Fall 2022 Capstone Project

Progress Report 1

Automatic landcover change detection and classification from Satellite images

JPMC 3

Team members:

Binny Naik, bn2341
Gangadhara Reddy Velagala, gv2310
Nishi Amish Modi, nam2238
Sai Prashanth Pathi, sp3915
Vishwas Reddy Thuniki, vt2356

Mentor:

Saba Rahimi

Table of contents

1. Introduction

1.1 Project Scope and Goal

This Capstone project is a collaboration between Columbia University at the Data Science Institute and JPMorgan Chase & Co. The objective of this Capstone Project is to build a Machine Learning model to automate the land cover change detection and classification from multitemporal, multiresolution, and multispectral satellite imagery over the Kigali region of Africa.

The 5m resolution NICFI imagery dataset from 2016 and 2019 (input images) and the Dynamic World Labels (target images) are being used for model training. The scope of the project is to classify the land cover into Tree Canopy, Buildings/Impervious land and Water for each pixel by building a model and then using it to detect land cover change for each class from satellite images of 2016 and 2019. We rely on Dynamic World images for model training purposes, however, we also manually annotate a set of 50 images to ensure that the model performance does not deviate with real world images.

We also focus on applying exploratory data analysis strategies to explore our data and find any interesting and meaningful patterns in the images. After researching from multiple resources we decided to train a modified UNet (to reduce training time) as our baseline model for detecting land cover change. The project will include the use of Google Earth Engine (GEE) for the manual extraction of images. We decided to use Google Earth Engine as it combines a multi-petabyte catalog of satellite imagery and geospatial datasets with planetary-scale analysis capabilities. The result of our project will be a machine learning model that will enable us to automatically detect the regions affected by deforestation in Kigali in 2019 as compared to 2016.

2. Data

All the data has been manually extracted from Google Earth Engine using NICFI (Norway's International Climate and Forest Initiative) and Dynamic World V1 satellite images. For this project we have considered two different time periods of 2016 and 2019 to capture the effect of land cover change in the region of Kigali, Rwanda. We have specifically only selected the data for the month of June/July from both years. The reason for selecting the same season/time frame is for true comparison of land change ignoring the effect of seasonality. Example: Spring season cannot be compared to Winter season as the vegetation cover will be different because of natural reasons.
The Area of Interest covering 3875.32 km2 area has been split into 256 smaller images for the purpose of exploratory data analysis and modeling.

2.1 NICFI

This image collection provides access to 5m resolution satellite monitoring of the tropics for the primary purpose of reducing and reversing the loss of tropical forests, contributing to combating climate change, conserving biodiversity, contributing to forest regrowth, restoration and enhancement, and facilitating sustainable development.

Dataset name:
Planet & NICFI (Norway's International Climate and Forest Initiative) Basemaps for Tropical Forest Monitoring - Tropical Africa

Earth Engine snippet:
ee.ImageCollection("projects/planet-nicfi/assets/basemaps/africa")

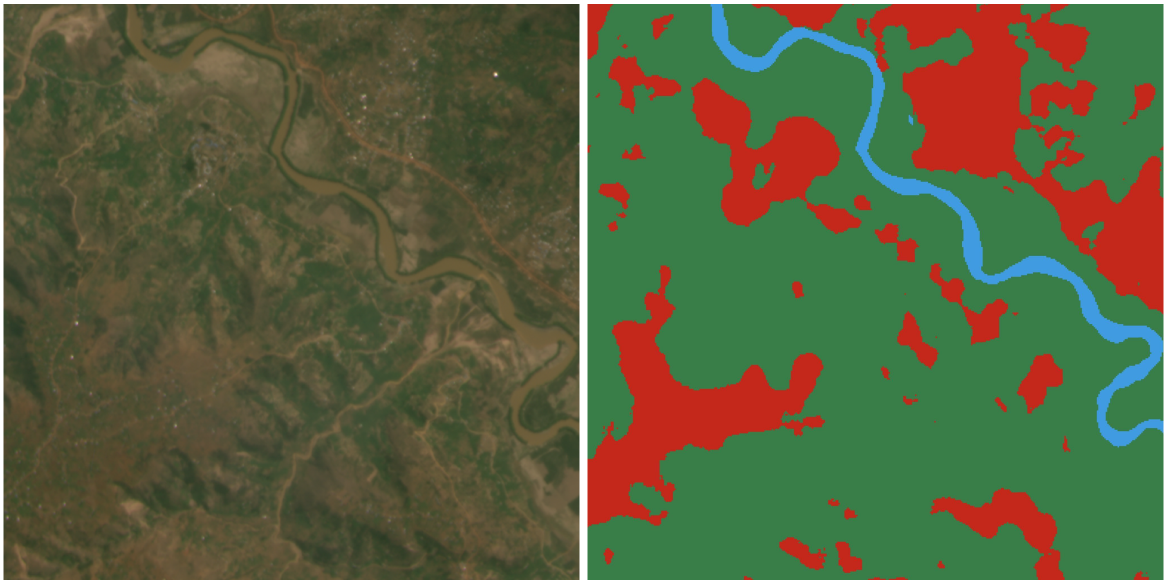


Figure 1: Figure on the left is a NICFI image and on the right is the corresponding Dynamic world image using 3 classes.

