Evaluating Machine Learning Tools for Humanitarian Road Network Mapping

by

Robert Smith

A Thesis Presented to the
Faculty of the Dornsife College of Letters, Arts and Sciences
University of Southern California
In Partial Fulfillment of the
Requirements of the Degree
Master of Science
(Geographic Information Science and Technology)

August 2021

Table of Contents

List of Figures

List of Tables and Equations

List of Abbreviations

AI	Artificial Intelligence
DNN	Deep Neural Networks
Esri	Environmental Systems Research Institute
GIS	Geographic Information Systems
HOT	Humanitarian OpenStreetMap Team
MCC	Matthew's Correlation Coefficient
ML	Machine Learning
MLC	Maximum Likelihood Classifier
OCHA	United Nations Office for the Coordination of Humanitarian Affairs
OSM	OpenStreetMap
S3	Simple Storage Service
SVM	Support Vector Machine

Abstract

The majority of the world's poorest populations live in areas where the road network is unmapped, leaving them isolated from help during disasters. The Humanitarian OpenStreetMap Team (HOT) aims to solve this problem through volunteered geographic information. Although Humanitarian OpenStreetMap Team volunteers have traditionally digitized new roads manually, the organization is collaborating with Microsoft and Facebook AI to explore the application of machine learning to computer-assisted mapping workflows. As interest in machine learning applications grows within the humanitarian community, new organizations and volunteers may seek to carry out road classification projects for poorly-mapped regions. This thesis evaluates the major considerations of a classification workflow – primarily training data engineering and model selection. The study showcases the implications of training and model decision on a subset of the SpaceNet Road Network Detection dataset using five models: Random Forest, Support Vector Machine, Maximum Likelihood Classifier, UNet, and DeepLabV3. Models are evaluated in extensive experiments, assessing the impact of varying width of the road features used to label the training data. Each scenario is evaluated using a Matthew's Correlation Coefficient, confusion matrix, and visual inspection of the predictions. Based on these results, the project reports valuable lessons-learned that can inform humanitarian organizations' design for road classification workflows.

Chapter 1 Introduction

Despite technological advancements in geographic information systems and the increasing scope, quality, and accessibility of geospatial data, many of the world's most vulnerable communities remain unmapped. The lack of geospatial data in developing countries makes it difficult to respond to humanitarian crises, exacerbating the hardship of impoverished people. The Humanitarian OpenStreetMap Team carries out large-scale volunteer mapping campaigns to expand open-source data to vulnerable and disaster-stricken regions, enabling relief organizations to reach those affected.

For example, in the city of Mwanza, Tanzania, the Humanitarian OpenStreetMap Team is partnering with a variety of cartographic charitable organizations to map new, high-risk urban growth. During 2019 and 2020, the city experienced 5.56 percent population growth, mostly in the form of unplanned communities (Macrotrends 2021). 81 percent of Mwanza's residents are beneath the global poverty line, and these unplanned communities have even less access than their neighbors to basic utilities such as drainage and sanitation (Erman, et al. 2019). These circumstances exasperate the ever-present risk of flooding and landslides presented by Mwanza's topography of steep slopes and draws with rivers draining into Lake Victoria (Adinani and Kovacic 2020). The Humanitarian OpenStreetMap Team and its partners are working with university students in the region to manually digitize road and building footprint maps from overhead imagery, simultaneously creating data to empower response to near-term disasters and transferring knowledge to local GIS professionals to build long-term resilience. In addition to campaigns that focus on vulnerable locations like Mwanza, the organization also supports massive mapping global campaigns. The Humanitarian OpenStreetMap Team has partnered with DigitalGlobe and the United States Agency for International Development (USAID) to

provide geospatial support to malaria prevention campaigns. From 2016 to 2017, five thousand 5000 volunteers mapped over 560,000 square kilometers in Botswana, Zambia, Zimbabwe, Cambodia, Laos, Guatemala, and Honduras, mapping over five million buildings for the Clinton Health Access Initiative. During 2017, volunteers mapped an additional 30,000 square kilometers and another million buildings in Mali and Rwanda for Indoor Residual Spray sanitation under the President's Malaria Initiative (Humanitarian OpenStreetMap Team 2020).

In addition to these fully manual digitization campaigns, the organization has explored options for greater computer-assistance, including collaborations with Microsoft, Facebook, and a variety of smaller and more regionally focused technology research organizations (Percival, Delattre and Eckle 2020). One of the most fruitful of these efforts has been Facebook AI's development of the MapWithAI tools (Basu, Bonafilia, et al. 2019). These tools integrate with the Humanitarian OpenStreetMap Team's Tasking Manager and allow volunteers to quickly digitize roads by simply confirming or rejecting predictions made by a deep learning model trained for road classification. These mapping campaigns expand the OpenStreetMap Foundation's world map using both manual digitization and machine learning techniques. A growing number of humanitarian mapping efforts has also developed adjacent to HOT which are also evaluating machine learning workflows. One such effort is UNICEF's recent collaboration with Development Seed to map over seven thousand previously unmapped schools in Colombia and eleven Caribbean nations (Yi 2019). As the sphere of organizations involved in ML-assisted humanitarian mapping grows, it will become important to identify best practices for the development of certain workflows, to prevent redundant work. This is especially true for road classification and extraction, a problem for which many solutions have been proposed and implemented.

Figure 1: *Road extraction workflow. This project focuses on optimizing the 'Classify Pixels' step.*

Figure 1 lays out a general template for the workflow an organization could employ to map roads with machine learning. The automated component of this workflow are pixel classification, raster segmentation, and feature extraction. The organization would begin by using a machine learning technique to classify the pixels in a given area. The output from this step would be a raster layer containing the predicted class ('road' or 'non-road') for each pixel in the original imagery. The next step would be to segment the predicted raster – to separate the regions where each class dominates and smooth these areas, then convert the resulting raster to features (Hay, et al. 2005). The initial road map would be extracted from these features, and it is at this point that volunteers would become involved in the process by manually editing the predicted features. The hand-corrected data would then be ready for production mapping. An organization implementing such a workflow would have numerous decisions to make, with some of the most difficult ones pertaining to the pixel classification process (Hay, et al. 2005). This study focuses on this first step in the workflow.

1.1 Research Objectives

The objective of this project is to evaluate best practices for the design of machine learning workflows for humanitarian road network mapping. Common problems when designing a pixel classification process include model selection and training data preparation. Many existing machine learning models may be applicable to the problem of road detection, but some perform better than others. For this reason, it is important to evaluate the comparative performance of different existing machine learning models for unique use cases. Another

important step in the workflow, training data preparation is studied as it has direct influence on the model prediction performance.

