MountainScape Semantic Segmentation of Historical and Repeat Images

by

Aniket Mahindrakar
B.Tech, Jawaharlal Nehru Technological University, 2019

A Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

in the Department of Computer Science

We acknowledge and respect the Lək^{w}əŋən (Songhees and X^{w}sepsəm/Esquimalt) Peoples on whose territory the university stands, and the Lək^{w}əŋən and W̱SÁNEĆ Peoples whose historical relationships with the land continue to this day.

MountainScape Semantic Segmentation of Historical and Repeat Images

by

Aniket Mahindrakar
B.Tech, Jawaharlal Nehru Technological University, 2019

Supervisory Committee

Dr. George Tzanetakis, Co-supervisor
(Department of Computer Science)

Dr. Eric Higgs, Co-supervisor
(School of Environmental Studies)

ABSTRACT

Semantic segmentation of ultra-high resolution images is challenging due to high memory and computation requirements. Current approaches to this problem involve cropping the ultra-high resolution image into small patches for individual processing in order to provide local context, or under-sampling the images to provide global context, or following a combination of both which gives rise to global-local refinement pipelines. In this thesis, we present the MountainScape Segmentation Dataset (MS2D) which comprises high-resolution historic (grayscale) manually segmented images of Canadian mountain landscapes captured from 1861 to 1958 and their corresponding modern (colour) repeat images. Additionally, we analyze the characteristics of the dataset, define evaluation criteria, and provide a baseline to serve as a reference benchmark for automated land cover classification using the Python Landscape Classification Tool (PyLC), an existing software tool. The main contribution of this thesis is the experimental exploration of various deep learning architectures to address the tiling artifacts and spatial context loss faced by PyLC in its tile-based processing of ultra-high-resolution images, alongside a comprehensive investigation using a larger dataset than that employed in the original PyLC study to solve this tiling problem.

Contents

List of Tables

List of Figures

ACKNOWLEDGEMENTS

I sincerely thank my supervisors, Dr. George Tzanetakis and Dr. Eric Higgs, for granting me a research assistantship on the Mountain Legacy Project and for their unwavering support throughout my research. Their patience, motivation, and vast knowledge have guided me through every stage of this journey. Their mentorship has been invaluable in shaping my research and the writing of this thesis, and I am truly grateful for the opportunity to work under their guidance—this experience has been a dream come true.

I am also grateful to Claire Wright for her mentorship and to Ben Wright for his insightful feedback, constant encouragement, and support throughout my research.

My gratitude extends to fRI Research, Mitacs, and Social Sciences & Humanities Research Council for funding this research and to Intact Financial Corp. for the co-op opportunity, which provided me with practical experience relevant to my thesis. I appreciate the invaluable contributions of Sebastian Yerex, Elizabeth Arich, and Darcy Benham for their work on land cover annotations.

I would like to express my heartfelt thanks to my fellow lab members for making me feel at home and keeping me motivated. The camaraderie over the past two years has been truly special. I also appreciate Dr. Eric Higgs for providing space in the Higgs Lab during my M.Sc. and for involving me in both professional and social activities within the lab community.

None of this would have been possible without the dedicated efforts of the administrative staff at the University of Victoria and the Department of Computer Science. I sincerely thank them for their invaluable support throughout my M.Sc.

I would also like to thank my friend Reuben Sinha for being part of this incredible journey. Your unwavering support and presence outside the lab have been a constant source of motivation.

Finally, I am profoundly grateful to my grandfather, Purushottam Mahindrakar, my grandmother, Meena Mahindrakar, my father, Manish Mahindrakar, my mother, Vidya Mahindrakar, and my brother, Rohit Mahindrakar, for their unwavering moral and emotional support throughout my life. Special thanks to my parents for their steadfast belief in me, for prioritizing my education, and for encouraging me to pursue my M.Sc. at the University of Victoria. I am also deeply thankful to my uncle, Rahul Mahindrakar, and my aunt, Sayali Mahindrakar, for organizing a memorable trip to Japan, and to my aunt, Vandana Kakade, for her invaluable support during my toughest moments while working on this thesis.

Aniket Mahindrakar

DEDICATION

This thesis is dedicated to my beloved family and friends: my grandfather, Purushottam Mahindrakar; my grandmother, Meena Mahindrakar; my father, Manish Mahindrakar; my mother, Vidya Mahindrakar; my brother, Rohit Mahindrakar; and my dear friend, Reuben Sinha.

Chapter 1

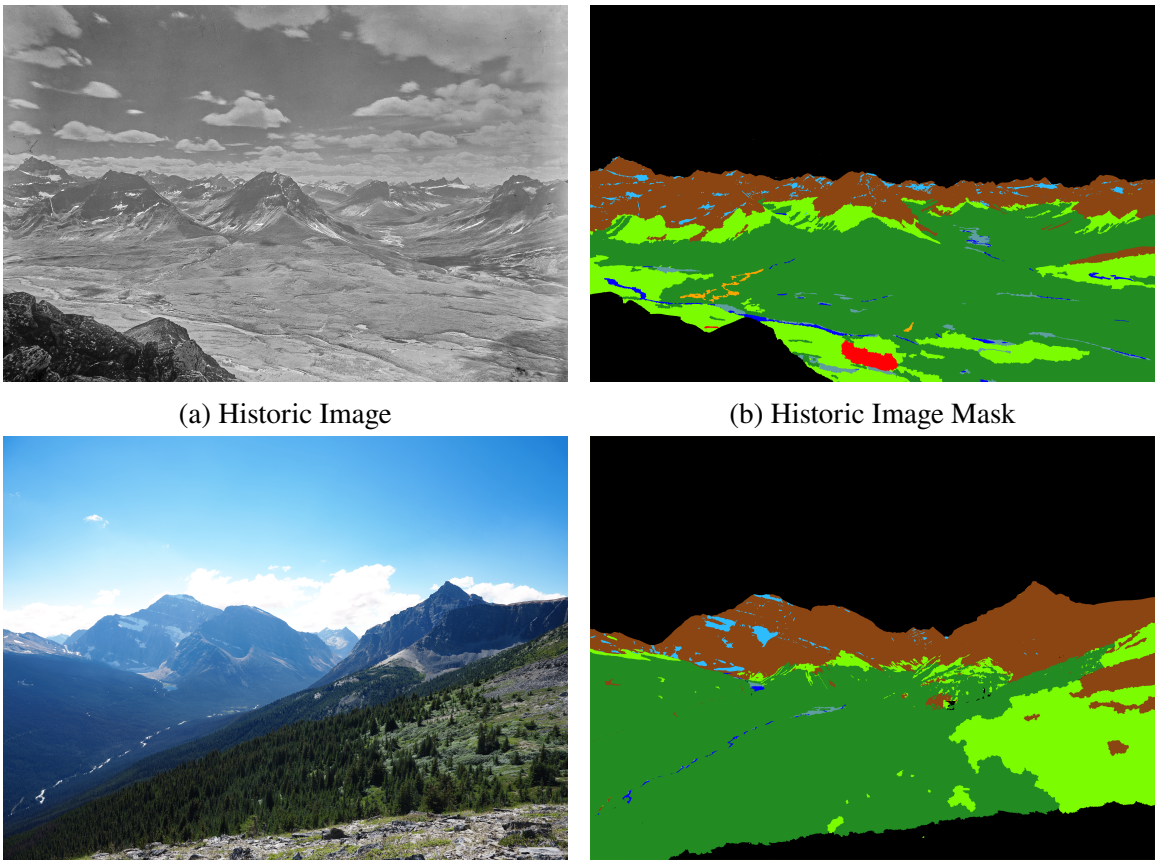
Introduction

1.1 Background

Recent advancements in photography and sensor technologies have enabled access to ultra-high resolution images with millions or even billions of pixels. Performing semantic segmentation on such images is challenging as it involves pixel-wise parsing of the images into different semantic categories. General segmentation methods cannot handle ultra-high resolution images due to memory and computational constraints. To handle large images, deep learning models typically downsample images or separately segment partitioned patches and merge their results into a high-resolution image, which can compromise segmentation performance due to loss of spatial details, boundary artifacts, or inconsistencies between patches. Lightweight models, on the other hand, perform poorly since their simple architecture struggles to capture long-range dependencies, such as relationships between distant regions in an image, and high-level semantic cues [1]. As a result, some specially designed ultra-high resolution segmentation architectures have been proposed that mainly follow the principle of global and local refinement [2, 3].

1.2 Dataset Overview

In this paper, we present a novel ultra-high resolution image dataset comprising of 284 image pairs of historic and repeat oblique images of Canadian mountain landscapes that have been manually segmented by field experts for performing land cover classification, as illustrated in Figure 1.1. Each pair of historic and repeat images have been manually aligned using control points to allow for change detection. The images are segmented into 8



(a) Historic Image

(b) Historic Image Mask

(c) Repeat Image

(d) Repeat Image Mask

Figure 1.1: Manually annotated images and masks of historical and repeat photographs of the Canadian Rocky Mountains, with color codes detailed in Table 1.1

broad categories of habitats as presented in Table 1.1 wherein ”Not Classified” is a special category that represents pixels not described by a land cover class, such as image artifacts or sky.

1.3 Problem Statement

The land cover classification problem that can be performed on the MountainScape Segmentation Dataset (MS2D) is defined as a multi-class segmentation task to detect areas of classes mentioned in Table 1.1 [4]. Each input is an oblique terrestrial image, taken at an angled perspective rather than a vertical or nadir view. The expected result is a land cover mask of same size in pixels as the input image, where the color of each pixel indicates its class label.

Category	Color
Coniferous Forest	
Rock / Barren Ground	
Water	
Recently Burned	
Herbaceous / Shrub	
Wetland	
Broadleaf / Mixedwood Forest	
Snow / Ice	
Not Classified	

Table 1.1: Habitat categories

1.4 Tiling Challenges in Python Landscape Classifier (PyLC)

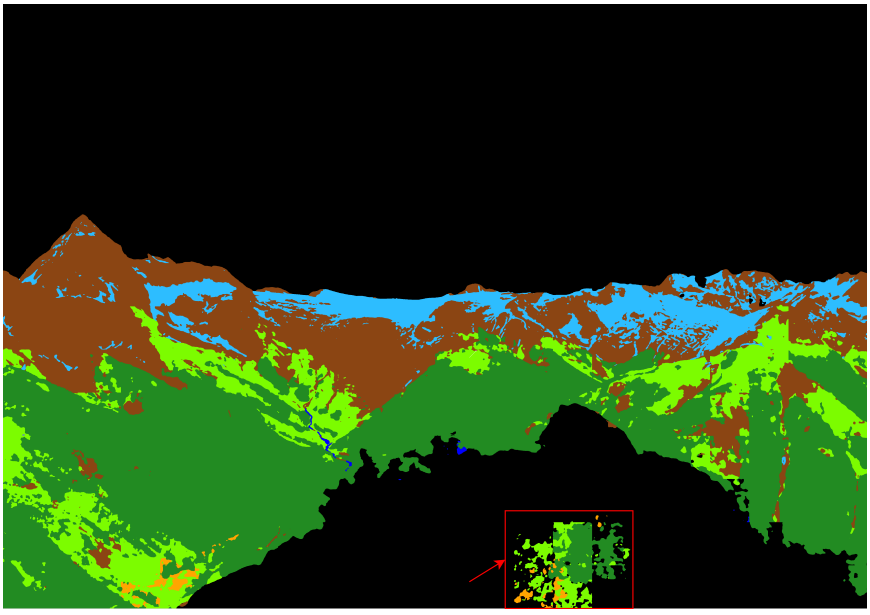


Figure 1.2: Tiling problem faced by PyLC

Python Landscape Classifier (PyLC) is a Pytorch-based trainable segmentation network for automated classification of land cover in oblique images. Implemented with DeepLabV3+,

PyLC builds on previous work by Rose [5]. During the segmentation process, PyLC faces a tiling problem, necessitated by the large, high-resolution images in the dataset, as illustrated in Figure 1.2. To fit the images into memory for deep learning models, they are divided into smaller tiles. However, this tiling approach can introduce several issues. The division of images can disrupt the spatial context, as relationships between landscape features that span multiple tiles may be lost. Additionally, tiling may create edge artifacts, where artificial boundaries at tile borders lead to inconsistencies in classification. Addressing these challenges is crucial for improving the robustness of PyLC's landcover classification.