2.2 Dynamic World

Dynamic World is a 10m near-real-time (NRT) Land Use/Land Cover (LULC) dataset that includes class probabilities and label information for nine classes. The 9 class Dynamic world images have been modified into 3 classes for our use case. The three new classes are 0 - Water, 1 - Tree and 2 - Impervious/Buildings

Dataset name:
Dynamic World V1

Earth Engine snippet:
[ee.ImageCollection("GOOGLE/DYNAMICWORLD/V1")]

2.3 Ground Truth Labels

The images sourced from NICFI are manually labeled into 3 classes using Ground Works. For the annotation we manually mark regions which are trees, water and buildings/impervious using NICFI as reference. This is a tedious and time consuming process as each image has to be looked at in detail and labeled accordingly. These manually labeled images are used as ground truth labels.

From the ground truth labels we extract each class by using the color of each label. The mask for each class is created using the CV2 technique of inRange. Since this technique itself is an approximation, we can expect low performance for the final validation of images. Also we need to note that NICFI is only a 5m resolution image which is much lower than high resolution images of 0.8m. The low resolution of NICFI images makes it difficult to differentiate small buildings, barren lands against bushes and crops which could lead to misclassification. This misclassification of built to trees and trees to built class can result in low performance of the model.

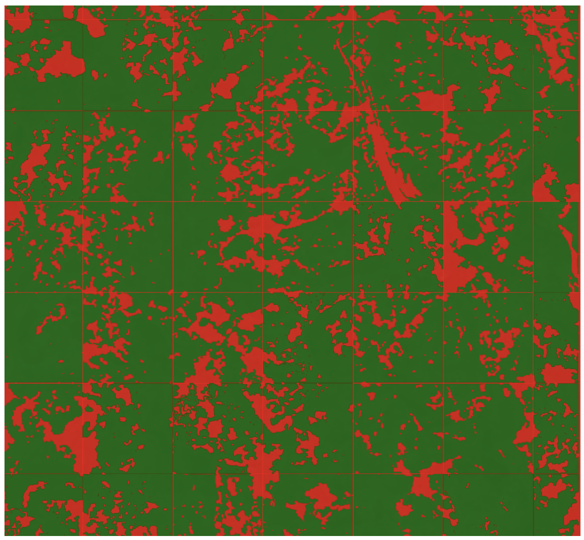


Figure 2: The above image is a manually labeled ground truth image

2.4 Data wrangling and cleaning

All the images had to be cleaned and standardized before processing. The cleaning and wrangling tasks included -

- Selecting the desired number of pixels and area covered per image
- Cloud cover
- Constant time frame of extracted images for each year
- Selecting the appropriate bands from each satellite
- Extracting image in multiple formats as TIF, PNG, etc
- Each satellite/dataset has separate preprocessing

Since Google Earth Engine was used most of the preprocessing was done using GEE software and specific libraries for each satellite which was time consuming.

3. Exploratory Data Analysis

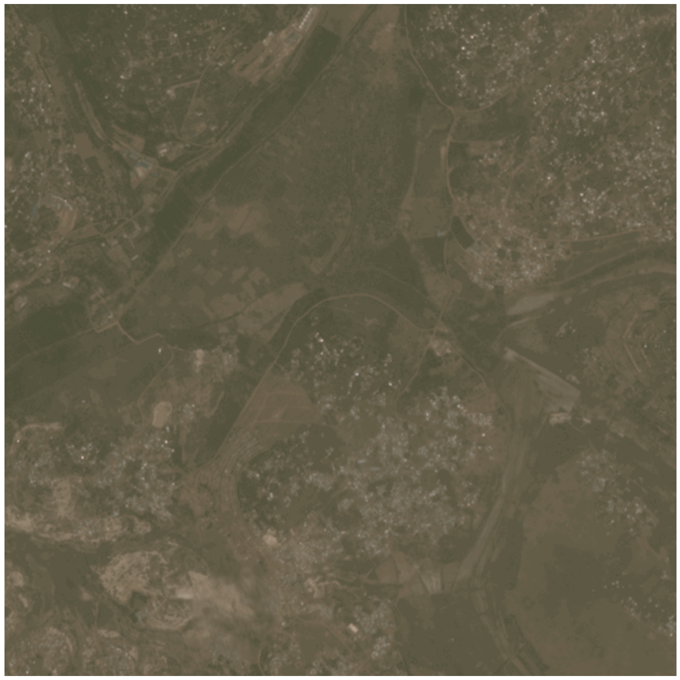


Figure 3: GIF which shows the land cover of two different time periods using satellite imagery.

