



**UNIVERSIDAD  
DE ANTIOQUIA**

**A machine learning methodology for land use/land  
cover classification in tropical areas using medium  
resolution satellite imagery, case: Colombia**

Autor

Alberto Mario Ceballos Arroyo

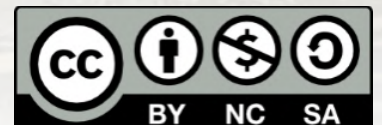
Universidad de Antioquia

Facultad de Ingeniería

Departamento de Ingeniería de Sistemas

Medellín, Colombia

2021



A machine learning methodology for land use/land cover classification in tropical areas  
using medium resolution satellite imagery, case: Colombia

**Alberto Mario Ceballos Arroyo**

Monografía presentada como requisito parcial para optar al título de:  
**Especialista en Analítica y Ciencia de Datos**

Asesor:

**Raul Ramos Pollán, Ph.D.**

Universidad de Antioquia  
Facultad de Ingeniería  
Departamento de Ingeniería de Sistemas  
Medellín, Colombia  
2021.

# A machine learning methodology for land use/land cover classification in tropical areas using medium resolution satellite imagery, case: Colombia

Alberto Mario Ceballos-Arroyo, advised by Prof. Raul Ramos-Pollan

June 11, 2021

## Contents

<b>1</b>	<b>Problem statement</b>	<b>2</b>
1.1	Business problem . . . . .	3
1.2	Machine learning approach . . . . .	4
1.3	Previous work . . . . .	4
<b>2</b>	<b>Proposed methodology</b>	<b>6</b>
2.1	Data sources . . . . .	6
2.2	Google Earth Engine . . . . .	9
2.3	Data pre-processing (S2) . . . . .	10
2.4	Data pre-processing and filtering (S1) . . . . .	11
2.5	Data fusion . . . . .	12
2.6	Data labelling . . . . .	12
2.7	Metrics . . . . .	12
<b>3</b>	<b>Methods</b>	<b>13</b>
3.1	Hardware and software . . . . .	13
3.2	Preparing data for ML algorithms . . . . .	13
3.3	Baseline Random Forest . . . . .	14
3.4	Convolutional Neural Network . . . . .	14
<b>4</b>	<b>Evaluation</b>	<b>15</b>
4.1	Results on the 500 m Amazonas dataset . . . . .	15
4.2	Results on the 30 m Antioquia dataset . . . . .	16
<b>5</b>	<b>Discussion</b>	<b>17</b>
5.1	Conclusions . . . . .	17
5.2	Future work . . . . .	17
<b>6</b>	<b>Acknowledgements</b>	<b>18</b>

# 1 Problem statement

Since the launch of the Landsat (1972) and other remote sensing missions, multi-spectral satellite imagery data from sites across the world has become available not only to government agencies, but also to the general public. Currently, petabytes of information from both free and commercial satellites are uploaded to digital repositories every day. Processing this amount of information is extremely difficult but provides opportunities for many areas, including geo-processing solutions for climate, biodiversity, and agriculture [10].

South Pole has helped avoid the emission of more than 170 million tons of CO<sub>2</sub> and protected and restored almost 55,000 km<sup>2</sup> of forests in tropical areas by carrying out projects in collaboration with industries from sectors as forestry, agriculture, and energy. More specifically, companies like South Pole develop projects which implement conservation and restoration, carbon capture, or emission avoidance activities for entities dedicated to agriculture, forestry, and other land use (AFOLU).

Such projects have had a positive impact both in the communities and the environment, and they have been made possible largely by the use of imagery from missions public missions such as Landsat 7 and Landsat 8 (30 m spatial resolution) and Sentinel 1 and Sentinel 2 (10 m spatial resolution), as well as commercial imagery from missions like WorldView (0.3 to 3.7 m spatial resolution). The reason for South Pole's success is that by implementing land use/land cover (LULC) analysis methodologies (see Figure 1 for an example), it is possible to identify suitable areas for implementing such process and to monitor, report, and verify (MRV) ongoing AFOLU projects.

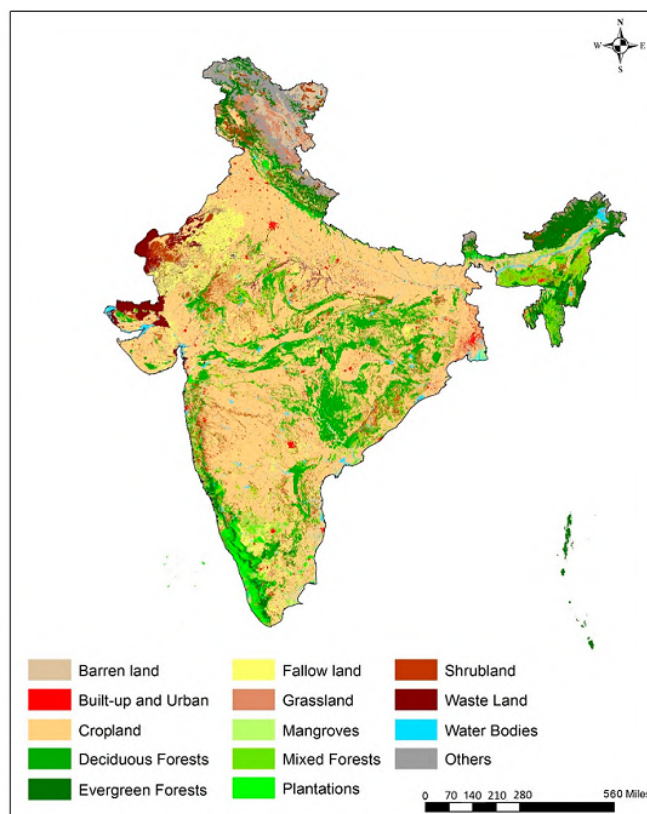


Figure 1: LULC classification map for India: Source: [15].

The typical LULC analysis workflow is semiautomatic and depends on the skills of the technical team and on access to specialized, usually expensive software. Since the personnel tasked with carrying out the non-automatic steps are prone to making either commission or omission errors, inconsistencies often arise in terms of the quality of the final products. Likewise, disparities in experience and proficiency for the tasks associated with such steps result in difficulties

from the project management standpoint, such as an inability to appropriately gauge the times and costs associated with each project.

Motivated by this, environmental and forest engineering experts at South Pole and other organizations (both private and public) have started a transition into the use of end-to-end toolboxes that can be employed for automatically classifying vast swathes of territory in a reliable manner. This will require automatizing the data acquisition pipeline (which is currently based on the JavaScript and browser-based Google Earth Engine IDE) and developing Machine Learning models that better exploit the spatial patterns in higher resolution satellite images, particularly in tropical areas.

However, freely available information on LULC for tropical zones such as Colombia is either low resolution (available maps for Colombia are in 100 m resolution), not comprehensive enough (too few labels), or not sufficiently precise/consistent. Thus, there is insufficient reference data for properly evaluating the performance of newly implemented methods. Due to this, the GeoAI team at South Pole is also required to develop appropriate data preparation methodologies for producing satellite imagery products which can be manually labelled to obtain 10 m spatial resolution ground truth labels.

## 1.1 Business problem

One of the main goals of South Pole when it comes to producing LULC maps is to carry out time series analyses across several months or years and to determine whether environments such as forest and mangroves have been compromised by human activities and other factors. While the time series analysis component is outside the scope of this work, it is a key component of the business problem and sets the framework for discussing all of the issues that motivate this research.

As stated at the start of this section, the current resolution of the imagery employed in LULC projects in tropical areas is a critical aspect. Even ambitious projects such as Mapbiomas, which involved dozens of public and private institutions and resulted in a LULC map for the whole Amazonian forest across several countries, are based on Landsat 7 and Landsat 8 imagery and limited to a spatial resolution of 30 m, whereas other publicly available sources like Sentinel 1 and 2 have a spatial resolution of 10 m.