To tailor insight to humanitarian applications, the experiment will focus thematically on a historical case study – HOT's response to the massive flooding that occurred in Khartoum, Sudan in 2013 (OCHA 2013). With few complete public maps available, HOT collaborated with regional NGOs to add roads and buildings to OpenStreetMap to aid in damage assessments (OpenStreetMap Wiki Contributors 2018). This project will assess that scenario by classifying roads in Khartoum using satellite imagery from the SpaceNet dataset, with OpenStreetMap roads acting as ground-truth labels for both training and validation.

The experiment will evaluate model selection and training data preparation by running five machine learning models for multiple iterations, each time varying the buffer distance used to create training data labels from the OpenStreetMap roads data. The results of this experiment will inform best practices for mapping organizations that integrate machine learning in their responses to future disasters like the 2013 Khartoum floods.

1.2 Motivation

Accurate road mapping is crucial to humanitarian response to natural disasters. Massive volunteer mapping campaigns like those carried out by the Humanitarian OpenStreetMap Team support infrastructure damage assessments and help relief workers locate and navigate to those in need of support (Humanitarian OpenStreetMap Team 2020). Integrating machine learning in mapping workflows holds the potential to multiply the positive impact of each volunteer mapper's effort, in turn helping governments and NGOs provide relief faster and more effectively to people in disaster-stricken regions. This project advances that end by developing

best practices for integrating machine learning with a future disaster response scenario analogous to the 2013 Khartoum floods.

Chapter 2 Related Work.

This project focuses on developing best practices for applying state-of-the-art machine learning tools and techniques to road mapping for humanitarian response. This study informs which models are appropriate for use in this experiment.

A variety of machine learning techniques are applicable to imagery classification problems (Mondal, et al. 2012). The concept of a random forest classifier is that it is an aggregation of the results from several decision trees (Breiman 2001). Each decision tree makes predictions by applying successive conditions until the classifications fit the training data. Individual decision trees are prone to overfitting – random forests mitigate this by generalizing across many instances of decision trees (Breiman 2001). Support vector machines (SVM) are similar to random forest classifiers in that they solve a classification problem by estimating a feature to most effectively distinguish target classses. SVMs are reported to more accurately separate classes that cannot be distinguished with linear classifiers with the use of kernels (Pupale 2018). Maximum likelihood classifiers (MLC) are another general-purpose machine learning model. This technique calculates the likelihood that any point within a data model will fall into each possible class, then assigns each point to its most likely class. In remote sensing, these models have been applied to land cover classification as well as segmentation of manmade objects (Mondal, et al. 2012).

Deep Neural Networks (DNNs) are emerging as the standard technique for image segmentation and pixel classification (Langkvist, et al. 2016). These models are based on image convolution, where a system iterates across an image tile and aggregates the values from each raster cell, producing a lower-resolution output that generalizes the features of the original input (Caia, Dimitriou and Arandjelovic 2020). UNet is one such technique, which implements a long

series of convolutional layers and pooling operations to aggregate an image, and then a reverses the pattern to generate segments from down-convolution (Ronnenberger, Fischer and Brox 2015). DeepLab is another DNN technique that improves on the original convolution method by introducing Atrous convolution (Zhou, Zhang and Wu 2018). This technique introduces gaps in the cell of the image that is being aggregated, thereby capturing a wider area in each cell of the output (Chen, et al. 2018)

Many previous studies have compared different types of machine learning models for specific practical applications (Langkvist, et al. 2016). This project does the same for road detection for humanitarian applications, with the purpose of identifying best practices. The work of Bonafilia, Yang, Gill, and Basu (2019) has been prominent in this area. Adopting a technique developed for the DeepGlobe 2018 challenge (Zhou, Zhang and Wu 2018), Basu et al. (2019) produced a dataset of road predictions spanning over 1.8 million square kilometers, accurately enough for immediate navigational use in many places (Basu, Bonafilia, et al. 2019). This dataset has been integrated with HOT's mapping tools and applied to large scale mapping campaigns in Thailand (Basu, Bonafilia, et al. 2019) and elsewhere.

Training data preparation is a key consideration in addition to model selection, and numerous satellite image classification experiments have developed novel labelling techniques (Langkvist, et al. 2016). Approaches have often combined automated and manual processes – for example, segmenting an imagery dataset then manually labeling each segment according to a classification schema (Langkvist, et al. 2016). Unsupervised feature labeling processes have also been applied to images (Singh, et al. 2018), with promising results in road network extraction (Basu, Bonafilia, et al. 2019). Due to the wide availability and coverage of OpenStreetMap roads, much attention has been given specifically to converting them into suitable labels for

machine learning. One such approach includes creating a buffer around each street-centerline feature, with a width determined by the class assigned by the volunteer according to OpenStreetMap's guidance for the region (Zuraban, Wightman and Brovelli 2019). One outcome of this experiment is to evaluate such a technique by measuring the effect of different buffer widths on prediction accuracy.

Chapter 3 Data and Methods

The application focus of this project is to support humanitarian mapping campaigns by identifying best practices for applying machine learning to road network extraction. The project focuses on state-of-the-art machine learning methods, and the free and publicly available SpaceNet Roads and OpenStreetMap datasets. A comparison of machine learning tools and training data preparation methods help identify best practices for organizations tackling road detection problems.

