Road Extraction from Time Series Satellite Images

Liyao Tang

Supervisor: Dr. Lexing Xie

A thesis submitted in partial fulfilment of the degree of
Bachelor of Advanced Computing (Honours) at
The Department of Computer Science
Australian National University

May 2018

Except where otherwise indicated, this thesis is my own original work.

Liyao Tang
May 24, 2018

Acknowledgements

I would like to appreciate my supervisor, Dr Lexing Xie, for providing my with this interesting project, for the continuous support through the project and for her patience and vast knowledge. She is always willing to help whenever I have any questions about research, thesis writing or career planning. I would like to thank Alex Mathews for guiding me through the jungle with his experience, helping me tackle technical difficulties and arranging all sort of research resources. I would also like to thank Pablo Rozas Larraondo and other collaborators from NCI for helping me collect and pre-process the data and provide me with their valuable insight and also computability. I thanks other friends for their valuable assistance and comments on my project. At last, I would say a special thank you to my family for supporting me through all my studies at universities.

Abstract

Road extraction from time-series satellite images is an important problem because it enables the analysis of road network evolution and urbanisation processes. In this thesis, we attempt the task from multiple perspectives, ranging from patch classification, sliding window (logistic regression) and image segmentation (fully convolutional networks and U-net). We analyse the performance from different approaches and propose two segmentation models that produce promising result. Our models cope with images of a medium resolution and multiple spectral bands, which are common properties of time-series satellite image collections. Despite our model containing fewer layers than most deep architecture, our model demonstrates stability across different scenes with various atmospheric conditions and is applicable for time-series analysis.

Contents

Chapter 1

Introduction

Urbanisation is a fundamental process in human history [Mumford 1961] and evolution of road network is found to be a fundamental driver in the process of urbanisation [Southworth 2013]. Meanwhile, road extraction from satellite images is one of the fundamental yet challenging tasks in remote sensing under a variety of demands ranging from urban planning to automated road navigation, self-driving vehicle etc.

In monitoring urbanisation, recent works have focused on utilising records of road networks in time series to understand the process of urbanisation. [Strano 2012] studies the evolution of road networks over almost 200 years to quantitatively characterise the growth of a city in Italy. [Xu et al. 2014] analyses the road network in a developing area and finds it to be positively correlated with built-up land areas.

In road extraction, recent works mostly focus on extracting roads from high-resolution images and various approaches have been taken. Statistical model with mathematical morphology methods are preferred traditionally. For instance, [Son 2004] proposes a hybrid approach consisting of support vector machine (SVM) and region growing method to segment the roads; and [Rajeswari et al. 2011] explores Level Set and Mean Shift methods.

Road extraction from satellite images can viewed as a semantic segmentation task from the prospective of image processing. Meanwhile, there is a trend to apply deep learning technology in image processing because they generally achieve better performance [Krizhevsky et al. 2012; Simonyan and Zisserman 2014]. Following the growth of deep learning, [Sai 2016] attempt to extract objects including roads and buildings from raw satellite images with Convolutional Neural Network (CNNs). Going deeper, [Zhang et al. 2017] combines the advantage of U-net and Residential Net to extract road network from high-resolution satellite images.

We notice that road network currently used in urban monitoring works are all historical records. Whereas areas without such record can not rely on current models for road extraction because they are designed for high-resolution images, not for historical collection of satellite images with usually a low or medium resolution. Hence, in this thesis, we propose two deep learning models for extracting road network from a collection of satellite images and reveal the road networks evolution in our local areas. Specifically, our main contributions can be summarised as:

1. We propose two models to extract roads from satellite images of medium resolution (30m), inspired by some famous architectures such as FCN and Unet.
2. We analyse and compare the performance of our models and logistic regression; and demonstrate that it is possible to extract roads under a medium resolution, compensated by extra bands (4 more bands apart from RGB bands).
3. We finally illustrate with two scenes that our models are able to cope with varying noise in a long time span and to provide a stable output revealing the evolution of the road networks.

The thesis is ordered as follows. First we will give a background knowledge of remote sensing and road extraction from satellite images. Then an overview of our dataset will be provided because there are no existing dataset in this field and it is thus carefully constructed by ourselves. Afterwards, four models for road extraction will be presented, with their corresponding experiments and discussions in the following chapter. Finally, an analysis of road network evolution on time-series collection of satellite images is given.

Chapter 2

Background

This chapter first introduces the background knowledge for remote sensing and road extraction and then review some classical and state-of-the-art approaches proposed in the recent years related to our work.

2.1 Introduction to Remote Sensing

Remote sensing has been defined by many times with different emphasis. An excessively definition of remote sensing can be given as the gathering of information at a distance [Campbell and Wynne 2011]. Yet in current usage, the term "remote sensing" generally refers to the use of satellite- or aircraft-based sensor technologies to detect and classify objects on Earth. In the rest of the thesis, we will restrict the remote sensing definition into satellite remote sensing of the Earth for our purpose.

Remote sensing can be concluded into two facets, which are acquiring information through a device and the analysis of the acquired data [Gupta 2018]. Regarding data acquisition, remote sensing uses electromagnetic (EM) radiation from an overhead perspective in one or more regions of the electromagnetic spectrum, reflected or emitted from the Earth's surface [Campbell and Wynne 2011]. With the thought of how we may utilize the acquired information, information theoretical principles provide that information is potentially available at an altitude from the Earth's surface, and in particular from the spectral, spatial and temporal variations of those fields [Landgrebe 2005].

Based on the restricted definition, we will give a summary of remote sensing regarding its acquisition of information and data analysis based on information provided by NASA [NASA 2018e; NASA 2012; USGS 2018a] and some famous books in remote sensing realm [Schowengerdt 2007; Gupta 2018].

2.1.1 Acquisition of Information

In remote sensing, the information is gathered and recorded by the sensor in the form of EM radiation, which serves as the communication link between the sensor and the object [Gupta 2018]. The following sections will give an overview of the different types of sensors and different spectral features of ground objects.

2.1.1.1 Sensor

Sensing instruments are of two primary types, passive and active.

Passive sensors employ sensors to measure radiation naturally reflected or emitted from the object, which is Earth in our case. Thus, the frequency bands for passive sensor measurements are determined by fixed physical properties of the substance being measured. This can be used to collect visible, NIR (Near Infrared), and SWIR (Short-Wavelength Infrared) because the earth reflects more such waves than it emits by itself.

Active sensors provide their own source of energy to illuminate the objects they observe. More specifically, an active sensor emits radiation in the direction of the target to be investigated and then detects and measures the radiation that is reflected or backscattered from the target. The motion of the sensor platform creates an effectively larger antenna, thereby increasing the spatial resolution. Active sensors mainly operate in the microwave spectrum, which enables them to penetrate the atmosphere in most conditions.

2.1.1.2 Collecting the EM Radiation

When collecting EM radiation of the Earth from the space, it is not possible to collect the whole spectrum, because various constituents in the atmosphere will absorb, diffract and reflect the radiation and these effects can be wavelength-dependent. For example, water vapor and carbon dioxide absorb waves from 2.5–3 μm and 5–8 μm while some microwave and radar waves are able to penetrate clouds, fog, and rain quite well. Those portions of the EM spectrum that can pass through the atmosphere with little or no attenuation are thus known as atmospheric windows.

After the radiation (originally from the active sensors or other sources such as the Sun) travelling through the atmosphere, it strikes objects on the Earth and similar effects, absorption, diffraction and reflection, happen, which bounce a portion of the radiation back to the space. Different objects have their own reflectance and absorption signatures, depending on their constituents.

After EM radiation are collected by satellites, though the whole EM spectrum is continuous, sensors need to separate the energy of radiation by filters into different discrete spectral bands in order to record the measurement. Such measurement are converted into values and stored as an image with the spatial information retained. The pixel values in such satellite image then denote the collected EM radiation reflected by corresponding ground objects, containing both useful features and noise.

As discussed above, factors affecting the collected radiation depends on both wavelength and objects on the Earth. Thus different parts of the EM spectrum are best suited for different tasks. For example, water bodies and vegetations transmit solar radiation and materials at normal temperatures begin to actively emit thermal radiation at longer wavelengths.

2.1.2 Data Analysis

There are two main interests in the analysis of remote sensing data, which are spatial relationships of the acquired feature maps and features measured via different waves on one single place, or one pixel if in digital data.

Nowadays application of remote-sensing data usually includes both interests, such as long-term monitoring of the environment, man's effect on the environment and global change monitoring.

Combined with other data, there are more and more great results from analysing those remote-sensing data. For example, [J. Vernon Henderson and Weil 2012] use satellite night lights data together with local income growth data to estimate the economic growth at both temporal and geographic scales, e.g. they found that coastal areas in sub-Saharan Africa have not grown faster than non-coastal areas over the last 17 years.

With the computer vision technology rising, it is possible to quantitatively monitor changes on the Earth surface. For example, [Fretwell PT 2014] count the breeding of southern right whales; [Abelson et al. 2014] quantitatively recognise roofs of different materials in Africa to detect extreme poverty in Kenya and Uganda.

2.2 Road Extraction on Satellite Images

Road extraction is one of the major application in remote sensing area because of its profound uses in traffic management, city planning, GPS navigation and map updating, etc [Shi et al. 2014]. This section first discusses the challenge in road extraction on satellite image and then the features one can use to detect roads and finally reviews some previous works in the road extraction with more details.

2.2.1 Challenge

As discussed in previous Section 2.1, satellite image are affected by sensor type, satellite position, sensor resolution and atmosphere interference, etc. The major noise caused by those interference can be summarized as follow [Weixing Wang 2016; Shi et al. 2014; Herumurti et al. 2013].

1. Shadows and other occlusion on the object will make segmentation more difficult and such occlusion can not be safely considered as a trivial Gaussian noise.

2. Under bad weather, the road and its background blurred into each other because of the added gray value brought by the clouds, even though waves able to penetrate clouds might be used. The fuzzy scene then results in a bad segmentation result.

3. There are different types of roads. Some roads are not paved and are similar to the background. Some other roads might be too small and can not be effectively segmented or even might not be represented in a single pixel in a satellite image

when the resolution is larger than the width of some selected roads. Hence, a mixture of different types of road can easily confuse the model.

4. Roads might be discontinuous from the view of satellites because of not only the shadow but also tunnels, buildings and other physical obstacles above the roads.

5. The noise distribution on a satellite image may be spatially different from place to place [Huazhong et al. 2014]. It is because firstly satellite images are collected at different time to construct an overview of the Earth and thus collected EM waves are under unique influence of atmosphere condition at the time of its measurement. Secondly, spatial variation of surface and atmospheric conditions may leads to a unique noise added to the EM waves collected from different places. Yet model generally assumes that the whole set of images share the same distribution of noise.

2.2.2 Features for Roads

As discussed in [G. Vosselman 1995; Wang et al. 2014] and [Weixing Wang 2016], the road in a satellite image generally contains following features that can help the extraction.

1. Spectral features

   Spectral features are probably the most direct features in satellite images as each pixel value denotes the recorded EM radiation bounced back by the object at that location. In modern satellite images where multiple sensors at different EM spectrum are usually used, EM signature of object at each pixel might be constructed.

2. Texture features

   Textures in image processing describes a spatial distribution [Stockman and Shapiro 2001], giving out information of spatial arrangement of pixel values and its intensities of a given image. The local texture will be different depending on the density of the roads nearby, indicating the probability of the appearance of the roads.

3. Contextual features

   The road on a satellite image has two obvious edge lines where the pixel value change sharply, with a larger gradient. At the same time, the pixel value within the road remains relatively consistent and is still largely different from those of the background. This contrast between roads and its surroundings can be helpful in the road extraction.

4. Topological features

Roads on the ground are usually thin lines crossing a large area from the view of satellites, compared to rivers which might suddenly run into a lake. Besides, roads usually forms a road networks and are not suddenly interrupted.

2.2.3 Review on Road Extraction

2.2.3.1 Mathematical and statistical models

There are many exciting mathematical and statistical tools that can be useful in road extraction from satellite images.

To utilise the topological features described above, mathematical morphology are used to extract lines from the image. For example, [Chanussot and Lambert 1998] assumes roads appear on the image as thin, elongated structures with a maximum width; they are locally rectilinear, with each road pixel belonging to a line segment that is longer than a minimum length; and each road segment is considered as a dark structure with respect to its surroundings. With those assumptions, several morphological filters are developed to filter out the valleys and peaks using the prior information of maximal width of roads and nonlinearity of the selected lines. At last, the roads network is generated by a simple thresholding on the output of the sequence of filters.

After straight line extraction conducts a local analysis on the image, line grouping helps with modelling a contextual constraint of the roads networks. Since low-level road extraction results will generally be fragmented and contain false hypotheses, grouping in road extraction involves selecting the correct roads from the extraction results as well as connecting them to construct the road network[Steger et al. 1997].

Markov random field (MRF) provides an efficient framework to introduce contextual features about the shape of road objects. [Tupin et al. 1998] and [Katartzis et al. 2001] extend the straight-line-based model discussed above based on this. Apart from using different set of filters for local straight lines detection, their models propagate the output from filters to an MRF to group lines using contextual constraints of the roads and to construct the final road networks.

Apart from MRFs, graph theory can also help extracting roads. [Unsalan and Sirmacek 2012] benefits from graph theory to represent the extracted road segments in a parametric and structural form. In their system, there are three main modules: probabilistic road center detection, graph-theory-based road network formation and road shape extraction. The system outputs the final result by applying the road shape extraction module to the constructed graph, which is based on the output of probabilistic road center detection.

[Bai et al. 2013] use graph cut (GC) theory to make fully use of the spatial information to achieve fine results. In their model, a pixel-wise fuzzy classification based on support vector machine (SVM) is first applied to select pixels with high probabilities, which then forms a pseudouser strike map. This map is then employed for graph cut theory to evaluate the truthful likelihoods of class labels and propagate them to the MRF for refinement.

2.2.3.2 Road extraction with neural networks

After neural networks and its back propagation (BP) [E. Rumelhart et al. 1986] algorithm were proposed in late 80's, it has been developing rapidly in various fields, including natural language processing, computer vision and social computing etc. With further development of neural network, some networks with specific features, such as convolutional neural network (CNN), are found to be remarkably successful in image processing. The tasks that are performed by those neural networks can be concluded into two parts, patch classification and semantic segmentation.

We will first briefly introduce those neural networks and then review major applications of those neural networks on road extraction.

1. Convolutional Neural Networks

   Convolutional neural networks (CNN) [LeCun et al. 1999] are a type of neural networks featuring the convolution layer, which is able to extract the right features for the given task. The convolution layer applies a convolution operation on its input, which is usually an image with multiple channels and is also called feature maps. The convolution operation convolves learn-able filters on the feature map and gives out the output. The convolution layer may includes multiple convolution operations and is usually followed by a pooling layer and a non-linearity activation function. The pooling layer is a sub-sampling process [Lee et al. 2009], which intuitively represents a local neighbourhood on a feature map with one value, thus shrinking the size of the feature maps. The non-linearity activation function is applied to feature maps element-wise and is commonly chosen to be Rectified Linear Units (ReLU) [Krizhevsky et al. 2012]. A common CNN architecture starts with several convolutional layers and then is followed by several fully connected layer.

   The CNN is powerful mainly because it can learn a set of useful filters to extract useful features for the given task and also can combine the local contextual information with the global view to give the output [Zeiler and Fergus 2013].

2. Fully Convolutional Networks

   As mentioned above, CNN usually has fully-connected layer before the output, which causes the loss of spatial information in the construction of model's final output. To overcome this, CNN can be modified into fully convolutional network (FCN) to directly generate an image as output, which suits the requirement of road extraction and other image segmentation task. FCN was first introduced by [Long et al. 2014] and contains no fully connected layer but an up-sampling process. The up-sampling process in FCN is implemented with learn-able filters as well, which can be viewed as the decoder in a encoder-decoder architecture [Badrinarayanan et al. 2015]. To improve the FCN further, model can combine the strengths of FCN with probabilistic graphical modeling to produce better result. [Zheng et al. 2015] and [Zhou et al. 2016] stack a conditional random field (CRF) as an additional layer after the FCN and [Liu et al. 2015] stack a MRF in-

stead. All three models are able to perform end-to-end training and are better than using FCN on it own.

Based on FCN, [Ronneberger et al. 2015] propose a U-net architecture which has become popular recently. Their architecture consists of not only the usual encoder-decoder structure in FCN but also a series of skip connections. Hence, at each up-sampling stage, the model can combine both spatial pixel-level information from the feature maps before down-sampling and the contextual information from the feature maps after down-sampling and corresponding up-sampling.

1. Patch Classification for Road Recognition

   As CNN can account for both contextual and spectral features in the patch, it becomes a powerful tool to classify patches into roads or other objects in satellite images. For example, [Zhong et al. 2017] proposes an agile CNN architecture with 3 convolutional layers and 2 fully connected layers to classify satellite images into different land-cover classes, including barren land, trees, grassland, roads, buildings and water bodies.

   In addition, road recognition can be combined with other application. For example [Liu et al. 2017] use deep convolutional networks to build an automatic system to evaluate and rate the urban visual environment based on street views and road conditions, which are first recognised from a large collection of random images sampled from Google Map Street Views.

2. Semantic Segmentation for Road Extraction

   As road extraction on satellite image needs to classify every pixel into a class, either road or background [Kirthika and Mookambiga 2011]. It can thus be viewed as a semantic segmentation task [Shapiro and Stockman 2001] with two classes.

   Early works of applying neural networks on road extraction, such as [Heermann and Khazenie 1992], were mostly based on spectral and contextual information of each pixel. Later, [Kirthika and Mookambiga 2011] integrates texture features including contrast, energy, entropy and homogeneity into the spectral features and extracts the roads in a sliding window fashion.

   [Kahraman et al. 2015] proposes a simple fully connected neural network with two hidden layers of 12 neurons and a 27-dimension input, that is, a input of a 3-by-3 region of R,G,B channels flatten into a vector. [Mnih and Hinton 2010] proposes a neural network with a single hidden layer and large number of units to extract roads from high-resolution satellite images using RGB channels. Their models also impose a pre-processing to reduce input dimensionality and a post-processing similar to line grouping.

   Apart from applying classifier on the central pixel of each pack in a sliding window fashion [Kirthika and Mookambiga 2011; Yin et al. 2010; Schroff et al. 2008], there are types of CNN that can output an image directly.

[Saito et al. 2016] uses a common CNN architecture described above to classify pixels into multiple classes including road. Their model outputs an 1-D vector, which is then reshaped into an image when calculating the loss or forming a prediction.

As FCN gains its popularity, [Maggiori et al. 2016] applies an FCN to segment satellite images into buildings and non-buildings area and outperforms [Saito et al. 2016]'s approach.

In road extraction fields, [Zhong et al. 2016] applies FCN on building and road extraction and demonstrates its promising results. [Wei et al. 2017] proposes an VGG [Simonyan and Zisserman 2014] -based FCN and a special loss function which can penalize the wrong prediction close to the road more heavily and thus models the geometric structure of roads. Their model gives a better result than the [Zhong et al. 2016]'s. [Zhang et al. 2017] proposes a deep network architecture combining the U-net and the residual network [He et al. 2015] and also reports a better result than [Saito et al. 2016] and original U-net on road extraction task.

Chapter 3

Dataset

This chapter first introduces the source of our dataset and then the way we construct the final dataset for training our model.

3.1 Source Data

As the development of remote sensing progresses, there are more and more valuable datasets released into open-source community. Our dataset are constructed based on satellite images from Landsat 8 observatory and road raster map from the Open Street Map (OSM) [OpenStreetMap contributors 2017]. This section introduces the overviews and features about these two source data.

3.1.1 LANDSAT 8

Landsat 8 is the eighth satellite in the Landsat series, launched under the Landsat Data Continuity Mission (LDCM), a collaboration between NASA and the United States Geological Survey (USGS) [USGS 2013]. Landsat represents the only source of global, calibrated, moderate spatial resolution measurements of the Earth's surface that are preserved in a national archive and freely available to the public. The data from the Landsat spacecraft constitute the longest record of the Earth's continental surfaces as seen from space. It is a record unmatched in quality, detail, coverage, and value [USGS 2016]. According to [USGS 2013] The mission has three main objectives.

1. It aims to collect and archive medium resolution (30-meter spatial resolution) multi-spectral image data affording seasonal coverage of the global landmasses for a period of no less than 5 years.

2. It ensures that LDCM data are consistent with the data from the earlier Landsat missions, in terms of acquisition geometry, calibration, coverage characteristics, spectral and spatial characteristics, output product quality, and data availability to permit studies of land cover and land use change over multi-decadal periods.

3. It distributes standard LDCM data products to users on a non-discriminatory basis and at no cost to the users.

The instruments on Landsat 8 and its data processing procedure are introduced in the following of this section, summarised from [Irons et al. 2012], Landsat 8 data user's handbook [USGS 2016], LDCM press kit [USGS 2013], its Data Format Control Book (DFCB) [USGS 2018c] and its official site [NASA 2018a].

3.1.1.1 Instruments

There are two types of sensors on the Landsat 8 observatory, which are Operational Land Imager (OLI) and the Thermal Infrared Sensor (TIRS). These two sensors coincidently collect multi-spectral digital images of the global land surface at a spatial resolution of 30 meters (visible, NIR, SWIR); 100 meters (thermal); and 15 meters (panchromatic).

OLI sensor provides the measurement of nine spectral bands with a spatial resolution of 30m (15m panchromatic band) over a 185km swath from the nominal 705km LDCM spacecraft altitude [NASA 2018d]. More specifically, it operates in coastal/aerosol, visible (RGB), near infrared, short wavelength infrared, panchromatic and cirrus bands.

TIRS measures land surface temperature in two thermal bands with a spatial resolution of 100m across a 185km swath from the nominal 705km altitude [NASA 2018b]. More specifically, it detects the thermal infrared, a long-wavelength light emitted by the Earth depending on the temperature on the surface. Its data can be used to track how land and water are being used.

The Landsat 8 system also includes a ground system responsible for scheduling the observation and managing the science data following its transmission from the spacecraft.

The entire Earth will fall within view once every 16 days due to Landsat 8's near-polar orbit.

3.1.1.2 Bands information

As introduced above, there are in total eleven bands recorded by the Landsat 8, summarised in Table 3.1.