There are several reasons for this to be the case, which we will list next:

- For every Landsat pixel, there are around 9 Sentinel-2 pixels (for the higher resolution bands). If we also take into account that features are less detailed in Landsat imagery, this means that manual classification consumes roughly ten times more effort.
- A Sentinel-2 image covering the same area as a Landsat image occupies much more disk space and also requires more RAM and processing power in order to be visualized, which forces South Pole to employ more powerful computers.
- Noisy elements such as clouds are easier to handle in Landsat imagery, making it easier to employ in cloudy or very cloudy areas like the Amazon rain forest.
- Most remote sensing experts are more familiar with Landsat imagery, than with the more recent Sentinel-2 products. Thus, the open source tools and workflows developed by the community for Landsat are more mature and accessible.

In addition to the above, most available LULC products at 10 m spatial resolution or better only cover regions such as the United States and Europe, which have less cloud incidence throughout the year, and whose environmental agencies have more resources than the equivalent Colombian institutions, such as IDEAM.

At the commercial level, having higher resolution labels is particularly important since each squared meter of forest that is inaccurately classified by a model can result in around 50 dollars of losses for South Pole. In addition, being able to properly classify other classes is extremely important too, as different measures can be proposed depending on whether forest was lost to agricultural, mining, human expansion, or other kinds of activities.



## 1.2 Machine learning approach

LULC classification can be addressed as well-posed supervised ML problem where  $X$  comprises a set of multi-spectral satellite images of a given size with a given spatial resolution, and  $y$  is a set of labelled images of the same size and spatial resolution where each pixel is assigned a class from a set of classes determined (although not necessarily set in stone) during the data labelling phase. As such, a model  $M$  can be developed and fitted using a loss function  $L$  and an optimization algorithm  $\theta$  in order to minimize the error  $E$  between the predictions output by  $M$ , deemed  $\hat{y}_i$  and each ground truth image  $y_i$ , where  $i$  is an index unique to each pair of satellite and labelled images (see Figure 2 for an example).

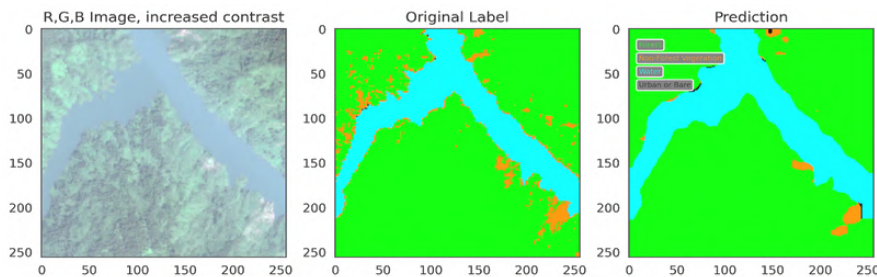


Figure 2: From left to right:  $X$ ,  $y$ , and  $\hat{y}$  as described above. Source: the authors.

Due to the relatively small amount of data at 30 m spatial resolution, we also propose addressing the issue of low data availability by carrying out weakly supervised pre-training using 500 m spatial resolution labels of a bigger region.

There are a few relevant discrepancies in the use of terminology by remote sensing experts and ML experts, which we will also discuss in this section:

- When discussing resolution, the ML community often refers to the size of each image in pixels, whereas the remote sensing community often uses the term resolution to reference the amount of territory represented in each pixel. Thus, a smaller sensor resolution often means that each pixel represents a smaller region, so that the full image is more detailed overall. For purposes of clarity, we will use the term spatial resolution for describing sensor/pixel resolution, and the term resolution for describing image size.
- In the ML community, supervised image classification consists in outputting a single label  $y_i$  for a given image  $X_i$ . However, in order to be useful for the purposes of South Pole, LULC methodologies must produce an independent label  $\hat{y}_{i,j,k}$  for every pixel  $X_{i,j,k}$ , where  $j$  refers to the row and  $k$  refers to the column where the pixel is located within the image. Nonetheless, the remote sensing community uses the term classification for this kind of problem, whereas it is in fact a case of semantic segmentation. In this work, we will use the term LULC classification (and not classification by itself) when referencing the learning problem at hand.
- Remote sensing experts often employ different names for metrics derived from the confusion matrix (although not for the confusion matrix itself). Since this work constitutes a first approximation, we will use the terminology from the ML community.

By considering the above and the points discussed in Subsection 1.1, we intend to develop ML models that can classify medium spatial resolution imagery with enough precision to improve South Pole's processes, and at the same time to communicate our results in an adequate manner so the tools can be put into use within the organization. The proposed business workflow can be seen in Figure 3.

## 1.3 Previous work

Many authors have addressed LULC classification using different approaches:

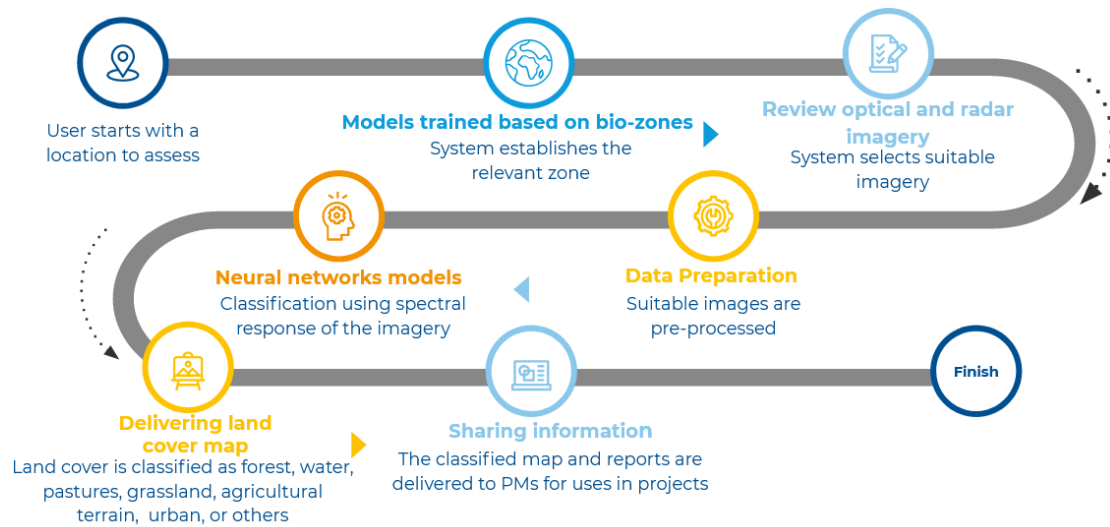


Figure 3: The objective at South Pole is to produce a workflow such as this, in which users only have to specify a location and provide the input data and minimal additional input is required. This work will cover steps 1 to 6. Source: South Pole.

- **Business knowledge-based:** these approaches comprise the use of remote sensing knowledge regarding the spectral response of different bands to assign classes to terrain [12, 14]. Often, this implies developing a decision tree algorithm where the thresholds are defined based on statistical insight and the experience of the researchers. This approach can lead to many pixels being classified incorrectly, so it is usually required to implement label post-processing and to carry out manual validation.
- **Machine learning-based:** most remote sensing developers employ traditional machine learning models such as support vector machines (SVMs) and random forest (RF) algorithms to classify relatively big regions based on small, manually-labelled regions (called 'seeds') [9, 11, 13, 3, 1]. Such models are often fed with the values of each spectral band, although some authors have carried out neighborhood-wise feature extraction to capture more spatial information. Usually, only a few tens of thousands labelled pixels are used for training, so these methods can be trained quickly. However, the relatively limited amount of training data, coupled with the difficulties associated with capturing spatial patterns, often cause these methods to require more post-processing and validation [18].
- **Deep learning-based:** in the last few years, more researchers and developers have employed deep convolutional networks for carrying out LULC classification [5, 19]. These networks have the advantage of being compatible with huge volumes of data, thanks to GPU acceleration. In addition, by exploiting the properties of convolutions, these models are able to capture spatial patterns in data without requiring any feature extraction procedures [16]. Conversely, they require huge amounts of labelled data, which is not always available. However, results from recent ML competitions using remote sensing imagery show that it is possible to produce high-resolution labels from low-resolution labelled data, which is easier to procure.