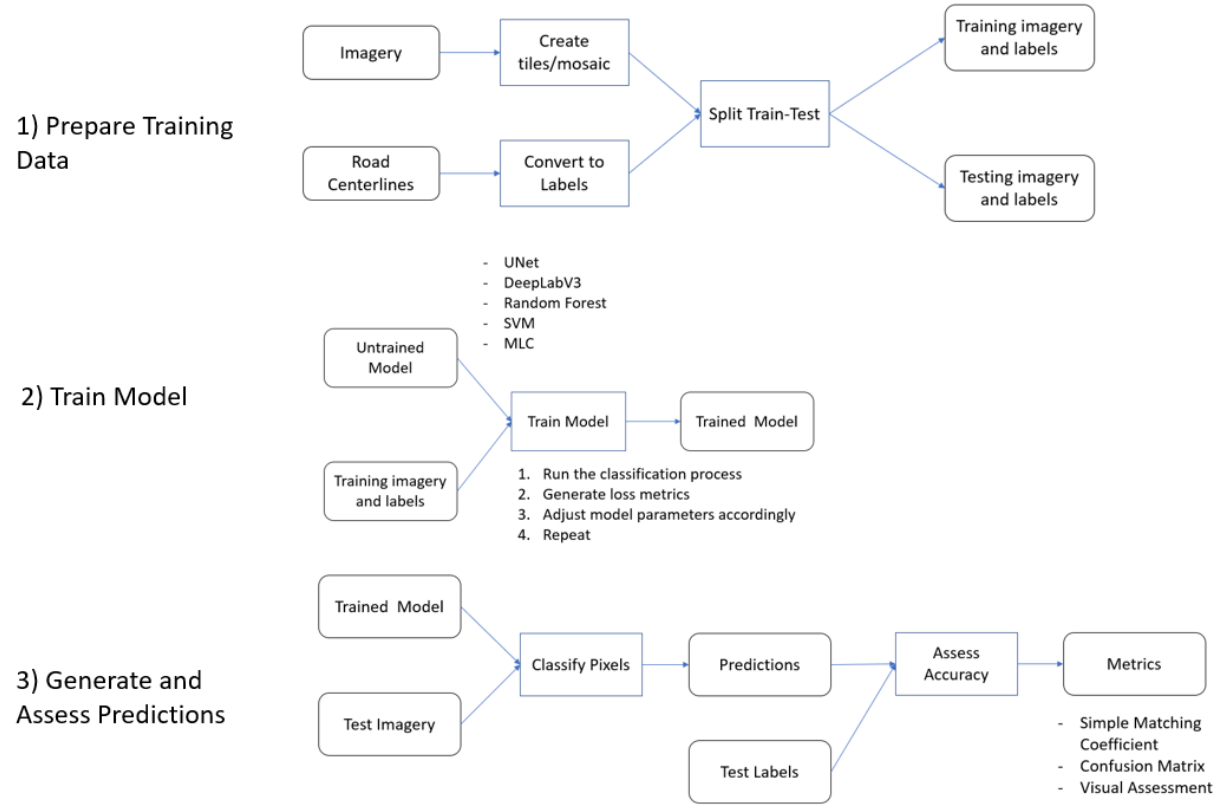

***Figure 2**: Template for an imagery classification workflow.*

Imagery classification workflows typically follow the outline shown in **Figure 2** The workflow begins with training data preparation. For a road network extraction problem like the one evaluated here, the data sources normally include remotely sensed imagery and road centerline data. The first step is to prepare the road centerlines for use as labels for the imagery.

This is important because regardless of the data format accepted by the software platform used to interface with a machine learning model, the models themselves will almost invariably require data to be in the format of an n-dimensional array, also known as a tensor (Stevens, Antiga and Viehmann 2020). The array is a single data structure that contains all the variables the model will consider for each pixel – its classification (road or non-road in this case), as well as the frequency sensed for each band in the imagery (eight in this case). The classification stored in the array for each pixel will depend on whether it is overlaid by the road label.

Road datasets typically consist of polyline geometries, so without any pre-processing only pixels which intersect the street's centerline would receive the road label classification. Creating a buffer around the centerline turns the roads into polygons that can more accurately label road pixels. Depending on the tools selected for implementation, it may be necessary to convert the road buffers to a raster layer, so that the two datasets can be converted to arrays and combined. One way to execute this step is to replace the values in the satellite imagery with ones, then clip the resulting raster to the road polygons. Since this project ran the models in their ArcGIS Pro implementations, some special considerations were made to label creation and data wrangling that Section 3.2 describes.

The second step is to train the model itself. Each model evaluated in this experiment approaches the actual problem of pixel classification differently, but for deep neural networks (DNNs) the learning process generally stays the same. Each DNN has a set of classification tools in place, as well as a set of optimizers – parameters that can be adjusted to change the model's outputs. The goal of the training process is to optimizes models in a manner that minimizes loss – the difference between predicted and ground-truth classes for a target dataset. Models are trained by repeating the classification process many times against the training

dataset. At the end of each iteration, known as an epoch, the model calculates the loss based on its predictions, then makes an adjustment to the optimizers (Stevens, Antiga and Viehmann 2020). This process repeats for a specified number of epochs, with loss decreasing after each iteration.

With the model trained, the final step is to generate and assess predictions. This means running the model against a new imagery dataset, then calculating metrics to see how it performs. The test data will typically be a section of the original dataset of imagery and roads, selected at-random and set aside at the beginning of the workflow (the 'Split Train-Test' step in **Figure 2**).

The project's experimental component is a sensitivity analysis of factors in a pixel classification workflow for road detection that require subjective choices. The first experimental variable is the distance used to create buffers from the street centerlines represented in the OpenStreetMap data. Scenarios will vary the width of the polygon road features used to label the training data, thereby exploring a wider search space in training the models that predict roads in the test iteration. The second experimental variable is the model being employed to predict the roads. By testing each model after training on a variety of road buffers, the experiment provides insight into how to optimize those parameters in a road classification workflow (**Table 1**).

Variable	Description	Expected Effect
Model	The type of machine learning model employed – Random Forest, Support Vector Machine, Maximum Likelihood Classifier, UNet, or DeepLabv3.	Some models are general-purpose while others are designed specifically for image classification problems, so some should perform better than others.
Road Label Width	The radius used to create the buffers used for road labels.	Buffers that are too wide will produce false positives, while buffers that are too narrow will produce false negatives

Metric	Description	Assesses
Matthew's Correlation Coefficient	Overall accuracy metric based on confusion matrix (see Equation 1).	Tests the overall predictive power of the model and quality of the results
Confusion Matrix	Compares predicted and ground-truth labels to calculate the rate of true-positive, false-positive, true-negative, and false-negative predictions.	Breaks down true predictions and errors to help identify causes of issues.
Visual Inspection	Visualize and review the results.	Validate and identify any issues with the other metrics.

Table 1: *Experimental variables and metrics.*

The experiment will evaluate the models' performance in each scenario (combination of classifier and buffer distance). The first is an accuracy score representing the proportion of accurate predictions. The accuracy score is a Matthew's Correlation Coefficient (MCC), calculated as

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

Equation 1: *Matthew's Correlation Coefficient.*

where *TP* is the number of true positive predictions, *TN* is the number of true negative predictions, *FP* is the number of false positive predictions, and *FN* is the number of false

negative predictions (**Equation 1**) (Chicco and Jurman 2020). MCC metric summarizes a classifier's predictive power.

The confusion matrix compares the number of true- and false-positive predictions against the number of true- and false-negative predictions. This will allow a more in-depth analysis of the accuracy of each model's predictions and is likely to assist with understanding each model's sensitivity to widening or narrowing the road labels. Visual inspection of the predictions generated by each scenario will serve as a final validation metric, identifying any obvious issues that the MCC and confusion matrix may not reflect.