1.5 Contributions

In this thesis, we present a reduction in tiling artifacts through an experimental exploration of various deep learning architectures tailored for ultra-high-resolution images, alongside a comprehensive investigation using the MountainScape Segmentation Dataset (MS2D), a dataset significantly larger than that used in the original PyLC study.

1.6 Structure

The remainder of this thesis is structured as follows. Related works are presented in Section 2. Section 3 explains the characteristics of images, details the annotation process, and introduces the division of training and test set. Section 4 describes the task of land cover classification in detail and proposes the evaluation metric used for the task. Section 5 provides an overview of the baseline approach, followed by the experiments conducted with various model architectures discussed in Section 6. Finally, Section 7 presents the conclusions.

Chapter 2

Background

This section outlines the key techniques used for landscape classification in repeat photography (the technique of photographing the same subject from the same or a similar vantage point at different times) [6], focusing on the challenges of oblique imagery. It reviews advancements from early manual methods to modern deep learning approaches, incorporates relevant benchmarks from orthogonal image classification such as the DeepGlobe 2018 challenge [4], and traces how these developments have contributed to tools like PyLC for automated land cover classification.

2.1 Introduction to MountainScape Segmentation Dataset (MS2D) and Repeat Photography

MS2D is based on the practice of repeat photography drawing on the Mountain Legacy Project (MLP) collection, the world's largest collection of high-resolution mountain images. In recent years, studies using repeat photography have developed various techniques to extract information from oblique images and assess changes between the photographed periods. Key to this process is the identification and/or classification of landcover details in the series of images. The majority of studies have relied on manual classification which is time consuming and generally limits the extent of the research [7]. This contrasts with landscape change studies using orthogonal imagery (satellite and aerial images) where automated classification is common [8]. Oblique image classification is complicated by the variation in scale within and between images (e.g. trees in near foreground look extremely different to those in far background, etc.), compared to nadir imagery (images taken from a directly overhead perspective) which tends to be taken at relatively consistent distance from

the subject. Progress in automated classification of oblique images is necessary to unlock the full potential of the images for landscape change analysis.

2.2 DeepGlobe 2018: Land Cover Classification Benchmark

While oblique image classification presents unique challenges, advances in land cover classification from orthogonal imagery offer useful methodological insights. One such milestone is the DeepGlobe 2018 challenge introduced by Demir et al. [4], which helped standardize the evaluation of deep learning models on high-resolution satellite imagery. The challenge included three tasks: road extraction, building detection, and land cover classification. Among these, the land cover classification task is most relevant to the present work.

The land cover classification track involved assigning one of seven classes—urban, agriculture, rangeland, forest, water, barren, and unknown—to each pixel in orthogonal imagery. The dataset comprised 1,146 training and 533 test images, each at 2448 × 2448 resolution, covering semi-rural and rural regions in Thailand, Indonesia, and India. Ground-truth labels were created from national land cover datasets and refined through expert annotations.

The task highlighted several key challenges: intra-class variability, ambiguous boundaries between land types, and a strong class imbalance—with agriculture dominating the dataset and classes like water and barren being sparsely represented. To address these, participants commonly adopted encoder-decoder architectures, such as U-Net and DeepLab, along with techniques like data augmentation, class-weighted loss functions, and multi-scale feature extraction.

Benchmark models using DeepLab with a ResNet backbone achieved a mean Intersection-over-Union (mIoU) of 0.433 on the validation set. Urban and agriculture classes achieved relatively high IoUs (0.6 and 0.5, respectively), while water and barren classes were much lower (0.2 – 0.3). These results highlighted the strengths and limitations of standard CNN-based approaches in handling complex landscape compositions.

Though the DeepGlobe dataset features nadir images unlike the oblique MS2D images, it offers a valuable benchmark. It demonstrates the potential of deep learning in automating land cover classification and underscores the need for specialized strategies when dealing with diverse terrain and imbalanced class distributions—challenges that are even more pronounced in oblique imagery.

2.3 Early Attempts and Techniques

In an early attempt, a subset of MS2D was introduced by Jean et al. [9], which included 60 manually annotated high-resolution image pairs of historic and repeat photographs from the Canadian Rocky Mountains. This paper also proposed a baseline algorithm using texture analysis and machine learning techniques for binary habitat segmentation. The study found that Histogram of Oriented Gradients (HOG) [10] descriptors performed better than Histograms of Local Binary Patterns (HLBP) [11] on historic images, while HLBP was more effective on repeat images, with an average Matthews Correlation Coefficient (MCC) [12] of 0.609 for repeat images and 0.430 for historic images in the binary classification of forest and non-forest categories.

The MCC is utilized here as a key metric to evaluate the binary classification accuracy of habitat segmentation in both historic and repeat images. MCC provides a balanced measure of performance for classifying forest and non-forest categories across the two texture-based descriptors. For a given class j (forest or non-forest), the MCC for each image i is calculated as:

$$\mathrm{MCC}_j = \frac{\sum_{i=1}^{n} \left(\mathrm{TP}_{ij} \times \mathrm{TN}_{ij} - \mathrm{FP}_{ij} \times \mathrm{FN}_{ij}\right)}{\sqrt{\left(\sum_{i=1}^{n} \left(\mathrm{TP}_{ij} + \mathrm{FP}_{ij}\right)\right)\left(\sum_{i=1}^{n} \left(\mathrm{TP}_{ij} + \mathrm{FN}_{ij}\right)\right)\left(\sum_{i=1}^{n} \left(\mathrm{TN}_{ij} + \mathrm{FP}_{ij}\right)\right)\left(\sum_{i=1}^{n} \left(\mathrm{TN}_{ij} + \mathrm{FN}_{ij}\right)\right)}} \tag{2.1}$$

where:

- TP_{ij} represents the count of true positives (pixels correctly identified as class j) for image i,
- TN_{ij} represents the count of true negatives (pixels correctly identified as not belonging to class j) for image i,
- FP_{ij} represents the count of false positives (pixels incorrectly identified as class j) for image i,
- FN_{ij} represents the count of false negatives (pixels of class j incorrectly identified as another class) for image i.

2.4 Deep Learning for Landscape Classification

Separately, a study by Buscombe et al. [12] evaluated the effectiveness of Deep Convolutional Neural Networks (DCNNs) in both image recognition and semantic segmentation of natural landscape images. It leveraged the lightweight and efficient MobileNetV2 DCNN framework for semantic segmentation tasks, showcasing its utility in processing smaller datasets rapidly. Additionally, the paper introduced a method for using structured prediction through a fully-connected Conditional Random Field (CRF) in a semi-supervised approach to efficiently generate ground truth labels and DCNN training datasets. The study culminated in proposing a hybrid approach for semantic segmentation that combined the DCNNs' capability to classify small image regions with the pixel-level precision of fully-connected CRFs, aiming for enhanced accuracy in performing segmentation.

Similarly, Bayr et al. [8] explored the application of a Convolutional Neural Network (CNN) for the automatic recognition of woody regrowth vegetation in repeat landscape photographs. Their research highlighted the utility of CNNs in processing large quantities of images, quantifying changes between repeat photographs, and identifying broader trends in landscape change.

Another study by Okamoto et al. [13] introduces an automated approach for converting time-lapse images of a Japanese alpine region into georeferenced vegetation classification maps. The authors address the challenge of classifying vegetation in time-lapse digital camera images, limited to three visible bands, by utilizing temporal variations in fall leaf colors. The proposed techniques proved effective, achieving a mean F1 score of 0.937 for vegetation classification.

2.5 PyLC Development

None of these approaches are fully applicable to the goal of multi-class image segmentation of the MS2D dataset for both colour and grayscale high-resolution images. To address this gap, Rose [5] developed the Python Landscape Classification Tool (PyLC). PyLC leverages recent advancements in DCNN to improve land cover classification in time-series remote-sensing data. Rose [5] evaluated the performance of two DCNNs, specifically U-net [14] and Deeplabv3+ [15], on historical and modern images from the MLP collection. A novel data augmentation method was introduced to address data limitations and class imbalance, achieving F1 scores of 0.839 and 0.909 for historic and repeat models, respectively.

2.6 PyLC Application and MIAS Integration

PyLC was subsequently applied by Tricker et al. [16] to extract landcover data from 19 images of the MLP collection. These data were processed using georeferencing tools in the Image Analysis Toolkit [17] and integrated to create a contemporary landcover map for the Athabasca valley in Jasper National Park, Canada, achieving a classification accuracy of 68% for the landcover map derived from oblique images. This workflow enhances the use of repeat photographs for obtaining quantitative landcover data. It offers several benefits including the ability to quickly and consistently classify images with minimal human intervention.

PyLC is now included in the Mountain Image Analysis Suite (MIAS), a plugin recently developed by Wright et al. for QGIS [18] that combines automated classification, image alignment/pixel mapping, and georeferencing tools to enable the efficient conversion of historical and repeat oblique images into landcover maps [19].

2.7 Conclusion

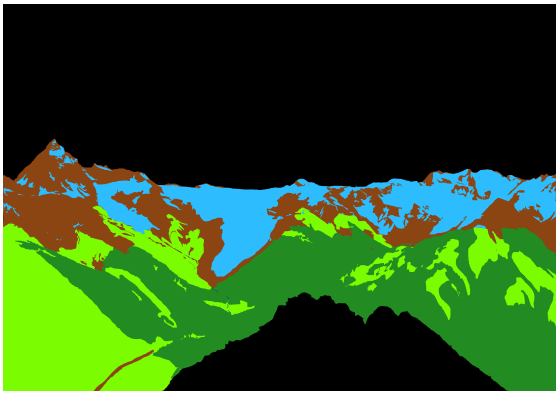
In summary, the landscape classification techniques have advanced significantly from manual methods to sophisticated deep learning approaches, particularly in the context of oblique imagery challenges. The development of tools like the Python Landscape Classification Tool (PyLC) marks a pivotal step in addressing the limitations of previous methodologies, enabling more accurate multi-class segmentation of the MS2D dataset. With its application in creating contemporary land cover maps and its integration into the Mountain Image Analysis Suite (MIAS), PyLC enhances the ability to leverage historical and repeat photography for monitoring landscape changes. These advancements lay a solid foundation for further research into various deep learning architectures for land cover classification.