Within South Pole, previous projects have been carried out based on a combination of approaches 1 and 2. In particular, these approaches were employed for the generation of the labelled data we will use in this study.

## 2 Proposed methodology

The proposed methodology seeks to address several of the problems at South Pole and is not limited to the supervised classification. Our intention is to enable the organization to carry out a mostly automatized LULC analysis workflow with minimal human intervention. Most of the steps were developed as IPython Jupyter Notebooks, with functionalities separated into independent Python modules. Data acquisition and pre-processing was carried out with the Google Earth Engine platform through its Python API, while the rest of the processing was done using Python and standalone libraries (numpy, rioarray, gdal, pytorch) locally. The more computationally intensive steps were carried out in GPU-enabled Azure virtual machines, and the data was stored in an Azure Blob Storage account.

An overview of the proposed methodology, including future steps, is depicted in Figure 4.

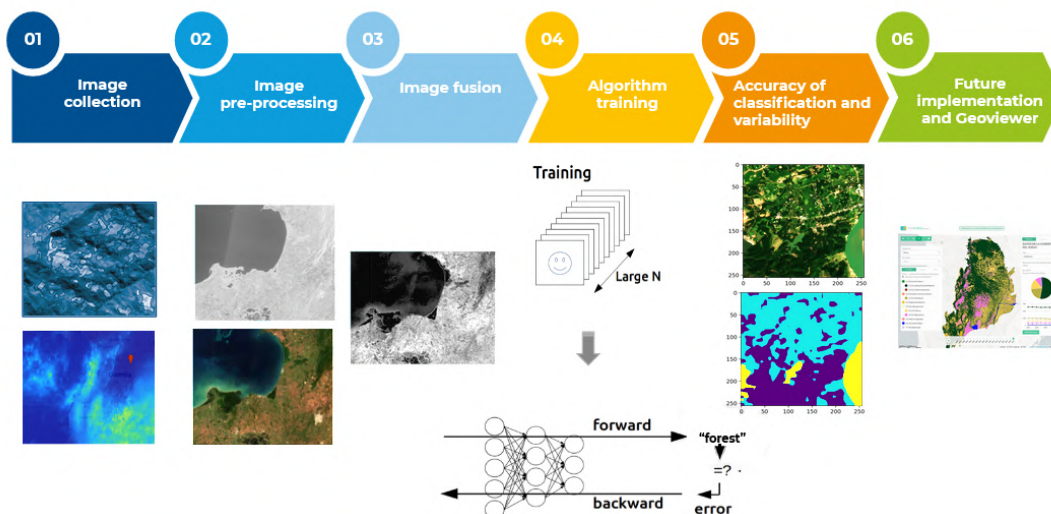


Figure 4: The steps comprising the ML methodology described in this work (01 to 05), as well as the proposed future implementation of a geoviewer tool. Source: South Pole.

### 2.1 Data sources

The data employed for this project comprises imagery from several sources. For the images that comprise our  $X$  set, we employ satellite data that comes from the Sentinel 1 radar satellite and the Sentinel 2 multi-spectral satellite, as well as elevation data coming from the ALOS digital elevation map (DEM). For the labels in our  $y$  set we employ a two datasets: first, a 500 m resolution LULC map developed internally by South Pole and which covers the whole Colombian Amazonian region; second, a smaller, 30 m spatial resolution dataset which covers part of the Antioquia department north of Medellin.

#### 2.1.1 Sentinel 2 data

The Sentinel-2 mission consists of two multi-spectral satellites, Sentinel 2A and Sentinel 2B, launched by the European Space Agency (ESA) in 2015 and 2017. Both satellites have fixed orbits and acquire pictures of segments of the Earth's surface called granules or tiles. Each tile covers around  $10 \text{ km}^2$  and is identified by a unique alphanumeric code (*e.g.*, 18NZF, see Figure 5). The median revisit time of each satellite for a given tile is 10 days, which is cut down to 5 if we consider both satellites. This means that, for most areas in the world, there are 74 Sentinel-2 images available per year (see Figure 6 for a detailed overview).

As mentioned above, Sentinel-2 satellites are multi-spectral, which means that several sensors with different wavelengths capture an image (band) whenever a tile is imaged. In total, Sentinel-2 products are comprised by 13 bands, which range from 10 m to 60 m spatial resolution and



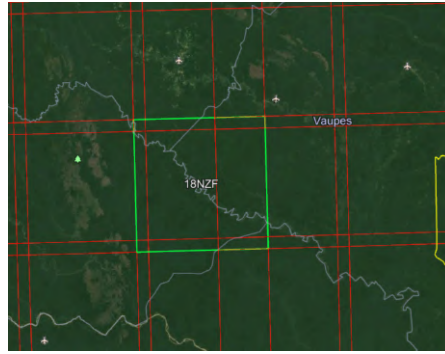


Figure 5: Several Sentinel-2 tiles overlaid on the Colombian Amazonian region. The green lines represent the outline of tile 18NZF, red lines represent the outlines of other tiles. Notice that some couples of tiles overlap to a higher degree than others, resulting in twice the revisit frequency for the overlapping segments. Source: Google Earth and ESA.

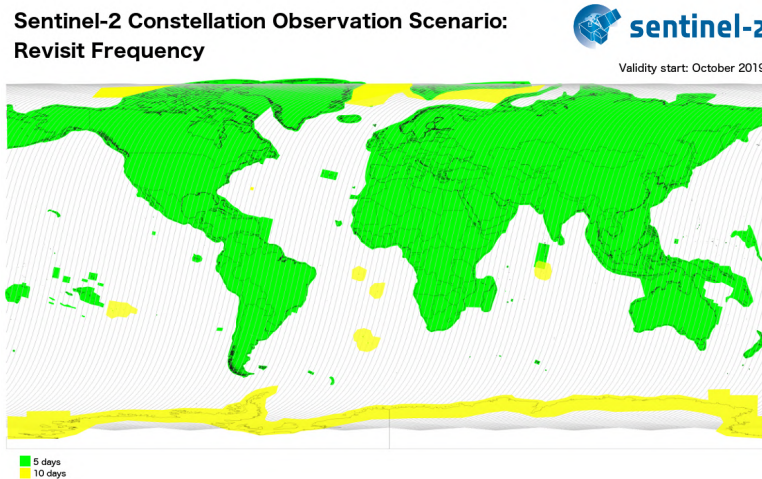


Figure 6: Combined revisit times for the Sentinel-2 satellites across the world. Source: ESA.

are useful for various tasks. Furthermore, the product includes a rudimentary cloud detection mask at 60 m resolution. However, the commission error in areas such as the Amazonian rain forest is such that most authors do not recommend using it [4]. Instead, we will use the more detailed s2cloudless cloud mask product which is uploaded by ESA.