3.1 Data Description

Table 2 summarizes the project's data sources. The SpaceNet Roads dataset consists of over 765sqkm of 31cm-resolution GSD WorldView3 imagery for Khartoum, Shanghai, Paris, and Las Vegas. The dataset is freely available through an Amazon Web Services S3 service, with download instructions on the SpaceNet project website. In addition to presenting the most difficult classification problem due to its many unpaved roads, Khartoum is a fitting study area candidate because it was the site of an extensive HOT mapping campaign after a series of destructive floods in 2013 (OpenStreetMap Wiki Contributors 2018). The imagery comes with some preprocessing done. It is divided into training and test datasets, each of which is divided into tiles and processed into four formats – multispectral (8-band), pan-sharpened, pansharpened-multispectral, and pan-sharpened-RGB. The training data consists of 471 TIFF images. Each pixel in the pansharpened-multispectral imagery contains sixteen bits of data and covers twenty-seven square centimeters. The tiles are projected to WGS1984, and each one takes up 25.79 megabytes of memory uncompressed.

DigitalGlobe's WorldView-3 satellite collects a panchromatic band, eight multispectral bands (blue, green, yellow, red, red-edge, and two near-infrared bands), eight short-wave infrared bands, and twelve bands capturing clouds, aerosols, water vapor, ice, and snow (CAVIS). To maximize the resolution and number of input variables for the training data, this project used the pansharpened-multispectral imagery. This imagery consists of the eight-band multispectral imagery combined with the panchromatic band. The multispectral imagery is collected at a 124cm resolution, and the panchromatic band is collected at a 31cm resolution, so combining the two datasets produces eight-band imagery with 31cm pixels (DigitalGlobe 2014).

Name	Source	Accuracy/Resolution	Description
OpenStreetMap Highways for Africa	OpenStreetMap, Esri	Visual inspection confirms suitability as ground-truth reference.	Hand-digitized road network produced by OpenStreetMap volunteers
SpaceNet Roads Dataset - imagery	SpaceNet	31cm pixel-size	765sqkm of GSD WorldView3 multispectral imagery, pre-sorted into training and test datasets.

Table 2: *Data sources..*

The project uses OpenStreetMap roads to label both training and test imagery. A visual inspection of the roads confirms their quality as a ground-truth reference for this project (**Figure 3**). OpenStreetMap data is available through many sources – for this project, it was accessed via

the ‘OpenStreetMap Highways for Africa’ Living Atlas Feature Layer from ArcGIS Online, then clipped to the SpaceNet imagery extent.

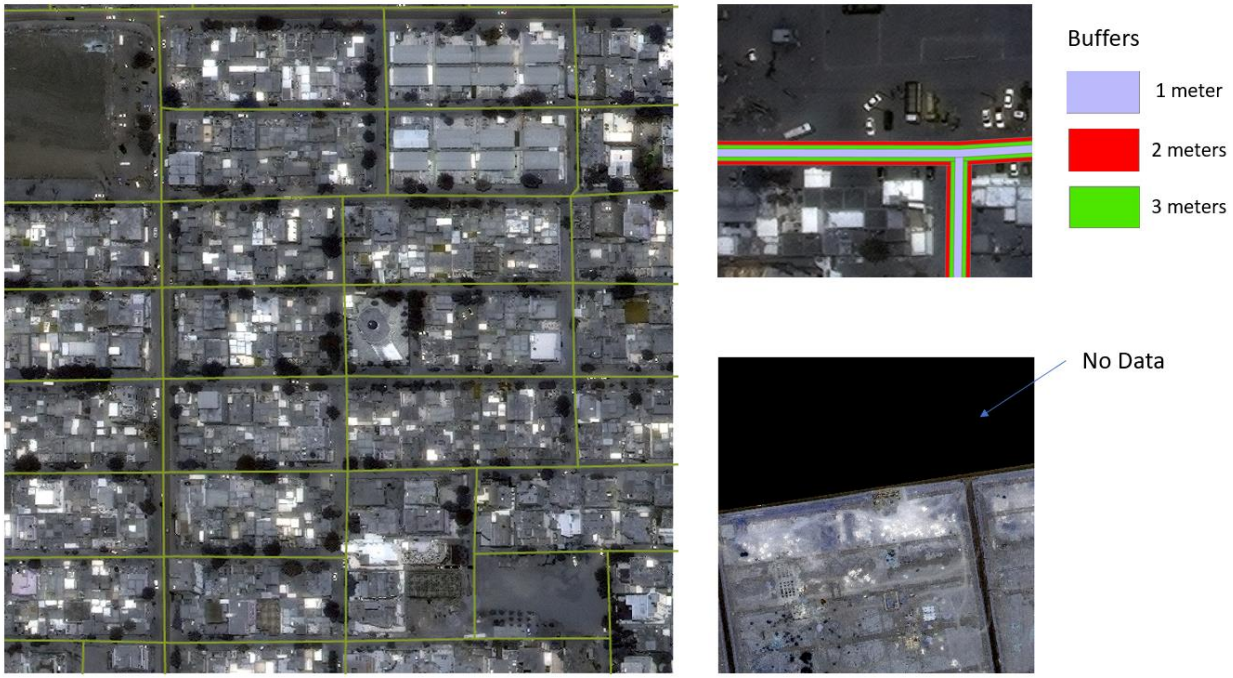

Figure 3: *Sample of the data used in this study. To the left is an image of the OpenStreetMap roads overlaying SpaceNet imagery. Top-right is an image of a section of the OSM roads buffered to the distances tested here – these buffers served as the road labels. Bottom-right is an example of a no-data section of the imagery.*

Figure 3 highlights key characteristics of the source data. First, the OpenStreetMap roads appear complete and very accurate for the study area (left). This is likely due to the attention this section of the map received during the Humanitarian OpenStreetMap Team’s response to the Khartoum floods in 2013 (OpenStreetMap Wiki Contributors 2018). The image on the top-right illustrates how different buffer distances fit roads in the study area. Getting a buffer distance that accurately captures the roads is a key aspect of preparing the training data, and this experiment will assess the impact of different buffer distances. Image on the bottom-right displays an area of a tile where no data was available. The SpaceNet dataset contains numerous such dark spots, which consist of pixels containing a zero for each raster band. Tiles with large dark spots, corresponding with zeros on all bands, were omitted.