1. Coastal/aerosol band senses deep blues and violets. Because blue light is easily scattered by tiny bits of dust and water in the air and even by air molecules themselves, it however has two main uses suggested by its name: imaging shallow water, and tracking fine particles like dust and smoke.

2. Visible bands contain visible red, green and blue (RGB) bands, to which typical human eyes will respond.

3. Near infrared (NIR) bands provide a important measurement for ecology because healthy plants reflect it. More precisely, the water in their leaves scatters the wavelengths back into the sky.

4. Short wavelength infrared (SWIR) band is sliced into two bands on Landsat 8. This band is useful in distinguishing the dry and wet lands. For example, rocks and soils that look similar in other bands often have strong contrasts in SWIR.

Table 3.1: Summary of bands collected by Landsat 8 [USGS 2016]

Spectral Band	Instrument	Wave Length (μm)	Spatial Resolution (m)
Band 1 - Coastal/Aerosol	OLI	0.435 - 0.451	30
Band 2 - Blue	OLI	0.452 - 0.512	30
Band 3 - Green	OLI	0.533 - 0.590	30
Band 4 - Red	OLI	0.636 - 0.673	30
Band 5 - Near Infrared	OLI	0.851 - 0.879	30
Band 6 - Short Wavelength Infrared	OLI	1.566 - 1.651	30
Band 7 - Short Wavelength Infrared	OLI	2.107 - 2.294	30
Band 8 - Panchromatic	OLI	0.503 - 0.676	15
Band 9 - Cirrus	OLI	1.363 - 1.384	30
Band 10 - Long Wavelength Infrared	TIRS	10.60 - 11.19	100
Band 11 - Long Wavelength Infrared	TIRS	11.50 - 12.51	100

5. Panchromatic band collects the visible spectrum as well, but instead of recording it into three RGB channels, it combines them into one. Because it receives more light at once, it is able to offer a sharper image and the pan sharpening process to provide more detail to RGB channels.

6. Cirrus band records only clouds, especially the cirrus clouds, because the atmosphere underneath it absorbs almost all of this band. This band is especially useful when removing the clouds from the satellite images.

7. Thermal infrared (TIR) band is recorded into two bands on Landsat 8. They record the heat on the ground and are useful to track water consumption such as irrigation.

3.1.1.3 Data correction

There are multiple types of Landsat data available from online portals, such as EarthExplorer, with different level of preprocessing and correction done for the data. The preprocessing is important as the raw data recorded by the Landsat 8 is subject to noise from both sensors and the environment, though instrument calibration are consistently performed through out the mission. This section briefly introduces both the instrument calibration on the Landsat 8 satellite and a general preprocessing procedure for its data correction.

1. Instrument Calibration

   The instrument calibration are recorded in detail in the Landsat 8 Data Users Handbook [USGS 2016] and this introduction is largely based on the handbook. The instruments are calibrated both before the launch and after the launch to ensure the instrument performance requirements are met or exceeded. These include

(a) radiometric requirements:

The system radiometric characterisation and model are recorded and monitored. More specifically, they include detectors' biases (i.e. default values or dark levels), their relative gains and absolute gains on receiving energy (i.e. when measuring the reflected EM radiation within the specified bands) and their degradation over the time. Hence, the striping and other detector-to-detector uniformity issues in the imagery is reduced and the sensor stability is monitored and maintained. Through the validation, individual detector calibration coefficients can be generated to improve the pixel-to-pixel uniformity as well.

(b) geographical requirements

To monitor the stability of the system's geometric and spatial performance and to identify and characterise any systematic variations in the system's geometric parameters as a function of time, temperature, and location; knowledge of multiple alignments and registrations are monitored and refined. The sensor-to-spacecraft alignment is monitored to estimate and reduce attitude error. The band-to-band registration assessment measures the relative alignment of the nine OLI and two TIRS spectral bands. The image-to-image registration assessment is to verify the Landsat 8 requirement that multi-temporal images of the same scene can be successfully co-registered to a 12-meter accuracy. The slope and width of the sensor's view is also characterised.

2. Data Preprocessing

[Young et al. 2017] provides a great overview over the data preprocessing of Landsat imagery and this introduction is based on their work.

The data preprocessing on the Landsat imagery is highly related to some common units measured by the Landsat satellite, shown in Figure 3.1.

Different preprocessing work-flows for some typical Landsat product are shown in Figure 3.2. Some typical preprocessing steps include:

(a) Geometric correction includes georeferencing (alignment of imagery to its correct geographic location) and orthorectifying (correction for the effects of relief and view direction on pixel location) This step ensure the exact positioning of an image.

(b) Conversion to radiance is a preprocessing step whereby the digital numbers recorded by sensors are converted back to radiance (often termed at-sensor radiance). This step is necessary for time-series and multi-spectral analysis due to the sensor degradation and differences between sensors.

(c) Solar correction accounts for solar influences on pixel values, converting at-sensor radiance to top-of-atmosphere (TOA) reflectance. Because the effect of sun depends on Earth-sun distance and solar elevation angle etc., which

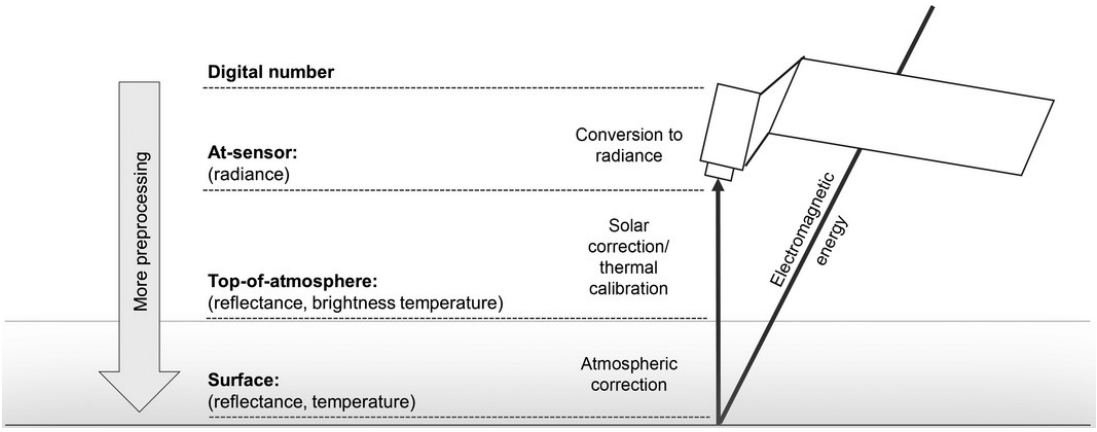


Figure 3.1: The figure is reproduced from [Young et al. 2017] and shows the common units of Landsat imagery used in ecological analysis. The units change as each step of absolute correction is performed: conversion to radiance, solar correction/thermal calibration, and atmospheric correction.

vary with time and latitude, it is necessary to perform this step in analysis across multiple images.

(d) Atmospheric correction attempts to account for influence from the Earth's atmosphere, converting the TOA reflectance to the surface reflectance. However, due to the complex, highly variable and spectrally dependent atmosphere effects, this step is generally hard to perform and can introduce additional errors [Schroeder et al. 2006; Vermote et al. 2016]. Hence, it is not recommended to perform unless necessary, such as comparison with ground reflectance data.

(e) Topographic correction considers the noise in reflectance values caused by illumination effects from slope, aspect and terrain, etc..

(f) Relative radiometric correction converts spectral values into values consistent and comparable across space and times, though the values do not represent the true surface reflectance. This step usually operates on digital numbers collected by sensors and the TOA reflectance to acquire spatially and temporarily consistent images in place of the atmospheric correction.

(g) Absolute radiometric correction accounts for sensor, solar, atmospheric, and topographic effects. This step tries to approximate the "true" and comparable values across time, space and sensors. Although this step can be performed at different stages, it is more suited to perform the processing at, for example, surface reflectance/temperature than at-sensor radiance.

3. Tile Grid System

   The satellite images are processed to a common tiling scheme to provide a consistent global view. Each tile contains 5000 x 5000 30-meter pixels acquired in a given day within its extents. For some tiles, though sometimes there may not be

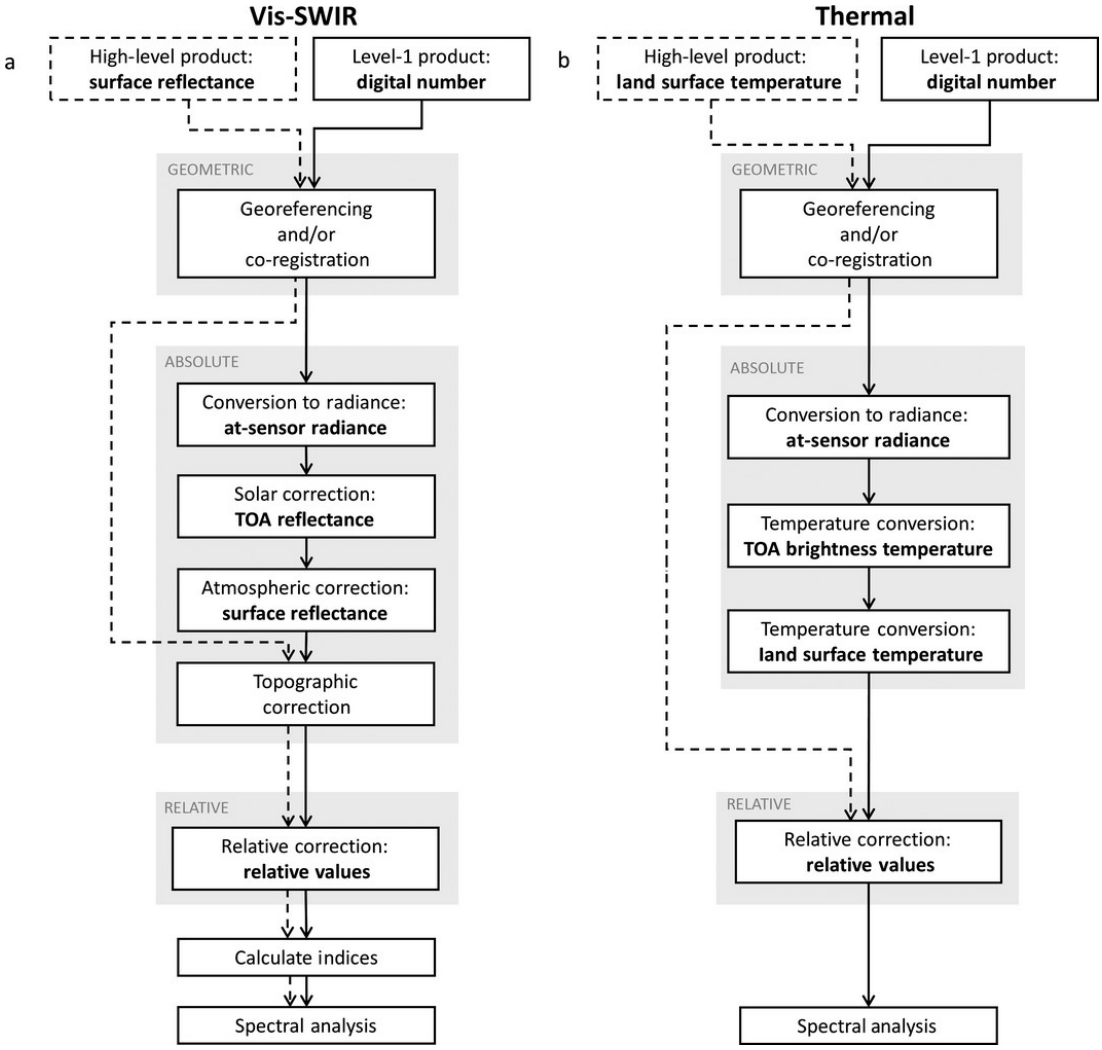


Figure 3.2: The figure is reproduced from [Young et al. 2017] and shows the common units of Landsat imagery used in ecological analysis. The units change as each step of absolute correction is performed: conversion to radiance, solar correction/thermal calibration, and atmospheric correction.

sufficient data to fill the entire tile, the partially filled tile is still recorded. Such stacks of tiles create a time-series satellite images collection and are archived and released under different product level according to its quality and preprocessing procedure.

3.1.2 OpenStreetMap

OpenStreetMap (OSM) [OpenStreetMap contributors 2017] is a project that creates and distributes free geographic data for the world. This section introduces the roads data from OSM, which is used to generate the road raster map as the ground truth.

The OSM data is collected and updated based on crowd sourcing from a variety of data sources [OSM 2017a], including personal survey/walk/journey, local knowledge, public GPS tracelogs, digital or textual data and other non-copyrighted resources. The OSM also launches a mobile application, called Geo Data Collect [HOTID 2017], for people to contribute.

After collecting the data, OSM structures the data into a hierarchical way, where the category, tags and relations between geographical points (defined by its latitude and longitude) are maintained and provided [OSM 2017b]. OSM also reserves the history of its database at at irregular intervals [OSM 2018], providing a possibility for time-series analysis based on OSM data, though extra care need to be taken because there is no suitable database schema for a full history OSM database.

Although the data is presented in a clean and neat format, data contributed from people can be of a various quality, due to lack of practice and knowledge which can be improved by practice and experience in map making [Rehrl and Gröchenig 2016]. As a result, the quality assessment of the OSM data is essential and has been carried out by many researchers. From various assessments, it is generally concluded that Openstreetmap is quite developed and mature as compared to geodata from commercial vendors and its number of absolute and relative errors is falling.[Sehra et al. 2014; Sehra et al. 2013; Kounadi 2009].

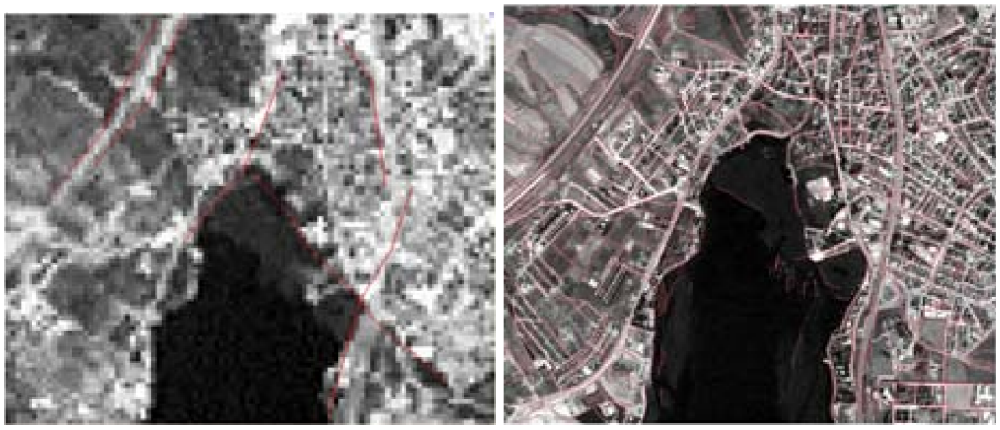
3.2 Combining into Dataset

This section briefly describes how the final dataset is constructed and its influence on models that will apply on it.

3.2.1 Satellite images

We choose a scene in Australian Capital Territory (ACT) as it is our local area, where Australian National University (ANU) sits.

We use the first seven bands of Landsat 8 product of Tier 1 (specifically, L1TP) because other bands are of different resolution (15m for panchromatic band and 100m for TIRS bands) or related to atmospheric processes (Cirrus band). The data has gone through the terrain (geographic) correction with clouds removed. The georegistra-



(a) Landsat (b) Quickbird

Figure 3.3: The figure is reproduced from [Gecen and Sarp 2018]. The results are from the same road extraction algorithm (line module of PCI Geomatica software) on low-resolution (Landsat of 30m resolution) and high-resolution (Quickbird of 2.4m resolution) images.

tion is consistent and within prescribed tolerances ($<$ 12m root mean square error (RMSE)). This data is suitable for pixel-to-pixel time series analysis [USGS 2018b].

Although carefully preprocessed, the data is still not perfect, containing undesirable noise such as illumination, occlusion and distortion etc.. Figure 3.4 gives an example of those noise in our dataset.

The 30m resolution in road extraction is considered to be low-resolution [Gecen and Sarp 2018] and brings inherited blurring effect. In theory, the sensor spatial resolution needs to be at least one-half the diameter of the smallest object of interest [Myint et al. 2011]. As a result, road extraction from Landsat images are considered hard in general and Figure 3.3 gives a comparison of road extraction result on image from Landsat and Quickbird [Gecen and Sarp 2018].

With spatial resolution of 30m, spectral features of roads are largely lost because they are buried with other background information inside the same pixel. Yet its texture feature and topological feature remain useful because the roads will influence the statistical information inside each pixel and such influence will follow the topology of roads. Hence, the uitilizing sub-pixel and contextual information is vital in this case [Blaschke 2010].

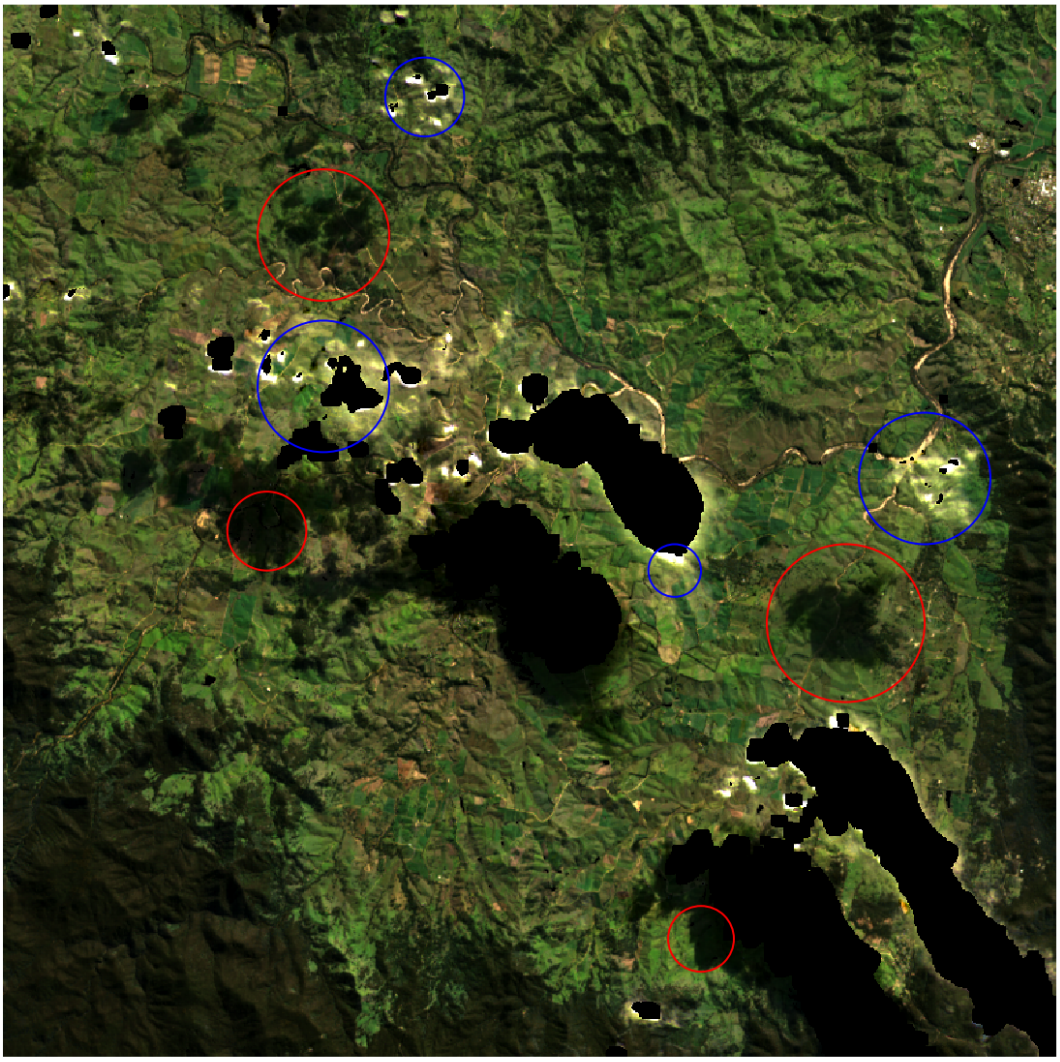


Figure 3.4: An example patch from our Landsat 8 dataset. The scene covers the western suburb of Bega, in the bottom-right corner in our satellite image. The black holes are clouds after removal, filled with invalid value. The fluffy white around those holes are clouds not removed, some of which are labelled by blue circles. Those clouds block the sun and result in obvious occlusion under it, some of which are labelled with red circles. The mountain peaks in the bottom-right corner also block the sun, causing the back-slope to be much darker than others.

3.2.2 Road Raster Map

There are different types of roads data available on OSM. OSM specifies all roads, street and paths using the tag "highway". We limit our candidates into the roads available for vehicles. Though OSM provides pedestrian and cycling ways, they are generally too tiny and has various obstacles overhead, which makes it easily buried into woods or buildings and thus is hard to be recognized from the satellite view.

Regarding the roads available for vehicles, OSM provides a ordered list from the most important to least important. Besides the roads in the list, we also includes some roads under the OSM's special road types into our candidates. Table 3.2 gives a summary on the roads we consider for extraction, though the dataset may use different combination of those roads.

3.2.3 Constructing the Dataset

The roads data from OSM come as shape files, while the satellite data from Landsat 8 product comes as raster maps in Hierarchical Data Format 5 (HD5F) file [NASA 2018c]. It is important to make sure these two set of data to be geographically aligned as they may use different coordinate system. We use the command line tool from Geospatial Data Abstraction Library (GDAL) version 2.1.3 [GDAL Development Team 201x] to project the road data into raster map. The re-projection of vector data (OSM road data in shape file) to the coordinate system of the raster data (Landsat 8 data) is handled by the tool.

After the whole satellite image and road raster map are aligned, they are segmented into corresponding training, validation and test set. Patches without invalid values or with a maximal amount of invalid values are then sampled from the segmented images.

Figure 3.5 is the split of the whole satellite image that we use in practice, where the Canberra city is sliced into all three sets. We take such segmentation because we would like each set to contain both urban and mountain areas. Typically, we collect the pixel-level positive-to-negative ratio in all three set and they are approximately $1:72$, $1:32$ and $1:65$ for respectively training, validation and test set.