### 2.1.2 Sentinel-1 data

The Sentinel-1 mission consists of two C-band synthetic aperture radar (SAR) satellites which operate in four imaging modes with various spatial resolutions. SAR has the advantage of acquiring images at wavelengths which are not impacted by cloud cover or illumination. Depending on the configuration of the satellite, images are acquired in single-swath or multi-swath modes, which introduces additional considerations when consuming the S1 products. In addition, exploiting these raw products can be difficult. For this reason, we employ the Sentinel-1 Ground Range Detected (GRD) product, which has been pre-processed by ESA with its Sentinel-1 Toolbox by applying the following steps [17]:

- Apply Orbit file to update orbit metadata
- Remove additive thermal noise from images acquired in multi-swath mode
- Remove GRD scene border noise

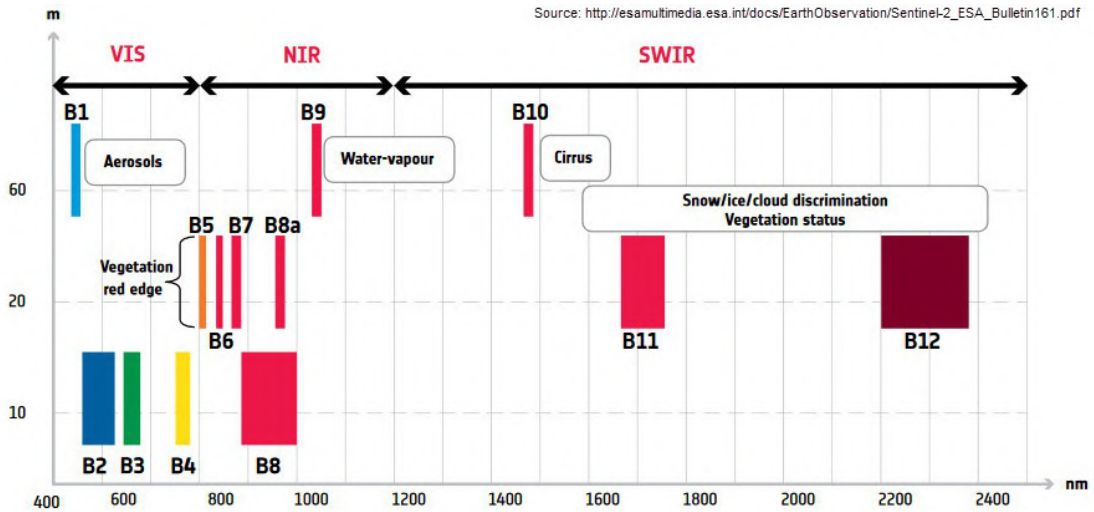


Figure 7: An overview of the bands in a Sentinel-2 product. Source: [8].

- Apply radiometric calibration to  $\sigma^0$
- Apply Range-Doppler terrain correction

The GRD product uploaded to GEE as the geocoded back scatter bands calibrated to the normalized radar cross section  $\sigma^0$  in dB units:  $10 * \log_{10}(\sigma^0)$ . This is done due to the fact back scatter values can differ by orders of magnitude within an image.

### 2.1.3 ALOS

The ALOS World 3D product is a global digital surface model (DSM) dataset with a spatial resolution of around 30 m. It was developed by the Japan Aerospace Exploration Agency (JAXA) based on the World 3D Topographic Dataset and allows for accurately measuring the elevation of specific locations across the world, which can be helpful in identifying areas like hills and rivers.

Some artifacts (impossibly low elevation values) exist in ALOS for areas of the world such as Antarctica, Brazil, and Indonesia, among others. We carefully reviewed the list of locations and concluded that there was no overlap with our area of study.

### 2.1.4 Colombian Amazonian region LULC map

As part of its projects for environmental vigilance in Colombia, South Pole produced a 500 m spatial resolution map covering the whole Amazonian region of Colombia. This map was developed by environmental and forest engineering experts based on Landsat 7 and 8 imagery from the year 2016, which then validated the results on critical areas using high-resolution Worldview imagery.

The original labelled dataset comprised more than 40 classes. However, some of these either had too few instances (less than 1000 pixels) or were not within the scope of South Pole's medium term objectives. Thus, we decided to combine many of the original classes into the following super classes:

- Forest
- Agricultural terrain
- Grassland
- Built-up terrain

- Water
- Others

In order to reduce the evident majority of forest pixels, we selected 6 of the tiles with the most LULC class variety: 18NUF, 18NWG, 18NYH, 18NYF, 18NCA and 19NED. The tiles, as well as most of the original 500 m spatial resolution label dataset, are shown in Figure 8.

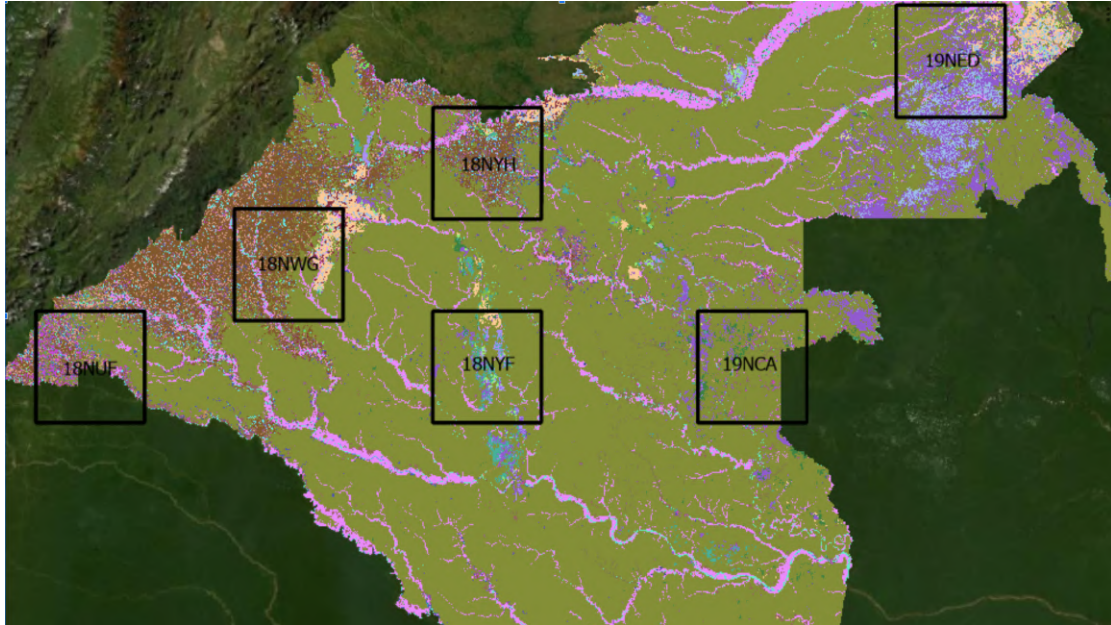


Figure 8: Part of the Colombian Amazonian region 500 m resolution LULC map developed by South Pole. Source: Ana Maria Zapata, South Pole.

### 2.1.5 North Antioquia region LULC map

South Pole also developed a smaller scale map centered on the area north of Medellín, in the Antioquia department. This smaller map As part of its projects for environmental vigilance in Colombia, South Pole produced a 500 m spatial resolution map covering the whole Amazonian region of Colombia. This map was developed by environmental and forest engineering experts based on Landsat 7 and 8 imagery from the year 2016, which then validated the results on critical areas using high-resolution Worldview imagery.

### 2.1.6 On combining Sentinel-2 imagery with 500 m and 30 m spatial resolution labels

Our motivation to use 500 m and 30 m spatial resolution labels is two-fold: first, the software solutions implemented at South Pole for labelling data are still being tested and thus the company is unable to produce enough data to cover more than a few small test territories; second, some research projects and remote sensing ML competitions have shown that certain models are capable of generalizing to higher spatial resolution imagery when trained on enough lower resolution imagery.

Since 30 m spatial resolution labels are fewer, we will use the 500 m labels as a pre-training set and then fine-tune on the 30 m dataset.

## 2.2 Google Earth Engine

The complete data pre-processing pipeline was implemented on Google Earth Engine (GEE) using the Python API. Thus, we were able to exploit the high-performance computing capabilities

of GEE and the flexibility of Python as a programming language.

### 2.2.1 Database structuring

We developed a small .csv file-based database in which we stored general information from the wider Amazon area and information belonging to each tile. The database is based on two tables: the general region (country, area, labelled data location, classes, associated tiles) and the specific tiles (country, area, tile, year, coordinates, and associated orbits). This allowed us to quickly set up experiments by considering factors specific to each location, based on a simple inner join operation on the country and area fields.