3.2 Experimental Design

The experimental design's purpose is to refine a pixel classification workflow by testing combinations of buffer distances and machine learning models. These scenarios produce classification diagnostics (**Figure 6**) that will indicate the best model and the most optimal road buffer distance to produce the most accurate and efficient predictions. The diagnostics will also highlight the effect of buffer-distance and model selection, evaluating the importance of these choices. The method described below was implemented with the GIS data processing tools and machine learning models available through ArcGIS Pro. The workflow consisted of three sections: data preparation, pixel classification, and accuracy assessment.

Preparing the data for classification and accuracy assessment entails staging the imagery and preparing the ground-truth labels. This process consists of creating the road buffers and splitting the imagery tiles further into 256 pixel x 256 pixel chips, along with their corresponding labels. Once the imagery and labels are in the appropriate format, they are randomly split into training and test datasets for model training (**Figure 4**). Two complications during implementation are worth noting here and will receive further discussion in Chapter 5. The first is that the standard machine learning models (Random Forest, Support Vector Machine, and Maximum Likelihood classifiers) required labels from each class in the classification schema in order to train. The solution to this was to label the training data for those models with two buffers – the first denoting 'road' pixels, and the second denoting 'non-road' pixels, as shown in the right-side image of **Figure 4** Another issue was that the standard machine learning models as implemented for this experiment were not compatible with tiled images – only single raster layers or mosaics. However, with the hardware available, only a single 1.69-megapixel image

could be processed at a time. This limited the training and testing of these models to a single SpaceNet imagery tile.

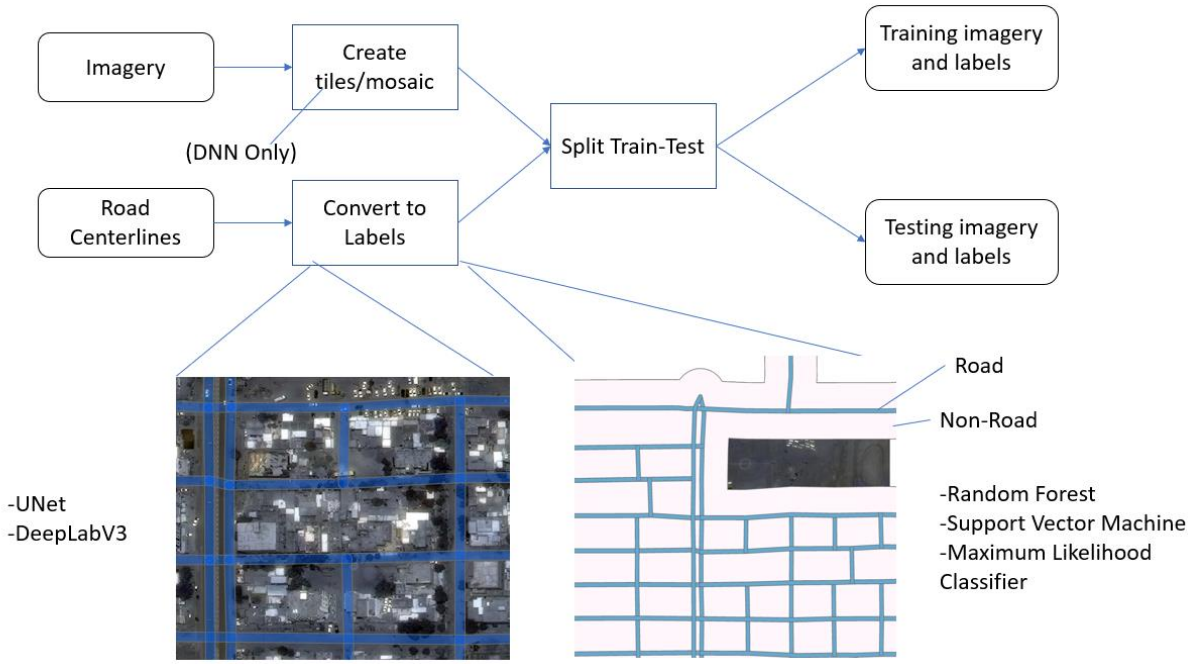

Figure 4: *Training data preparation process.*

With the training and test data prepared, the classifiers are run against test imagery, with varying combinations of models and buffer distances. The method of accuracy assessment in this project is an Matthew's Correlation Coefficient and confusion matrix for each scenario run. After a set of predictions is generated, the workflow defines five-hundred random points on the classified raster which serve as a representative sample of predictions. The result is an accuracy assessment points feature class that includes the predicted and actual classification for the pixel at each point. Based on this data, the experiment calculates a Matthew's Correlation Coefficient (**Equation** 1) and confusion matrix.

3.2.1 Overview of Models

This project compares five models for image classification: Random Forest, Support Vector Machine, Maximum Likelihood Classifier, UNet Classifier, and DeepLabV3 classifier.

Each has strengths and weaknesses in both predictive power and performance. Random Forest classifiers, Support Vector Machines, and Maximum Likelihood Classifiers are general-purpose machine learning models which see application across a variety of industries. UNet and DeepLabv3 are both deep neural networks (DNNs) which were designed specifically for processing images. The Random Forest Classifier aggregates the results of several different decision trees. A decision tree classifier fits to a dataset by successively adding and adjusting conditions that split the data into its respective classes. Decision trees tend to be extremely fast and performant on large datasets, but are generally prone to overfitting (Breiman 2001). Random forest models mitigate this by running several decision trees on different subsets of the training data and then aggregating the results, at the expense of time (Liberman 2017).

A support vector machine (SVM) attempts to separate two classes by fitting a line, similarly to how a decision tree classifier works. SVMs improve over decision tree and random forest classifiers by projecting data into higher dimensions when no line separating classes can easily be fitted. For example, if points of two classes are not linearly separable, the SVM may project those points into three dimensions with use of kernels. As a result, non-linear decision surfaces can be defined with linear functions (Pupale 2018). In previous comparisons, SVMs have outperformed random forest classifiers when a large number of features are present in imagery (Sheykhmousa, et al. 2020).

Maximum Likelihood Classifiers (MLC) are another commonly employed classification technique. An MLC calculates the likelihood of a pixel falling into each class based on the training data, and then assigns each pixel the most likely class. SVMs have been shown to be more accurate than MLCs across many classification problems, including separation of humanmade structures (Mondal, et al. 2012).

The two Deep Neural Network models tested in this project are UNet and DeepLabV3. UNet was originally developed to segment microscope images for medical research. The original design consists of 23 convolutional layers with pooling layer interspersed between. The convolutional layers generalize an image by moving across each row of pixels and summarizing the values in each section until, for example, a 586x586-cell image tile is reduced to 284x284. This process then gets reversed, and each layer successively learns to build a more accurate image (Ronnenberger, Fischer and Brox 2015). DeepLabV3 reflects a more traditional 'sliding-window' convolutional neural network. However, its developers introduced Atrous convolution, which widens the perspective of a convolution layer by introducing holes in the window (Chen, et al. 2016).