The scheme of creating patches are deterministic: Given a window size and a step size, the window rolls across the image segments and all the valid patches are collected into the dataset. The concrete dataset for each model are described in further detail in their Experiment Dataset sections in Chapter 5.

Table 3.2: Summary of candidate roads under tag 'highway' described in [OSM 2017c]

Tag value	Description
	Roads (from most important to least important)
motorway	A restricted access major divided highway, normally with 2 or more running lanes plus emergency hard shoulder. Equivalent to the Freeway, Autobahn, etc..
trunk	The most important roads in a country's system that aren't motorways. (Need not necessarily be a divided highway.)
primary	The next most important roads in a country's system. (Often link larger towns.)
secondary	The next most important roads in a country's system. (Often link towns.)
tertiary	The next most important roads in a country's system. (Often link smaller towns and villages)
unclassified	The least most important through roads in a country's system - i.e. minor roads of a lower classification than tertiary, but which serve a purpose other than access to properties. Often link villages and hamlets. (The word 'unclassified' is a historical artefact of the UK road system and does not mean that the classification is unknown)
residential	Roads which serve as an access to housing, without function of connecting settlements. Often lined with housing.
service	For access roads to, or within an industrial estate, camp site, business park, car park etc.
	Special Road Types
living street	For living streets, which are residential streets where pedestrians have legal priority over cars, speeds are kept very low and where children are allowed to play on the street.
track	Roads for mostly agricultural or forestry uses. Although tracks are often rough with unpaved surfaces, this tag is not describing the quality of a road but its use.
road	A road/way/street/motorway/etc. of unknown type. It can stand for anything ranging from a footpath to a motorway. This tag should only be used temporarily until the road/way/etc. has been properly surveyed.

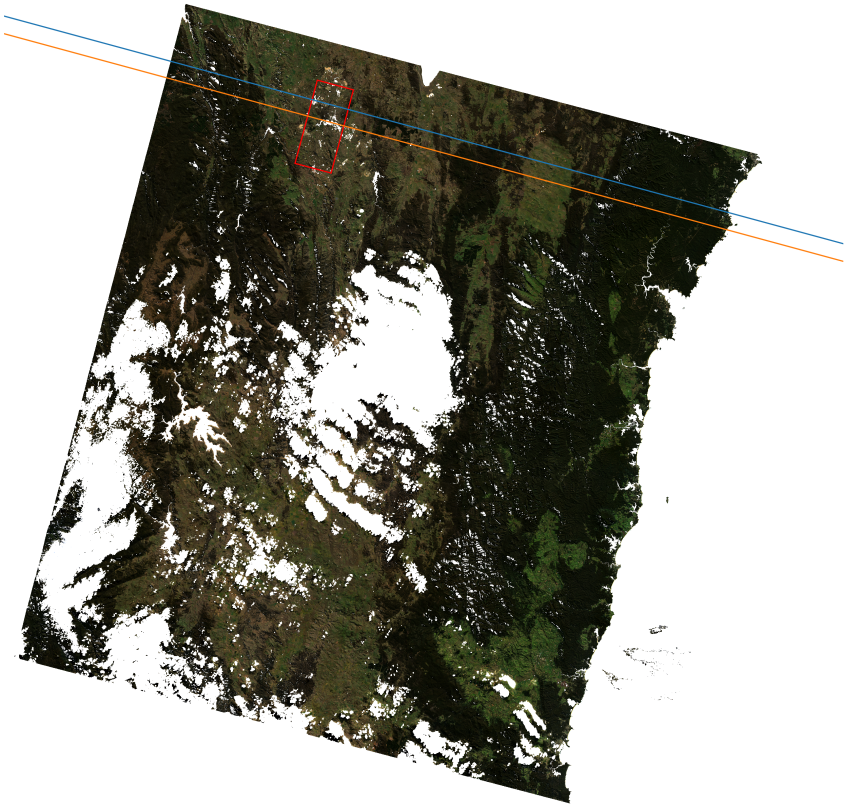


Figure 3.5: The image segment above the blue line are used to construct the test set; the segments between blue and orange line the validation set; and everything below orange line the training set. All invalid values are replaced with the white colour, which are large water body and clouds. It is worth nothing that all segments contain a part of the Canberra city (in the top left) so that they all cover a portion of urban area. The black holes are mainly resulted from clouds and large water body, such as sea water, removal; and they are filled with invalid values. The Canberra city locates roughly in the red box.

Chapter 4

Models for Road Extraction

4.1 Patch Classification for Road Recognition

4.1.1 Problem and Goals

As an initial attempt to the road extraction on our dataset, we borrow the ideas from both center-pixel classification and segmentation in high-resolution image. In high-resolution image, pixels are significantly smaller than object. Hence, objects consist of a group of pixels and each of them reveals a part of the object's information, as shown in the Figure 4.1. In this case, the result of patch classification (Figure 4.1a) can broadcast to every pixel inside it (Figure 4.1c). More specifically, we may approximate the probability of a pixel being road by the probability of its patch containing a road. In this way, we construct a prediction map of a resolution lower than that of original raw image (30m).

In this approach, the problem becomes a classification on patches constructed from the dataset. As there are different types of roads available for extraction, the main goals for the model are to analyse the characteristics of different roads in the satellite images and to produce a road map with lower resolution than our original image, indicating that there are useful information in the satellite image which may help us extract the roads. The results from this model are also a guideline for further analysis.

4.1.2 Model Description

The image patch are standardised to be mean-centered with uniform variance, using Equation 4.1, and then are flattened into a row vector. The model takes the flattened vectors and passes it through three hidden layers and finally output a probability of containing a road using sigmoid function.

$$X(c,i,j) = \frac{1}{\sigma(c)}[X(c,i,j) - \mu(c)] \tag{4.1}$$

where $X(c,i,j)$ denotes the pixel of image patch X in its channel c at the index i, j; $\mu(c)$ and $\sigma(c)$ are the mean and standard deviation of the channel c across the whole dataset, which are calculated using the following Equation 4.2 and Equation 4.3.

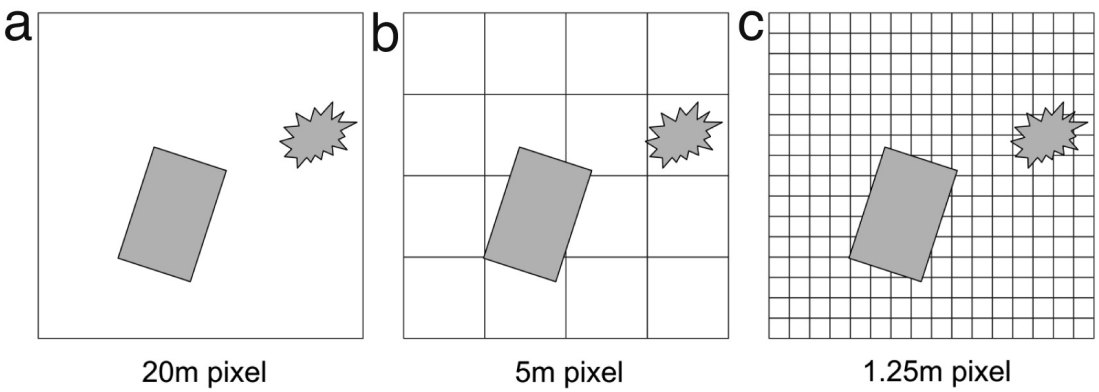


Figure 4.1: The figure is reproduced from [Blaschke 2010]. The grey area forms the objects under consideration and are sliced into the pixels.

$$\mu(c) = \frac{1}{N \times m^2} \sum_{n=1}^{N} \sum_{i,j=0}^{m} X_n(c, i, j) \tag{4.2}$$

$$\sigma(c) = \sqrt{\frac{1}{N \times m^2} \sum_{n=1}^{N} \sum_{i,j=0}^{m} (X_n(c, i, j) - \mu(c))^2} \tag{4.3}$$

where N is the number of image patches in the whole dataset and X_n is the n^{th} image patch in the dataset, of the shape (c, m, m), denoting a square image patch with c channels and m pixels in height and width.

4.2 Semantic Segmentation for Road Extraction

4.2.1 Overall Problem and Goals

As discussed in Section 2.2.3, road extraction can be viewed as an image segmentation task, where each pixel is classified into a class so that the resolution pertains. Hence, Different pixel-level classification approaches can be considered. These include central-pixel classification in a sliding window fashion and fully convolutaional neural network approach.

In the following sections, we first adapt a logistic regression with a sliding window approach onto out road extraction task; then secondly design an FCN-based model designed for the task; and finally further propose a model based on U-net architecture.

4.2.2 Logistic Regression with Sliding Window

4.2.2.1 Problem and Goals

While being aware that such approach has been focused on by [Kirthika and Mookambiga 2011] and several others, we still experiment with a simple model on our dataset in order to demonstrate that there are enough information to support a pixel-level classification despite the 30-meter resolution.

4.2.2.2 Model Description

We implement a logistic regression model using only the linear combination of its input. The input is a flattened vector of the image patch across all available bands. The model then outputs the probability of the central pixel in this image patch. Before being fed into the model, the image patch are either not preprocessed at all, preprocessed to be mean-centered using Equation 4.4 or standardised using Equation 4.1.

$$X(c, i, j) = X(c, i, j) - \mu(c) \tag{4.4}$$

where $X(c, i, j)$ denotes the pixel of image patch X in its channel c at the index i, j; $\mu(c)$ is the mean of channel c across the whole dataset, calculated using Equation 4.2.

4.2.3 Segmentation with Fully Convolutional Network

4.2.3.1 Problem and Goals

After demonstrating that there are enough information for recognising roads against the background, it is natural to improve the result with a more complex model. As fully convolutional networks has become the state-of-the-art technique in image semantic segmentation [Garcia-Garcia et al. 2017], we expands our experiment to use an agile fully convolutional network (referred as *Agile FCN*) for road extraction on our dataset.

The goal for Agile FCN is to improve the result to a reasonable stage and to provide a usable road extraction tool for the later time-series analysis.

4.2.3.2 Model Description

Agile FCN is based on convolutional layers only. The model is thus able to take an image patch as input and output the probability map of roads for that patch directly. Each layer in the model contains a set of convolution kernels of the size 3x3 and/or 1x1. The 1x1 kernel can pertain the pixel-level information whereas 3x3 convolution is included to provide contextual information. Such design of layer with multiple kernels are originally inspired by inception module [Szegedy et al. 2014].

It is referred as Agile FCN because it contains only two or three hidden layers and is quite shallow, compared to fully convolutional networks with more than 10 layers in most literature [Long et al. 2014; Badrinarayanan et al. 2015]. Such a design is because we are working on a dataset of 30m resolution. Thus, it is expected that the roads are fine on our dataset, taking up a few pixels in width. Hence, a small network with its relatively small receptive field is still expected to capture enough spacial context.

An example of Agile FCN is shown in Figure 4.2, which has 3 hidden layers and each hidden layer contains both 3x3 and 1x1 kernels.

The output of the agile FCN is given by a 1x1 convolution followed by a softmax. The last 1x1 convolution is necessary because it can reduce the channel numbers to that of a prediction map before the softmax, while focuses more on the pixel-level information passed through for the purpose of resolution.

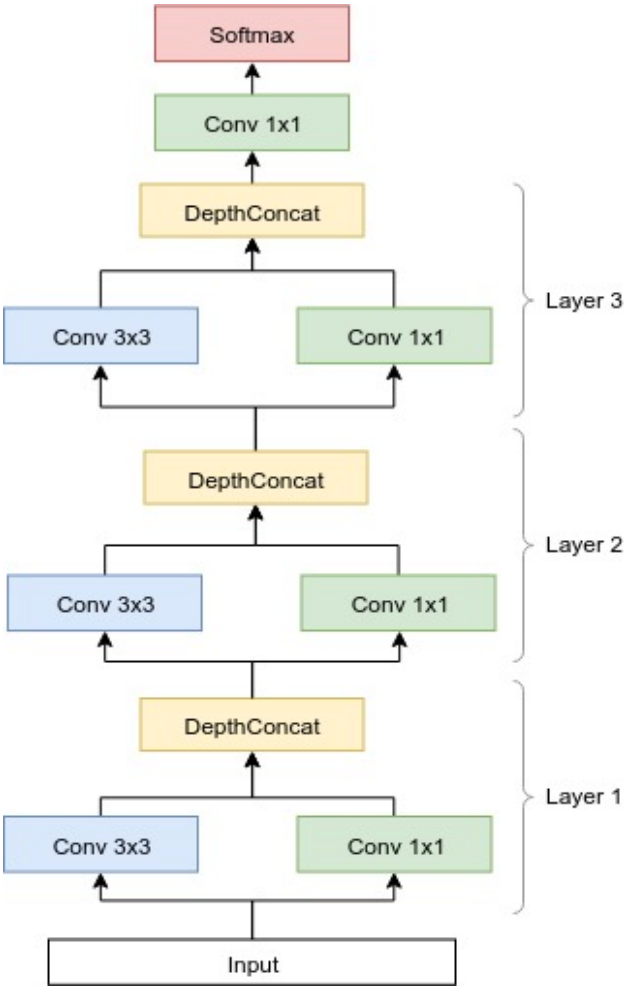


Figure 4.2: FilterConcat is the concatenation operation that stacks the output feature maps of all previous kernels, forming the input of next layers.

4.2.4 Segmentation with U-net

4.2.5 Problem and Goals

Though the Agile-FCN performs reasonably better than logistic regression model, it is possible to further improve the result on our dataset. As the Agile FCN is designed based on the idea of combining both pixel-level and contextual information, some previous work, such as U-net [Ronneberger et al. 2015] and SegNet [Badrinarayanan et al. 2015], have extended this idea and has become state-of-the-art models in semantic segmentation. Hence, we would like to combine our Agile FCN with those successful models to achieve a better result.

4.2.6 Model Description

The model proposed here combines the structure of original U-net architecture and our Agile FCN. The U-shaped structure in U-net is referred as *U connection* and components from Agile FCN acts as an additional *input bridge*. Hence our model is referred as *BridgeUnet* and can be summarised from following two perspectives:

1. U connection

   Similar to original U-net, the U connection is of an encoder-decoder structure, containing a down-sampling path, a tunnel and an up-sampling path. The input image is first down-sampled into a compact representation, and after gone through a tunnel it is up-sampled back to its original size. Such structure provides high-level information with its localisation. To compensate low-level detail and localisation to the high-level semantic features, model concatenates the feature maps in the encoder with corresponding maps in the decoder.

2. Input bridge

   To extends the idea of providing low-level detail to the model, a path independent from the U connection connects the input image directly to the output segmentation map. Such independent path is not influenced by the performance of high-level semantic features and thus can focus on those pixel-level details that may be missed in U connection. Due to its role, the path consists mostly of 1-by-1 convolutions.

The BridgeUnet has only a few more convolution kernels than Agile FCN but a much larger receptive fields, due to its U connection. We choose to enlarge the receptive fields because the topology of roads can be useful and needs a larger receptive fields than Agile FCN to be utilised.

Figure 4.3 shows an example BridgeUnet with a input of 128-by-128 patch. The bridge in the example consists of only 1-by-1 convolutions.

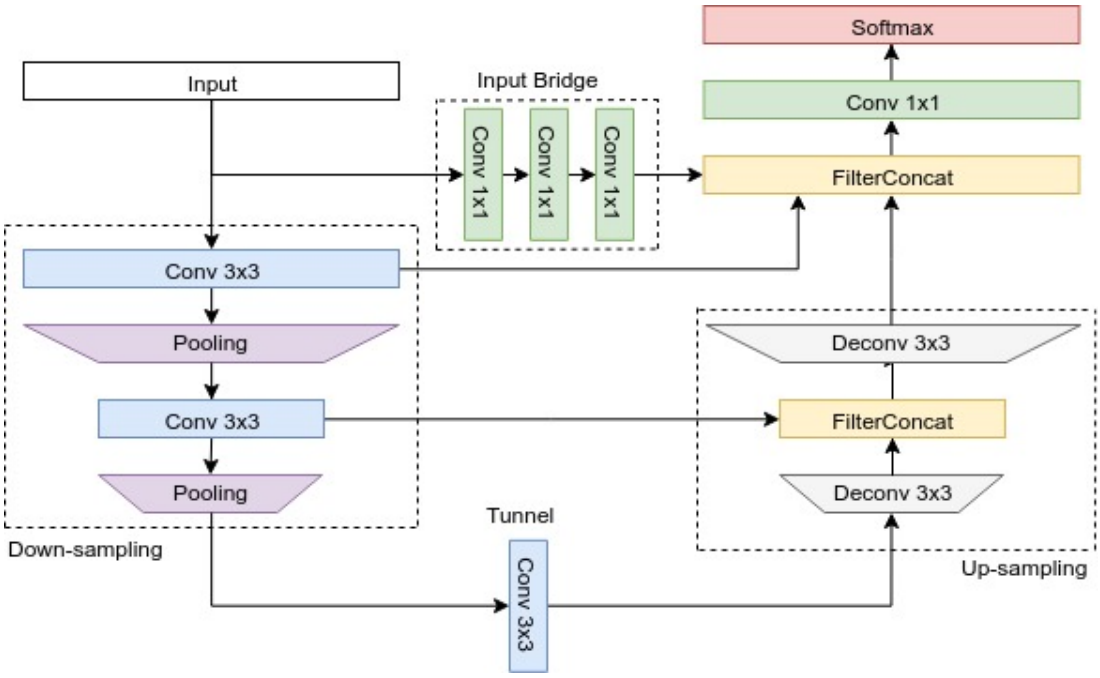


Figure 4.3: FilterConcat is the concatenation operation that stacks the output feature maps of all previous kernels, forming the input of next layers.

Chapter 5

Experiments and Results

This chapter presents the results of our experiments with their set-ups. As we tried several different architecture to extract the road, some insights are gained both within each architecture and across different architectures. In this chapter, two architectures and their experiments will be discussed separately and a comparison between them will be made.

Our implementation is based on TensorFlow [Abadi et al. 2015], including the use of scikit-learn [Pedregosa et al. 2011], numpy [Jones et al. 01] and matplotlib [Hunter 2007].

5.1 Patch Classification on Road Recognition

5.1.1 Model Experimental Settings

The size of input image patches, before flattened out, is fixed into 28x28 pixels across all the 7 bands. The choice of a width of 28 pixels is because it can cover a reasonably large area and provide enough information to recognise the main object within that area. It is also the choice of SAT-4/6 dataset [Zhong et al. 2017]. Three hidden layers are fixed to contain 128, 256 and 512 neurons respectively. It is chosen to use 64 image patches as a batch with a learning rate of $1e-5$. The model is trained for 10 epochs in the training phase with an Adam optimiser [Kingma et al. 2014]. The experiment are performed on different dataset with a common model. Datasets with different roads will be introduced in the next section.

To visualise the result, a prediction map is produced by approximating the probability of each pixel being of road class. The approximation is done by assigning the probability of a patch containing a road to every pixel within that patch and the pixel-level probability is averaged over all assignments for pixels covered by multiple patches. As a result, a heat map indicating the appearance of roads can be shown against the original satellite image.

5.1.2 Experiment Dataset

The patches are sampled using the scheme described in Section 3.2.3 with a 8-by-8 window and a stride of 28 pixels at both directions. Patch is considered to be valid if

Table 5.1: Road Classes

Road Classes	OSM Type	Reasons
”Big roads”	motorway trunk primary	These roads are generally as wide as 4 lanes for both directions due to their usage and requirement.
”Medium roads”	secondary tertiary	These roads usually have 2 lanes for both directions and often link towns.
”Small roads”	unclassified track	These roads are usually used in the local area to access properties and link small villages
”Urban roads”	residential service living street road	These roads usually appear in urban area and around buildings. The roads of unknown road type from OSM tag ”roads” are considered as urban area because those roads only appear around urban area in the are covered by our satellite images.

it contains no invalid value. In order to avoid confusing the model, patch is labelled into containing roads only when it contains more than 1% pixels in road class. The patches are sample from the whole satellite image. The first 75% of the constructed dataset are used for training and the next 10% are used for validation and the last 15% are used for testing.

To analyse different road types, multiple datasets are constructed using different combinations of roads. All the roads under consideration are discussed and listed in Section 3.2.2. We group similar roads in to classes. Table 5.1 describes those classes with corresponding reasons according to OSM description, which is listed in Table 3.2.

5.1.3 Evaluation Metrics

We typically use Area Under the Curve (AUC) and average precision (AP) to measure the performance of models because they operate on different dataset and output probabilities for their prediction. Using AUC and AP can remove the influence of a threshold value and is comparable across models on different datasets.

AUC measures the area under Receiver Operating Characteristic (ROC) curve [Fawcett 2006], while the ROC curve is created by plotting the true positive rate (TPR) on the Y axis against the false positive rate (FPR) on the X axis at various threshold settings (ranging from $-\infty$ to $+\infty$ with sufficient small steps). Then, AUC is given by

equation 5.1.

In ROC space, point (0,0) denotes the model predicting only to be negative; point (0,1) denotes a perfect classification; and (1,1) the model predicting only to be positive. The $y = x$ line in ROC space denotes the performance of a random-guessing classifier when evaluated on sufficiently large dataset because when it predicts positive it is expected to predict correctly in half of its time (in a binary classification task as in our case) [Fawcett 2006], whereas a better classifier will have more points in the top-left corner, closer to point (0,1) and thus covering more space under its curve. Hence, any classifier better than random guessing will have an AUC score greater than 0.5. ROC is also invariant to the class skew and is inherited by AUC [Fawcett 2006]. AUC can summarise the overall location of an ROC curve relative to the diagonal [Hanley and Mcneil 1982] and gives an indication of the amount of "work done" by a classification scheme [Bradley 1997].

$$AUC = \int_0^1 T(F)dF \tag{5.1}$$

where $T(F)$ is the true positive rate when false positive rate equals F.