In the reference paper [1] there are 4 classes for land cover each being water, tree cover, low vegetation and impervious. Let's look at the class distribution for our Area of Interest (AOI).

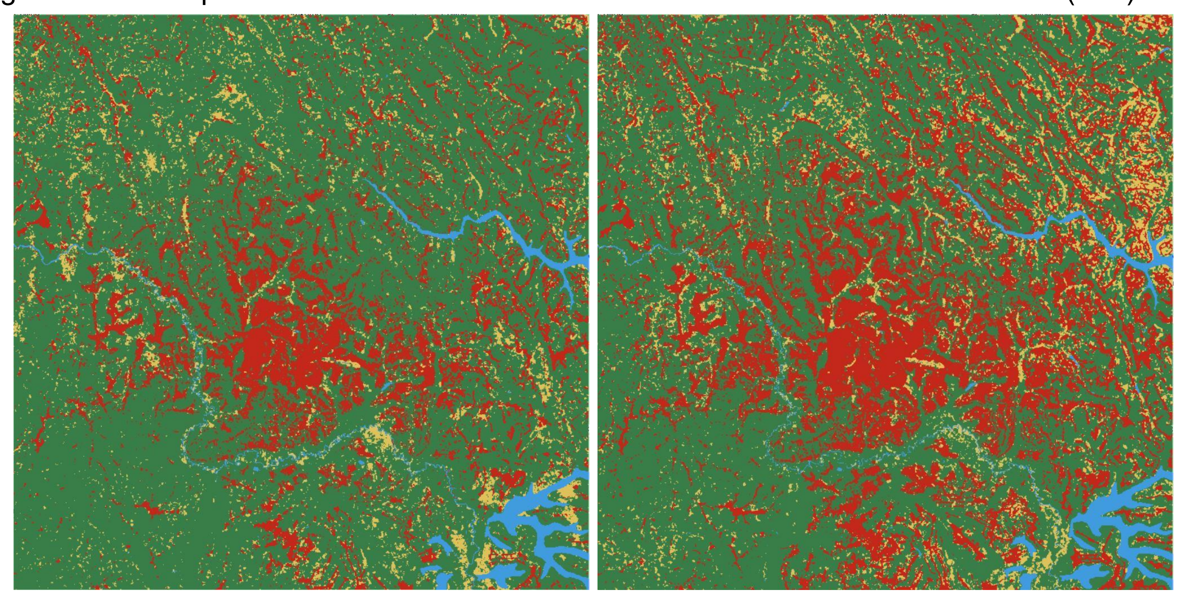


Figure 4: Figure on left is 2016 and on right 2019 Dynamic world images of AOI (Kigali) using 4 classes

If you look at the image you can see that the low vegetation cover/class is very low.

Coming to NICFI satellite image it is of 5m resolution, which is not a high resolution satellite image, which makes it difficult for ground truth labeling of land cover. Hence, we are going ahead with 3 classes in total for our analysis and project. The three classes are tree canopy, water and impervious/built area.

Dynamic World				Our use case	
Value	**Color**	**Color_as_seen**	**Description**	**New Class Desc**	**New Color**
0	419BDF		water	Water	
1	397D49		trees	trees	
2	88B053		grass	trees	
3	7A87C6		flooded_vegetation	trees	
4	E49635		crops	trees	
5	DFC35A		shrub_and_scrub	trees	
6	C4281B		built	impervious	
7	A59B8F		bare	impervious	
8	B39FE1		snow_and_ice	Water	