### 2.2.2 Data acquisition metadata

For every data download job submitted to GEE, all parameters associated with the acquisition are first set up in a Python dictionary which is stored along the images after they are downloaded. This allows for different parameter combinations to be tested and for the parameters employed for every tile that is downloaded to be reproducible.

## 2.3 Data pre-processing (S2)

There are several objectives to be accomplished when pre-processing the Sentinel-2 data:

- Removing images that are cloudy to the point of being unusable or outside the relevant period.
- Translating the original pixel intensity values, which are top-of-atmosphere (TOA) values, into surface reflectance (SR), topographically-corrected values.
- Flagging cloud-covered or haze-corrupted pixels so that they are not included in further analyses.

After carrying out the above steps, we are capable of producing a mosaic. Mosaics are combinations of several images of the same tile at different dates in which flagged values are ignored. Since we have several images per tile, we can fill the holes left by clouds and still have a coherent representation of the region by using robust aggregation methods such as the median or another quantile.

### 2.3.1 Filtering

We considered two criteria for selecting the images for each tile:

- Cloud cover percentage as per the default cloud segmentation 30%: images that are too cloudy contain very little usable data, so we discard them altogether.
- Images belonging to years 2018-2020: only data from the first Sentinel-2A satellite is available for years 2015 and 2016. Furthermore, the revisit frequency only reached its current state well into 2018, so there are not enough images with which to produce a cloudless mosaic for most tropical regions. We consider that any changes in the land cover between 2016 and 2020 won't have a considerable impact due to the comparatively low resolution of the labels, so we employ the best available mosaic from years 2018 to 2020.
- Orbit: Sentinel-2 tiles are often assigned two orbits which cover partially overlapping sections of the tiles. If images captured in different satellite orbits are combined into a mosaic to fill holes left by clouds, the overlapping area will end up filled with artifacts due to the variations in lighting, so we decided to process orbits independently.

In most cases, the filtering above results in between 10 and 20 images per orbit from the original 74 across the full year, which is a sufficient amount for producing an acceptable cloudless mosaic (few artifacts and few no-data areas).

### 2.3.2 Cloud and shadow detection (S2)

As stated before, Sentinel-2 images come with a poor default cloud mask at a poor resolution. Since 2018, ESA started uploading re-processed, 10 m spatial resolution cloud probability masks (0-100) for most Sentinel-2 scenes using its s2cloudless tool. These masks are a separate dataset, so in order to combine them with our base Sentinel-2 TOA data we perform a join operation with the image date as the key.

Careful inspection of the s2cloudless masks shows that, while they are much more detailed, these masks are not perfect: the cloud threshold that defines what is and what is not a cloud must be carefully selected in a trade-off between false positives (coastlines and some rooftops) and false negatives (haze and small clouds not being detected). We decided to employ a threshold of 15, which correctly detects most clouds but results in a few commission errors or false positives.

In order to remove false positives in cloud detection, we employ the cloud displacement index [7] (CDI), which exploits the parallax effect in the bands (there is a slight delay in time between each sensor acquiring its respective image) to discriminate objects in the surface (like rooftops) from objects which are far above the surface (such as clouds). Once these false positives are detected, they are promptly removed from the cloud mask.

Next, we apply a morphological erosion operation on the cloud mask with a 3 pixel diameter circular structuring element in order to remove very small objects that are not likely to be clouds. After, we carry out a morphological dilation operation with a 15 pixel diameter in order to increase the size of the clouds and more appropriately flag hazy areas.

After producing the final cloud mask, we detect shadows by projecting clouds onto the surface at several candidate heights based on the position of the sun and the Sentinel-2 satellite at the time of acquisition [12]. We filter out spurious shadows by applying a normalized darkness index, which often misidentifies water areas as shadows. We correct the shadow mask by applying a normalized water index on the shadow mask. The final shadow mask is corrected by applying the same morphological operations as in the cloud detection step.

Finally, the two masks are combined into a single cloud-shadow mask. All pixels considered to belong either to a shadow or a cloud as per the above steps are masked out and replaced with a no-data value (akin to a None in Python).

### 2.3.3 Atmospheric correction

In order to carry out atmospheric correction and obtain SR values, we employ the Py6S AC model as per [12]. Essentially, Py6S functions as a lookup table for obtaining the equivalent SR value for a given TOA value as per a radiative transfer model.

### 2.3.4 Topographic correction

When images are acquired in areas with variations in elevation, the angle of the acquisition can cause variations in reflectance for similar sites in different positions. For this purpose, we apply topographic correction as per [12], resulting in the elimination of most shadows projected in slopes and similar surfaces.

### 2.3.5 Compositing (S2)

After masking clouds and shadows and correcting the image, we produce a mosaic for a given year by taking the 15th (Amazonas) or the 40th (Antioquia) percentile for every pixel in all 10-20 images. Whenever a pixel is masked out at every image, we flag it as a no-data pixel to remove it from further steps in the pipeline.

## 2.4 Data pre-processing and filtering (S1)

Sentinel-1 products consist of 4 polarization bands. However, these are not always available due to differences in acquisition modes. In order to standardize our data acquisition process, we select only the VV and VH dual polarization bands (well suited for vegetation analysis [6], both at a resolution of 10 m. Furthermore, in addition to the pre-processing applied by GEE, we



applied a speckle filter with a kernel radius of 5 pixels in order to remove any remaining noise from the images.

### 2.4.1 Compositing (S1)

We take the mean of the Sentinel-1 bands across the same time range as the Sentinel-2 images. Since Sentinel-1 tiles are not the same as Sentinel-2 tiles, this means that images from several S1 orbits are combined into one.

## 2.5 Data fusion

After obtaining all pre-processed images, we select the B4, B3, B2, B5, B6, B7, B8, B8A, B11, and B12 bands from Sentinel-2, the VV and VH bands from Sentinel-1, and the elevation from the ALOS DSM. We align them and reproject them to EPSG:4326 at a scale of 10 m by re-sampling the lower spatial resolution bands (nearest neighbor method). The resulting pixels are roughly 9.85-9.95 meters in width and height, which is an aspect that should be corrected in order to put the models in production. Figure 9 shows an example of the extracted RGB bands from S2 and one of the S1 radar bands after reprojection and alignment.

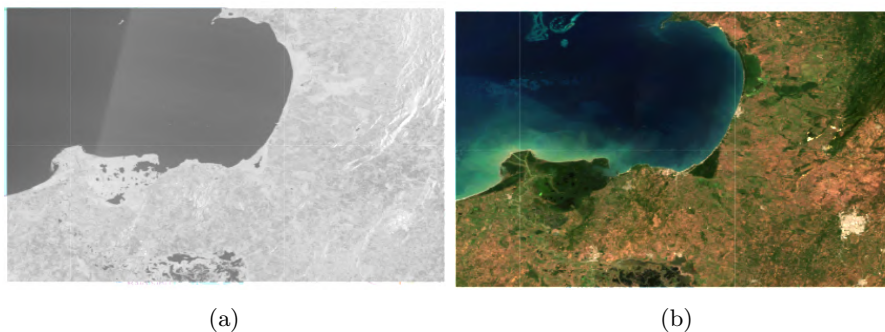


Figure 9: a) Sentinel-1 composite, VV band. b) Sentinel-2 composite, B4, B3, and B2 bands (RGB). Source: the authors.

## 2.6 Data labelling

All of the images downloaded were also sent to the labelling team at South Pole. Currently, they are implementing a SVM and seed labelling-based approach coupled with manual validation to generate 10 m spatial resolution labels based on the provided imagery. Once the labelling is finished, we hope that we can carry out the fine-tuning step on 10 m instead of 30 m spatial resolution data.

## 2.7 Metrics

We tested two metrics for this project: first, raw accuracy as a very general measure of performance; second, balanced accuracy as a way to assess the overall accuracy of the method by weighing all classes equally, which is necessary given the imbalanced nature of the dataset.