Chapter 3

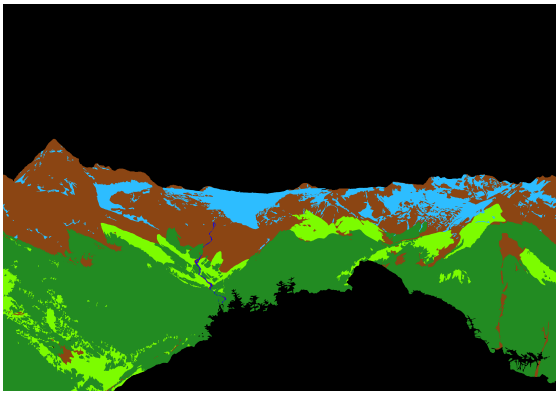
Dataset, Task and Evaluation Metrics

(a) Historic Image

(b) Repeat Image



(c) Manually Segmented Historic Image's Mask



(d) Manually Segmented Repeat Image's Mask

Figure 3.1: Sample MoutainScape Segmentation Image

In this section, we discuss the dataset and imagery characteristics, provide an explanation of the annotation methodology used to obtain training labels, formally define the land cover classification task, and explain the evaluation metrics in terms of their computation

and implementation.

3.1 Dataset

3.1.1 Overview

The MS2D dataset comprises 284 manually annotated ultra-high resolution oblique terrestrial images, divided into historical images (144) and repeat images (140). The historical and repeat images are split into train and test sets with 9 test images each. The image resolutions range from 16 to 80 megapixels (one million pixels) and the corresponding masks are RGB images with 8 landcover classes. The pixel-wise class distribution of the dataset for both historical and repeat images is available in Tables 3.1 and 3.2. The range in image size presents an additional challenge for classification compared to datasets with standardized image dimensions. The MS2D landcover classification scheme, as detailed in Table 1.1, was developed based on visual assessment of the underlying dataset, previous manual segmentation work [9, 20–22], and landcover classifications from other remote sensing products for similar landscapes [23].

Category	**Train (millions)**	**Test (millions)**	**Total (millions)**	**Proportion (%)**
Coniferous Forest	545.44	37.61	583.05	19.90
Rock / Barren Ground	312.08	18.98	331.06	11.30
Water	14.89	0.46	15.35	0.52
Recently Burned	59.66	3.77	63.43	2.16
Herbaceous / Shrub	351.17	20.75	371.92	12.69
Wetland	28.12	3.06	31.18	1.06
Broadleaf / Mixedwood Forest	5.79	0.04	5.83	0.20
Snow / Ice	50.53	3.57	54.10	1.85
Not Classified	1,372.59	101.50	1,474.09	50.31

Table 3.1: Class distributions for historical images

3.1.2 Landcover Classes

- *Coniferous Forest:* Areas with more than 6% tree cover and where coniferous trees make up over 75% of the vegetation.
- *Rock/Barren Ground:* Areas with less than 6% vegetation cover, consisting mainly of soil, sand, gravel, or rock.

Category	Train (millions)	Test (millions)	Total (millions)	Proportion (%)
Coniferous Forest	762.52	42.80	805.32	28.76
Rock / Barren Ground	303.11	15.94	319.05	11.39
Water	13.25	0.22	13.47	0.48
Recently Burned	22.47	1.12	23.60	0.84
Herbaceous / Shrub	168.38	13.72	182.10	6.50
Wetland	24.80	2.08	26.88	0.96
Broadleaf / Mixedwood Forest	16.91	0.94	17.86	0.64
Snow / Ice	16.95	1.51	18.46	0.66
Not Classified	1295.44	98.03	1393.47	49.76

Table 3.2: Class distributions for repeat images

- *Water:* Areas covered by standing or flowing water, such as rivers, lakes, and reservoirs.
- *Recently Burned:* Areas showing clear evidence of burned timber on the landscape.
- *Herbaceous/Shrub:* Areas with more than 6% vegetation cover but less than 6% tree cover, characterized by 'dry' or 'mesic' moisture conditions.
- *Wetland:* Areas with more than 6% vegetation cover and less than 6% tree cover, typically exhibiting 'wet' or 'aquatic' moisture conditions.
- *Broadleaf/Mixed Wood:* Areas with more than 6% tree cover, with broadleaf trees comprising more than 25% of the vegetation.
- *Snow/Ice:* Areas covered by visible snow or glacial ice, typically devoid of vegetation.
- *Not Classified:* Sections not included in the classification, usually comprising foreground elements, the sky, or image imperfections.

3.1.3 Annotation Process

The pixel-wise segmentation masks were annotated by experts knowledgeable about the landscapes described in the images. A custom MLP software facilitated the manual classification of the first set of oblique images [20]. Subsequent manual segmentation occurred in the photo editing software Affinity Photo. Example masks are demonstrated in Figure 1.1.

3.2 Land Cover Classification

Land cover classification is a problem in remote sensing and computer vision that involves segmenting an image into multiple regions, each representing a specific land cover type. Semantic segmentation aims to classify each pixel in an image into a category, where the goal is to assign each pixel to one of the pre-defined classes [24]. This segmentation task is evaluated based on the accuracy of the predicted class labels compared to the manually segmented class labels.

An oblique terrestrial image serves as input and the expected solution is a pixel-wise classification of the image into land cover categories. For visual assessment, this classification can then be represented as an RGB mask of the input image, with the same pixel dimensions as the image, where each pixel's colour represents a class designation as specified in Table 1.1.

3.3 Evaluation Metrics

We use three measures to evaluate the success of classification, namely: F1 Score, Intersection over Union (IoU), and weighted IoU. Assuming there are n images, for a given class j the metrics are defined as:

$$F1Score_j = \frac{2 * \sum_{i=1}^{n} TP_{ij}}{2 * \sum_{i=1}^{n} TP_{ij} + \sum_{i=1}^{n} FP_{ij} + \sum_{i=1}^{n} FN_{ij}} \tag{3.1}$$

$$IoU_j = \frac{\sum_{i=1}^{n} TP_{ij}}{\sum_{i=1}^{n} TP_{ij} + \sum_{i=1}^{n} FP_{ij} + \sum_{i=1}^{n} FN_{ij}} \tag{3.2}$$

where TP_{ij} represents the count of pixels in image i that are correctly identified as class j, FP_{ij} represents the count of pixels in image i that are incorrectly identified as class j, and FN_{ij} represents the count of pixels with true label j that are incorrectly identified as any class other than j. Assuming there are k land cover classes, the final score is defined as the weighted average IoU among all classes.

$$WeightedIoU = \sum_{j=1}^{k} w_j * IoU_j \tag{3.3}$$

where,

$$w_j = \frac{Number of pixels in class j}{Total number of pixels in manually segmented image} \tag{3.4}$$

Note that there is a ”Not Classified” class which is excluded from the evaluation (i.e., predictions for these pixels are not included in the calculation and thus do not influence the final score).

Chapter 4

Baseline

In this section, we establish baseline accuracy metrics for oblique terrestrial images using the MS2D dataset with PyLC.

4.1 Overview of PyLC

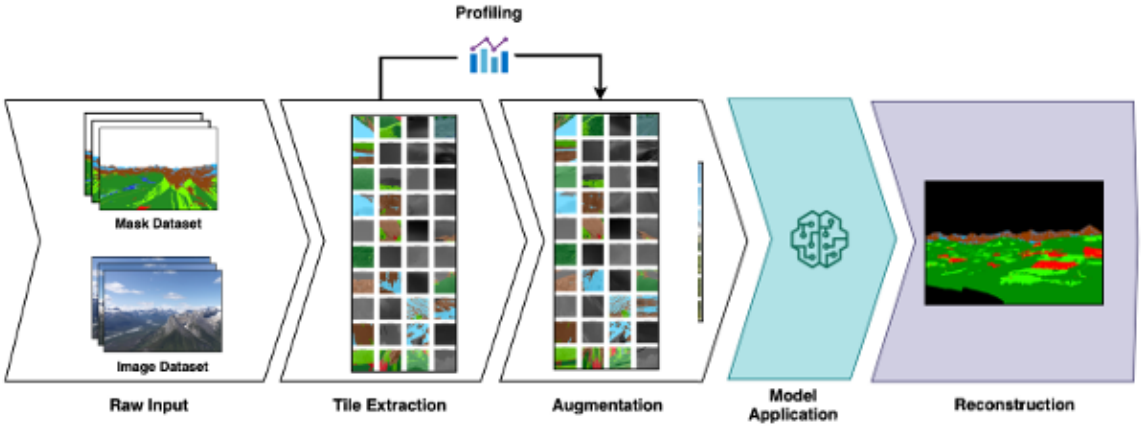


Figure 4.1: PyLC Model Architecture

PyLC is a trainable segmentation network that automates the classification of land cover types in grayscale and color oblique photographs [5], as shown in Figure 4.1. PyLC is built on the DeepLabV3+ [15] architecture and processes ultra-high-resolution landscape images by dividing them into smaller 512×512 pixel tiles, which are individually segmented and later reconstructed into full-sized masks using a smoothing algorithm that reduces edge artifacts. At the core of PyLC is a ResNet101 backbone that extracts deep spatial features

through residual learning, improving training stability and accuracy. These features are then passed to an encoder equipped with Atrous Spatial Pyramid Pooling (ASPP), which captures multiscale context without downsampling the input. This is particularly important in oblique images, where objects vary dramatically in size between the foreground and background. The decoder module refines these features by upsampling them and recovering sharp object boundaries, helping the model distinguish fine-scale land cover transitions.

To improve segmentation performance, PyLC uses a weighted multi-loss function that combines cross-entropy loss, Dice loss, and focal loss. This loss design improves learning on underrepresented classes and enhances overlap-based accuracy. PyLC also incorporates extensive data augmentation and an oversampling strategy inspired by SMOTE to address severe class imbalance in the training dataset. In addition, the network includes normalization routines tailored to historical grayscale and modern RGB images, improving consistency across lighting and image quality conditions.

Overall, PyLC is specifically designed to address the key challenges of oblique image segmentation, including high resolution, variable scale and perspective, limited training data, and class imbalance.

4.2 Implementation Details

PyLC was trained using 144 pairs of historical and 140 pairs of repeat photographs from the MLP collection, each paired with their corresponding landcover classifications. Out of these, 9 images were designated as test images and excluded from the training phase. The overall training process took 8 hours to complete.