Table 1: Class labels

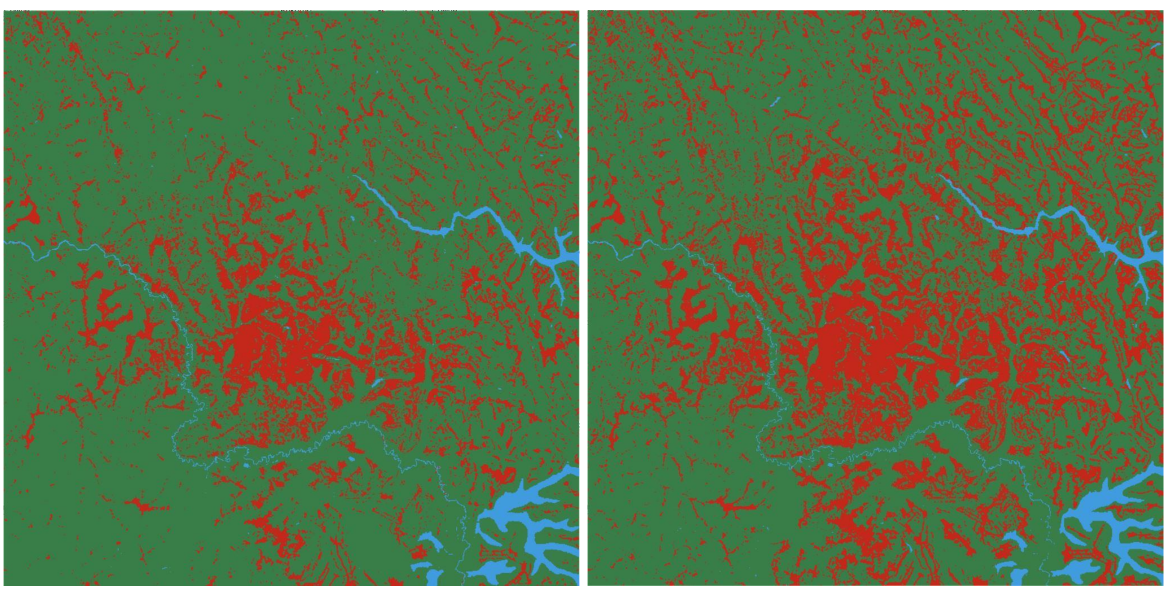


Figure 5: Figure on left is 2016 and on right 2019 Dynamic world images of AOI (Kigali) using 3 classes

The whole AOI region covering 3875 km2 has been split into 256 images covering roughly 15.2 km2 square area each. We have analyzed the class percentage and cumulative sum of pixels for each class.

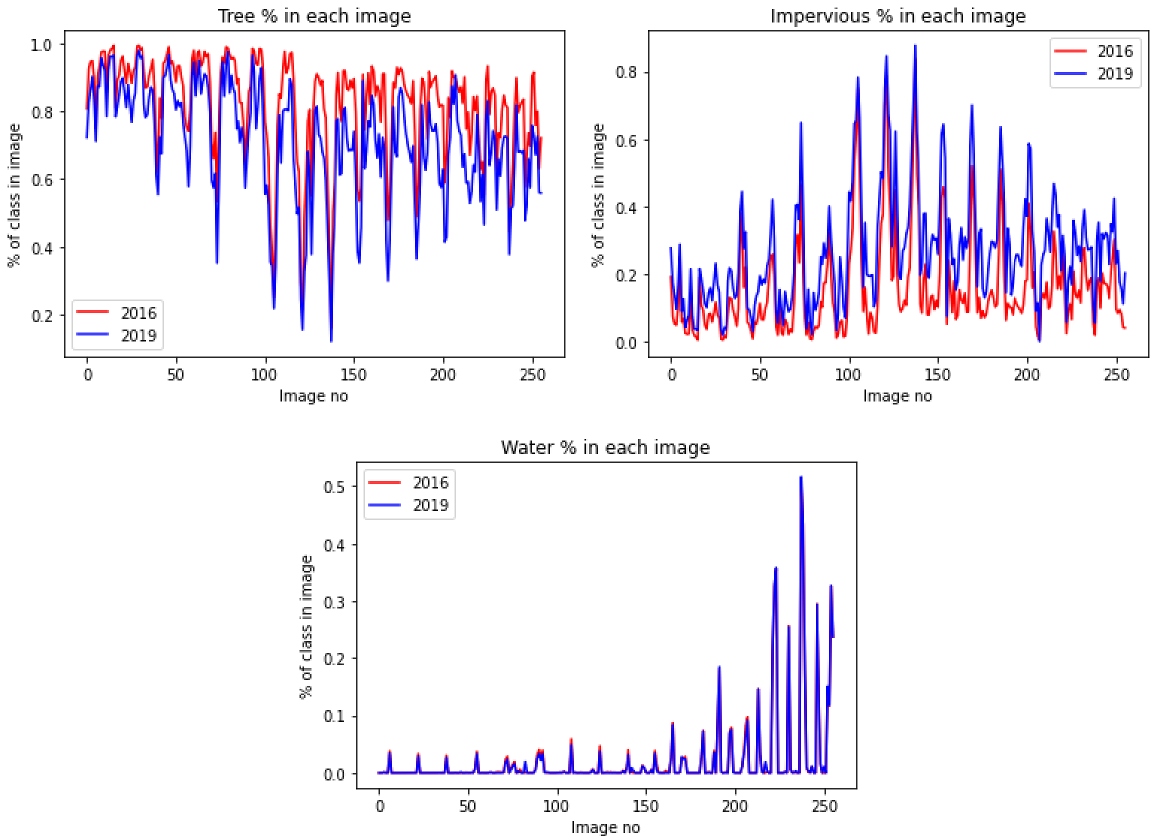


Figure 6: Class percentage per image

From the above images we can see how the percentage of each class has changed in each image. Tree percentage has decreased and the percentage of impervious has increased which is an acceptable result as population grows more forests and grasslands are converted into buildings. Water percentage has remained the same.