Although DeepLabV3 and UNet represent the cutting edge and industry standard for image segmentation, the literature suggests that SVMs may present strong competition from the domain of more general-purpose machine learning models (Mondal, et al. 2012).

Chapter 4 Results

This project classified road and non-road pixels using different techniques with varying parameters and training data. From these results, the project derived lessons learned for imagery analysis in small-scale GIS implementations. These issues themselves provide valuable insight to practitioners seeking to implement a GIS for humanitarian road network mapping.

In addition to lessons learned from implementation, the outputs of the experiment provide insight that can inform future workflows. The quantitative results generated for each scenario are a confusion matrix displaying true and false predictions for road and non-road pixels, Matthew's Correlation Coefficients calculated for each confusion matrix, and the raster datasets containing the predictions themselves made by each model.

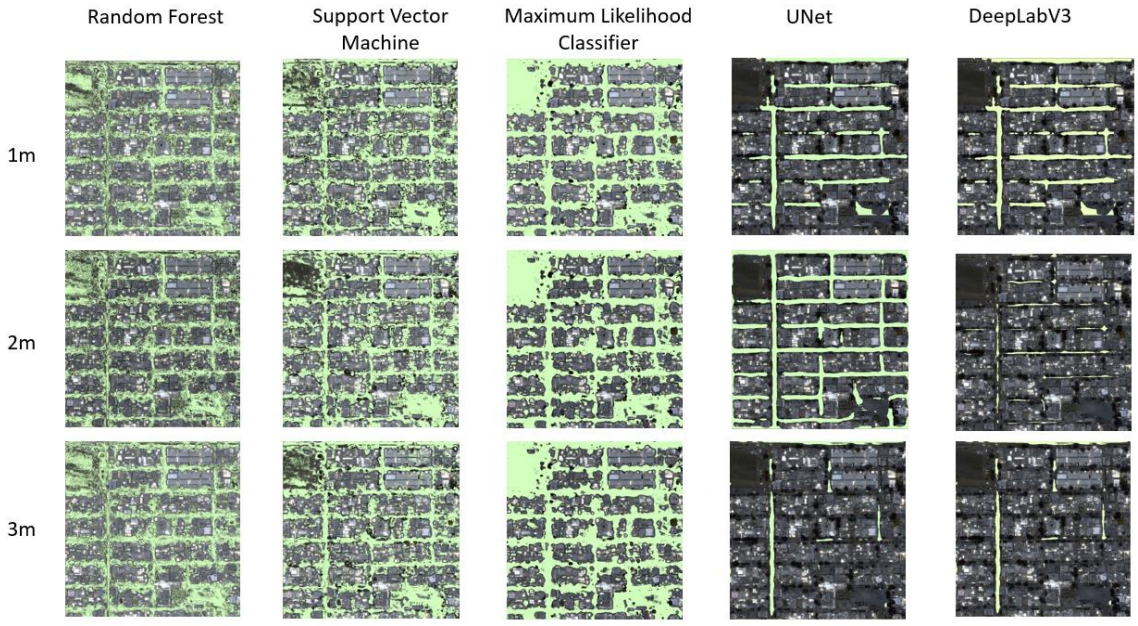

Figure 5: *Classification predictions.*

The first result that stands out on examination of the predictions (**Figure 5**) is that the standard machine learning models appear far more prone to false positives than the DNNs, whereas the DNNs appear to tend toward false negatives – the tiles for RF, SVM, and MLC are crowded with positive predictions, whereas the UNet and DeepLab tiles appear sparse. The confusion matrices (**Figure 6**) support this result, reporting an average false positive rate for the standard models twenty-seven percent higher than the DNNs. The confusion matrices also report average false negative rates 36% lower for the DNNs than for the standard models.

The effect of road label width also appeared to have a far stronger impact on the DNNs than on the standard machine learning models. The standard models' average true positive rate increases only three percent between one-meter buffer and three meter buffers, whereas the average true positive rate drops by forty two percent for the DNNs. This difference is clear when observing the predictions – while RF, SVM, and MLC produce similar predictions across all buffer distances, UNet and DeepLab both produce far fewer positive predictions with three-meter buffers than with one-meter buffers.