The computational resources utilized for this process are detailed in Table 4.1, which includes a CPU with 16 GB RAM and a GPU featuring 32 GB VRAM. These resources were provided by Digital Alliance Canada's cloud infrastructure, facilitating efficient and scalable model training.

Component	**Model**	**Specifications**
CPU	-	16 GB RAM
GPU	NVIDIA V100	32 GB VRAM

Table 4.1: Specifications of Computational Resources Utilized for PyLC Training

4.3 Results

PyLC yielded an overall F1 score of 0.77, and a weighted mean IoU of 0.78 for repeat photographs in the test set. For the historical images, the network achieved an overall F1 score of 0.60, and a weighted mean IoU of 0.62 as demonstrated in Table 4.2. The class-specific metric scores for historic and repeat images are reported in Table 4.3 and Table 4.4 respectively.

Metric	**Historical Images**	**Repeat Images**
F1 Score	0.60	0.77
IoU	0.48	0.64
Weighted IoU	0.62	0.78

Table 4.2: Performance metrics of PyLC on the test dataset

Category	**IoU**	**F1 Score**
Coniferous Forest	0.79	0.88
Rock / Barren Ground	0.58	0.74
Water	0.45	0.62
Recently Burned	0.29	0.45
Herbaceous / Shrub	0.50	0.66
Wetland	0.13	0.22
Broadleaf / Mixedwood Forest	0.01	0.03
Snow / Ice	0.67	0.80
Not Classified	0.95	0.97

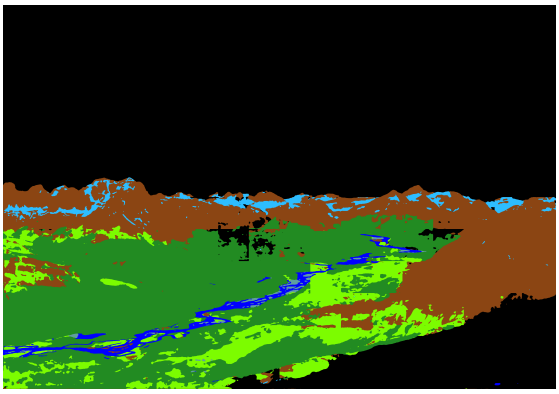
Table 4.3: Class-specific metric scores of PyLC for historical images

Example results are shown in Figure 4.2, with our result at the top, the terrestrial image in the middle, and manually segmented image at the bottom. The results and IoU scores are from PyLC's direct segmentation without post-processing. PyLC accurately distinguishes areas of Coniferous Forest, Barren Ground, and Snow & Ice. It also effectively identifies areas of Herbaceous/Shrub, Wetland, and Water but sometimes struggles to identify the complete area present in an image. It identifies Recently burned and Broadleaf/Mixedwood

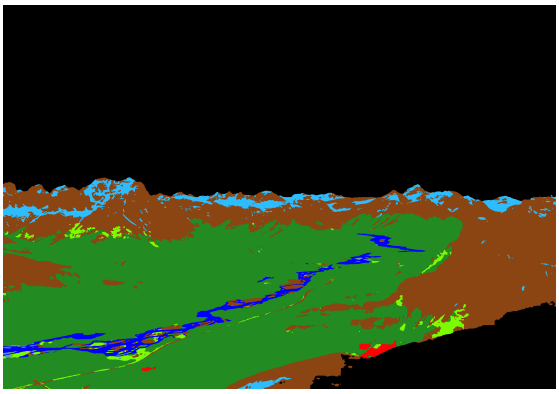
Category	**IoU**	**F1 Score**
Coniferous Forest	0.90	0.95
Rock / Barren Ground	0.74	0.85
Water	0.57	0.73
Recently Burned	0.35	0.52
Herbaceous / Shrub	0.58	0.73
Wetland	0.56	0.72
Broadleaf / Mixedwood Forest	0.39	0.57
Snow / Ice	0.73	0.84
Not Classified	0.98	0.99

Table 4.4: Class-specific metric scores of PyLC for repeat images

areas less accurately, which is attributable to their rarity in the training data and often-similar appearance to surrounding classes. It performs less accurately on historical images than their modern counterparts due to a combination of less information being presented in greyscale images as well as frequent cases of mixed or poor image quality.



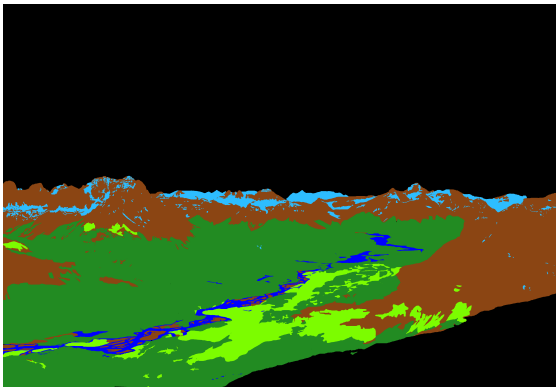
(a) Historic Image's Predicted Mask



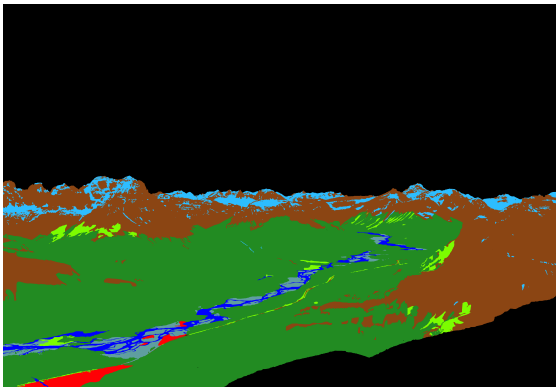
(b) Repeat Image's Predicted Mask

(c) Historic Image

(d) Repeat Image



(e) Manually Segmented Historic Image's Mask



(f) Manually Segmented Repeat Image's Mask

Figure 4.2: Baseline Results

Chapter 5

Experiments

In this section, we share our experiment results for the MS2D dataset with various model architectures specifically designed for ultra-high-resolution images.

5.1 Global Local Network (GLNet)

The Global-Local Network (GLNet) is a deep learning architecture designed to perform semantic segmentation on ultra-high-resolution images in a memory-efficient manner as show in Figure 5.1. Large images typically pose a challenge for conventional models, as they cannot be processed in full resolution without exceeding GPU memory limits. GLNet addresses this by combining a global branch that operates on a downsampled version of the full image, and a local branch that processes selected high-resolution patches [2]. The global branch captures broad semantic context across the entire image, while the local branch focuses on detailed object boundaries and textures.

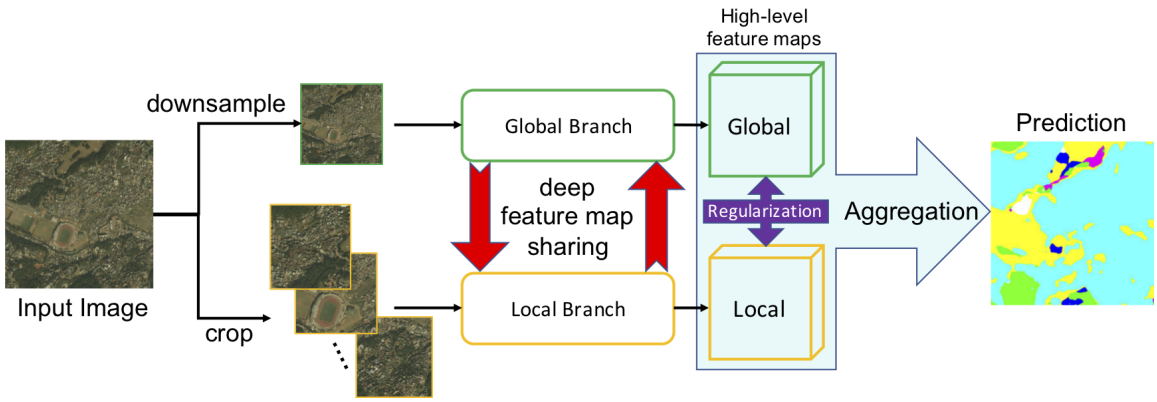


Figure 5.1: GLNet Model Architecture

During operation, GLNet begins by downsampling the full input image and dividing it into a set of overlapping or non-overlapping patches. The downsampled image is passed through the global encoder, and the high-resolution patches are fed to the local encoder. Both branches typically use a shared backbone encoder (e.g., ResNet), meaning the same network architecture and weights are applied to extract features from both the global and local views. This ensures that both branches produce consistent and compatible feature representations.

To relate the features from these two branches, GLNet performs feature alignment, where the local patch features are mapped back to their corresponding positions within the global context. This may involve coordinate transformations or positional embeddings to maintain spatial coherence. The model also applies mutual regularization between the global and local feature maps during training — for example, by enforcing consistency between predictions at overlapping regions or by sharing gradients — so that each branch benefits from the supervision and learned features of the other.

Once the features from both branches are processed and aligned, they are merged using a fusion module, which combines them either via simple concatenation or more advanced mechanisms like attention-based fusion. This final set of fused features is then passed through decoding layers to produce the final pixel-wise segmentation map.

Overall, GLNet's operation enables it to efficiently segment ultra-high-resolution images by capturing both large-scale semantic structure and fine local details, while significantly reducing memory usage during training and inference.

5.1.1 Implementation Details

GLNet was trained on 144 pairs of historical and 140 pairs of repeat photographs from the MLP collection, each linked to corresponding landcover classifications. Nine images were withheld for testing and not included in training. The training process took approximately 80 hours, allowing the model to effectively learn from the dataset.

The computational resources utilized for this experiment comprised a high-performance CPU with 64 GB of RAM and a powerful GPU, specifically the NVIDIA V100 equipped with 32 GB of VRAM. These specifications were essential for handling the intensive computations required by GLNet, ensuring efficient execution. The cloud infrastructure provided by Digital Alliance Canada facilitated the management of these resources.

Component	Model	Specifications
CPU	-	64 GB RAM
GPU	NVIDIA V100	32 GB VRAM

Table 5.1: Specifications of Computational Resources Utilized for GLNet Training

5.1.2 Results

The method produced an overall F1 score of 0.45 and a weighted mean IoU of 0.67 for repeat photographs in the test set. For the historical images, the model achieved an overall F1 score of 0.48 and a weighted mean IoU of 0.47, as shown in Table 5.2. The class-specific metric scores for historical and repeat images are presented in Tables 5.3 and 5.4, respectively.