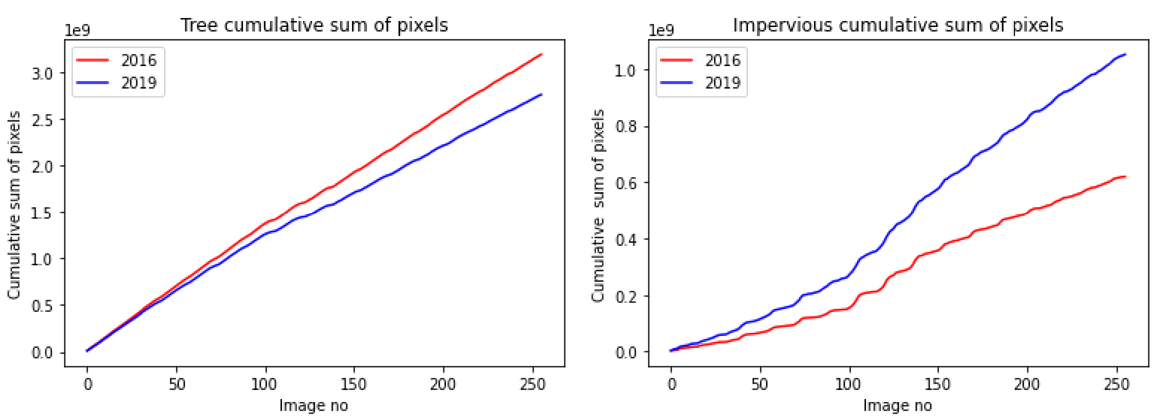


Figure 7: Cumulative sum of pixels/area

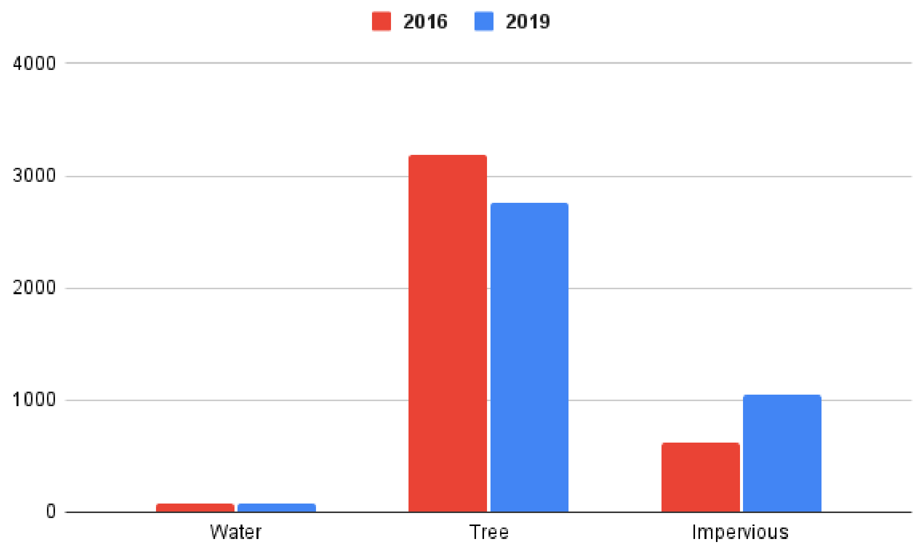
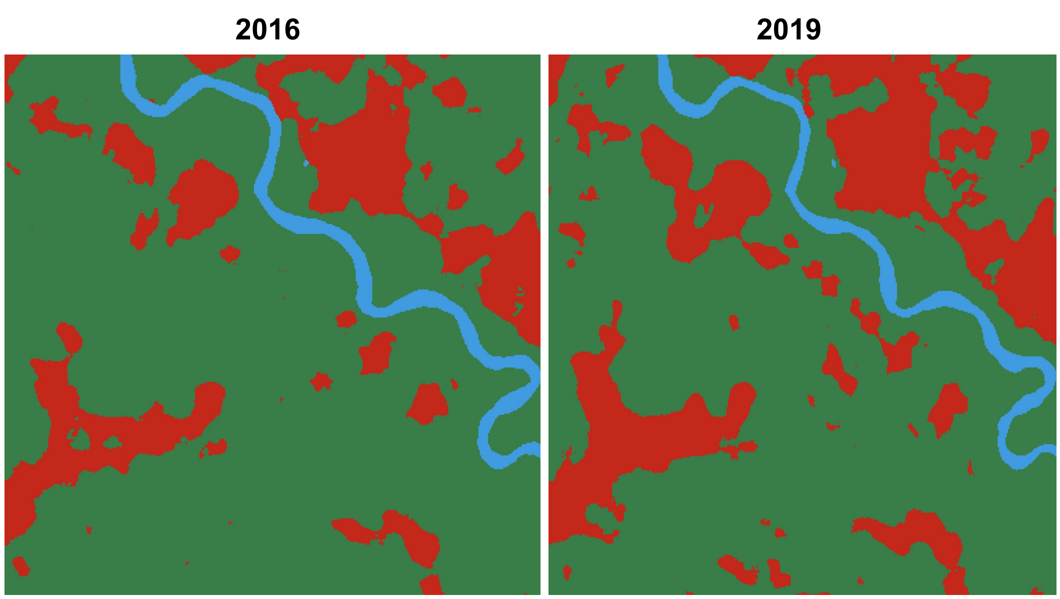


