days with large GPU memory (32 GB). Users/developers should be aware of these requirements when they decide to implement semantic segmentation method. In addition, the ability to quantify SCP areas using a regional-scale algorithm is also a challenge, and a more general model is required. The high number of SCP areas across Iowa introduces the variability of SCP conditions (orientation, size, and shapes) in the training dataset, and the results show that 500,000 samples give such generalization for this application. Conceptually, the mapping presented in this study assumes that spatial-spectral patterns of SCP areas are identifiable for model interpretation and learning. Once the model understands these patterns in the training data, we can predict new areas with adequate spatial resolution. Following that, large-area application
at 2-m resolution presents high spatial detail and heterogenous landscape and we observed that terraces and grassed waterways can be confused with other features. For instance, false-positive areas were observed in the irrigated systems, wetter areas, and cropland with non-homogeneous surface (Fig. 4.8). Recognizing these limitations, our findings should be interpreted with caution at farm-level because we did not conduct a manual editing of this product, and the visual inspection is recommended when applied for specific farmland.

In addition, this study presents a binary map of SCP areas for overview distribution in the entire region, but some researchers can be more interested in multi-class records. This semantic segmentation with three false-color bands does not allow accurate distinction of SCP types since the spectral/spatial features are quite similar in some cases, such as grassed terraces versus narrow waterways. Also, users should notice that other practices than terraces and grassed waterways were eventually classified in this product, such as filter strips and riparian buffer zone. As mentioned, SCP classification requires high spatial resolution data for target identification, and the data availability imposes limitations for further improvements in this large-area application. For example, a potential improvement (not proven) in this classification is the addition of topographic variables such as hillshade. However, high-resolution elevation models are only available for some states, and they have different quality protocols, vertical accuracy and time acquisition. These aspects play a crucial role to achieve reliable and comparable results among states, and our classification framework does not include these LiDAR-derived data to avoid potential bias in the overall analysis. Beyond that, improvements to the current study could be explored with the extension of this methodology by including new labeled samples from other states, evaluation of other deep learning methods, post-processing with manual editing, and classification strategies for statewide projects with high-resolution LiDAR data. While this discussion highlights aspects to be
improved, this study was able to illustrate a promising framework for operational monitoring of SCPs by providing the overview of SCP distribution over Midwest United States.

**Conclusion**

This study presented the first mapping of structural conservation practice over Midwest United States croplands. Results indicate the capability of this framework for regional monitoring of SCP areas using adapted U-Net model and NAIP imagery (2018-2019). In general, SCP mapping achieved 76.8% of overall accuracy, and presents fine-detail mapping across states. The results show large variability in SCP occurrence among Midwest states, with high occurrence in Iowa, Illinois and Nebraska. In contrast, Michigan, North Dakota and Minnesota present the lowest percentage of SCP in their croplands. Overall, we observe that cluster of SCP areas are associated with certain soil and terrain conditions: high-high cluster of SCPs were observed in the slope higher than ~1.5 %, bulk density lower than 1.4 g/cm³, and Ksat lower than 7 µm/s. Our findings also show the agreement in spatial pattern of SCP areas and high erosion-prone areas, which documents farmer efforts towards soil conservation. Although preliminary analysis shows the applicability of this framework, the farm-level analysis requires some cautions since there are known source of errors such as discontinuity or mislabeling of SCP areas. Finally, a spatially explicit inventory of SCP areas is useful for a variety of scientific and policy applications such as (i) the understanding of farmer’s contribution for soil and water conservation, (ii) the improvement of soil erosion risk modeling by considering structural conservation practices, and (iii) the evaluation of priority areas by national programs. Considering the importance of soil and water conservation in agricultural areas, this new product becomes a useful for further assessment of these practices in the Midwest U.S. croplands.
Acknowledgements

We thank the USDA's Farm Service Agency (FSA) for publicly NAIP imagery, and other USDA resources for accomplishment of this study. The national cropland product is available at https://nassgeodata.gmu.edu/CropScape/. We thank the California Soil Resource Lab for the availability of soil products (https://casoilresource.lawr.ucdavis.edu/). We also thank the JRC\ESDAC for the availability of Global Soil Erosion map from Borrelli et al. (2017). Also, Iowa BMP project was a key resource for this study (https://www.gis.iastate.edu/gisf/projects/conservation-practices), and we thank Robin McNeely for sharing the geodatabase. This study has benefitted from usage of ISU high performance computing resources (https://www.hpc.iastate.edu/).

References


Rabalais, N. N., Turner, R. E., & Scavia, D. (2002). Beyond Science into Policy: Gulf of Mexico Hypoxia and the Mississippi River: Nutrient policy development for the Mississippi River watershed reflects the accumulated scientific evidence that the increase in nitrogen loading is the primary factor in the worsening of hypoxia in the northern Gulf of Mexico. BioScience, 52(2), 129-142.


CHAPTER 5. GENERAL CONCLUSION

In this dissertation, deep learning methods were explored for land cover mapping and environmental analysis, such as object-based convolutional neural network (chapter 1), deep neural network (chapter 2) and (iii) fully-convolutional network (chapter 3). While a detailed conclusion is presented in each chapter, this section is focused on the research questions listed in Chapter 1:

Chapter 2. What are the benefits of multiscale object-based CNN approach for large-area land cover classification? How relevant is the integration of multiscale CNN models? Is the large reference dataset important for CNN training?