Metric	Historical Images	Repeat Images
F1 Score	0.48	0.45
IoU	0.37	0.38
Weighted IoU	0.47	0.67

Table 5.2: Performance metrics of GLNet on the test dataset

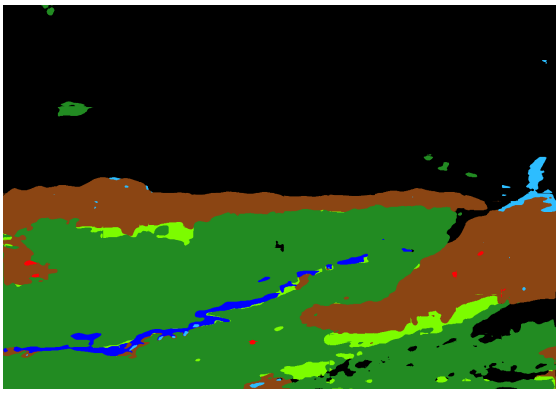
Figure 5.2 presents example results, with our segmentation at the top, the terrestrial image in the center, and manually segmented image at the bottom. The results and IoU scores reflect GLNet's direct segmentation without any post-processing. While GLNet effectively captures low-level features and identifies the various classes in both repeat and historical images, it faces difficulty capturing fine details in the segmented masks which can be attributed to its high complexity.

Category	IoU	F1 Score
Coniferous Forest	0.60	0.75
Rock / Barren Ground	0.50	0.67
Water	0.18	0.30
Recently Burned	0.23	0.38
Herbaceous / Shrub	0.32	0.49
Wetland	0.14	0.25
Broadleaf / Mixedwood Forest	0	0
Snow / Ice	0.39	0.56
Not Classified	0.92	0.96

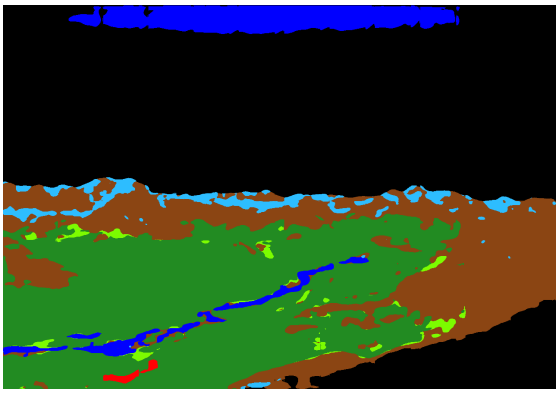
Table 5.3: Class-specific metric scores of GLNet for historical images

Category	IoU	F1 Score
Coniferous Forest	0.82	0.90
Rock / Barren Ground	0.64	0.78
Water	0.05	0.10
Recently Burned	0	0
Herbaceous / Shrub	0.48	0.65
Wetland	0.02	0.03
Broadleaf / Mixedwood Forest	0.01	0.01
Snow / Ice	0.48	0.65
Not Classified	0.95	0.97

Table 5.4: Class-specific metric scores of GLNet for repeat images



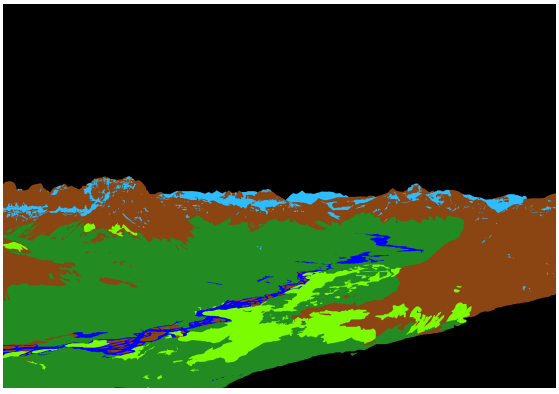
(a) Historic Image's Predicted Mask



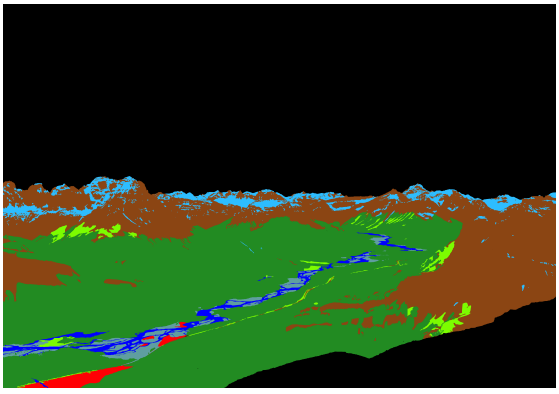
(b) Repeat Image's Predicted Mask

(c) Historic Image

(d) Repeat Image



(e) Manually Segmented Historic Image's Mask



(f) Manually Segmented Repeat Image's Mask

Figure 5.2: GLNet Model Results

5.2 From Contexts to Locality (FCtL)

From Contexts to Locality (FCtL) is a deep learning architecture developed for ultra-high-resolution image segmentation. Unlike traditional patch-based methods that treat each patch independently, leading to inconsistencies, FCtL models the context-to-local correlation between a patch and its neighboring regions as shown in Figure 5.3. This enables the network to make more context-aware predictions while preserving local detail.

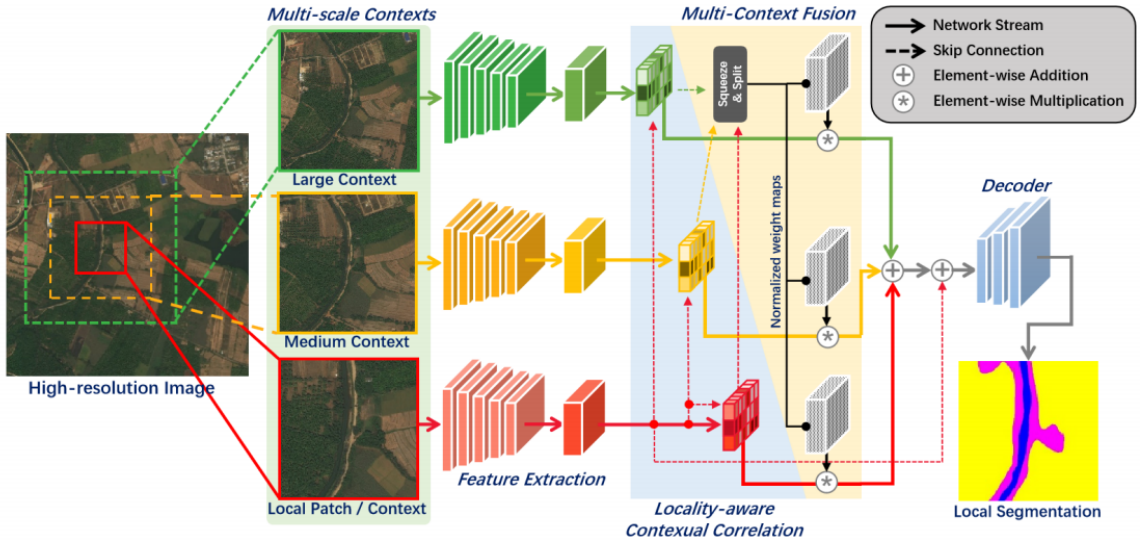


Figure 5.3: FCtL Model Architecture

The architecture consists of three main modules. The Locality-Aware Contextual Correlation Module (LCCM) calculates correlation maps between a target patch and its surrounding context, guiding the network to focus on relevant contextual cues. The Adaptive Context Fusion (ACF) module merges multi-scale contextual features using attention mechanisms, ensuring that both coarse and fine details are captured effectively. The Contextual Semantics Refinement (CSR) network further enhances segmentation quality by iteratively refining coarse predictions, especially along object boundaries. It reduces noise and sharpens contours through residual learning.

FCtL addresses several key challenges in ultra-high-resolution image segmentation. Context fragmentation, common in patch-based models, arises when patches lack information about the larger scene, leading to inconsistent labels. FCtL alleviates this by embedding contextual dependencies early in the pipeline. Boundary artifacts, which appear when merging independently processed patches, are minimized through context-aware prediction and refinement. Lastly, scale variation, a frequent issue in ultra-high-resolution imagery, is han-

dled through adaptive context fusion, which captures semantic cues at varying spatial resolutions.

Overall, FCtL combines patch efficiency with context sensitivity, producing high-quality, coherent segmentation masks suitable for large-scale remote sensing tasks [3].

5.2.1 Implementation Details

FCtL was trained using 144 pairs of historical and 140 pairs of repeat photographs from the MLP collection, with each image associated with corresponding landcover classifications. Nine images were reserved for testing and excluded from the training process. The entire training procedure took 108 hours to complete.

The computational resources utilized to train FCtL were identical to those used for GLNet as shown in Table 5.1, except the CPU had 32 GB of RAM rather than 64 GB.

5.2.2 Results

The model achieved an overall F1 score of 0.60 and a weighted mean IoU of 0.74 on the test set of repeat photographs. For the historical images, it recorded an overall F1 score of 0.49 and a weighted mean IoU of 0.54, as detailed in Table 5.5. Class-specific metrics for both historical and repeat images are provided in Tables 5.6 and 5.7, respectively.

Metric	**Historical Images**	**Repeat Images**
F1 Score	0.49	0.60
IoU	0.38	0.50
Weighted IoU	0.54	0.74

Table 5.5: Performance metrics of FCtL on the test dataset

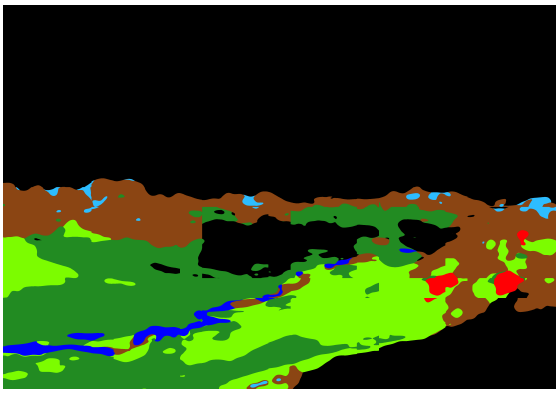
Figure 5.4 illustrates sample outcomes, featuring our segmentation at the top, the original terrestrial image in the middle, and manually segmented image at the bottom. Both the results and IoU scores represent FCtL's unrefined segmentation, as no post-processing steps were applied. Similar to the GLNet, FCtL efficiently captures low-level features and distinguishes between different classes in both repeat and historical images. However, MS2D's high complexity makes it challenging to capture fine details in the segmented masks.