AP are used in different fields with different meaning. Here, AP is introduced as measuring the area under precision-recall curves [Zhu 2004] because it is calculated by integrating the precision value across the corresponding recall value across (0,1) as shown in equation 5.2, while the precision-recall curve is created by plotting the precision on Y axis against the recall value on X axis at various threshold and is able to show the trade-off between precision and recall [Davis and Goadrich 2006]. The precision-recall curve has a larger difference between models when evaluated on a largely skewed dataset [Bockhorst and Craven 2005].

$$AP = \int_0^1 p(r)dr \tag{5.2}$$

where $p(r)$ is the precision of the model prediction when it has a recall of r.

5.1.4 Results and Discussion

The experiments start from the "big roads" and increasingly includes roads from lower classes. Experiments are also done on medium and small roads as well as urban roads. Table 5.2 summarises the results, in which average precision (AP) and area under the curve (AUC) are evaluated on the test set. The positive ratio in the table is the proportion of positive examples in the whole dataset. Figure 5.1 shows a set of prediction maps from models trained with different types of road.

It is observed that the model on "big roads" and "medium roads" (experiment 1,2) have relatively high AUC scores with low AP scores. High AUC scores are because that the "big and medium roads" usually pass by urban areas so that model can recall the majority of positive examples by only urban detection. The model can increase the recall while maintaining a low FPR because of the large number of negative examples due to the sparsity of "big and medium roads" in the large wild field. Whereas low AP

Table 5.2: Summary of patch classification on different combination of roads

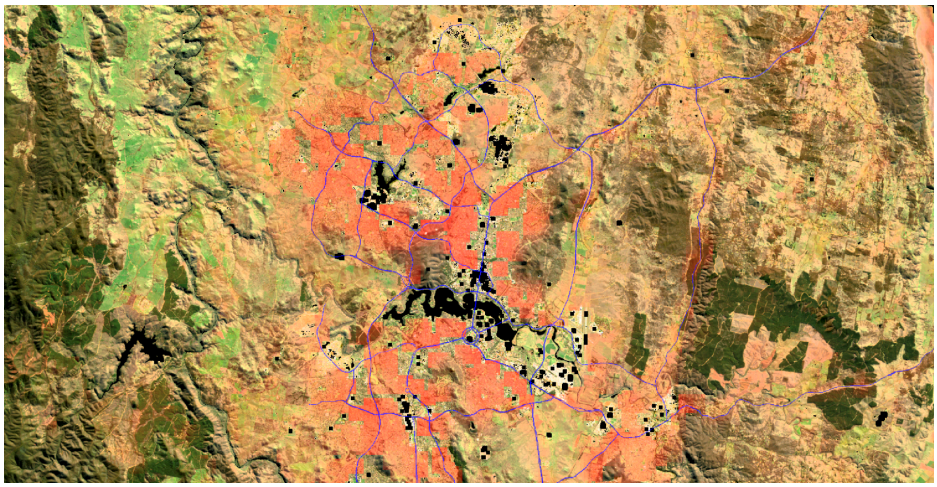
No.	Road Types	positive ratio	AP	AUC
1.	Big roads	0.03	0.136	0.801
2.	Big & medium roads	0.11	0.338	0.768
3.	Big, medium & small roads	0.34	0.518	0.671
4.	Big, medium, small & urban roads	0.36	0.548	0.676
5.	Medium & small roads	0.30	0.460	0.664
6.	Urban roads	0.04	0.413	0.856
7.	Big & urban roads	0.06	0.394	0.828

scores might be because model will risk having more false positive prediction when trying to recall more roads, thus precision decreases rapidly given a relatively small number of positive examples. As shown in Figure 5.1a, model becomes less confident when it comes to road in the up-right corner, where the background are similar to nearby mountains.

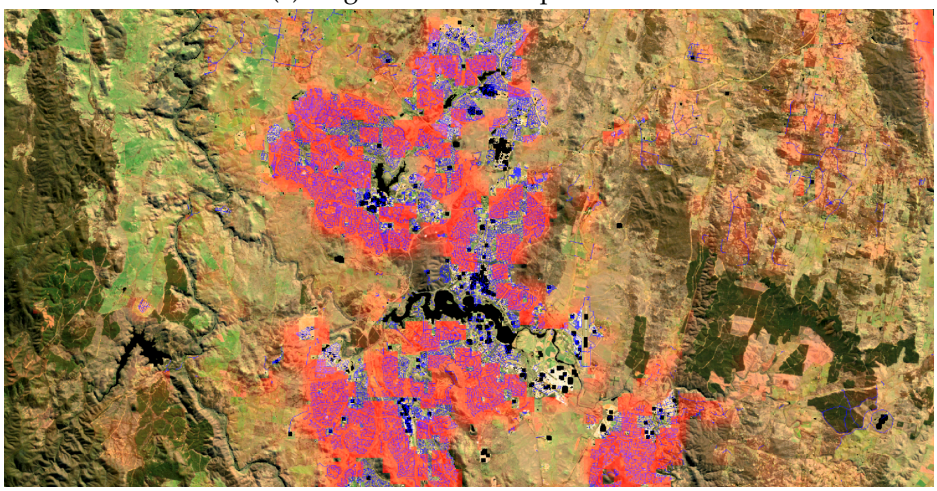
The above argument also applies on the model trained when ”urban roads” takes up the majority of roads (experiment 6,7). Yet the model gains a relatively higher AP score because ”urban roads” are more centralised in the urban area and contain less roads scattered in the fields, making it easier to recall more roads for the model, as shown in Figure 5.1b.

However, when the ”small roads” are considered (experiments 3,4,5), AUC scores drop to near 0.7 or even below. The drop in AUC scores is within expectation because surroundings of those ”small roads” cover most of the terrains and backgrounds, including mountain ridges, valleys, vegetation and barrier lands, etc. Various backgrounds, along with less negative examples (as a higher positive rate), prevent models from recalling the roads while maintaining a low false positive rate. As a result, it is suggested that ”small roads” might need a more powerful model to be more successfully recognised. This is also supported by Figure 5.1c, model is not confident as it tends to assign a similar probability of containing roads to almost every patch.

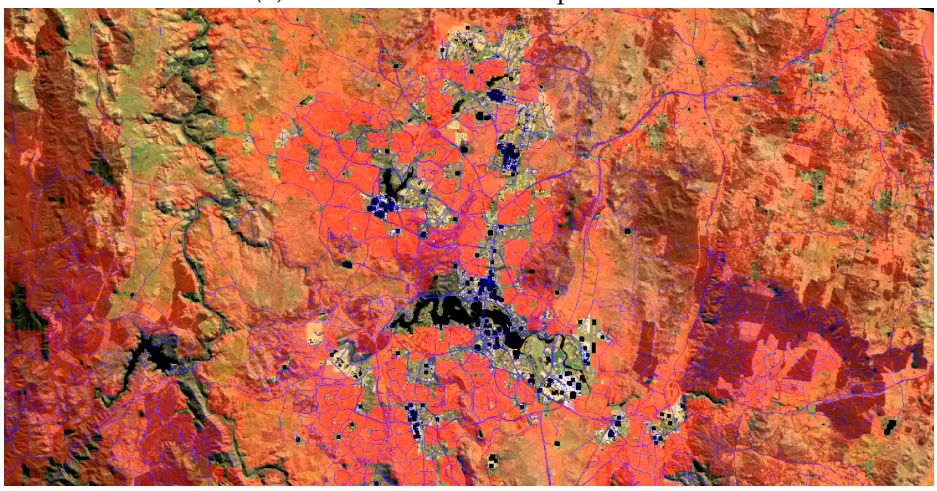
Overall, it is suggested that the model is able to learn to recognise where roads are more probable to appear and produce a reasonable heat map for different combinations of road types, as shown in Figure 5.1.



(a) ”big roads” from experiment 1



(b) ”urban roads” from experiment 6



(c) ”big, medium & small roads” from experiment 3

Figure 5.1: The probability of each pixel being road is added into its Red channel and more red means more probable of being roads; and blue lines denote the ground truth. Some area with no prediction are due to invalid values.

5.2 Semantic Segmentation for Road Extraction

5.2.1 Logistic Regression with Sliding Window

5.2.1.1 Model Experimental Settings

The logistic regression model takes a 8-by-8 pixels image patch as input, which is flatten into a row vector. Compared to 28-by-28 window, 8-by-8 window is much smaller. Hence, the simple model can focus more on the central pixel while provided with enough context information. Another reason for the small window lies in that we try to avoid too many parameters and irrelevant information to the classification of central pixel.

Though model structure are fixed, different preprocessing methods and balancing scheme are analysed. Balancing scheme are essential because of the imbalance in our dataset, which is discussed more in detail in the following section 5.2.1.2. Balancing scheme includes over-sampling [Bowyer et al. 2011] and weighting in our experiments.

The over-sampling is done on the minor class, road class in our case, to increase the number of that class so that it is encountered by model in training phase more often. The number of minor class are usually over-sampled to be equivalent to that of major class [Bowyer et al. 2011]. In our experiment, we use a scheme similar to [Yun et al. 2011]'s, where positive examples (roads) are selected more often than the negative (backgrounds) so that the number of road examples encountered by the model takes up a pre-defined proportion of all the examples it encounters. Reasons of our choice are as follow:

1. It retains all information in original dataset, compared to under-sampling [Qazi and Raza 2012]. Loss of information from original dataset is not desired in our case because satellite image covers a large range of terrains that can be of very different distributions.

2. It does not need to create synthetic images, compared to SMOTE [Yun et al. 2011], which can be hard regarding satellite image with 7 bands.

Apart from over-sampling or under-sampling, weighting the loss of misclassification of different classes are also an effective way to handle an imbalanced data set, which is also know as "cost-sensitive learning by example weighting" [Abe et al. 2004]. We adopt a common weighting scheme where the weight for positive class is the percentage of the negative class in the whole dataset and vies-versa, i.e. the weight for negative class is the percentage of the positive class in the whole dataset. We adopt such weighting scheme because it is dependent on the statistical information from the dataset and does not need to be tuned compared to a fixed weighting scheme.

For each model, regularisation parameters are tuned and preprocessing method or balancing scheme are chosen. The model are trained for 20 epochs with a learning

rates of $1e-5$ and Adam optimiser [Kingma et al. 2014]. The loss function adopts the mean cross entropy loss.

5.2.1.2 Experiment Dataset

The whole image is split into three slices and training, validation and test sets are constructed with the scheme, as described in Section 3.2.3. The window size is set to 8 pixels with stride of 1 pixel on both axis. Same as patch classification, a valid patch is a patch with no invalid value. The label for each patch is then the label of its central pixel.

The road types chosen in this experiment are "big, medium and small roads", described in Table 5.1. These roads include roads that are expected to be observable in our satellite images and roads worth further sub-pixel analysis, as discussed in the patch classification experiment (Section 5.1). We do not include "urban roads" because those roads may hold our model back to a model for urban area detection, instead of a model for road extraction.

After constructed, the training set contains in total approximately 20,000,000 image patches with positive class (road) taking up a proportion of approximately 0.014 and negative class (background) 0.986, that is, a positive-to-negative ratio of approximately $1:70$.

5.2.1.3 Evaluation Metrics

Apart from Area Under the Curve (AUC) and the Average Precision (AP), we also report the balanced accuracy (BA) [Brodersen et al. 2010] in this experiment because it is able to give a measurement of accuracy of a classifier while overcome the influence of a imbalanced test set. The balanced accuracy are calculated with equation 5.3. To obtain the true positive rate and true negative rate, the threshold of prediction is set to 0.5.

$$BA = \frac{1}{2}\left(\mathrm{TPR} + \mathrm{TNR}\right) \tag{5.3}$$

where TPR is true positive rate and TNR is true negative rate.

5.2.2 Results and Discussion

Table 5.3 summarises results from the experiments with different settings. Figure 5.3 shows some example results of road extraction.

From Table 5.3a, it is observed that model gives a better result when input is standardised than that to be mean-centered. And mean-centering the input helps model perform better on the test set, compared to the model with no preprocessing at all. Such observation also suggests that despite those software- and hardware-based data correction and processing on the satellite image, they are far from perfect and bring along their own noise [Paul M. Mather 2011].

From Table 5.3b, it is observed that the model with a weighted loss reports the best performance and over-sampling also helps the model to perform much better thean the one without any balancing scheme, which is not much better than randomly guessing. Although the performance increases along with the over-sampling rate, we argue that the weighting scheme performs better because if over-sampling at a rate of $0.5 + \epsilon$, where $\epsilon \in (0, 0.5)$, the previous major class will become the minor class and gives a similar result to the model with a over-sampling rate of $0.5 - \epsilon$.

We also experiments models with both over-sampling and weighting, which gives a similar result to that with no balancing scheme because models almost always predict examples to be roads. Thus those results are not shown in the table.

It is worth mentioning that results reported on Table 5.3a are achieved when models use the weighting scheme. Similarly, results reported on Table 5.3b are achieved when models use the standardising as preprocessing. That is to say, standardising are the best under all balancing schemes in our experiments and the weighing is the best under all preprocessing methods in our experiments.

We also observe that model performs better on test and validation sets, compared to training set. This might be caused by the different class balance in each set, resulted from the specific way we segment our dataset, discussed in Section 3.2.3.

Figure 5.3 shows the model, despite being naive, still learns to extract some apparent roads, e.g. the main roads going up to the top-right corner in Figure 5.3a. Yet small roads in the mountain area are seldom extracted, e.g. the left part in Figure 5.3b. Yet because the model performs only linear combination of those features, it is expected that it cannot segment the roads finely. For example, it produces a noisy output at the upper part of Figure 5.3a and a large area of red-ish "fog" above the mountain in the left part of Figure 5.3b.

Such result demonstrates that it is possible to extract roads from a 30m resolution satellite images with extra spectral features (7 bands compared to 3 bands in RGB image) and a more complicated model may be explored to elevate the result.

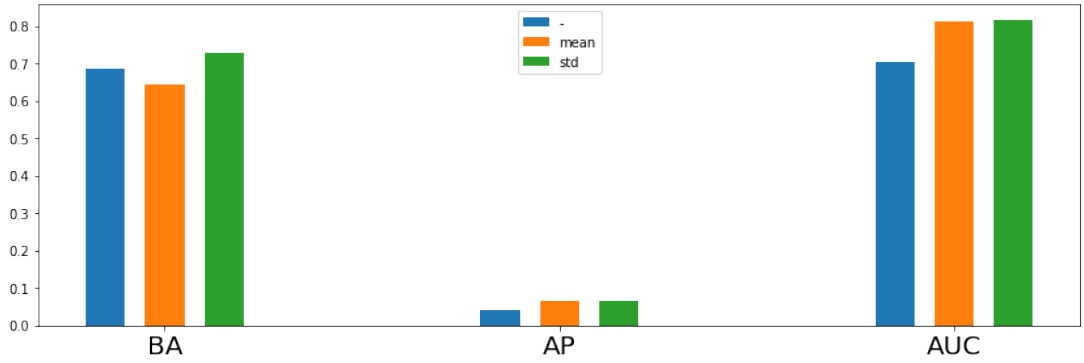
Table 5.3: Logistic Regression with Sliding Window

Preprocessing Method	BA			AP			AUC		
	train	val	test	train	val	test	train	val	test
–	0.631	0.651	0.687	0.028	0.065	0.040	0.647	0.669	0.706
mean	0.609	0.622	0.644	0.053	0.093	0.065	0.755	0.757	0.814
std	0.679	0.663	**0.727**	0.052	0.095	**0.066**	0.752	0.765	**0.817**

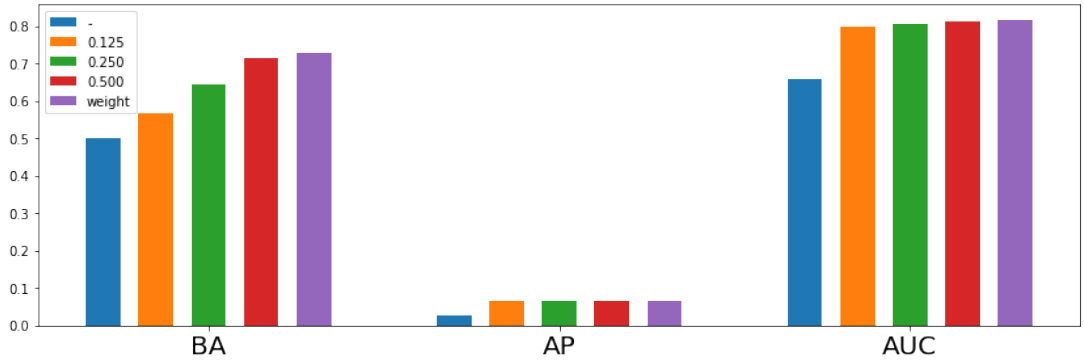
(a) For preprocessing method, '–' stands for no preprocessing, 'mean' for mean-centring and 'std' for standardising.

Balancing Scheme	BA			AP			AUC		
	train	val	test	train	val	test	train	val	test
–	0.500	0.500	0.500	0.022	0.044	0.026	0.615	0.613	0.658
0.125	0.536	0.540	0.567	0.057	0.092	0.065	0.749	0.745	0.800
0.250	0.590	0.591	0.645	0.057	0.091	0.065	0.754	0.748	0.805
0.500	0.675	0.653	0.715	0.057	0.093	0.065	0.751	0.753	0.814
weight	0.679	0.663	**0.727**	0.052	0.095	**0.066**	0.752	0.765	**0.817**

(b) '–' denotes no balancing is used; '0.500', '0.250' and '0.125' denote that the positive class is over-sampled to a proportion of, respectively, 50%, 25% and 12.5% of the whole data ever fed into the model in training phase; and 'weight' denotes the weighting scheme is used on the loss function.

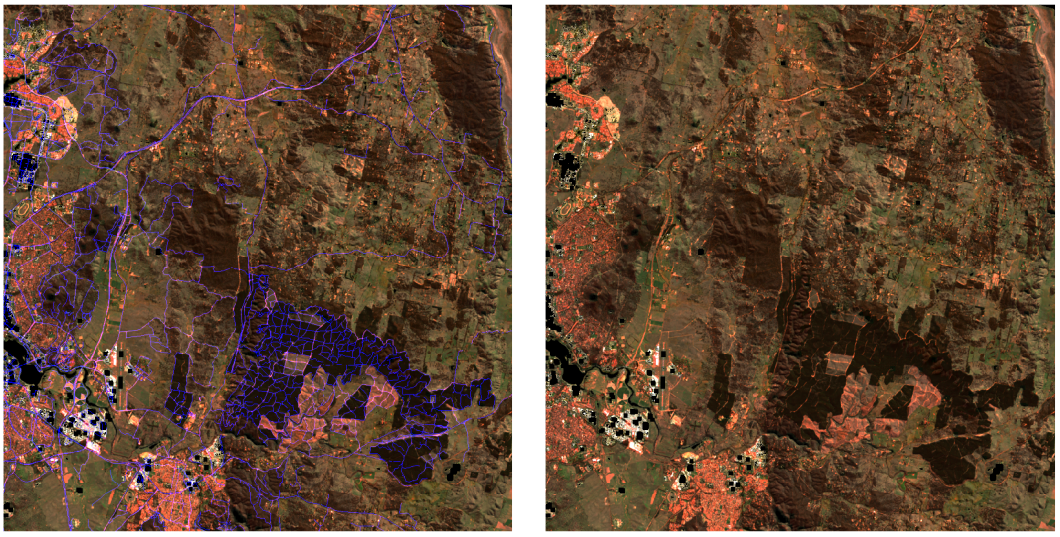


(a) Results on test set from models in Table 5.3a

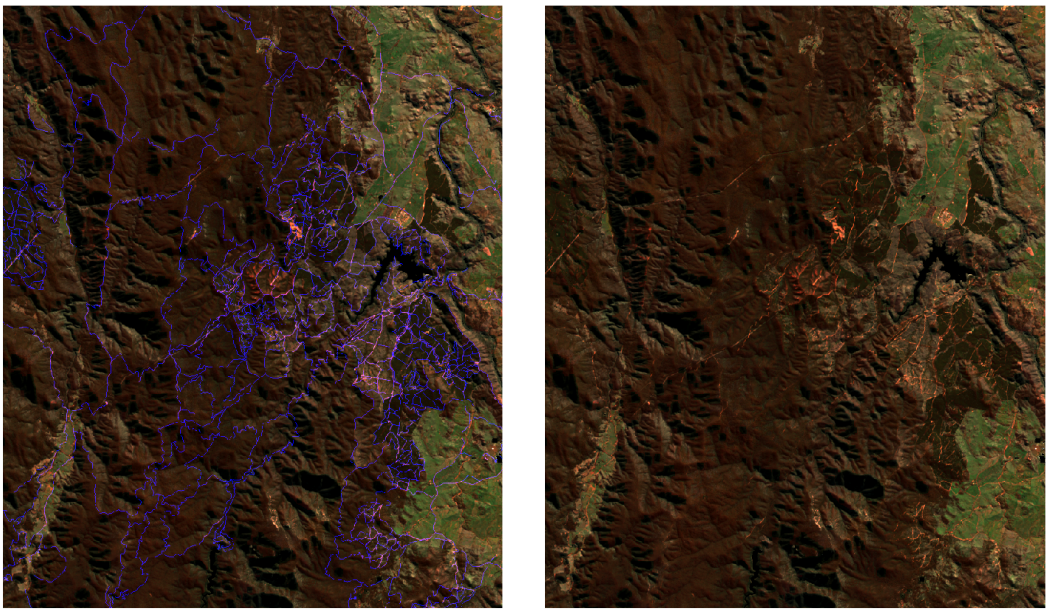


(b) Results on test set from models in Table 5.3b

Figure 5.2: Bar plots of results on test set from models in Table 5.3



(a) CBR airport (bottom-left corner) and its neighbour



(b) Mountain area

Figure 5.3: The prediction map is generated by model with "std" preprocessing and "weight" balancing. It is plot after a post-processing to highlight the roads, which is expressed by Equation 5.4. The processed probability of roads are added into the Red channel and the blue lines denote the ground truth. A prediction map without ground truth is also provided at the right column for comparison.

$$\begin{aligned} P' &= -log(-P + 1 + \epsilon) \\ P'_{norm} &= \frac{P' - \min(P')}{\max(P') - \min(P')} \end{aligned} \tag{5.4}$$

where P is the original prediction map and P'_{norm} is that after processing; ϵ is a negligible positive to prevent a $+\infty$ probability and is chosen to be $1e-9$ in the practice.