In general, the proposed multi-OCNN method integrates object analysis (skeletonize + window selection) and multiscale CNNs for land cover classification. This framework gives the flexibility to select the most appropriate input window for each convolutional location for any land target, while it also reduces the number of CNN predictions during the classification. The benefits of multi-OCNN approach are illustrated by finely-detailed mapping (preservation of boundaries) and fast processing of county-based images compared to standard CNNs. Our findings show that the multi-OCNN method presents the highest accuracy compared to fixed-CNN (single model) or fixed-OCNN. When we are working on regional classification, object analysis can reduce the number of CNN predictions, increase the speed, and improve the overall accuracy (Table 4.5, 4.6 and 4.7). Furthermore, the study illustrates the importance of multiscale models when the classification is performed in heterogenous landscape where the size and shape of targets vary in terms of magnitude and complexity. The multiscale CNN approach increases the agreement between input patch and observed target, but this is only possible with object-based classification
using further analysis of object geometry. Remote sensing users can benefit from this content to understand the challenges of traditional CNN and the potential strategy using object-based classification for better results. In addition, this study presents a new reference dataset with 1 million patches for 10 land cover classes. Several CNN experiments have been conducted with existing datasets (UCMerced Land-use, SIRI-WHU, WHU-RS19, SAT-6 and PatternNet), and high-resolution NAIP data are freely available across the U.S., and our reference dataset represents an opportunity for more experiments using CNNs, such as land cover products. The results shown in Figure 4.8 illustrated the model generalization with similar accuracy between training and testing datasets for 900,000 samples. Thus, this chapter also provides a 1-million patches as reference dataset and this publicly available dataset represents a new opportunity for another land cover methods using CNNs. Future recommendation: The NAIP program is a national and long-term records of high-resolution aerial data and the recommendation is the implementation of this multi-OCNN for land cover classification of other states.

Chapter 3. Is the deep neural network model appropriate for mapping of small wetlands using WorldView-3 image? What is the benefit of DNN application compared to other machine learning methods? What is the impact of spatial resolution to wetland mapping?

The results in Chapter 3 show that classification performance achieved 93.3% accuracy and high spatial overlap between reference and classified wetlands polygons (J-index ~ 0.801). The quality of results confirms the applicability of Worldview-3 and DNN for wetland mapping, primarily due to capturing small wetland areas. DNN performance was near-similar in comparison to other machine learning methods. This result suggests that efficient feature selection simplifies the classification problem (see input distribution on Fig. 3.8), and model limitations become less
evident in the results. However, we emphasized that the DNN algorithm offers a great flexibility in the architecture design (number of layers, nodes, activation functions and other) and this can be useful to improve classification performance when other machine learning methods may not work well. Lastly, the spatial resolution is one of the most relevant point for this study area where the classification accuracy (overlap area) can reduce from 90.36% (1.2 m) to 61.22% (30-m) as we decrease the spatial resolution (1.2 m → 30 m). The appropriate use of satellite data thus has great implications for mapping capabilities, especially for small wetlands. *Future recommendation:* The application of this framework (DNN + WV-3 + feature selection) is suggested in other areas using multiple dates.

Chapter 4. *Is the semantic segmentation algorithm (U-Net) useful for SCP classification? What are the benefits and limitations of this methodology? What is the spatial distribution of structural conservation practices across Midwest U.S. croplands?*

In the chapter 4, the semantic segmentation method was proven to be useful for SCP classification, achieving finely-detailed mapping of these targets. The advantage of this method is the exploration of spatial patterns to support the identification of SCPs, which overcomes the limitation of three spectral bands in such heterogeneous landscapes. However, computational power is highly demanded for implementation of this study, including parallel programming and GPUs. The limitation of this framework is mostly related to the nature of the optical data itself and the addition of high-resolution elevation product (e.g., hillshade) can potentially improve the mapping accuracy, but high resolution elevation datasets are limited in terms of spatial and temporal coverage across the Midwest U.S states. Regarding the SCP product, the results show a large variability of SCP areas across the entire region, and high extent of SCPs was observed in
Iowa, Illinois and Nebraska. Our findings show that these states present \( \sim 52\% \) of classified SCP areas and the largest number of these practices is observed in southwestern Iowa. In contrast, some states present lower records of SCP areas such as North Dakota and Michigan. As mentioned, the implementation of SCPs is motivated by a variety of factors, such as soil and topographic conditions. Our findings suggest high cluster of SCP (\%) over regions with moderate slope and low saturate hydraulic conductivity. Interestingly, we also observed that SCP areas are typically co-located with high soil erosion rates. These main results support further discussion about the conservation practices in the Midwest croplands, and this geospatial information is useful to define priority areas for incentives and projects conducted by conservation programs. *Future recommendation:* The main improvement in this study is the addition of high-resolution LiDAR-derived metrics (e.g., hillshade) to develop a multi-class product.