Figure 8: Percentage land cover by each class

From this graph we can see that tree cover of 31.89 km2 in 2016 has reduced to 27.58 km2 in 2019. Similarly impervious land has increased from 6.19 km2 in 2016 to 10.51 km2 in 2019. Note: Each pixel of the image covers 1 m2 area.

Example of Land Cover Loss / Gain:



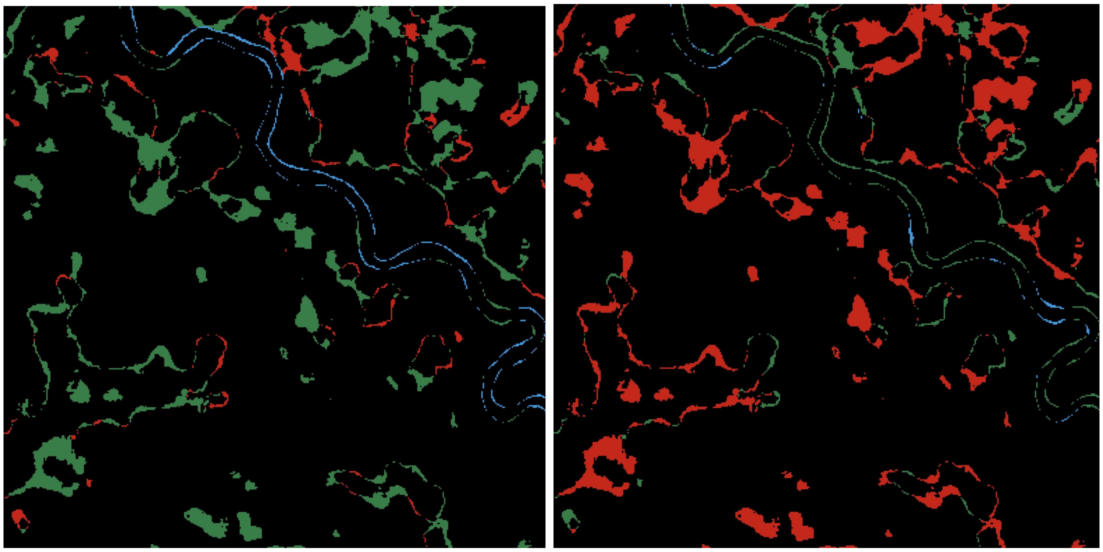


Figure 9: Example of Land Cover Loss/ Gain

Here, the green color in the bottom left image shows that Trees are lost in those pixels whereas Red in the bottom right image shows that corresponding tree loss led to gain in Impervious surface. Black indicates that there is no change in the land cover.

4. Methodology/Model

The task now is to generate a low resolution labeled image from high resolution satellite images. To achieve this, we have used a modified UNet model by changing the encoder with MobileNetV2. The class imbalance in the data as seen in the previous graph is something which makes the learning difficult as the network is easily prone to overfit. Following are the breakdown of steps performed for building this model:

1. **Training/Testing Data:** High resolution images of 2019 (256 images) are used for training the model. We did not include 2016 data since we wanted the model to be robust of the year in which the image was collected. Thus, we used the entire 2016 data (256 images) for validation. (However, looking at the baseline results, we plan to include some more images from 2016 in training dataset in the next iterations)

2. **Preprocessing:** Due to memory limitations and to reduce training time for baseline, we first compressed the images (both train and validation) to 128x128 resolution and then scaled the pixel values between [0,1].

3. **Prediction task:** Multi class classification with three labels (Trees, Impervious, Water)

4. **Model Architecture:** UNet architecture in general has two components, the first is the encoder which takes the input image and generates a lower dimensional feature map.

The second part is a decoder which decompresses this feature map and tries to generate the higher dimensional output image. In the modified UNet model [6], we used a MobileNetV2 architecture for encoder and pix2pix [7] upsample block for decoder. The training has been set to False for the encoder.