5.2.3 Segmentation with Fully Convolutional Network

5.2.3.1 Model Experimental Settings

Because we would like to retain the size of input image, all convolutional layers are configured to use padding with a stride of 1 along both axis and no pooling layer are used. The convolutional layer also includes an optional batch normalisation [Ioffe and Szegedy 2015] and a rectified linear activation [Krizhevsky et al. 2012]. The input is an image patch of 128-by-128 pixels. The learning rate is set to $1e-5$ with 1 image patch per batch. The loss is calculated with pixel-wise cross entropy, averaged over all the pixel inside a batch. The model is trained for 20 epochs with Adam optimiser [Kingma et al. 2014] before evaluated on the test set.

We standardise the input as a preprocessing step and weight the loss according to the classes' proportion in dataset, as those settings result in a better performance on our dataset as discussed in Section 5.2.1.

As described in Section 4.2.3, Agile FCN is built from layers that consist of one or more kernels with different number of output feature maps. As a result, each layer can be denoted by a bracket containing both the type of convolution and the number of their output feature maps. Typically, *k*x*k* : *n* denotes a *k*x*k* kernel outputting *n* feature maps and each bracket contain one or more such pairs.

With such notation, Agile FCN shown in Figure 4.2 can be denoted as:

$$\begin{pmatrix}3\text{x}3 : 32\\ 1\text{x}1 : 32\end{pmatrix}\begin{pmatrix}3\text{x}3 : 64\\ 1\text{x}1 : 64\end{pmatrix}\begin{pmatrix}3\text{x}3 : 128\\ 1\text{x}1 : 128\end{pmatrix}$$

where the model contains three layers as denoted by three brackets; and in total $32+32=64$ feature maps are concatenated into the output after the first layers, $64+64=128$ after the second and $128+128=256$ after the third, which is then followed by a 1x1 convolution that is dropped in the notation.

Table 5.4 lists different settings of Agile FCN with the above notation.

As shown in the table, four models have their own different settings and thus result in some different features among them, which is discussed as followed.

- Model A only uses spectral features of each pixel without its context. Such approach may learn to discover the percent of spectral reflectance contributed by roads within each pixel, sometimes also referred as per-pixel classification [Jensen and Lulla 1987].

- Model B has only 3-by-3 kernels, which is able to learn a series of filters and utilise the contextual information but may have trouble from lost pixel-level information.

- Model C has a receptive field of 5-by-5 window when calculating the logit, yet is able to utilise both pixel-level and contextual information.

- Model D are deeper than C and has extra pixel-level information specifically retained through 1-by-1 convolution.

Table 5.4: Agile FCN Structures in Experiments

Model Type	Model Structure	Description
A	$(1x1:32)\ (1x1:64)\ (1x1:128)$	The model contains three layers with only one 1x1 kernel for each.
B	$(3x3:32)\ (3x3:64)\ (3x3:128)$	The model contains three layers with only one 3x3 kernel for each.
C	$\binom{3x3:32}{1x1:32}\ \binom{3x3:64}{1x1:64}$	The model contains two layers with two kernels for each, which are 3x3 and 1x1 respectively.
D	$\binom{3x3:32}{1x1:32}\ \binom{3x3:64}{1x1:64}\ \binom{3x3:128}{1x1:128}$	The model contains three layers with two kernels for each, which are 3x3 and 1x1 respectively.

5.2.3.2 Experiment Dataset

The dataset is constructed in the way described in Section 3.2.3, with a 128-by-128 window and a stride of 16 pixels. To cover the major area of original satellite image, patch is considered invalid only when invalid pixels are more than 1% in the patch, compared to zero tolerance in logistic regression. To waive the influence of invalid pixels in patches, they are replaced by mean value of their channel.

The weighting scheme described in Section 5.2.1 is used. As each pixel becomes a training example here, the number of examples for each class becomes the number of valid pixels labelled into each class. Such scheme gives a positive-to-negative ratio of approximately $1:67$ in training set. Compared to that in logistic regression (approximately $1:70$),the class imbalance decreases slightly, because small areas surrounded by invalid values are excluded when using a relatively large image patch and those small areas are more often in the mountain area where roads rarely appear.

5.2.3.3 Evaluation Metrics

We use a similar evaluation metrics to that in logistic regression, which are balanced accuracy (BA), average precision (AP) and area under the curve (AUC). Metrics are evaluated on training, validation and test set.

5.2.3.4 Results and Discussion

Table 5.5 summarises results from experiments with different structures and Figure 5.4 gives an overview of performance from each structures on test set.

Agile FCN sometimes performs better on test set than training set, which again might be because the different class balance between sets as discussed in Section 3.2.3.

It is expected that model A, which is the traditional per-pixel classification, performs the worst and its result is similar to that from Logistic Regression (Table 5.3) be-

Table 5.5: Agile FCN

Model	BA			AP			AUC		
	train	val	test	train	val	test	train	val	test
A	0.633	0.684	0.722	0.062	0.133	0.086	0.724	0.780	0.825
B	0.790	0.786	0.802	0.202	0.232	0.153	0.877	0.872	0.889
C	0.715	0.730	0.764	0.147	0.189	0.119	0.815	0.840	0.838
D	0.799	0.792	**0.807**	0.222	0.239	**0.169**	0.887	0.878	**0.893**

* The notation for Model denotes models listed in Table 5.4

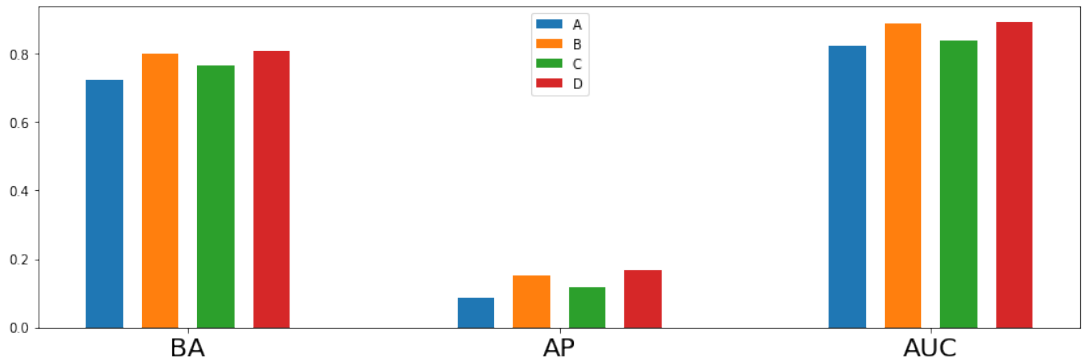


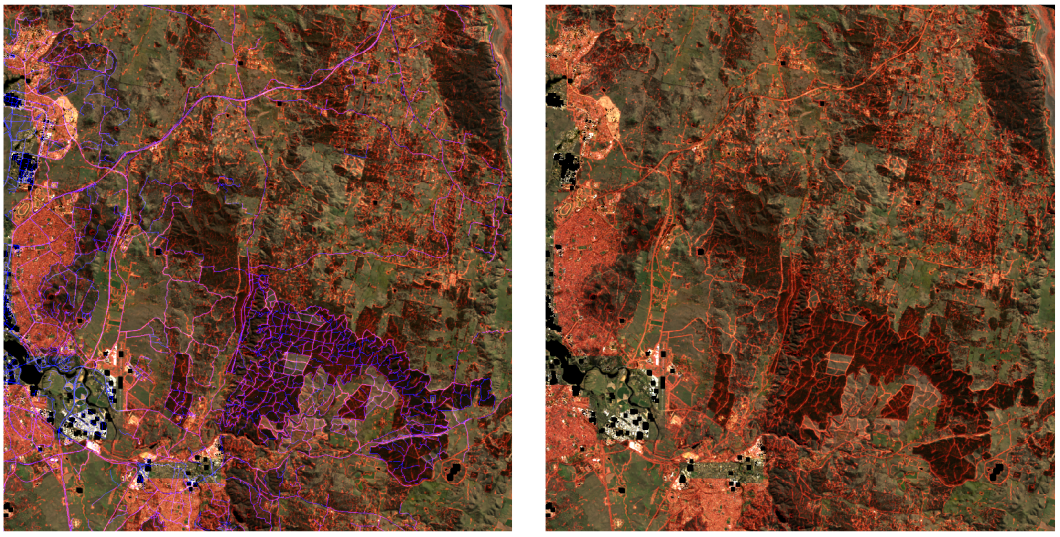
Figure 5.4: Bar plots of results on test set from models in Table 5.5

cause it only does not account for any contextual information. Agile FCN with other settings all outperforms Logistic Regression unsurprisingly; It is however surprising that model B, which essentially learns a series of filters, outperforms the model C, our Agile FCN with only two layers, because model B is originally expected to suffer from the loss of pixel-level information. It might indicates that road extraction on satellite images, though low in resolution yet with more channels, needs enough depth and representation power in the model to utilise those spectral information to compensate insufficient spatial information and to achieve a better result.

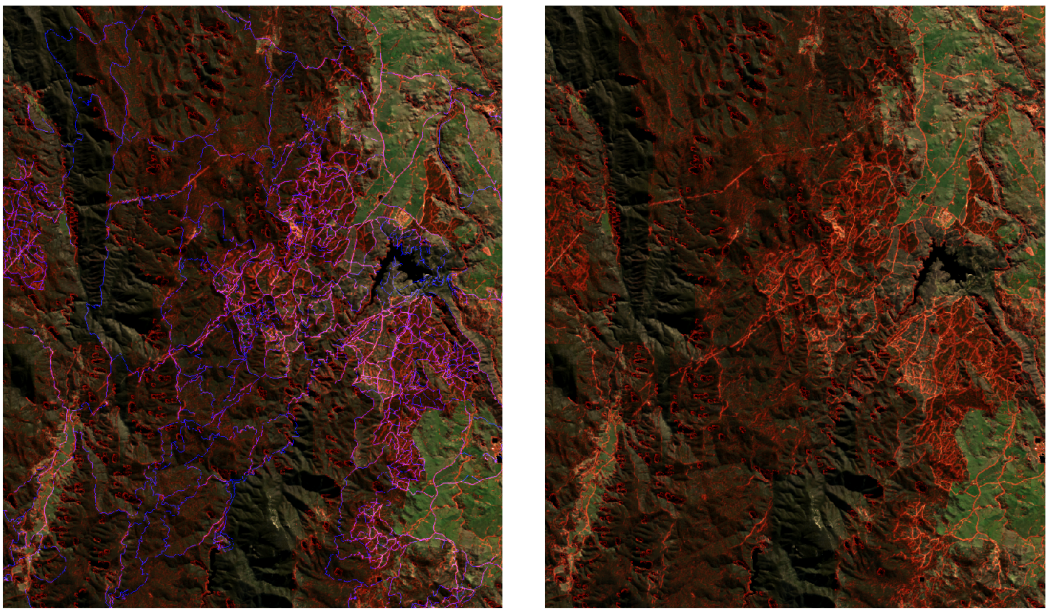
We are glad to see that model D outperforms B in all metrics, though by a small margin. Such result suggests that the per-pixel level information may not make a huge difference with a limited receptive field.

Figure 5.5 demonstrates that the Agile FCN is able to extract roads with much finer detail, compared to the sample prediction from the logistic regression model (Figure 5.3). In fact, prediction from Agile FCN covers most of the labelled roads. Meanwhile, the model are also more confident about its prediction; thus its enhancement for visualisation are lighter than that for the logistic regression.

Nonetheless, the model produce a noise output especially at the top-right part in Figure 5.5a. It might be because model has a limited knowledge about its surroundings so that it may not make full use of topological features of roads.



(a) CBR airport (bottom-left corner) and its neighbour



(b) Mountain area

Figure 5.5: Prediction maps are all generated by Agile FCN model D. The scenes are the same with that for Logistic Regression. For visualisation purpose, the probabilistic prediction is enhanced before being plotted. The enhancement is described by Equation 5.5. The enhanced probability of roads are added into the Red channel and the blue lines denote the ground truth. Prediction maps without ground truth are also provided at the right column for comparison.

$$P' = P^2 \tag{5.5}$$

where P is the probabilistic prediction and P' is the one being plotted.

5.2.4 Segmentation with U-net

5.2.4.1 Model Experimental Settings

As described in Section 4.2.4, model consists of two parts, input bridge and U connection that has down-sampling, up-sampling and a tunnel.

The input bridge contains various convolutions, yet they all use a stride of 1 and pad the image to retain the size of their feature maps, similar to the Agile FCN.

In U connection, the convolution layers in down-sampling path and tunnel are fixed to use 3-by-3 kernels with a stride of 1 and all pad the image to retain its size; each convolution is also followed by an optional batch normalisation layer and a rectified linear unit (ReLU). The pooling layer in down-sampling path is fixed to the common 2-by-2 max pooling with a stride of 2. In up-sampling path, deconvolution is used to perform up-sampling.

After all paths merge, a 1x1 convolution followed by softmax is used to produce the probabilistic prediction map.

Similar to Ailge FCN, loss is calculated as the average of pixel-wise cross entropy, our weighting scheme in Section 5.2.1 is adopted as described in Section 5.2.3 and input data is standardised.

The learning rate is set to $5e-6$, smaller than that of Agile FCN, because we observed over-fitting in the training phase. Hence, early stopping [Yao et al. 2007] is used and Figure 5.6 shows an example of over-fitting we observed.

In experiments, the U connection is fixed to the structure shown in Figure 4.3, except for the number of output channels of each convolutional and deconvolutional layer. Despite for the common U connection structures, variances of BridgeUnet are explored. Table 5.6 lists the settings for both.

5.2.4.2 Experiment Dataset

The experiment dataset we use for BridgeUnet is the same as that for Agile FCN, which is described in Section 5.2.3.

5.2.4.3 Evaluation Metrics

To be comparable to previous models, the same metrics are used and evaluated on training, validation and test sets. Those metrics include balanced accuracy (BA), average precision (AP) and area under the curve (AUC).

5.2.4.4 Results and Discussion

Table 5.7 summarises results of experiments on different settings and Figure 5.7 gives some examples of prediction of the model on actual scenes. Table 5.8 compares the original prediction maps from all three models (Logistic Regression, Agile FCN and BridgeUnet) on the same scene, with ground truth and input raw image provided.

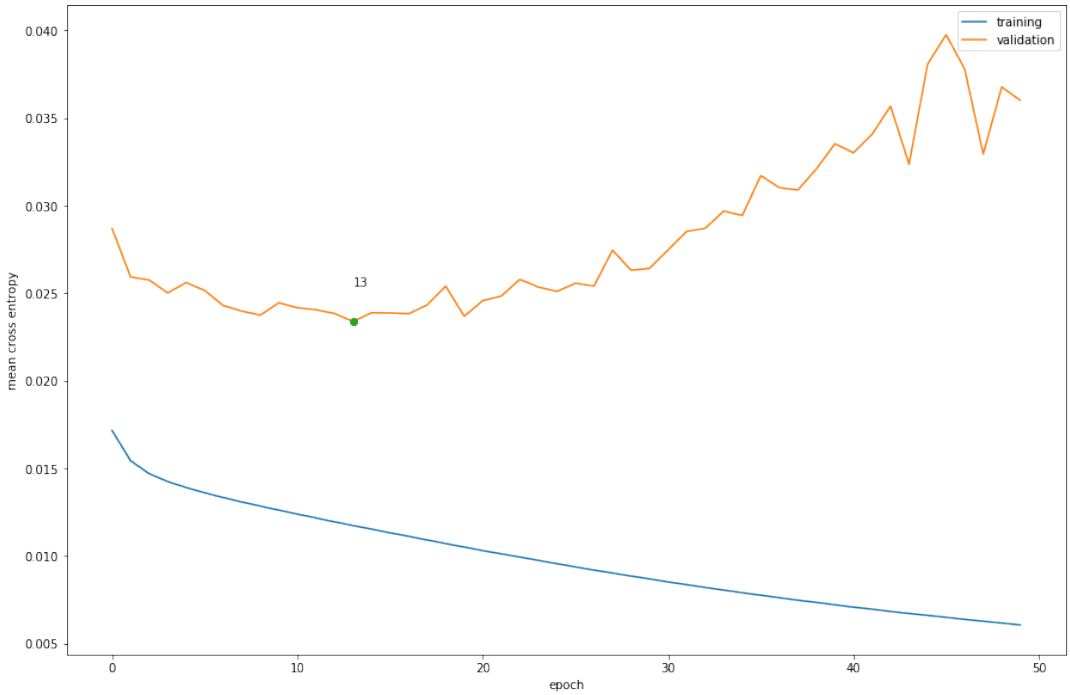


Figure 5.6: The green dot annotates the epoch number in which model's mean cross entropy on validation set achieves the lowest point and increases afterwards.

Table 5.6: BridgeUnet Structures in Experiments

(a) structure of U connection

Type	Channels after Layer					
	Down-sampling		Tunnel	Up-sampling		
	Conv1	Conv2	Conv3	Deconv1	Concat1=Deconv1+Conv1	Deconv2
A	32	64	128	64	128	32
B	64	128	256	128	256	64

(b) structure of Input Bridge

Type	Structure
1	$\begin{pmatrix}3x3:32\\1x1:32\end{pmatrix}$
2	$\begin{pmatrix}3x3:32\\1x1:32\end{pmatrix}\begin{pmatrix}3x3:32\\1x1:32\end{pmatrix}$
3	$(1x1:32)\,(1x1:32)\,(1x1:32)$

* The notation for Input Bridge is described when introducing Agile FCN model, in the Section 5.2.3.1.

Table 5.7: Numerical Metrics from BridgeUnet

Model	BA			AP			AUC		
	train	val	test	train	val	test	train	val	test
A1	0.787	0.764	0.780	0.133	0.176	0.096	0.872	0.850	0.870
A2	0.789	0.773	0.791	0.143	0.185	0.097	0.875	0.856	0.876
A3	0.791	0.764	0.779	0.144	0.176	0.100	0.876	0.851	0.874
B1	0.790	0.765	0.792	0.147	0.174	0.106	0.882	0.851	0.869
B2	0.845	0.786	**0.801**	0.197	0.203	0.113	0.923	0.870	**0.878**
B3	0.842	0.778	0.794	0.200	0.196	**0.117**	0.921	0.862	0.874

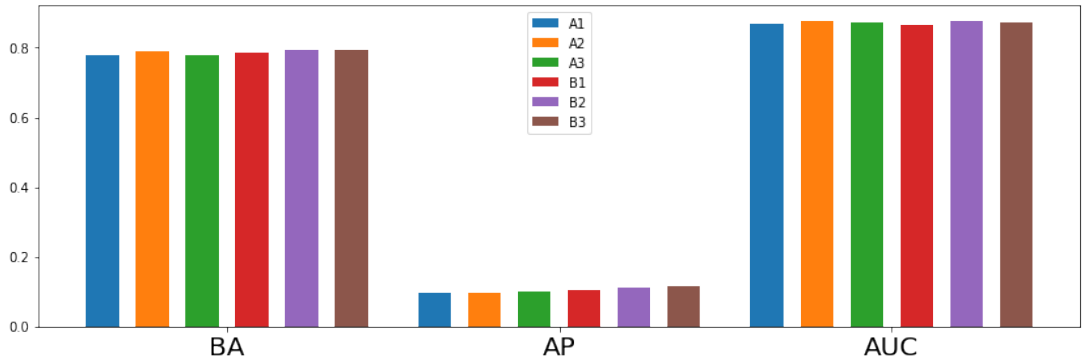
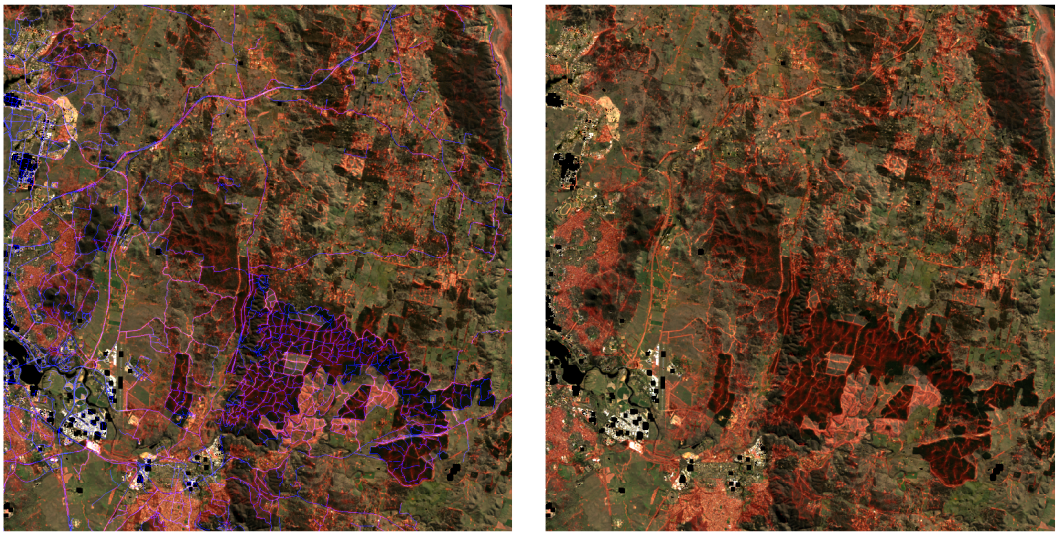


Figure 5.7: Bar plots of results on test set from models in Table 5.7

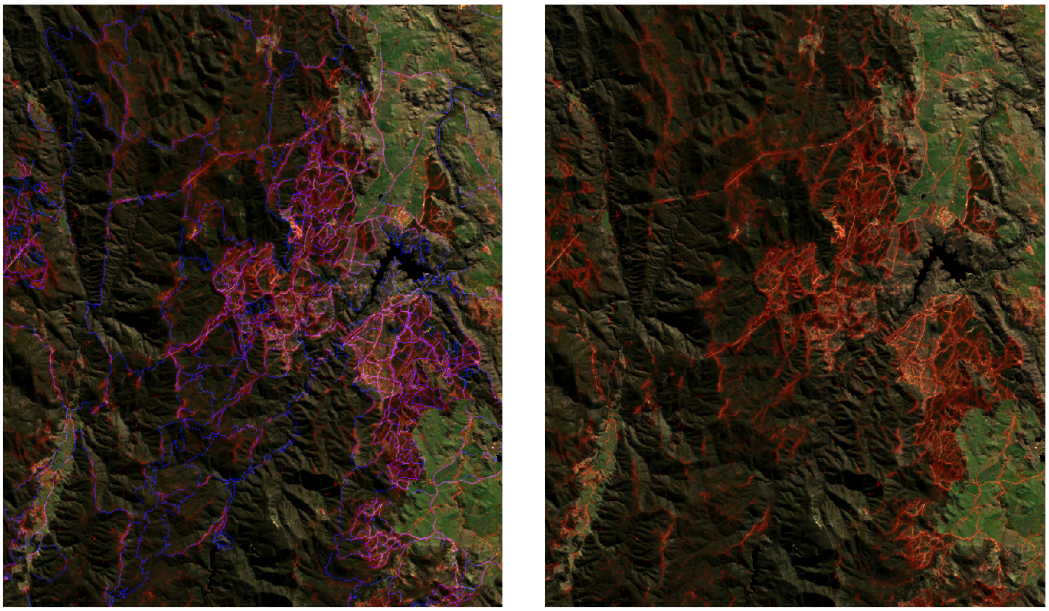
From the result (Table 5.7), models whose U connection has more feature maps (type Bx, $x \in (1, 2, 3)$) performs better than those corresponding model with less feature maps in each layer (type Ax $x \in (1, 2, 3)$) because type B has more representation power than type A. Meanwhile, regarding Input Bridge, type 2 and 3 are generally better than type 1 and only yield a bigger difference when using U connection of type B. Such observation are consistent with the suggestion in Agile FCN that per-pixel information may be valuable only when context information is sufficiently explored.