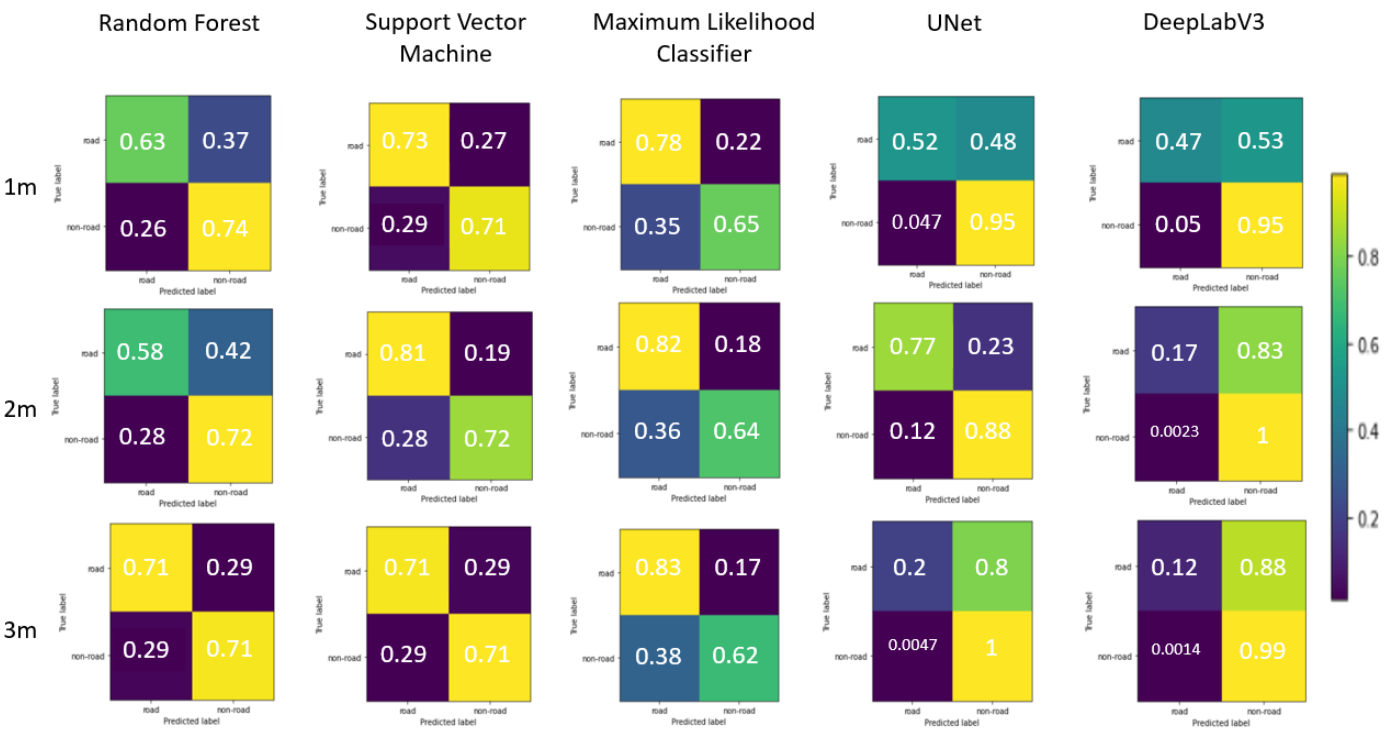

***Figure* 6:** *Confusion matrices.*

The MCCs for each scenario are displayed in **Table 3**. MCC scores range from negative one to one, with negative one indicating one hundred percent false predictions and one indicating a perfect set of true predictions. A range of .229 to .577 indicates a moderately strong positive correlation between model predictions and true values overall (Chicco and Jurman 2020). Across all fifteen scenarios, the average coefficient was .362. The MCCs do not appear to sharply divide the standard models from the DNNs, except for the UNet scores at one meter and two meters. UNet achieved the highest coefficient at .577 with the two-meter labels, the only model to reach a coefficient over .5. This score is sharply contrasted with DeepLabv3's score of .367, also with the two-meter buffers. UNet appears to peak with two-meter buffers, dropping to .382 with three-meter buffers. This peak in accuracy, as well as the contrast between UNet and DeepLabv3 with two-meter buffers, are also both clearly visible in **Figure 5**.

	RF	SVM	MLC	UNet	DeepLabv3
1m	.294	.341	.327	.522	.481
2m	.229	.402	.341	.577	.367
3m	.310	.312	.331	.382	.218

Table 3*: Matthew's Correlation Coefficients for each scenario.*

Comparing the mapped predictions themselves, confusion matrices, and MCCs both quantify and qualify the test's results – the visual inspection provides an intuitive insight into performance, and the confusion matrices and MCCs each provide quantitative evidence to support those intuitions. The confusion matrices confirm the visible overall difference in prediction accuracy between the standard models and the DNNs – the standard models were generally prone to false positives, whereas the DNNs were more prone to false negatives in some cases, though not all. UNet stands out as the most accurate classifier in this test, especially when trained with two-meter road buffers. The difference in accuracy between UNet and DeepLabv3 with two-meter buffers also stands out as anomalous. Although the MCCs differ between UNet and DeepLabv3 for all three buffer distances, two meters is the only distance where the predictions also differ visibly.

Chapter 5 Conclusion

Out of the models tested here, UNet performed with the highest accuracy, producing outputs that could be mistaken for pre-processed road features rather than mere classified rasters. Road label width was a strong factor in accuracy for the DNNs (though less so for the standard machine learning models). UNet's performance peaked when trained on two-meter buffers, then fell sharply when the buffers increased to three meters.

Further review of the implementation in this project reveals several useful lessons. Buffering multiple rings around an object can be an effective way to convert existing data into training labels with built-in 'non-road labels for binary classification. However this is not necessary for more feature detection-oriented models like UNet and DeepLabV3. Excluding 'non-road' labels for these models results in only the positive road predictions being included in the output, which could streamline the post-processing workflow. The accuracy assessment method – calculating confusion matrices from a stratified random sample of points, then generating Matthew's Correlation Coefficients – proved effective at quantifying and supporting the results which were already evident upon visual inspection of the predictions.

The primary limitation of this experiment was the processing capacity available for articulating the imagery. Although it was still possible to train the DNN models on the full SpaceNet Khartoum training dataset, this was not possible for the standard ML models, which instead trained on a single 400m x 400m tile. This presents a confounding factor when interpreting the experiment's results, as it is possible that standard ML models could have performed much better if trained on a larger dataset.

Figure 7: *Road labels created from three-meter buffers. The three-meter buffer appears to fit the roads in Khartoum well - any wider and the label would likely start catching too many non-road pixels and cause false positives.*

Training the DNN models took upwards of three hours for each scenario, not to mention the classification itself, meaning that running all ten buffer distances for the two DNNs alone could have more than a week to complete. This issue forced the reduction from ten buffer distances to three, as shown in **Figure 3** in Chapter 3. Regardless, this change does not likely impact the experiment's outcome dramatically, because upon visual inspection it appears that any buffer beyond three meters would have resulted in false positives (**Figure 7**). Adding increments of less than a meter (i.e 1m, 1.5m, 2m, 2.5m, 3m) likely would have not provided valuable results for the time it would have taken to run those scenarios.

As for the standard machine learning models, as noted in Chapter 3, attempting to train the models on anything larger than a single 1.69MP tile caused the tool to fail, likely due to lack of available computing resources. Because of this, these models were trained on a single tile and tested on another tile. The issue with this is that it puts the standard ML models at a disadvantage compared to the DNN models, which were able to train on the entire dataset. This is a consideration when evaluating the results.

Future work can pursue the goals of this project by adding scale to the experiment. One plausible approach is to implement the same experimental design with a system purpose-built for ingesting and processing a high volume of imagery, and to expand the sample size for training and prediction. This study focused on a single locale, residential roads in Khartoum, and can be extended to areas of the city at various states of development. Another way is to include models omitted here. For example, this experiment did not test the more recently developed Multi-Task Road Extractor, which improves the connectivity of detected roads by simultaneously learning road direction and pixel characteristics (Batra, et al. 2019). Additionally, future work could test the effects of different training data preparation techniques. For example, a binary mask could be used to remove no-data values from the training imagery, shown in **Figure 3**. A future experiment could also evaluate the results of labelling roads by creating buffers with different widths depending on the road segment's classification in the OpenStreetMap data (Zuraban, Wightman and Brovelli 2019). Finally, future work could extend these studies into production and evaluate the performance of mapping volunteers working with each extracted dataset (Percival, Delattre and Eckle 2020).