5. **Class Imbalance:** Class Imbalance is a major factor in this model since many initial trials with simple CNNs yielded the model predicting Trees for all pixels in all images. Thus, we adjusted the class weights parameter by experimenting with few cases and were able to handle imbalance to certain extent

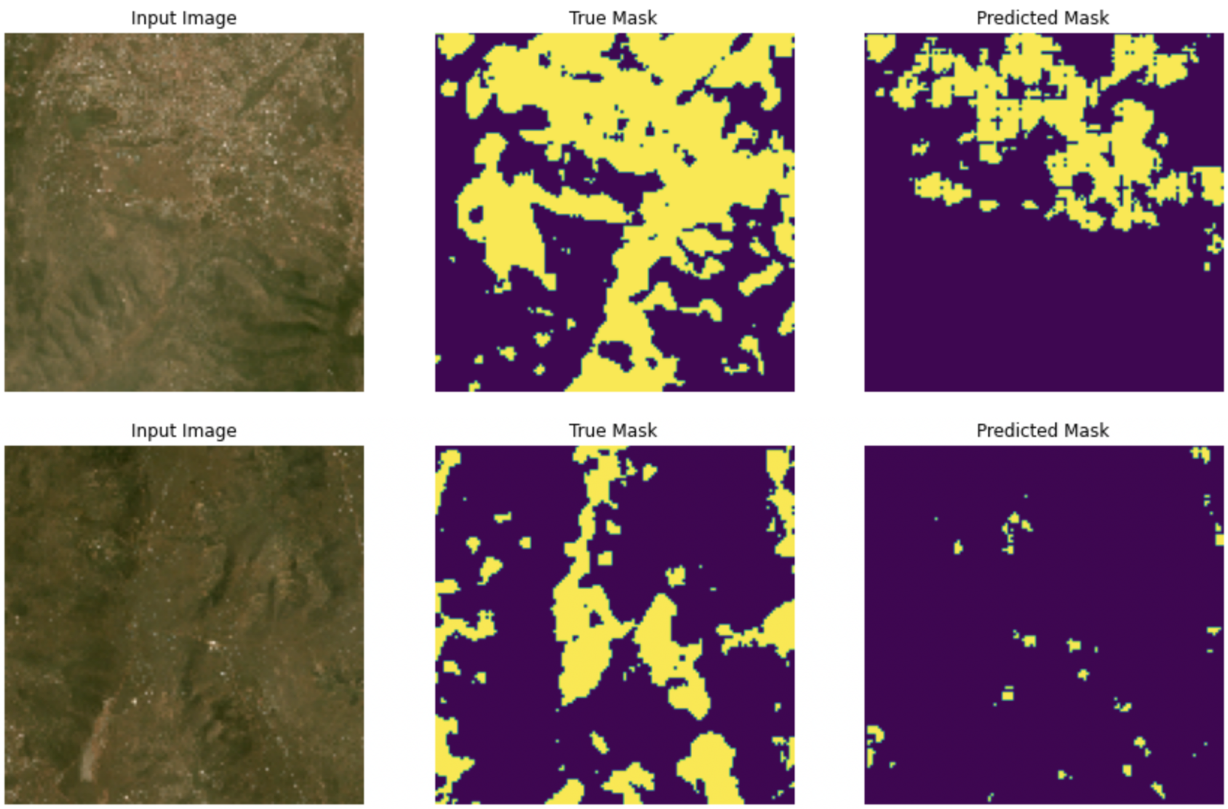


Figure 10: Sample Predictions

*Here, Violet color indicates Trees and Yellow color indicates Impervious.

As we can see, the model is still overfitting in a sense that the model is not able to predict Impervious in most of the cases. (Low recall of Impervious class)

5. Results

Evaluation is a tricky part in this model since we are classifying each pixel in the image. Also, the resolution of the images has been brought down from 4000x4000 to 128x128. However, even if we extrapolate the pixels after prediction, i.e. convert back 128x128 to 4000x4000, the percentage of area covered will not change. Thus, we can approximately rely on the results of 128x128 images as an alternative. Average Precision, Average Recall for all three classes are calculated in the following way:

1. Each predicted image has been converted to a list of 128x128 = 16,384 pixels

2. Now, precision and recall for each class are calculated by assuming we have 16,384 data points. Thus, we get Precision and Recall for each class for one given image

3. The Precision and Recall for each class of all the images are averaged to give Average Precision and Average Recall for each class

4. In addition, average accuracy is also calculated in the same manner

Class	Average Precision	Average Recall
Trees	86.28%	99.17%
Impervious	77.84%	18.24%
Water	33.82%	15.29%
Average Accuracy	87.34%	

Table 2: Accuracy, Precision and Recall for each class

Manual Validation of Ground truth:

In order to make sure that our training labels and predicted outputs are in line with ground truth, we have manually annotated 5 images (we plan on completing 50 images by next report) using a software called Groundwork.