Notwithstanding, although BridgeUnet has a larger receptive field and more layers than Agile FCN, it is worse than Agile FCN (Table 5.5) in most numerical metrics evaluated on test set.

From the sample prediction map (Figure 5.8), prediction maps from BridgeUnet is less noisy while capturing most details. For example, the noisy road prediction in the top-left corner of Figure 5.8a is much less than that from Agile FCN (Figure 5.5a); there is no red-ish fog resulted from prediction with low confidence in the back of mountain area (bottom-right part) in Figure 5.8b, compared with that from Agile FCN (Figure 5.5b).



(a) CBR airport (bottom-left corner) and its neighbour



(b) Mountain area

Figure 5.8: Prediction maps are all generated by model B3. The scenes are the same with that for Logistic Regression and Agile FCN. Before visualised, the probabilistic prediction are enhanced in the same way as Agile FCN, described by Equation 5.5. Again, enhanced probability of roads are added into the Red channel and the blue lines denote the ground truth. Prediction maps without ground truth are also provided at right column for comparison.

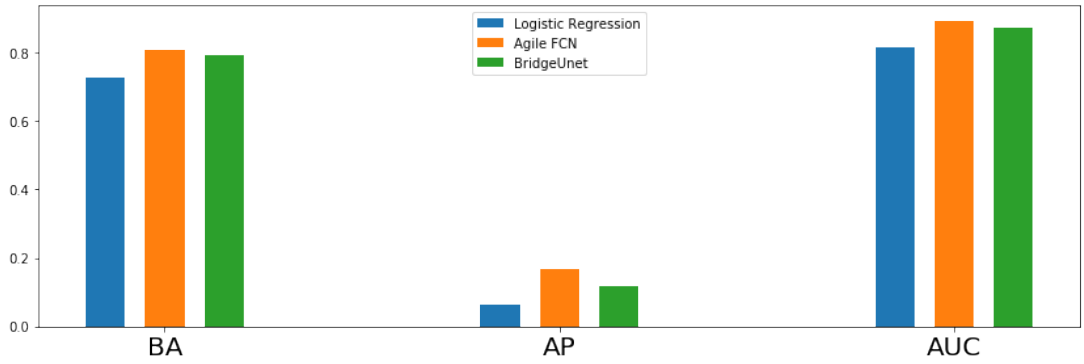
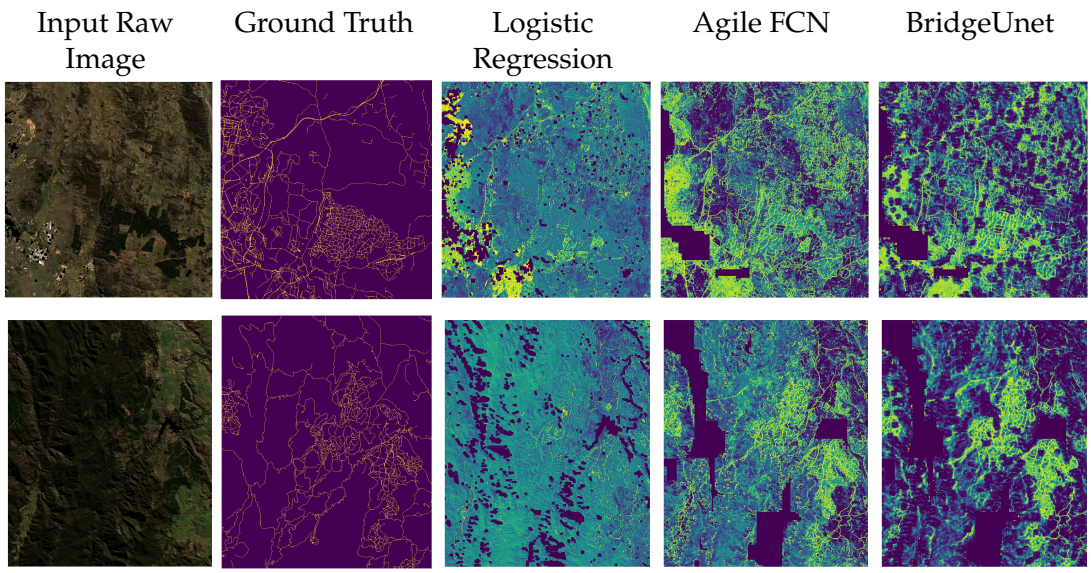


Figure 5.9: Bar plots shows metrics evaluated on test set from different models. The Logistic Regression model is the one with 'std' preprocessing and 'weight' balancing, from Table 5.3; the Agile FCN is the model D in Table 5.4; The BridgeUnet is the model B3 in Table 5.6.

Table 5.8: Comparison between Original Prediction Maps



The first row is the scene of Canberra airport (bottom-left corner) and its neighbours; and the second row is mountain area. Those scenes are the same as those previously presented in discussion section for each model. The model that generates each prediction name are placed above each column. All the probabilistic prediction maps are the original one before being enhanced, unlike the scene in their discussion sections; and brighter the colour is, larger the probability assigned into that pixel. Some completely black holes with no in prediction maps are because of invalid values and such holes are especially apparent in column for Logistic regression because it tends to predict all pixel with some probability of being roads.

5.3 Time-series Analysis on Road Networks Evolution

5.3.1 Goal

As models achieve competitive results on their dataset, we would like to demonstrate that our models are capable of unravelling the evolution of road networks from our time-series satellite images.

In this experiment, we show that our models are able to pick up roads from satellite images taken years ago, and can thus reveal roads change without seeing any ground truth of roads network in the past.

5.3.2 Models in Experiment

In this experiment, we use a trained BridgeUnet model from previous experiment (Section 5.2.4), specifically, the model B3 in Table 5.7. We also experiment with Logistic Regression model with 'std' preprocessing and weighted loss, described in Section 5.2.1. Additionally, the Agile FCN produces a similar result to BridgeUnet, thus we do not present its result. Results from mentioned two models are discussed and compared.

5.3.3 Experiment Dataset

We select two places around Canberra where roads construction are known, specifically, Coombs and Googong. These places are cropped from all available satellite images. It is worth noting that cropped image is used only if it contains a clear sight, typically, if valid values take up at least 80% of the whole sight. Finally, the scheme described in Section 3.2.3 is used to construct the dataset on both places along the time series for roads extraction.

5.3.4 Temporal De-noising

We notice that the atmosphere condition blocks the view of satellite from time to time. As a result, to obtain a stable and sufficient view and prediction over the selected area, we perform a post-processing referred as *temporal de-noising* or *de-noising over time*, explained by following Equation 5.6.

$$P'(i,j) = \frac{\sum_{n=1}^{N} P(i,j)}{\sum_{n=1}^{N} isvalid(n,i,j)} \tag{5.6}$$

where N is the total number of images within a given time span; P_n is the probabilistic prediction for the nth image in that time span; $P_n(i,j)$ is the probabilistic prediction for the pixel at position (i,j) in prediction map P_n; function $isvalid(n,i,j)$ is to check if the pixel (i,j) is contained by any valid patches in the nth satellite images, which can be expressed by the following Equation 5.7.

$$isvalid(n,i,j) = \begin{cases} 1, & \text{if pixel } (i,j) \text{ is in any valid patch in the } n\text{th images} \\ 0, & \text{otherwise} \end{cases} \tag{5.7}$$

In practice, the time span is chosen to be a year because the land cover changes significantly in different seasons. Hence, de-noising over a year is expected to relief such seasonal influence and also account for noise from occasional bad weather. However, the performance of such de-noising can be drawn back in practice. For example, some areas are especially cloudy in some typical seasons and may have more bad weathers in one year than the other. We also noticed that some areas can be blocked in the selected images throughout a particular year.

5.3.5 Results and Discussion

The model, either BridgeUnet or Logistic Regression, first predicts on each raw images selected into the dataset, and then both raw images and prediction maps are de-noised over time. Temporal de-noising for raw images uses the same process as explained in previous Section 5.3.4, but substitutes the probabilistic prediction map with each channel in the raw images.

To reveal the emerging and vanishing roads, temporarily de-noised prediction maps representing each years are subtracted from each other. Typically, we select 2013, 2015 and 2017 to show the evolution of road networks along the time.

As there is no road network record in the past as ground truth, only prediction result from the model and raw images are displayed.

5.3.5.1 BridgeUnet

Table 5.9 shows the scene and road network of Googong area in the time series of year 2013, 2015 and 2017. Table 5.10 shows the emerging and vanishing roads by showing the difference between model predictions in different years, i.e 2013 vs. 2015 and 2015 vs. 2017. Similarly, Table 5.11 shows the scene and road network in Coombs area, followed by emerging and vanishing roads there in Table 5.12.

Table 5.9: Road Changes in Googong (2013-2017)

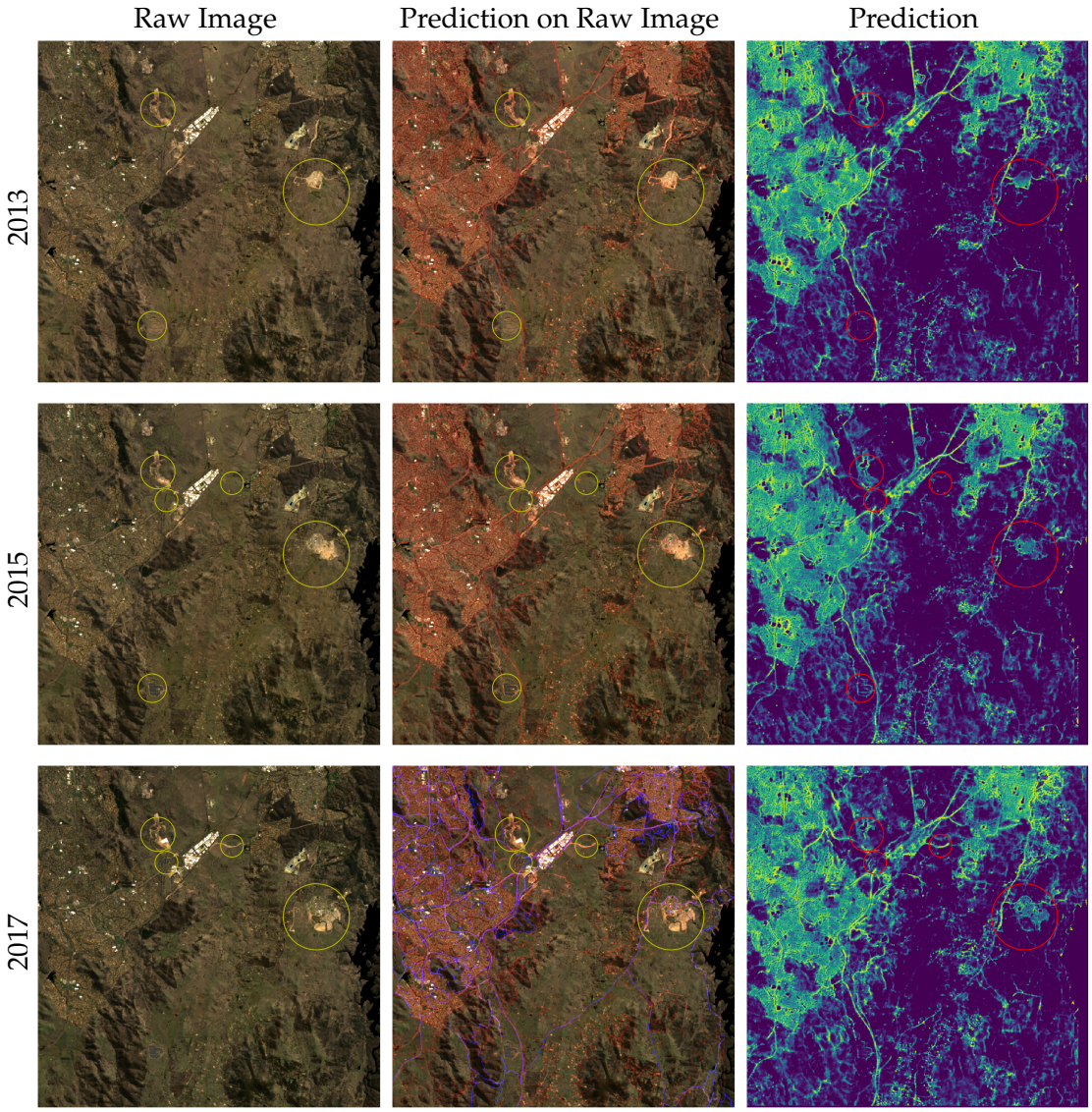
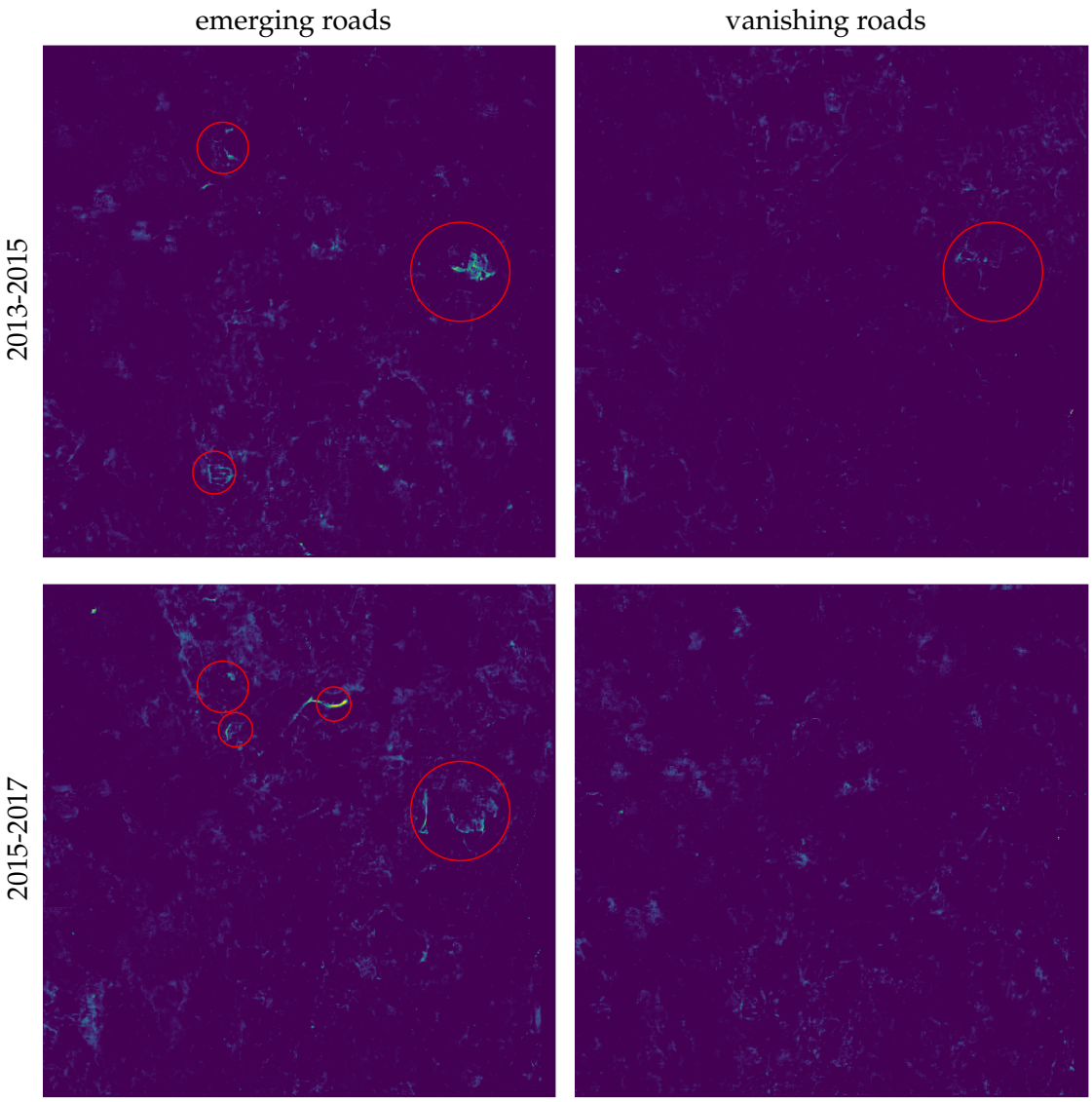


Table lists road changes in Googong from 2013, 2015 to 2017, predicted by BridgeUnet. Both raw images and prediction maps are shown; they are both temporarily de-noised as discussed in Section 5.3.5. The prediction maps are further enhanced by Equation 5.5 before being plotted. In column "Prediction on Raw Image", enhanced probabilistic predictions are added into the Red channel of corresponding raw images for better visualisation and our ground truth are added into the Blue channel of the most recent image (image for 2017). The places where road change are visible from raw image are circled. Please zoom in to see more detail.

Table 5.10: Road Changes in Googong (2013-2017)



The figures are produced by subtracting one prediction map in a year from the other, as discussed in Section 5.3.5. Hence, the brighter the place is, the more likely it is emerging / vanishing along the given time series. The emerging roads are labelled in red circle, corresponding to those circles in Table 5.9. There are few vanishing roads except for small roads in the red circle in 2013-2015, which vanishes because of the complete of construction at that part. Those parts with similar brightness in 2015-2017 are not considered vanishing because they do not stand out from their context and are indeed noise.

Table 5.11: Road Changes in Coombs (2013-2017)