Accuracy is calculated based on the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). Balanced accuracy is calculated based on the Sensitivity and Specificity metrics, which are also described below.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$Sensitivity = \frac{TP}{TP + FN} \quad (2)$$

$$Specificity = \frac{TN}{FP + TN} \quad (3)$$

$$BalancedAccuracy = \frac{Sensitivity + Specificity}{2} \quad (4)$$

## 3 Methods

### 3.1 Hardware and software

We pre-processed all of the data and trained all models on an Azure Standard NC6s\_v3 virtual machine with 112 GB of RAM, 6 CPU cores, and a Volta V100 16 GB GPU.

Satellite and labelled imagery manipulation was done using the rioxarray library, version 0.4.0. For the Random Forest algorithm, we used the scikit-learn library, version 0.24.2. The convolutional neural network implementation was done with the pytorch library, version 1.8.1.

### 3.2 Preparing data for ML algorithms

Our data preparation pipeline consisted of several steps, which are described next:

#### 3.2.1 Alignment

At this step, we align each satellite image with the labelled dataset and store the aligned images separately. See Figure 10 for an example.

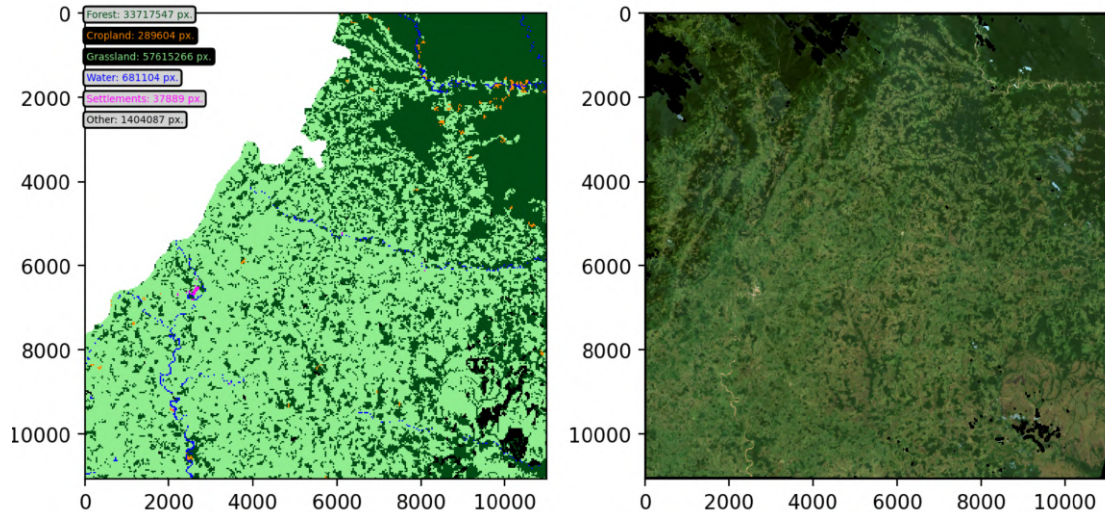


Figure 10: Aligned satellite imagery and labels for tile 18NWH, in the Amazonas region. Source: the authors.

#### 3.2.2 Slicing

At their original resolution, it is unfeasible to process the images with any model. For this reason, we take each image and slice it into 256x256 fragments with a step of 224, allowing for some overlap between adjacent slices. Importantly, we removed all slices that contain no-data values

### 3.2.3 Data augmentation

At the moment of slicing, we apply 7 data augmentation operations: two flips, three rotations, and two combined flip and rotation operations. This way, we noticeably increase the size of our datasets.

### 3.2.4 Resulting dataset sizes and class distributions

For the selected tiles in the Amazonas region, we obtained roughly 100528 256x256 images with their respective labels after data augmentation. For the Antioquia region, we obtained around 5072 samples. While doing the slicing, we also calculated the relative histogram of the classes for both regions, which can be seen in Figure 11.

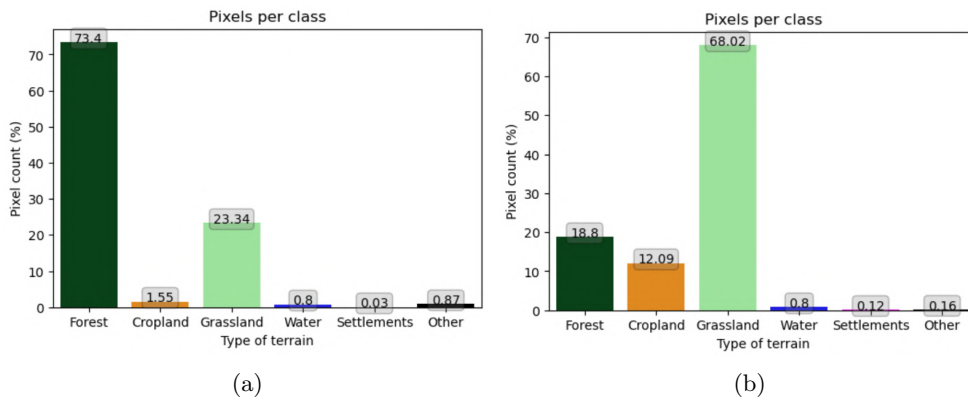


Figure 11: a) Relative class frequency for the dataset in Amazonas. b) Relative class frequency for the dataset in Antioquia. Source: the authors.

### 3.2.5 Standard scaling

During slicing, we also register and progressively update the mean and the standard deviation of all channels in the satellite imagery. These means and standard deviations are necessary for properly fitting CNN models. Ideally, the calculation of means and standard deviations should be done only with the training data as to not use leaked information during testing. However, we found that doing this on the fly for the training and test partitions (which are selected randomly) consumed too much time.

## 3.3 Baseline Random Forest

We employed a Random Forest method as a baseline for our comparisons. Random Forest models do not require any particular pre-processing, although it was required to reshape data into a tabular format in order to process it with the method.

### 3.3.1 Training

Since the Random Forest algorithm can not be retrained on new data, we fitted it on a set of 100 random samples (equivalent to 6553600 individual pixels) from the Amazonas dataset and tested it on the Antioquia dataset. We performed no feature extraction; instead, we employed the 13 bands from the satellite images as individual features. We used 100 estimators with a maximum depth of 10 to avoid overfitting.

## 3.4 Convolutional Neural Network

We employed the DeepLabV3+ [2] CNN architecture with a ResNet50 backbone as our CNN method (see Figure 12). The original architecture was designed to work with 3 input channels, so

we switched out the first convolutional module for a 13 channels one. The model was optimized with the Adabelief optimizer [20] (eps = 1e-16, with rectification and weight decoupling), using the weighted cross entropy loss (weights defined as per Figure 11). In order to achieve extra performance, we used mixed FP-16 and F-32 calculations using PyTorch’s automatic mixed precision module.

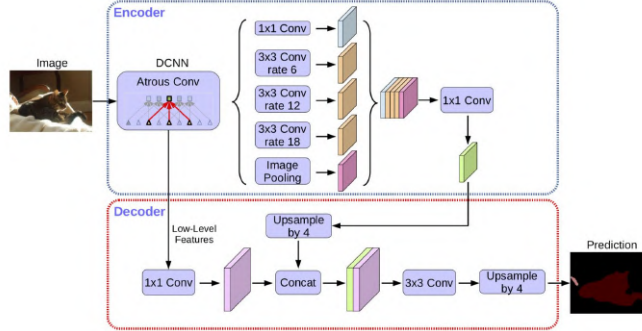


Figure 12: The DeepLabV3+ architecture is based on atrous convolutions, which help capture multi-scale context. Source: [2].

## 4 Evaluation

### 4.1 Results on the 500 m Amazonas dataset

Figure 16 depicts the training and test accuracy and loss curves for the CNN trained on the Amazonas dataset for 11 epochs. As expected, the model fits very slowly due to the difference in spatial resolution between imagery and labels.