Category	IoU	F1 Score
Coniferous Forest	0.72	0.84
Rock / Barren Ground	0.48	0.65
Water	0.19	0.32
Recently Burned	0.19	0.32
Herbaceous / Shrub	0.44	0.61
Wetland	0.01	0.01
Broadleaf / Mixedwood Forest	0	0
Snow / Ice	0.50	0.66
Not Classified	0.94	0.97

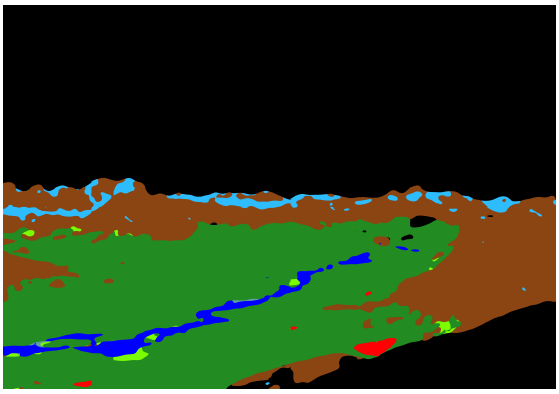
Table 5.6: Class-specific metric scores of FCtL for historical images

Category	IoU	F1 Score
Coniferous Forest	0.87	0.93
Rock / Barren Ground	0.70	0.82
Water	0.39	0.56
Recently Burned	0.04	0.08
Herbaceous / Shrub	0.52	0.69
Wetland	0.46	0.63
Broadleaf / Mixedwood Forest	0	0
Snow / Ice	0.57	0.73
Not Classified	0.98	0.99

Table 5.7: Class-specific metric scores of FCtL for repeat images



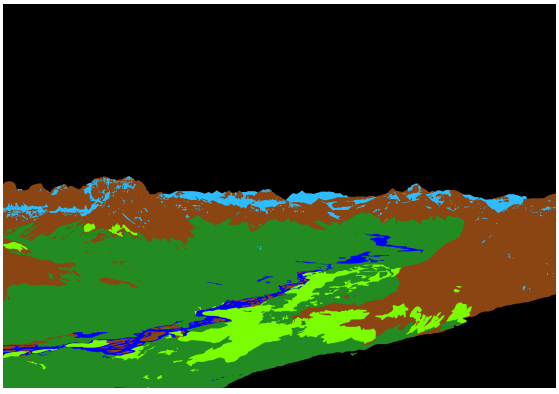
(a) Historic Image's Predicted Mask



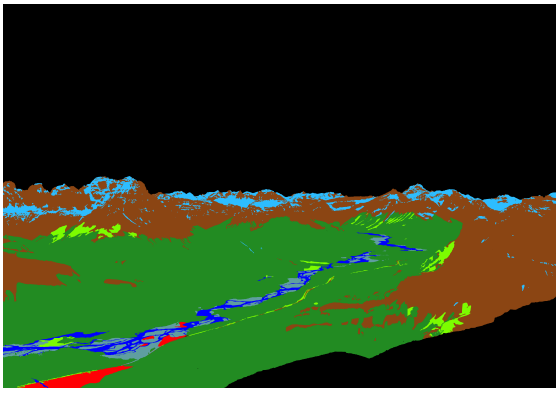
(b) Repeat Image's Predicted Mask

(c) Historic Image

(d) Repeat Image



(e) Manually Segmented Historic Image's Mask



(f) Manually Segmented Repeat Image's Mask

Figure 5.4: FCtL Model Results

5.3 Integrating shallow and deep networks (ISDNet)

ISDNet is an end-to-end image segmentation framework tailored for ultra-high-resolution remote sensing images, where memory constraints and loss of fine details present significant challenges. The model is built on a bilateral segmentation architecture that integrates both shallow and deep branches, each designed to process the input at different resolutions and levels of abstraction, as illustrated in Figure 5.5.

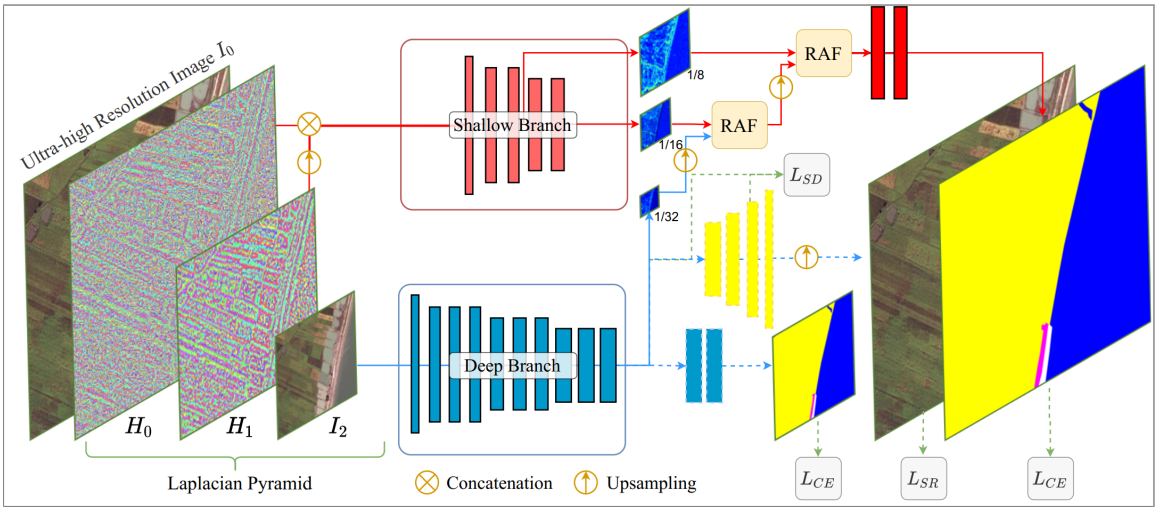


Figure 5.5: ISDNet Model Architecture

The shallow branch operates on the high-resolution input to preserve fine spatial details, which are often lost in deep network downsampling. Meanwhile, the deep branch works on a lower-resolution version of the input, allowing for efficient extraction of high-level semantic features with a broader receptive field. These complementary outputs are integrated using a feature fusion module, which aligns the spatial dimensions of both branches and merges them via channel-wise concatenation followed by convolutional refinement. This fusion allows the network to leverage both fine edge details and global semantic cues for precise segmentation.

A notable innovation in ISDNet is the incorporation of heterogeneous supervision signals—specifically, the use of an auxiliary learning task such as image super-resolution. This multitask learning strategy strengthens the shallow branch by guiding it to recover high-frequency details, improving overall segmentation accuracy [1]. Additionally, this approach alleviates issues like over-smoothing and semantic drift that are common in segmentation tasks involving large-scale data.

By jointly optimizing the segmentation and super-resolution tasks, ISDNet addresses

key challenges in remote sensing image segmentation: maintaining fine object boundaries, managing high computational demands, and improving generalization across varied terrains and scales.

5.3.1 Implementation Details

ISDNet was trained on 284 images of varying resolutions, containing 8 classes of landscape regions, as shown in Table 1.1. The 'Not Classified' class is excluded from the evaluation. The dataset was split into training and test sets, with 135 historical photographs and 9 test images, and 131 repeat images with 9 test images. The training process took approximately 16 hours, utilizing the same computational resources as FCtL.

5.3.2 Results

The framework achieved an overall F1 score of 0.71 and a weighted mean IoU of 0.80 on the test set of repeat photographs. For the historical images, it recorded an overall F1 score of 0.61 and a weighted mean IoU of 0.66, as shown in Table 5.8. Class-specific metrics for both historical and repeat images are presented in Tables 5.9 and 5.10, respectively.

Metric	**Historical Images**	**Repeat Images**
F1 Score	0.61	0.71
IoU	0.50	0.61
Weighted IoU	0.66	0.80

Table 5.8: Performance metrics of ISDNet on the test dataset

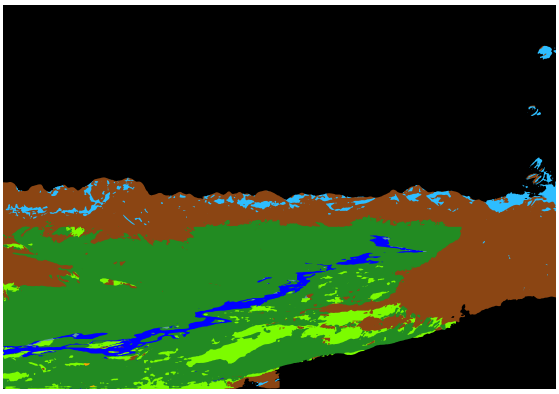
Figure 5.6 illustrates sample outcomes, featuring our segmentation at the top, the original terrestrial image in the middle, and manually segmented image at the bottom. Both the results and IoU scores reflect ISDNet's raw segmentation, as no post-processing was applied. ISDNet outperforms our baseline and successfully addresses the tiling problem present in the PyLC architecture. However, its longer training time can be attributed to its high complexity.

Category	IoU	F1 Score
Coniferous Forest	0.82	0.90
Rock / Barren Ground	0.62	0.76
Water	0.34	0.51
Recently Burned	0.62	0.77
Herbaceous / Shrub	0.52	0.68
Wetland	0.11	0.21
Broadleaf / Mixedwood Forest	0	0
Snow / Ice	0.49	0.66
Not Classified	0.96	0.98

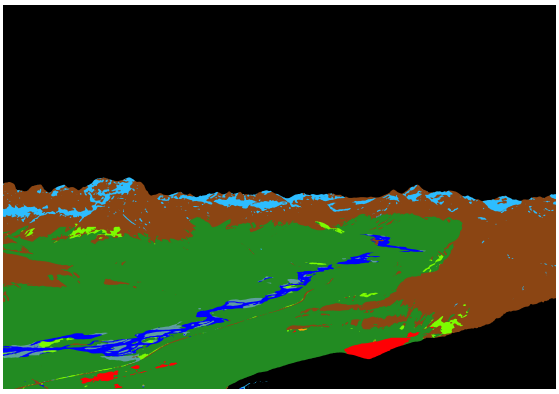
Table 5.9: Class-specific metric scores of ISDNet for historical images

Category	IoU	F1 Score
Coniferous Forest	0.90	0.95
Rock / Barren Ground	0.79	0.89
Water	0.53	0.69
Recently Burned	0.29	0.45
Herbaceous / Shrub	0.65	0.78
Wetland	0.62	0.76
Broadleaf / Mixedwood Forest	0.01	0.02
Snow / Ice	0.69	0.82
Not Classified	0.99	0.99

Table 5.10: Class-specific metric scores of ISDNet for repeat images



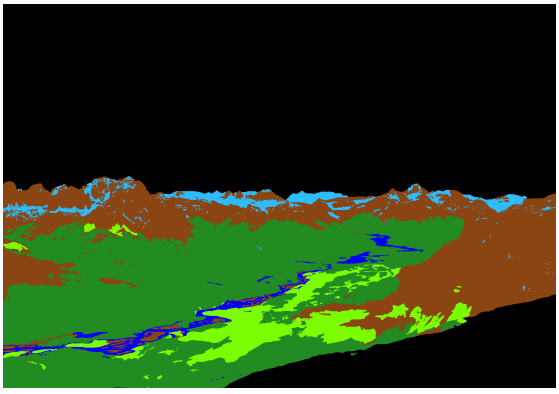
(a) Historic Image's Predicted Mask



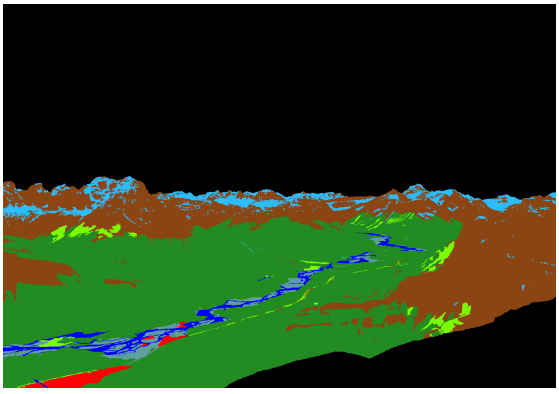
(b) Repeat Image's Predicted Mask

(c) Historic Image

(d) Repeat Image



(e) Manually Segmented Historic Image's Mask



(f) Manually Segmented Repeat Image's Mask

Figure 5.6: ISDNet Model Results

5.4 Overall Results

In this section, we summarize the performance of the models — GLNet, FCtL, ISDNet, and PyLC — on the MS2D dataset using F1 Score and Weighted IoU metrics, and discuss how the proposed approach mitigates the tiling issue.