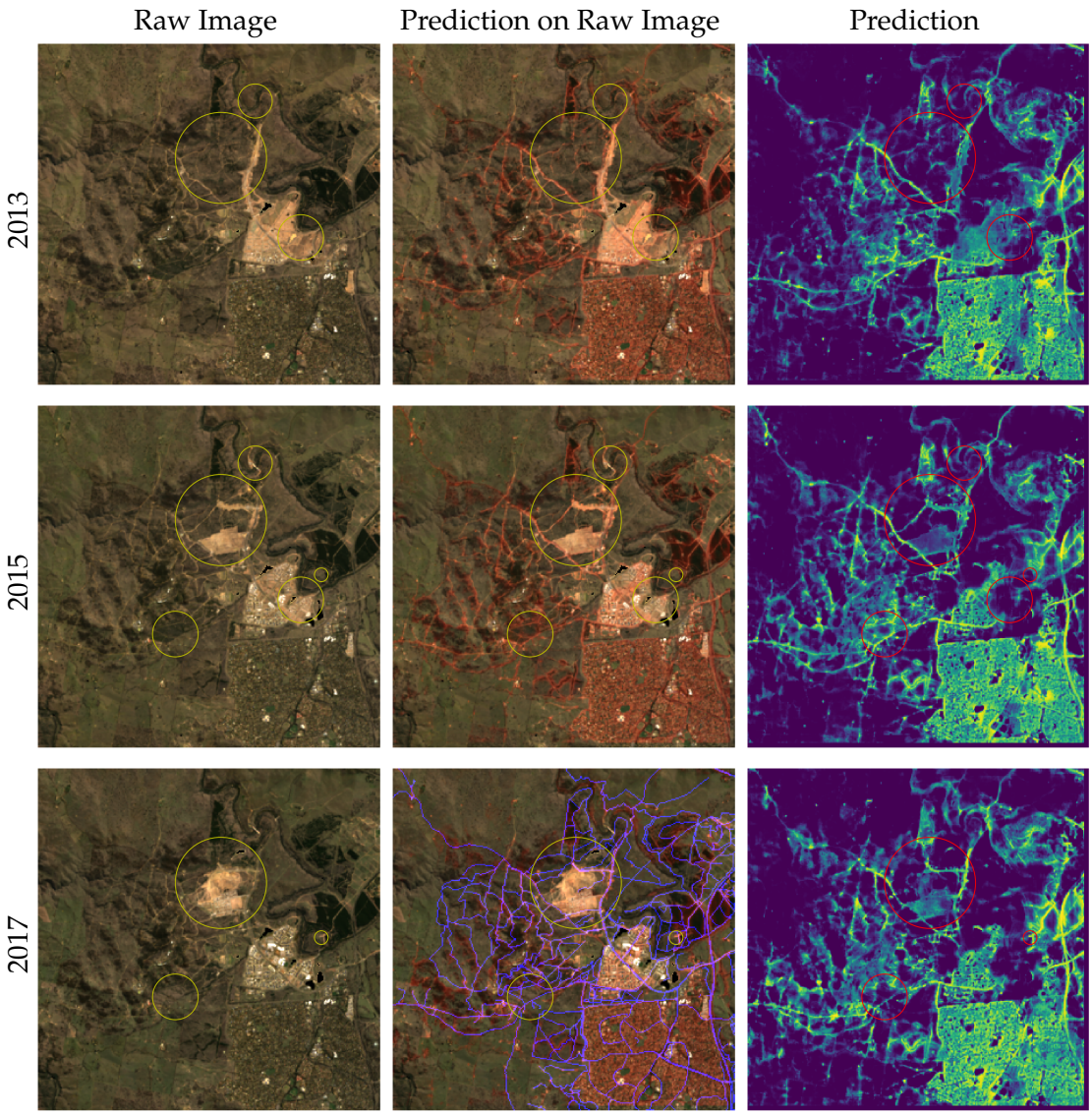
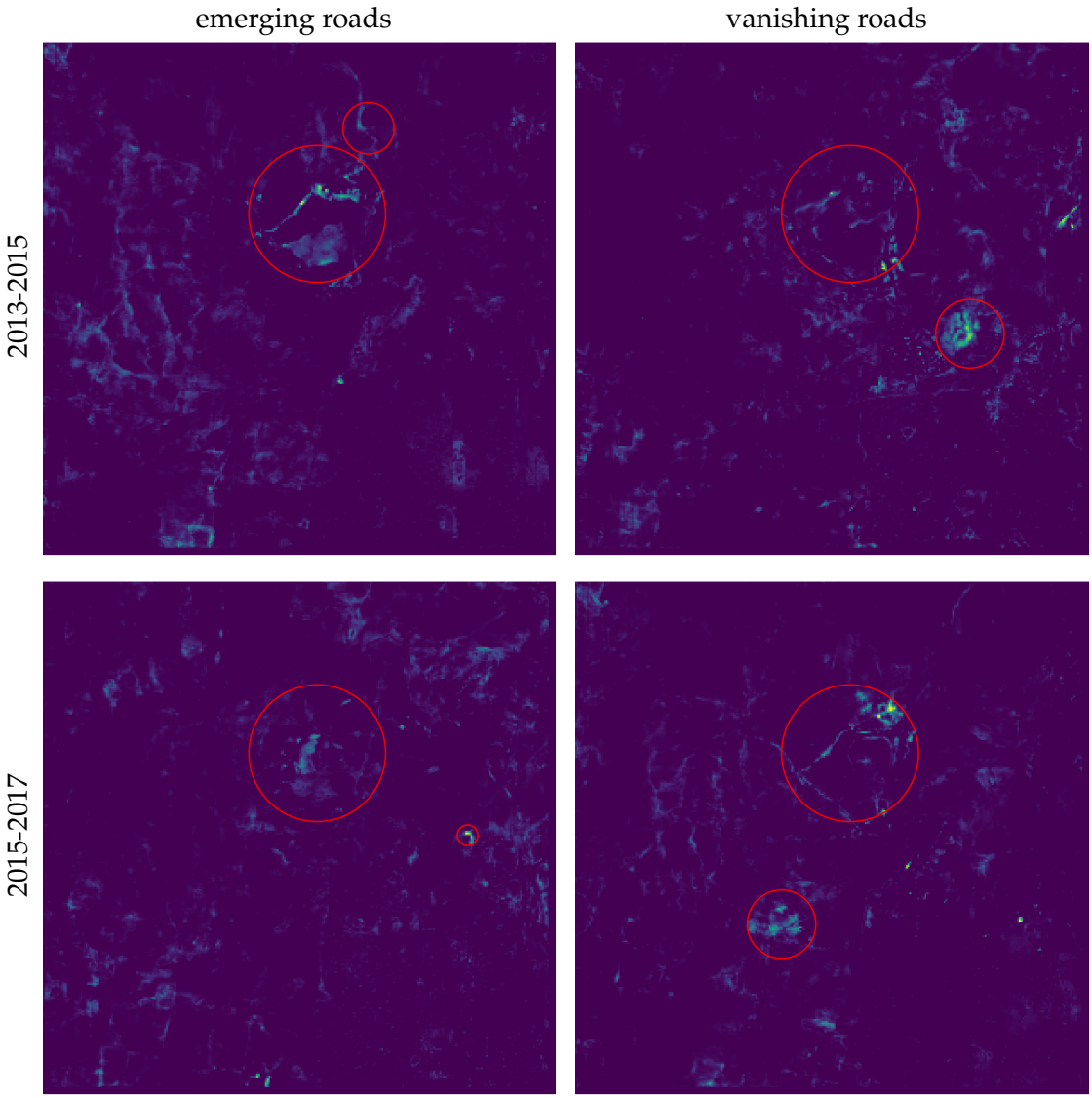


Table lists the road change in Coombs from 2013, 2015 to 2017, predicted by Bridge-Unet. The images are produced in the same fashion as for Googong, explained in Table 5.9.

Table 5.12: Road Changes in Coombs (2013-2017)



The figures are produced in the same fashion as Googong (Table 5.10). In vanishing roads of 2013-2015, The model is confused and predicts road in the small circle to vanish because the construction site in 2013 becomes building in 2015. Such land cover change between years, either from construction site into buildings or poor vegetation into rich vegetation, also confuses model to predict roads in 2015-2017 to vanish, including roads in circles and other hight-light dots (in vanishing roads of 2015-2017), because model tends to predict with less probability to those pixels after their changes. Another factor for some dramatic change is the pixels being invalid and valid throughout different years, which can be interpret as places being blocked and visible throughout different years.

It is yet interesting that model reasonably predict the edge of roads to vanish on the edge of the bigger circle in 2013-2015 because the road was under construction in 2013 and only the actual road remained into 2015.

5.3.5.2 The Stable Logistic Regression

We also experiments with the Logistic Regression, using the same process for BridgeUnet described in the beginning of Section 5.3.5.

Surprisingly, the Logistic Regression, though tends to assign every pixel in a patch with some probability of being road, are more stable across the time than the BridgeUnet.

We suspect that the stableness of Logistic Regression is because such a simple model lacks the ability to capture the tiny changes in the patch. That is to say, unlike BridgeUnet, the changes such as decreasing of vegetation may not be recognised by logistic regression and the model still predicts with a similar probability. Or, we could suspect that the Logistic Regression learns to recognise basic land cover instead of extracting roads.

Table 5.13 shows the scene and road network of Googong area in the time series of year 2013, 2015 and 2017. Table 5.14 shows the emerging roads in different years, i.e 2013 vs. 2015 and 2015 vs. 2017 and it compares the result from BridgeUnet and Logistic Regression. Similarly, Table 5.15 shows the scene and road network in Coombs area, followed by the comparison of emerging roads between BridgeUnet and Logistic Regression in Table 5.16.

Table 5.13: Road Changes in Googong (2013-2017)

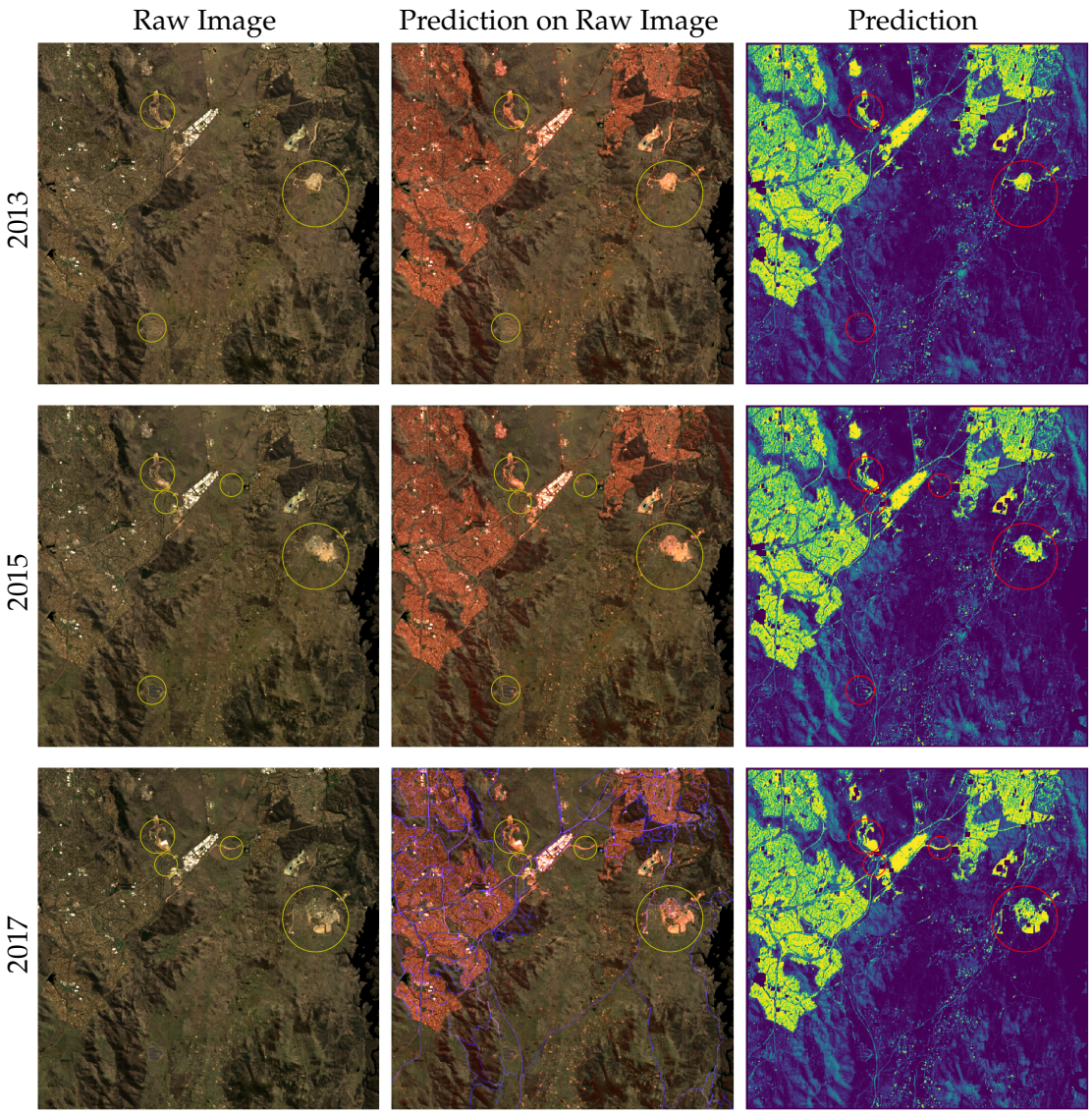
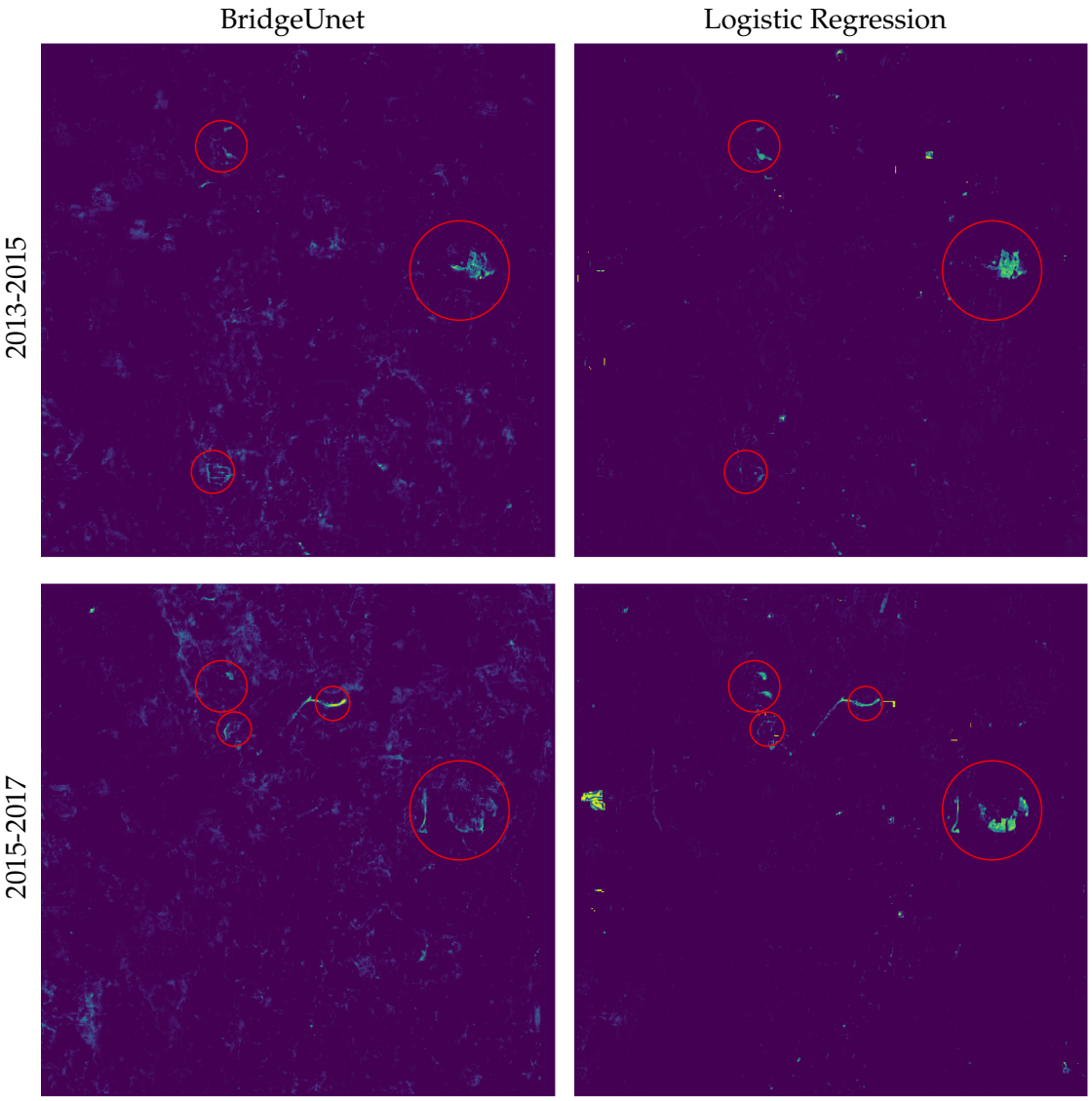


Table lists the road change in Googong from 2013, 2015 to 2017, predicted by Logistic Regression. The images are produced in the same fashion as previous, explained under Table 5.9.

Table 5.14: Emerging Roads in Googong (2013-2017)
BridgeUnet vs. Logistic Regression



The table shows the predicted emerging roads from 2013, 2015 to 2017 in Googong and compare the result from BridgeUnet (left column) and Logistic Regression (right column). It is worth noting that the brightness are comparable across above figures and brighter the pixel is, bigger the probability assigned to it. There is an obvious chunk predicted to emerge in 2015-2017 by Logistic Regression on the left edge of that image, but is disagreed by BridgeUnet. Such inconsistency is because that place is blocked by clouds throughout the 2015 in our dataset but not so in 2017, confusing the Logistic Regression because Logistic Regression does not handle any invalid values; whereas BridgeUnet is able to tolerate such changes because of its relatively large receptive fields and the consequent robustness learned in its training.

Table 5.15: Road Changes in Coombs (2013-2017)

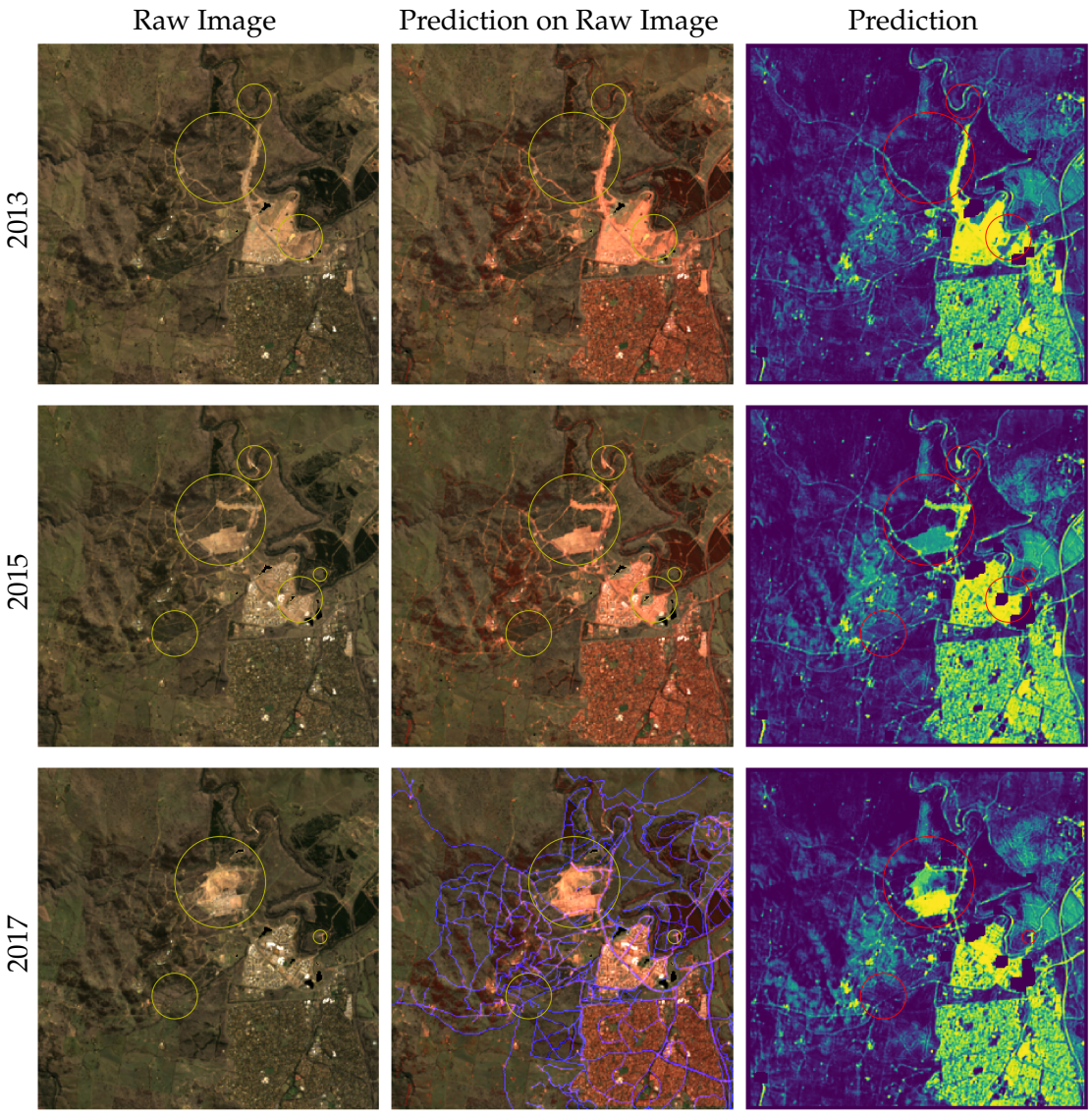
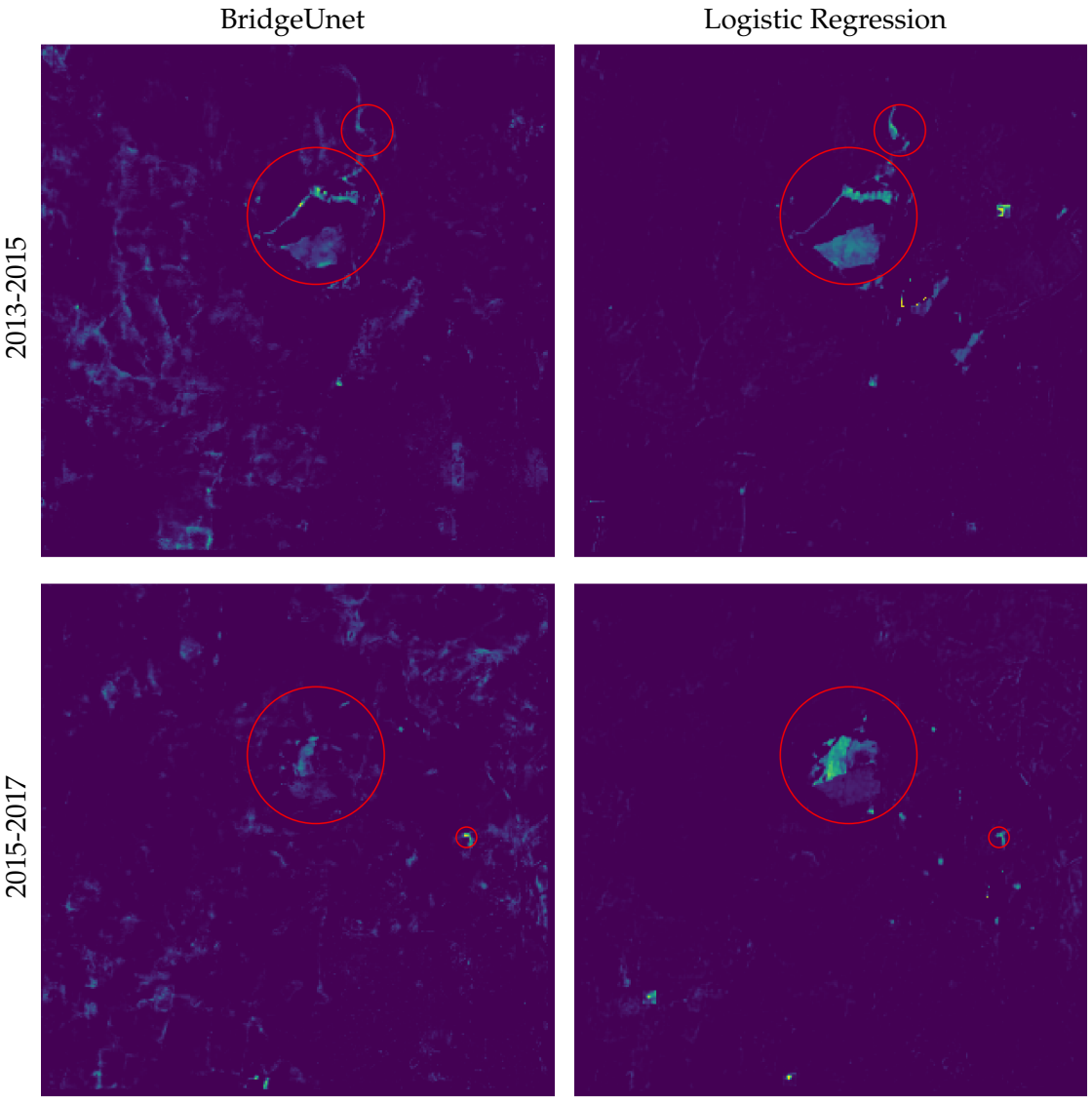


Table lists the road change in Coombs from 2013, 2015 to 2017, predicted by logistic regression. The images are produced in the same fashion as previous, explained in Table 5.9.