- The primary difficulty that we encountered was due to image quality. Since NICFI images are 5m resolution, we could not accurately distinguish between buildings and barren lands with shrubs. That led to a significant difference in the percentage of classes in Dynamin world labels (training labels) and Ground truth images.

- Second, we could not directly export the output in an image format and had to use some approximation techniques to get extracted images.

Following are the average precision and recall which are calculated as mentioned above:

1. Dynamic World Labels vs Ground Truth:

Class	Average Precision	Average Recall
Trees	80.84%	84.78%
Impervious	27.77%	22.43%
Water	NA	NA
Average Accuracy	72.15%	

Table 3: Dynamic World Labels vs Ground Truth
(*Water was not present in the validated images, thus precision and recall are unreported)

2. Predicted Label vs Ground truth:

Class	Average Precision	Average Recall
Trees	77.36%	99.89%
Impervious	37.71%	0.26%
Water	NA	NA
Average Accuracy	77.31%	

Table 4: Predicted Label vs Ground Truth

Here, the recall of impervious class in Predicted vs Ground truth is due to two factors:

1. Error from manual annotation. If we look at the recall of Impervious class in Dynamic world vs Ground truth, it is 24% which indicates a significant loss of correct Impervious labels in training data itself. We would like to recheck the manual validation to get better labels and check the difference between ground truth and predicted images

2. The current baseline model already has a very low recall rate for impervious class which is 18%. Thus, the model further predicts a lesser number of impervious pixels (compounding of error) and results in much lower recall.

6. Next Steps

As part of next steps, we would like to focus on the following areas to get better results:

1. Increase the training size of the model by including examples from 2016 data as well

2. Data Augmentation of the less representative class to handle imbalanced data

3. Try implementing the UNET without compressing the image to 128x128 by optimizing disk and memory storage

4. If the above does not work, we could try using sliding window of 128x128 over the 4000x4000 image to see if there is lift in model performance

5. Try other UNET architectures leveraging transfer learning techniques to get better model performance

6. Better annotation of ground truth as we try to improve the quality of labels and export the images without using approximation techniques

7. Challenges

- The first challenge faced by us was the dataset collection. We had to create a dataset of NICFI images using Google Earth Engine. We invested a lot of time in creating a dataset of the Kigali region for “2016” and “2019” to showcase the deforestation in this region. We need high resolution images which are not provided by NICFI for improving performance.

- The next challenge was to find an automated tool for data labeling. We tried a lot of tools such as Labelbox, Label Studio, Microsoft Land Cover and QGIS. We could not label the images precisely using any of these tools. Later on, we decided to use Ground Work for data labeling. For Instance, in QGIS, Model wasn't able to generate enough parameters from the zonal_statistics operation for the input image which resulted in the errors faced during the model training.

- Data Labeling using Ground Work was one the biggest challenges throughout the process. We have to manually label 50 images. It is extremely tedious to label as each image takes approximately 3-4 hrs to label a single image

- Another challenge in the process was the high computational cost of the images. We had to resize the images from “256X4000X4000X3” to “256X128X128X3” to avoid our session from being crashed. We are looking forward to finding a way to optimize the usage of disk storage vs memory storage as part of our next steps

8. References

[1] High-resolution land cover change from low-resolution labels: Simple baselines for the 2021 IEEE GRSS Data Fusion Contest https://arxiv.org/pdf/2101.01154.pdf.

[2] Planet Team (2017). Planet Application Program Interface: In Space for Life on Earth. San Francisco, CA. https://api.planet.com

[3] Brown, C.F., Brumby, S.P., Guzder-Williams, B. et al. Dynamic World, Near real-time global 10 m land use land cover mapping. Sci Data 9, 251 (2022). doi:10.1038/s41597-022-01307-4

[4] 2020 IEEE GRSS Data Fusion Contest, 2020. http://www.grss-ieee.org/community/technical-committees/data-fusion/2020-ieee-grss-data-fusion-contest/.

[5] 2021 IEEE GRSS Data Fusion Contest Track MSD, 2021. http://www.grss-ieee.org/community/technical-committees/data-fusion/2021-ieee-grss-data-fusion-contest-track-msd/.

[6] Tensorflow Image segmentation tutorial https://www.tensorflow.org/tutorials/images/segmentation

[7] Image-to-Image Translation with Conditional Adversarial Networks https://arxiv.org/abs/1611.07004

[8] Microsoft Land cover mapping project https://github.com/microsoft/landcover

[9] Google Earth Engine https://earthengine.google.com/