5.4.1 Comparative Summary

Metric	**Historical Images**				**Repeat Images**			
	GLNet	FCtL	ISDNet	PyLC	GLNet	FCtL	ISDNet	PyLC
F1 Score	0.48	0.49	**0.61**	0.60	0.45	0.60	0.71	**0.77**
IoU	0.37	0.38	**0.50**	0.48	0.38	0.50	0.61	**0.64**
Weighted IoU	0.47	0.54	**0.66**	0.62	0.67	0.74	**0.80**	0.78

Table 5.11: Performance Metrics

Table 5.11 provides a comparative analysis of the results for both historical and repeat images. Overall, ISDNet delivers the best performance, surpassing PyLC in terms of F1 score and weighted IoU for historical images by 1% and 4%, respectively. For repeat images, while ISDNet outperforms PyLC by 2% on weighted IoU, it falls short by 6% on the F1 score. ISDNet also resolves the tiling issue present in PyLC's architecture, though its higher complexity leads to longer training times. FCtL and GLNet, despite efficiently capturing low-level features and distinguishing between different classes in both image sets, struggle with fine detail segmentation due to MS2D's high complexity. ISDNet's performance is also comparable to DeepGlobe [4] benchmarks on the relatively simpler task of segmenting orthogonal satellite images. Figure 5.7 illustrates the results of all four models compared with manually segmented masks.

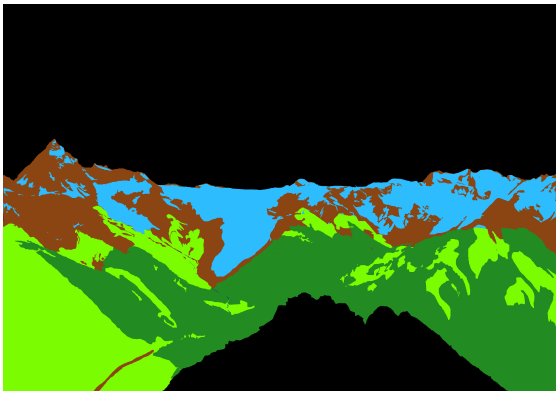
Model	CPU	GPU	Training time
GLNet	64 GB	NVIDIA V100 32 GB	80 h
FCtL	32 GB		108 h
ISDNet	32 GB		16 h
PyLC	16 GB		8 h

Table 5.12: System Configurations and Training Time

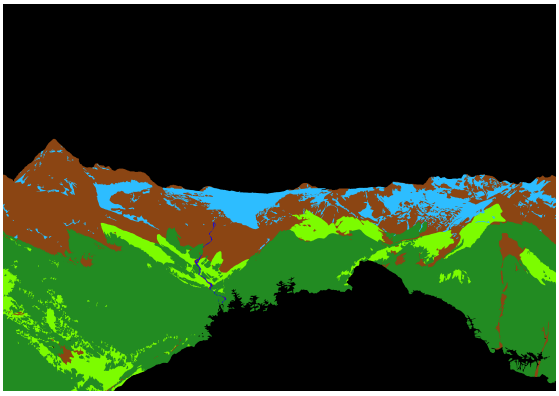
Table 5.12 provides a comparative analysis of system configurations and training times for all four models. PyLC is the most efficient, requiring just 8 hours with a 16 GB CPU, while ISDNet follows with a training time of 16 hours using a 32 GB CPU. Despite its training time, ISDNet's performance is superior to PyLC's. In contrast, FCtL has the longest training time at 108 hours with the same system requirements as ISDNet, but GLNet, though using a higher CPU, takes 80 hours.

(a) Historic Image

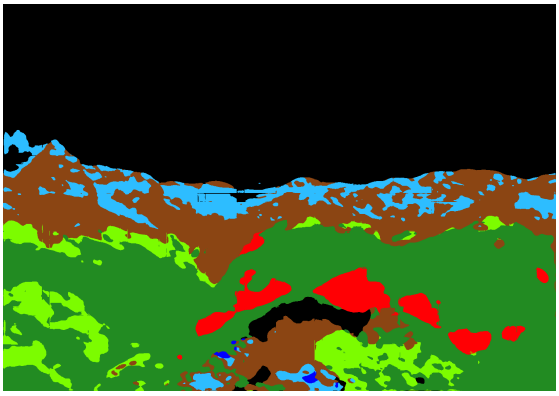
(b) Repeat Image



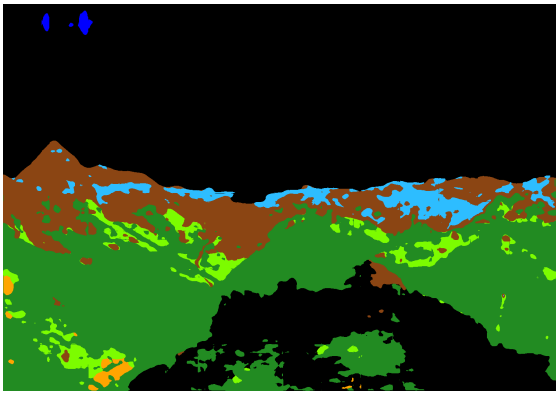
(c) Manually Segmented Historic Image's Mask



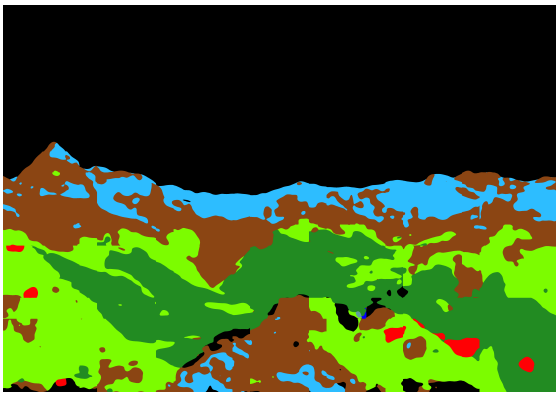
(d) Manually Segmented Repeat Image's Mask