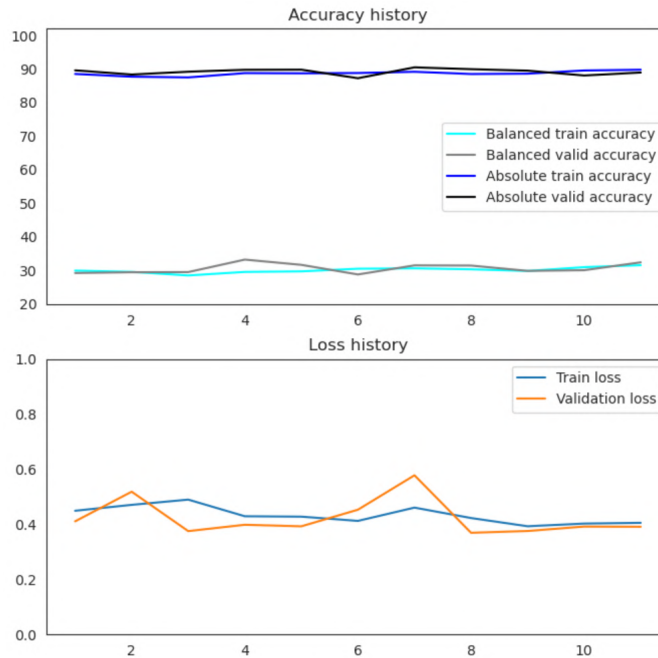


Figure 13: Training and test curves for the CNN on the Amazonas dataset. Source: the authors.

As depicted in Table 1, the DeepLabV3 model obtained better results overall. Note that the metrics were calculated by comparing the low spatial resolution labels with the masks output by the models, which are trained on much higher resolution satellite imagery.

Model	Dataset	Accuracy	Balanced accuracy
Random Forest	Amazonas	0.7458	0.1996
DeepLabV3+	Amazonas	<b>0.8880</b>	<b>0.3324</b>
DeepLabV3+	Antioquia+Amazonas pre-train	<b>0.9105</b>	<b>0.4701</b>

Table 1: Summarized validation results by training and then testing on the Amazonas and Antioquia+Amazonas pre-train dataset configurations.

When training and testing on the Antioquia dataset for 23 epochs, we see that the training procedure goes on much more quickly (see Figure 14). The pre-training also helps obtain a reasonable accuracy even in the first few epochs. We believe that this attests to the potential of this approach to generalize from lower to higher spatial resolutions.

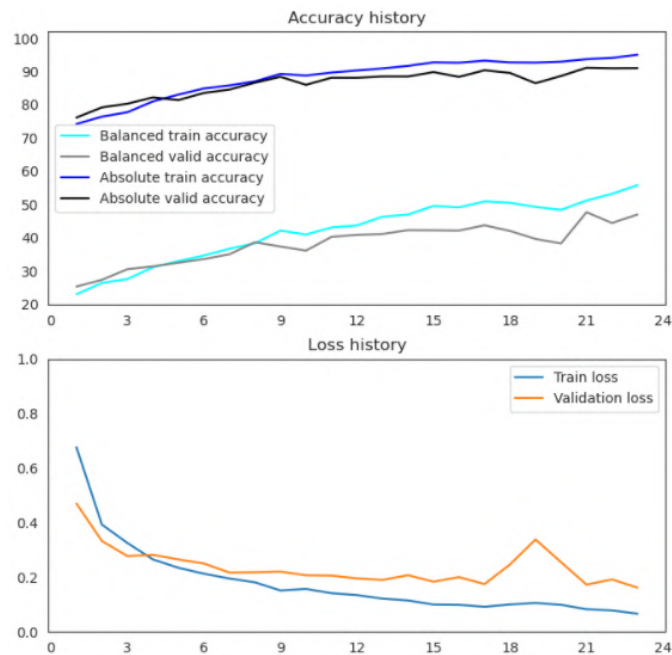


Figure 14: Training and test curves for the CNN on the Antioquia dataset, pre-trained on the Amazonas dataset. Source: the authors.

#### 4.1.1 Qualitative assessment

In general, despite the extremely low resolution of the labels, attempting to predict on the 10 m resolution imagery with the network results in more or less convincing labels which roughly follow the general distribution of the visual features in the images. However, the random forest model often produced very uniform labels with most terrain classified either as grassland or forest. See Figure 15 for an example.

## 4.2 Results on the 30 m Antioquia dataset

By fine tuning the CNN model on the Antioquia data, we were able to obtain better classification of several objects. While still not as precise (the Antioquia labels themselves are not as precise), we can see how the model detects some objects that were not in the labels (the settlement) are detected by the network.



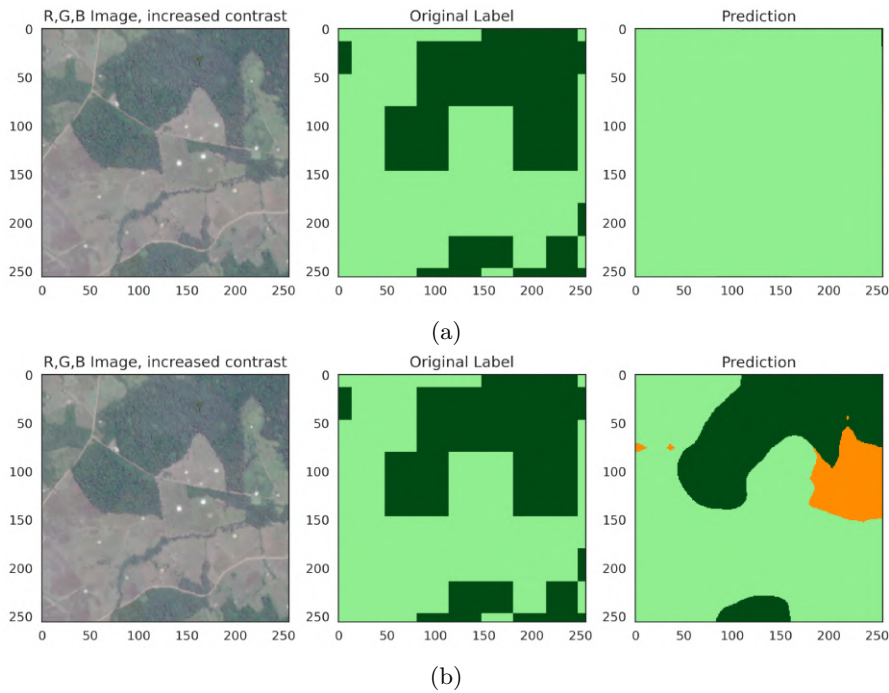


Figure 15: a) Prediction of an image in Amazonas by the RF model. b) Prediction of the same image by the CNN model. Source: the authors.

## 5 Discussion

### 5.1 Conclusions

As part of this work, we developed a methodology for acquiring cloud-free mosaics from tropical areas. This will be an invaluable resource for South Pole, since developments in tropical areas are sometimes impeded by the prevalence of clouds in satellite imagery acquisitions. Our approach to storing regions as a database and acquisition parameters as associated metadata will also help make the whole process traceable.

We also showed that CNN models outperform more classical remote sensing approaches when trying to predict 10 m spatial resolution labels from very low spatial resolution labels. However, the labels output by the CNN model using such a data configuration were only approximately similar to the outline of the areas in each image.

For this reason, we proposed using the augmented 500 m spatial resolution data (more than 100000 images) as a pre-training step for using the same network on a 30 m spatial resolution dataset. By doing this, we obtained reasonable results with only a few epochs of training, which proves the potential of weakly supervised learning for this kind of problem.

### 5.2 Future work

Future work will comprise four directions:

- Adjust the parameters of the pre-processing pipeline for each tile in order to reduce the amount of artifacts.
- Develop an intuitive, web-based tool for labelling satellite images based on the data produced by the pre-processing pipeline.
- Artificially improving the low spatial resolution labels by unsupervised methods such as clustering based on the higher spatial resolution labels. This has been the focus of the IEEE Global Land Cover Mapping With Weak Supervision competitions, where 1 m satellite imagery is used together with 30 m labels.