Table 5.16: Emerging Road in Coombs (2013-2017)
BridgeUnet vs. Logistic Regression



Similar to Table 5.14, this table compares the result from BridgeUnet and Logistic Regression on emerging roads from 2013, 2015 to 2017 in Coombs; and brightness are comparable across figures and denotes probability. Again, Logistic Regression gives much less noise than BridgeUnet in the prediction. It is however observed that, BridgeUnet tends to assign more probability to the changes of roads instead of the changes of land cover, compared to Logistic Regression, which is suspected to steadily uncover the change in land cover.

Chapter 6

Conclusion

In this thesis, we attempt to extract roads from time-series collection of satellite images from LANDSAT 8 project. We approach the problem from multiple perspectives, including patch classification, sliding window and image segmentation.

We firstly apply patch classification to produce a heat map for the probability of roads appearing, demonstrating that roads are recognisable on our dataset. We then select those challenging types of roads in patch classification to develop our model for road extraction as a semantic segmentation task. In approaching this task, we first adapt Logistic Regression with sliding window and its result of road extraction achieves a balanced accuracy of 0.73 on test set; we secondly propose Agile FCN model based on FCN and inception model, reaching a balanced accuracy of 0.81 on test set; finally we explore and propose BridgeUnet model based on U-net architecture, gaining a similar balanced accuracy of 0.80 on test set. We have thus demonstrated that road extraction from time-series satellite images, which are often of medium-resolution with extra spectral bands, are achievable.

Such achievement then enables us to analyse the evolution of road networks in area where road records are not available. We illustrate such evolution by revealing the changes in road networks in two local areas, Goongong and Coombs; we also compare the predictions from different models, Logistic Regression and our BridgeUnet, on these scenes.

In the future, we believe it is possible to improve the performance of the models, especially the BridgeUnet. Typically, it is important to advance the model to be invariant to different input across time and space, in order to produce a stable output across a long time span. We suggest such objectives may be approached by developing more specialised model or exploring more task-specific loss function, instead of the common cross entropy in this thesis.

Given the current result on road extraction, it is also possible to profile other geographical features from the time-series satellite images and it is also exciting for people to conduct more in-depth time-series analysis than that in this thesis. For any future research inspired by this, we also provide the tensorflow -based implementation of our work [1].

[1] Implementation of our work available at https://github.com/beihaifusang/Sat-Img-in-Time

Bibliography

2004. Road extraction using svm and image segmentation. *Photogrammetric Engineering and Remote Sensing 70*, 12. (p. 1)

2016. Multiple object extraction from aerial imagery with convolutional neural networks. *Journal of Imaging Science and Technology 60*, 1. (p. 1)

ABADI, M., AGARWAL, A., BARHAM, P., BREVDO, E., CHEN, Z., CITRO, C., CORRADO, G. S., DAVIS, A., DEAN, J., DEVIN, M., GHEMAWAT, S., GOODFELLOW, I., HARP, A., IRVING, G., ISARD, M., JIA, Y., JOZEFOWICZ, R., KAISER, L., KUDLUR, M., LEVENBERG, J., MANÉ, D., MONGA, R., MOORE, S., MURRAY, D., OLAH, C., SCHUSTER, M., SHLENS, J., STEINER, B., SUTSKEVER, I., TALWAR, K., TUCKER, P., VANHOUCKE, V., VASUDEVAN, V., VIÉGAS, F., VINYALS, O., WARDEN, P., WATTENBERG, M., WICKE, M., YU, Y., AND ZHENG, X. 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org. (p. 31)

ABE, N., ZADROZNY, B., AND LANGFORD, J. 2004. An iterative method for multi-class cost-sensitive learning. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04 (New York, NY, USA, 2004), pp. 3–11. ACM. (p. 36)

ABELSON, B., R. VARSHNEY, K., AND SUN, J. 2014. Targeting direct cash transfers to the extremely poor. (p. 5)

BADRINARAYANAN, V., KENDALL, A., AND CIPOLLA, R. 2015. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *CoRR abs/1511.00561*. (pp. 8, 28)

BAI, J., XIANG, S., AND PAN, C. 2013. A graph-based classification method for hyperspectral images. *IEEE Transactions on Geoscience and Remote Sensing 51*, 2 (Feb), 803–817. (p. 7)

BLASCHKE, T. 2010. Object based image analysis for remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing 65*, 1, 2 – 16. (pp. 18, 24)

BOCKHORST, J. AND CRAVEN, M. 2005. Markov networks for detecting overlapping elements in sequence data (2005). (p. 33)

BOWYER, K. W., CHAWLA, N. V., HALL, L. O., AND KEGELMEYER, W. P. 2011. SMOTE: synthetic minority over-sampling technique. *CoRR abs/1106.1813*. (p. 36)

BRADLEY, A. P. 1997. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition 30*, 7, 1145 – 1159. (p. 33)

BRODERSEN, K. H., ONG, C. S., STEPHAN, K. E., AND BUHMANN, J. M. 2010. The balanced accuracy and its posterior distribution. In *2010 20th International Conference on Pattern Recognition* (Aug 2010), pp. 3121–3124. (p. 37)

CAMPBELL, J. B. AND WYNNE, R. H. 2011. *Introduction to Remote Sensing, Fifth Edition*. Guilford Publications. (p. 3)

CHANUSSOT, J. AND LAMBERT, P. 1998. An application of mathematical morphology to road network extraction on sar images. In *ISMM '98 Proceedings of the fourth international symposium on Mathematical morphology and its applications to image and signal processing* (1998), pp. 399–406. (p. 7)

DAVIS, J. AND GOADRICH, M. 2006. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06 (New York, NY, USA, 2006), pp. 233–240. ACM. (p. 33)

E. RUMELHART, D., E. HINTON, G., AND J. WILLIAMS, R. 1986. Learning representations by back propagating errors. *323*, 533–536. (p. 8)

FAWCETT, T. 2006. An introduction to roc analysis. *Pattern Recognition Letters 27*, 8, 861 – 874. ROC Analysis in Pattern Recognition. (pp. 32, 33)

FRETWELL PT, F. J., STANILAND IJ. 2014. Whales from space: Counting southern right whales by satellite. *PLOS one*. (p. 5)

G. VOSSELMAN, J. K. 1995. *Road tracing by profile matching and Kalman filtering*. A. Gruen, E. Baltsavias, O. Henricsson (Eds.). (p. 6)

GARCIA-GARCIA, A., ORTS-ESCOLANO, S., OPREA, S., VILLENA-MARTINEZ, V., AND RODRÍGUEZ, J. G. 2017. A review on deep learning techniques applied to semantic segmentation. *CoRR abs/1704.06857*. (p. 25)

GDAL DEVELOPMENT TEAM. 201x. *GDAL - Geospatial Data Abstraction Library, Version x.x.x*. Open Source Geospatial Foundation. (p. 20)

GECEN, R. AND SARP, G. 2018. Road detection from high and low resolution satellite images. (p. 18)

GUPTA, R. P. 2018. *Remote Sensing Geology*. Springer-Verlag Berlin Heidelberg. (p. 3)

HANLEY, J. AND MCNEIL, B. 1982. The meaning and use of the area under a receiver operating characteristic (roc) curve. *143*, 29–36. (p. 33)

HE, K., ZHANG, X., REN, S., AND SUN, J. 2015. Deep residual learning for image recognition. *CoRR abs/1512.03385*. (p. 10)

HEERMANN, P. D. AND KHAZENIE, N. 1992. Classification of multispectral remote sensing data using a back-propagation neural network. *IEEE Transactions on Geoscience and Remote Sensing 30*, 1 (Jan), 81–88. (p. 9)

HERUMURTI, D., UCHIMURA, K., KOUTAKI, G., AND UEMURA, T. 2013. Urban road extraction based on hough transform and region growing. pp. 220–224.

HOTID. 2017. Geo data collect. (p. 17)

HUAZHONG, R., DU, C., LIU, R., QIN, Q., YAN, G., LI, Z.-L., AND MENG, J. 2014. Noise evaluation of early images for landsat 8 operational land imager. *22*. (p. 6)

HUNTER, J. D. 2007. Matplotlib: A 2d graphics environment. *Computing In Science & Engineering 9*, 3, 90–95. (p. 31)

IOFFE, S. AND SZEGEDY, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR abs/1502.03167*. (p. 41)

IRONS, J. R., DWYER, J. L., AND BARSI, J. A. 2012. The next landsat satellite: The landsat data continuity mission. *Remote Sensing of Environment 122*, 11 – 21. Landsat Legacy Special Issue. (p. 12)

J. VERNON HENDERSON, A. S. AND WEIL, D. N. 2012. Measuring economic growth from outer space. *American Economic Review*. (p. 5)

JENSEN, J. R. AND LULLA, D. K. 1987. Introductory digital image processing: A remote sensing perspective. *Geocarto International 2*, 1, 65–65. (p. 41)

JONES, E., OLIPHANT, T., PETERSON, P., ET AL. 2001–. SciPy: Open source scientific tools for Python. [Online; accessed ¡today¿]. (p. 31)

KAHRAMAN, I., TURAN, M., AND KARAŞ, I. 2015. Road detection from high satellite images using neural networks. *5*, 304–307. (p. 9)

KATARTZIS, A., SAHLI, H., PIZURICA, V., AND CORNELIS, J. 2001. A model-based approach to the automatic extraction of linear features from airborne images. *IEEE Transactions on Geoscience and Remote Sensing 39*, 9 (Sep), 2073–2079. (p. 7)

KINGMA, P., D., AND BA, J. 2014. Adam: A method for stochastic optimization. (pp. 31, 37, 41)

KIRTHIKA, A. AND MOOKAMBIGA, A. 2011. Automated road network extraction using artificial neural network. In *2011 International Conference on Recent Trends in Information Technology (ICRTIT)* (June 2011), pp. 1061–1065. (pp. 9, 25)

KOUNADI, O. 2009. Assessing the quality of openstreetmap data.

KRIZHEVSKY, A., SUTSKEVER, I., AND HINTON, G. E. 2012. Imagenet classification with deep convolutional neural networks. In F. PEREIRA, C. J. C. BURGES, L. BOTTOU, AND K. Q. WEINBERGER Eds., *Advances in Neural Information Processing Systems 25*, pp. 1097–1105. Curran Associates, Inc. (pp. 1, 8, 41)

LANDGREBE, D. A. 2005. *Signal Theory Methods in Multispectral Remote Sensing*. John Wiley, Sons. (p. 3)

LECUN, Y., HAFFNER, P., BOTTOU, L., AND BENGIO, Y. 1999. *Object Recognition with Gradient-Based Learning*, pp. 319–345. Springer Berlin Heidelberg, Berlin, Heidelberg. (p. 8)

LEE, H., GROSSE, R., RANGANATH, R., AND NG, A. Y. 2009. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09 (New York, NY, USA, 2009), pp. 609–616. ACM. (p. 8)

LIU, L., SILVA, E. A., WU, C., AND WANG, H. 2017. A machine learning-based method for the large-scale evaluation of the qualities of the urban environment. *Computers, Environment and Urban Systems 65*, 113 – 125. (p. 9)

LIU, Z., LI, X., LUO, P., LOY, C. C., AND TANG, X. 2015. Semantic image segmentation via deep parsing network. *CoRR abs/1509.02634*. (p. 8)

LONG, J., SHELHAMER, E., AND DARRELL, T. 2014. Fully convolutional networks for semantic segmentation. *CoRR abs/1411.4038*. (pp. 8, 26)

MAGGIORI, E., TARABALKA, Y., CHARPIAT, G., AND ALLIEZ, P. 2016. Fully convolutional neural networks for remote sensing image classification. In *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)* (July 2016), pp. 5071–5074. (p. 10)

MNIH, V. AND HINTON, G. E. 2010. Learning to detect roads in high-resolution aerial images. In K. DANIILIDIS, P. MARAGOS, AND N. PARAGIOS Eds., *Computer Vision – ECCV 2010* (Berlin, Heidelberg, 2010), pp. 210–223. Springer Berlin Heidelberg. (p. 9)

MUMFORD, L. 1961. *The City in History*. Harcourt, Brace and World. (p. 1)

MYINT, S. W., GOBER, P., BRAZEL, A., GROSSMAN-CLARKE, S., AND WENG, Q. 2011. Per-pixel vs. object-based classification of urban land cover extraction using high spatial resolution imagery. *Remote Sensing of Environment 115*, 5, 1145 – 1161. (p. 18)

NASA. 2012. What are passive and active sensors.

NASA. 2018a. Landsat 8. (p. 12)

NASA. 2018b. Landsat 8. (p. 12)

NASA. 2018c. Landsat 8 data processing and archive system (dpas). (p. 20)

NASA. 2018d. Landsat 8 osi requirement. (p. 12)

NASA. 2018e. Remote sensors. (p. 3)

OPENSTREETMAP CONTRIBUTORS. 2017. Planet dump retrieved from https://planet.osm.org . https://www.openstreetmap.org. (pp. 11, 17)

OSM. 2017a. Beginners guide 1.1 - contribute map data. (p. 17)

OSM. 2017b. Elements. (p. 17)

OSM. 2017c. Key:highway. (p. 21)

OSM. 2018. History api and database. (p. 17)

PAUL M. MATHER, M. K. 2011. *Computer Processing of Remotely-Sensed Images: An Introduction*. Wiley. (p. 37)

PEDREGOSA, F., VAROQUAUX, G., GRAMFORT, A., MICHEL, V., THIRION, B., GRISEL, O., BLONDEL, M., PRETTENHOFER, P., WEISS, R., DUBOURG, V., VANDERPLAS, J., PASSOS, A., COURNAPEAU, D., BRUCHER, M., PERROT, M., AND DUCHESNAY,

E. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research 12*, 2825–2830. (p. 31)

QAZI, N. AND RAZA, K. 2012. Effect of feature selection, smote and under sampling on class imbalance classification. In *2012 UKSim 14th International Conference on Computer Modelling and Simulation* (March 2012), pp. 145–150. (p. 36)

RAJESWARI, M., GURUMURTHY, K. S., OMKAR, S. N., SENTHILNATH, J., AND REDDY, L. P. 2011. Automatic road extraction using high resolution satellite images based on level set and mean shift methods. In *2011 3rd International Conference on Electronics Computer Technology*, Volume 2 (April 2011), pp. 424–428. (p. 1)

REHRL, K. AND GRÖCHENIG, S. 2016. A framework for data-centric analysis of mapping activity in the context of volunteered geographic information. *5*, 37. (p. 17)

RONNEBERGER, O., FISCHER, P., AND BROX, T. 2015. U-net: Convolutional networks for biomedical image segmentation. *CoRR abs/1505.04597*. (pp. 9, 28)

SAITO, S., YAMASHITA, Y., AND AOKI, Y. 2016. Multiple object extraction from aerial imagery with convolutional neural networks. *60*, 10402–1/10402. (pp. 9, 10)

SCHOWENGERDT, R. A. 2007. *Remote Sensing: Models and Methods for Image Processing*. Elsevier Science. (p. 3)

SCHROEDER, T. A., COHEN, W. B., SONG, C., CANTY, M. J., AND YANG, Z. 2006. Radiometric correction of multi-temporal landsat data for characterization of early successional forest patterns in western oregon. *Remote Sensing of Environment 103*, 1, 16 – 26. (p. 15)

SCHROFF, F., CRIMINISI, A., AND ZISSERMAN, A. 2008. Object class segmentation using random forests. In *Proc. British Machine Vision Conference (BMVC)* (January 2008).

SEHRA, S. S., SINGH, J., AND RAI, H. S. 2013. Assessment of openstreetmap data - A review. *CoRR abs/1309.6608*.

SEHRA, S. S., SINGH, J., AND RAI, H. S. 2014. A systematic study of openstreetmap data quality assessment. In *2014 11th International Conference on Information Technology: New Generations* (April 2014), pp. 377–381. (p. 17)

SHAPIRO, L. G. AND STOCKMAN, G. C. 2001. *Computer Vision*, pp. 279–325. Prentice Hall, New Jersey. (p. 9)

SHI, W., MIAO, Z., AND DEBAYLE, J. 2014. An integrated method for urban mainroad centerline extraction from optical remotely sensed imagery. *IEEE Transactions on Geoscience and Remote Sensing 52*, 6 (June), 3359–3372. (p. 5)

SIMONYAN, K. AND ZISSERMAN, A. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. *ArXiv e-prints*. (p. 10)

SOUTHWORTH, M. 2013. *Streets and the Shaping of Towns and Cities*. IslandPress. (p. 1)

STEGER, C., MAYER, H., AND RADIG, B. 1997. The role of grouping for road extraction. In A. GRUEN, E. P. BALTSAVIAS, AND O. HENRICSSON Eds., *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)* (Basel, 1997), pp. 245–256. Birkhäuser Basel. (p. 7)

STOCKMAN, G. AND SHAPIRO, L. G. 2001. *Computer Vision* (1st ed.). Prentice Hall PTR, Upper Saddle River, NJ, USA. (p. 6)

STRANO, N. V. L. V. P. S. B. M., EMANUELE. 2012. Elementary processes governing the evolution of road networks. *2*. (p. 1)

SZEGEDY, C., LIU, W., JIA, Y., SERMANET, P., REED, S. E., ANGUELOV, D., ERHAN, D., VANHOUCKE, V., AND RABINOVICH, A. 2014. Going deeper with convolutions. *CoRR abs/1409.4842*. (p. 26)

TUPIN, F., MAITRE, H., MANGIN, J. F., NICOLAS, J. M., AND PECHERSKY, E. 1998. Detection of linear features in sar images: application to road network extraction. *IEEE Transactions on Geoscience and Remote Sensing 36*, 2 (Mar), 434–453. (p. 7)

UNSALAN, C. AND SIRMACEK, B. 2012. Road network detection using probabilistic and graph theoretical methods. *IEEE Transactions on Geoscience and Remote Sensing 50*, 11 (Nov), 4441–4453. (p. 7)

USGS. 2013. The landsat 8 (ldcm) press kit. (pp. 11, 12)

USGS. 2016. *Landsat 8 Data Users Handbook*. USGS. (pp. 11, 12, 13)

USGS. 2018a. Atmospheric transmittance information. (p. 3)

USGS. 2018b. Landsat collections. (p. 18)

USGS. 2018c. *U.S. LANDSAT ANALYSIS READY DATA (ARD) DATA FORMAT CONTROL BOOK (DFCB)*. USGS. (p. 12)

VERMOTE, E., JUSTICE, C., CLAVERIE, M., AND FRANCH, B. 2016. Preliminary analysis of the performance of the landsat 8/oli land surface reflectance product. *Remote Sensing of Environment 185*, 46 – 56. Landsat 8 Science Results.

WANG, J., QIN, Q., YANG, X., WANG, J., YE, X., AND QIN, X. 2014. Automated road extraction from multi-resolution images using spectral information and texture. *2014 IEEE Geoscience and Remote Sensing Symposium*, 533–536.

WEI, Y., WANG, Z., AND XU, M. 2017. Road structure refined cnn for road extraction in aerial image. *IEEE Geoscience and Remote Sensing Letters 14*, 5 (May), 709–713. (p. 10)

WEIXING WANG, Y. Z. F. W. T. C. P. E., NAN YANG. 2016. A review of road extraction from remote sensing images. *Journal of Traffic and Transportation Engineering (English Edition)*. (pp. 5, 6)

XU, R., QIN, Q., LIN, C., ZHANG, Y., AND LI, J. 2014. Investigating the impact of road network development on land cover change in lijiang river basin. In *2014 IEEE Geoscience and Remote Sensing Symposium* (July 2014), pp. 4243–4246. (p. 1)

YAO, Y., ROSASCO, L., AND CAPONNETTO, A. 2007. On early stopping in gradient descent learning. *Constructive Approximation 26*, 2 (Aug), 289–315. (p. 45)

YIN, Z., BISE, R., CHEN, M., AND KANADE, T. 2010. Cell segmentation in microscopy imagery using a bag of local bayesian classifiers. In *2010 IEEE International Symposium on Biomedical Imaging: From Nano to Macro* (April 2010), pp. 125–128.

YOUNG, N. E., ANDERSON, R. S., CHIGNELL, S. M., VORSTER, A. G., LAWRENCE, R., AND EVANGELISTA, P. H. 2017. A survival guide to landsat preprocessing. *Ecology 98*, 4 (3), 920–932. (pp. 14, 15, 16)

YUN, Z., NAN, M., DA, R., AND BING, A. 2011. An effective over-sampling method for imbalanced data sets classification. *20*. (p. 36)

ZEILER, M. D. AND FERGUS, R. 2013. Visualizing and understanding convolutional networks. *CoRR abs/1311.2901*. (p. 8)

ZHANG, Z., LIU, Q., AND WANG, Y. 2017. Road extraction by deep residual u-net. *CoRR abs/1711.10684*. (pp. 1, 10)

ZHENG, S., JAYASUMANA, S., ROMERA-PAREDES, B., VINEET, V., SU, Z., DU, D., HUANG, C., AND TORR, P. H. S. 2015. Conditional random fields as recurrent neural networks. *CoRR abs/1502.03240*. (p. 8)

ZHONG, Y., FEI, F., LIU, Y., ZHAO, B., HONGZAN, J., AND ZHANG, L. 2017. Satcnn: satellite image dataset classification using agile convolutional neural networks. *8*, 136–145. (pp. 9, 31)

ZHONG, Z., LI, J., CUI, W., AND JIANG, H. 2016. Fully convolutional networks for building and road extraction: Preliminary results. In *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)* (July 2016), pp. 1591–1594. (p. 10)

ZHOU, H., ZHANG, J., LEI, J., LI, S., AND TU, D. 2016. Image semantic segmentation based on fcn-crf model. In *2016 International Conference on Image, Vision and Computing (ICIVC)* (Aug 2016), pp. 9–14. (p. 8)

ZHU, M. 2004. Recall, precision and average precision. (p. 33)