The results found in this assessment are of value to any organization that intends to integrate machine learning in a road mapping workflow, especially for humanitarian purposes. There is a growing community of NGOs, corporations small and large, and government agencies involved in this work, with increasing interest in machine learning. These organizations have many factors to consider when planning a road extraction workflow, of which pixel classification is small but crucial component. This experiment indicates that when implementing a pixel classification process, organizations should consider two-meter centerline buffers for residential

roads for their training labels, and that UNet will provide more accurate results than comparable DNN models and more general-purpose machine learning techniques.

Works Cited

Adinani, Hawa, and Primoz Kovacic. 2020. "Commmunity Mapping for Urban Risk in Mwanza, Tanzania." *Humanitarian OpenStreetMap Team.* October 22. Accessed 2021. https://www.hotosm.org/updates/spatial-collective-humanitarian-openstreetmap-team-and-openmap-development-tanzania-extend-their-engagement-to-mwanza/.

Basu, Saikat, Derrick Bonafilia, James Gill, and David Yang. 2019. *Building High-Resolution Maps for Humanitarian Aid and Development with Weakly- and Semi-Supervised Learning.* Menlo Park: Facebook Research.

Batra, Anil, Suriya Singh, Saikat Basu, Guan Pang, C. V. Jawahar, and Manohar Paluri. 2019. "Improved Road Connectivity by Joint Learning of Orientation and Segmentation." *Conference on Computer Vision and Pattern Recognition.* Long Beach: Conference on Computer Vision and Pattern Recognition.

Breiman, Leo. 2001. *Random Forests.* Berkeley, CA: University of California Berkeley Statistics Department.

Caia, Peter D., Neofytos Dimitriou, and Ognjen Arandjelovic. 2020. "Precision medicine in digital pathology via image analysis and machine learning." In *Artificial Intelligence and Deep Learning in Pathology*, 149-173. Amsterdam: Elsevier.

Chen, Liang-Chieh, George Papandreou, Kevin Murphy, Alan L. Yuille, and Iasonas Kokkinos. 2016. "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs." *ArXiv.*

Chen, Liang-Chien, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. 2018. "Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation." *arXiv.* https://arxiv.org/abs/1802.02611v2.

Chicco, Dave, and Giuseppe Jurman. 2020. "The advantages of the Matthews correlation coefficient (MCC) over F1 score an daccuracy in binary classification evaluation." *BMC Genomics* 21 (6).

DigitalGlobe. 2014. *WorldView-3 Data Sheet.* Technical Specifications, London: DigitalGlobe.

Erman, Alvina, Mercedeh Tariverdi, Marguerite Obolensky, Xiaomeng Chen, Rose Camille Vincent, Silvia Malgioglio, Jun Rentschler, Stephane Hallegatte, and Nobuo Yoshida. 2019. *Wading Out the Storm: The Role of Poverty in Exposure, Vulnerability and Resilience to Floods in Dar Es Salaam.* Policy Research Working Paper, Washington D.C.: World Bank. https://documents1.worldbank.org/curated/en/788241565625141093/text/Wading-Out-the-Storm-The-Role-of-Poverty-in-Exposure-Vulnerability-and-Resilience-to-Floods-in-Dar-Es-Salaam.txt.

Hay, Geoffrey J, Guillermo Castilla, Michael A Wulder, and Jose R Ruiz. 2005. "An automated object-based approach for the multiscale image segmentation of forest scenes." *International Journal of Applied Earth Observation* 7: 339-359.

Humanitarian OpenStreetMap Team. 2020. *What We Do.* https://www.hotosm.org/what-we-do.

Langkvist, Martin, Andrey Kiselev, Marjan Alirezaie, and Amy Loutfi. 2016. "Classification and Segmentation of Satellite Orthoimagery Using Convolutional Neural Networds." *Remote Sensing.*

Liberman, Neil. 2017. "Decision Trees and Random Forests." *Towards Data Science,* January 26.

Macrotrends. 2021. *Mwanza, Tanzania Metro Area Population 1950-2021.* https://www.macrotrends.net/cities/22899/mwanza/population.

Mondal, Arun, Sananda Kundu, Surendra Kumar Chandniha, Rituraj Shukla, and P. K. Mishra. 2012. "Comparison of support vector machine and maximum likelihood classification technique using satellite imagery." *International Journal of Remote Sensing and GIS* 1 (2): 116-123.

OCHA. 2013. *Sudan: UN emergency funds bolster support for flood hit communities.* New York City: United Nations Office for the Coordination of Humanitarian Affairs. https://www.unocha.org/story/sudan-un-emergency-funds-bolster-support-flood-hit-communities.

Percival, Bo, Felix Delattre, and Melanie Eckle. 2020. *How we measure the effects of AI-assisted mapping.* Web Page, Washington DC: Humanitarian OpenStreetMap Team.

Pupale, Rushikesh. 2018. "Support Vector Machines (SVM) - An Overview." *Towards Data Science*, June 16.

Ronnenberger, Olaf, Philipp Fischer, and Thomas Brox. 2015. "U-Net: Convolutional Networks for Biomedical." *ArXiv.*

Sheykhmousa, Mohammadreza, Masoud Mahdianpari, Hamid Ghanbari, Pedram Ghamisi, Saeid Homayouni, and Fariba Mohammadimanesh. 2020. "Support Vector Machine Versus Random Forest for Remote Sensing Image Classification: A Meta-Analysis and Systematic Review." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 13: 6308 - 6325. https://ieeexplore.ieee.org/document/9206124/.

Singh, Suriya, Anil Batra, Guan Pang, Lorenzo Torresani, Saikat Basu, Manohar Paluri, and C. V. Jawahar. 2018. "Self-Supervised Feature Learning for Semantic Segmentation of Overhead Imagery." *British Machine Vision Conference.* Newcastle: British Machine Vision Conference.

Stevens, Eli, Luca Antiga, and Thomas Viehmann. 2020. *Deep Learning with PyTorch.* Shelter Island, NY: Manning Publications Co.

Yi, Zhuangfang. 2019. *Finding unmapped schools from space with AI.* Medium.com, May 20. https://medium.com/devseed/finding-unmapped-schools-from-space-with-ai-28459f68c2f3.

Zhou, Lichen, Chuang Zhang, and Ming Wu. 2018. "D-LinkNet: LinkNet with Pretrained Encoder and Dilated Convolution for High Resolution Satellite Imagery Road Extraction." *Conference on Computer Vision and Pattern Recognition.* Salt Lake City: Conference on Computer Vision and Pattern Recognition.

Zuraban, M. A., P. Wightman, and M. A. Brovelli. 2019. "A machine learning pipeline articulating satellite imagery and openstreetmap for road detection." *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 26-30.