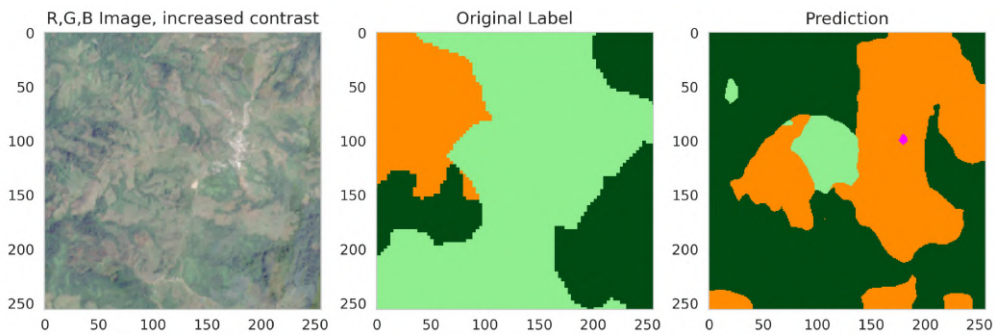


Figure 16: Example of an image classified using the model fine-tuned on Antioquia. Source: the authors.

- Testing our proposed approach on a small 10 m resolution subset of the Amazonas labels, since differences between the two datasets could be having an impact on fine-tuning.

By following these steps, we expect that South Pole will be capable of producing 10 m spatial resolution LULC products in the near future.

## 6 Acknowledgements

This work was supported by internal funds of South Pole starting on July 2020 and, since April 2021, by Microsoft’s AI 4 Earth program, which allocated \$15,000 USD in Azure credits for this project.

I want to thank current and previous members of the GeoAI team at South Pole, such as Eliana Molina, Ana Maria Zapata, Alejandro Gomez, David Velasquez, and Juan Carlos Vasquez for their role as mentors and collaborators during most of the steps detailed in this documents. I also grateful to Jaime Benjumea for his invaluable methodological insight and Sergio Perez for his technical support in managing the resources employed for this project. In addition, I am thankful for all the guidance I received from Professor Ramos-Pollan and the rest of the Analytics and Data Science Specialization team while carrying out this project.

And last, but not least, I would like to thank my wife Geraldine for her unconditional support during this project.

## References

- [1] Jesús A. Anaya, René R. Colditz, and Germán M. Valencia. Land cover mapping of a tropical region by integrating multi-year data into an annual time series. *Remote Sensing*, 7(12):16274–16292, 2015.
- [2] Liang Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 11211 LNCS, pages 833–851, 2018.
- [3] Nicola Clerici, Cesar Augusto Valbuena Calderón, and Juan Manuel Posada. Fusion of sentinel-1a and sentinel-2A data for land cover mapping: A case study in the lower Magdalena region, Colombia. *Journal of Maps*, 13(2):718–726, 2017.
- [4] Rosa Coluzzi, Vito Imbrenda, Maria Lanfredi, and Tiziana Simoniello. A first assessment of the Sentinel-2 Level 1-C cloud mask product to support informed surface analyses. *Remote Sensing of Environment*, 217(October 2017):426–443, 2018.
- [5] Nicholas M. Enwright, Lei Wang, Hongqing Wang, Michael J. Osland, Laura C. Feher, Sinéad M. Borchert, and Richard H. Day. Modeling barrier island habitats using landscape position information. *Remote Sensing*, 11(8), 2019.
- [6] A. Flores, K. Herndon, R. Thapa, and E. Cherrington. *SAR Handbook: Comprehensive Methodologies for Forest Monitoring and Biomass Estimation*. 2019.
- [7] David Frantz, Erik Haß, Andreas Uhl, Johannes Stoffels, and Joachim Hill. Improvement of the Fmask algorithm for Sentinel-2 images: Separating clouds from bright surfaces based on parallax effects. *Remote Sensing of Environment*, 215(April 2017):471–481, 2018.
- [8] GEOSAGE. Spectral Discovery for Sentinel-2 Imagery, 2020.
- [9] Nguyen Thanh Hoan, Ram C. Sharma, Nguyen Van Dung, and Dang Xuan Tung. Effectiveness of Sentinel-1-2 Multi-Temporal Composite Images for Land-Cover Monitoring in the Indochinese Peninsula. *Journal of Geoscience and Environment Protection*, 08(09):24–32, sep 2020.
- [10] Joe Morrison. How to Find the Latest Satellite Imagery in 2021. Technical report, Azavea, Philadelphia, PA, 2021.
- [11] Huong Thi Thanh Nguyen, Trung Minh Doan, Erkki Tomppo, and Ronald E. McRoberts. Land use/land cover mapping using multitemporal sentinel-2 imagery and four classification methods-A case study from Dak Nong, Vietnam. *Remote Sensing*, 12(9):1–28, 2020.
- [12] Minh D. Nguyen, Oscar M. Baez-Villanueva, Duong D. Bui, Phong T. Nguyen, and Lars Ribbe. Harmonization of Landsat and Sentinel 2 for Crop Monitoring in Drought Prone Areas: Case Studies of Ninh Thuan (Vietnam) and Bekaa (Lebanon). *Remote Sensing*, 12(2):281, jan 2020.
- [13] Tien Dat Pham, Naoto Yokoya, Junshi Xia, Nam Thang Ha, Nga Nhu Le, Thi Thu Trang Nguyen, Thi Huong Dao, Thuy Thi Phuong Vu, Tien Duc Pham, and Wataru Takeuchi. Comparison of machine learning methods for estimating mangrove above-ground biomass using multiple source remote sensing data in the red river delta biosphere reserve, Vietnam. *Remote Sensing*, 12(8):1334, apr 2020.
- [14] Ate Poortinga, Karis Tenneson, Aurélie Shapiro, Quyen Nquyen, Khun San Aung, Farrukh Chishtie, and David Saah. Mapping plantations in Myanmar by fusing Landsat-8, Sentinel-2 and Sentinel-1 data along with systematic error quantification. *Remote Sensing*, 11(7):1–19, 2019.

- [15] P S Roy, P Meiyappan, P K Joshi, M P Kale, V K Srivastav, S K Srivasatava, M D Behera, A Roy, Y Sharma, and R M Ramachandran. Decadal Land Use and Land Cover Classifications across India, 1985, 1995, 2005, 2016.
- [16] Andrei Stoian, Vincent Poulain, Jordi Inglada, Victor Poughon, and Dawa Derksen. Land cover maps production with high resolution satellite image time series and convolutional neural networks: Adaptations and limits for operational systems. *Remote Sensing*, 11(17):1–26, 2019.
- [17] Andreas Vollrath, Adugna Mullissa, and Johannes Reiche. Angular-based radiometric slope correction for Sentinel-1 on google earth engine. *Remote Sensing*, 12(11):1–14, 2020.
- [18] Kilian Vos, Kristen D. Splinter, Mitchell D. Harley, Joshua A. Simmons, and Ian L. Turner. CoastSat: A Google Earth Engine-enabled Python toolkit to extract shorelines from publicly available satellite imagery. *Environmental Modelling and Software*, 122:104528, 2019.
- [19] Fabien H. Wagner, Alber Sanchez, Marcos P.M. Aidar, André L.C. Rochelle, Yuliya Tarabalka, Marisa G. Fonseca, Oliver L. Phillips, Emanuel Gloor, and Luiz E.O.C. Aragão. Mapping Atlantic rainforest degradation and regeneration history with indicator species using convolutional network. *PloS one*, 15(2):e0229448, 2020.
- [20] Juntang Zhuang, Tommy Tang, Yifan Ding, Sekhar Tatikonda, Nicha Dvornek, Xenophon Papademetris, and James S. Duncan. AdaBelief Optimizer: Adapting Stepsizes by the Belief in Observed Gradients. (NeurIPS), 2020.