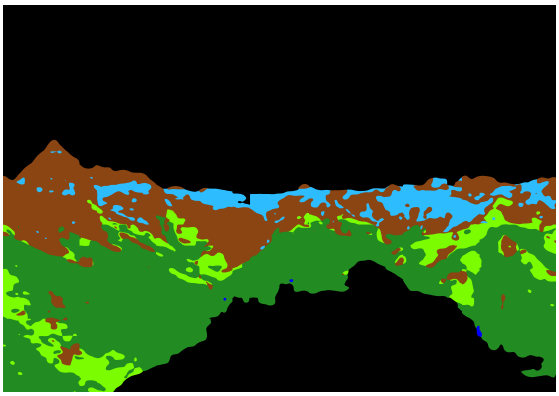
(e) GLNet Historic Image Mask



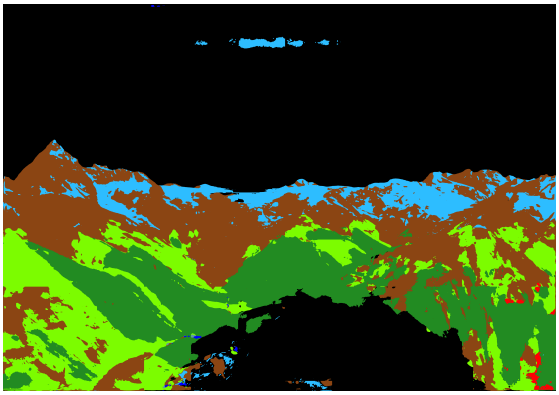
(f) GLNet Repeat Image Mask



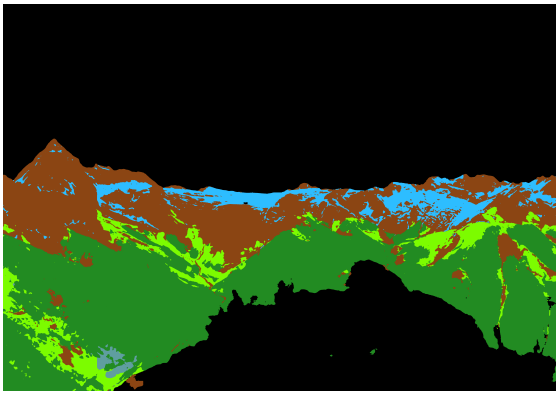
(g) FCtL Historic Image Mask



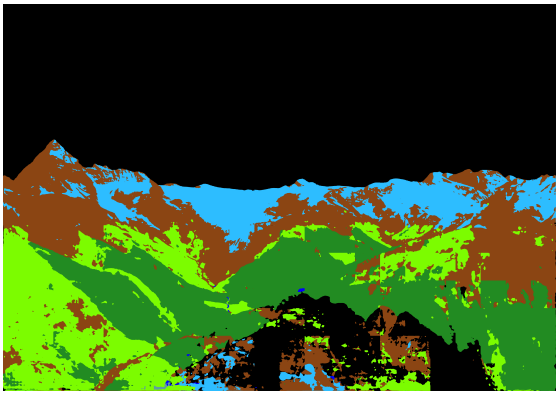
(h) FCtL Repeat Image Mask



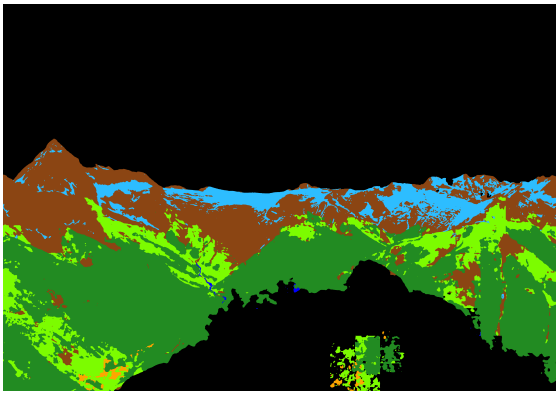
(i) ISDNet Historic Image Mask



(j) ISDNet Repeat Image Mask



(k) PyLC Historic Image Mask

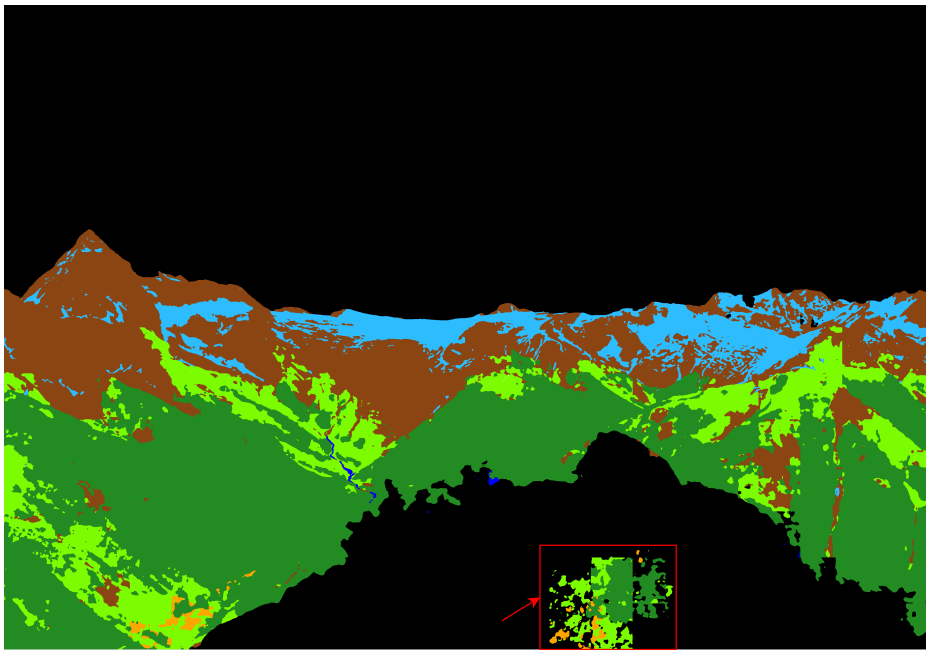


(l) PyLC Repeat Image Mask

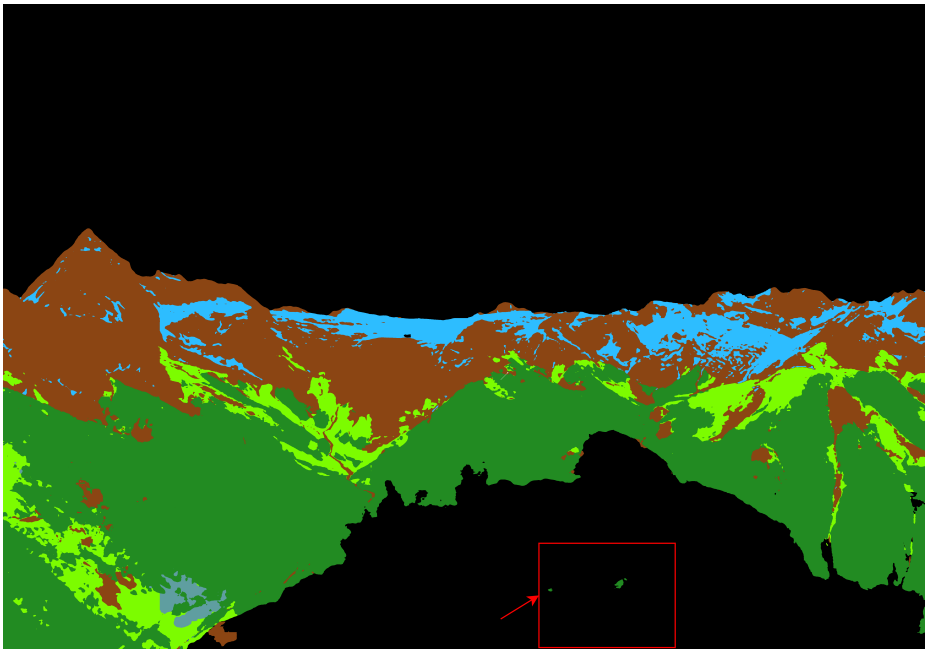
Figure 5.7: Model Results Synopsis

5.4.2 Tiling Artifacts Resolution

ISDNet effectively addresses the tiling problem faced in PyLC by enabling end-to-end segmentation of ultra-high-resolution images, as depicted in Figure 5.8. Its architecture balances detail preservation and semantic context without relying on patch-based inference,



(a) Titling Artifact encountered in PyLC



(b) Tiling Artifact resolved by ISDNet

Figure 5.8: Tiling Artifact disappears with the proposed approach

making it well suited for large-scale MS2D dataset.

Chapter 6

Conclusion

We introduced the innovative MountainScape Segmentation Dataset (MS2D), featuring 284 annotated image pairs of historical and repeat photographs for land cover classification. This paper examined the unique imagery characteristics of MS2D, outlined the annotation methodology, defined the land cover classification task, and detailed the evaluation metrics. It also establishes an initial performance benchmark using PyLC and provides a comparative analysis of various model architectures applied to the MS2D dataset. The ISDNet model achieved the highest performance, with F1 scores of 0.71 and 0.61, and weighted mean IoU scores of 0.80 and 0.66 for repeat and historical images, respectively, generating comparable results to DeepGlobe benchmarks, demonstrating the robustness of ISDNet on the MS2D dataset for land cover classification. Additionally, ISDNet effectively resolved the tiling issue encountered in PyLC, ensuring seamless segmentation across image boundaries. We anticipate that the MS2D dataset will continue to develop and evolve, serving as a crucial benchmark for land cover classification in oblique imagery, fostering collaborative interdisciplinary research. This benchmark will allow for accurate comparisons, paving the way for advancements at the intersection of computer vision, machine learning, remote sensing, and geosciences.

Bibliography

[1] S. Guo, L. Liu, Z. Gan, Y. Wang, W. Zhang, C. Wang, G. Jiang, W. Zhang, R. Yi, L. Ma, and K. Xu, ‘‘Isdnet: Integrating shallow and deep networks for efficient ultra-high resolution segmentation,’’ in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 4361–4370.

[2] W. Chen, Z. Jiang, Z. Wang, K. Cui, and X. Qian, ‘‘Collaborative global-local networks for memory-efficient segmentation of ultra-high resolution images,’’ in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[3] Q. Li, W. Yang, W. Liu, Y. Yu, and S. He, ‘‘From contexts to locality: Ultra-high resolution image segmentation via locality-aware contextual correlation,’’ in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 7252–7261.

[4] I. Demir, K. Koperski, D. Lindenbaum, G. Pang, J. Huang, S. Basu, F. Hughes, D. Tuia, and R. Raskar, ‘‘Deepglobe 2018: A challenge to parse the earth through satellite images,’’ in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, Jun. 2018. [Online]. Available: http://dx.doi.org/10.1109/CVPRW.2018.00031

[5] S. Rose, ‘‘An evaluation of deep learning semantic segmentation for land cover classification of oblique ground-based photography,’’ Master’s thesis, University of Victoria, 2020.

[6] E. Margolis and L. Pauwels, *The SAGE handbook of visual research methods*. Sage, 2011.

[7] K. Green, ‘‘Selecting and interpreting high-resolution images,’’ *Journal of Forestry*, vol. 98, no. 6, pp. 37–40, 2000.

[8] U. Bayr and O. Puschmann, ‘‘Automatic detection of woody vegetation in repeat landscape photographs using a convolutional neural network,’’ *Ecological Informatics*, vol. 50, pp. 220–233, 2019.

[9] F. Jean, A. B. Albu, D. Capson, E. Higgs, J. T. Fisher, and B. M. Starzomski, ‘‘The mountain habitats segmentation and change detection dataset,’’ in *2015 IEEE Winter Conference on Applications of Computer Vision*, 2015, pp. 603–609.

[10] N. Dalal and B. Triggs, ‘‘Histograms of oriented gradients for human detection,’’ in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, 2005, pp. 886–893 vol. 1.

[11] M. Pietikäinen, ‘‘Local binary patterns,’’ *Scholarpedia*, vol. 5, no. 3, p. 9775, 2010.

[12] B. Matthews, ‘‘Comparison of the predicted and observed secondary structure of t4 phage lysozyme,’’ *Biochimica et Biophysica Acta (BBA) - Protein Structure*, vol. 405, no. 2, pp. 442–451, 1975. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0005279575901099

[13] R. Okamoto, R. Ide, and H. Oguma, ‘‘Automatically drawing vegetation classification maps using digital time-lapse cameras in alpine ecosystems,’’ *Remote Sensing in Ecology and Conservation*, vol. 10, no. 2, pp. 188–202, 2024.

[14] O. Ronneberger, P. Fischer, and T. Brox, ‘‘U-net: Convolutional networks for biomedical image segmentation,’’ 2015. [Online]. Available: https://arxiv.org/abs/1505.04597

[15] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, ‘‘Encoder-decoder with atrous separable convolution for semantic image segmentation,’’ 2018. [Online]. Available: https://arxiv.org/abs/1802.02611

[16] J. Tricker, C. Wright, S. Rose, J. Rhemtulla, T. Lantz, and E. Higgs, ‘‘Assessing the accuracy of georeferenced landcover data derived from oblique imagery using machine learning,’’ *Remote Sensing in Ecology and Conservation*, vol. 10, no. 3, pp. 401–415, 2024. [Online]. Available: https://zslpublications.onlinelibrary.wiley.com/doi/abs/10.1002/rse2.379

[17] M. E. Sanseverino, M. J. Whitney, and E. S. Higgs, ‘‘Exploring landscape change in mountain environments with the mountain legacy online image analysis toolkit,’’ *Mountain Research and Development*, vol. 36, no. 4, pp. 407–416, 2016.

[18] Q. Org, ‘‘Qgis geographic information system,’’ *QGIS Association*, 2021.

[19] C. Wright, C. Bone, D. Mathews, J. Tricker, B. Wright, and E. Higgs, ‘‘Mountain image analysis suite (mias): A new plugin for converting oblique images to landcover maps in qgis,’’ *Transactions in GIS*, 2024.

[20] J. Fortin, J. Fisher, J. Rhemtulla, and E. Higgs, ‘‘Estimates of landscape composition from terrestrial oblique photographs suggest homogenization of rocky mountain landscapes over the last century,’’ *Remote Sensing in Ecology and Conservation*, vol. 5, 12 2018.

[21] M. Frederickson, ‘‘When the flame goes out: an exploration of landscape change using repeat photography related to indigenous burning in kananaskis country, alberta,’’ Ph.D. dissertation, 2022.

[22] Higgs, Sanseverino, Whitney, and Fortin, ‘‘Advances in visual applications: Visualizing quantifying landscape change in sw alberta using mountain legacy project photography,’’ fri Research, Tech. Rep., 6 2020.

[23] G. McDermid, R. Hall, G. Sanchez-Azofeifa, S. Franklin, G. Stenhouse, T. Kobliuk, and E. LeDrew, ‘‘Remote sensing and forest inventory for wildlife habitat assessment,’’ *Forest Ecology and Management*, vol. 257, no. 11, pp. 2262–2269, 2009. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378112709001595

[24] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, ‘‘Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs,’’ 2017. [Online]. Available: https://arxiv.org/abs/1606.00915