

BARRETT HONORS THESIS

# A U-Net to Identify Deforested Areas in Satellite Imagery of the Amazon

Joshua Giel and Liam Douglas

**Thesis Director:** Dr. Malena Español

**Second Committee Member:** Dr. Douglas Cochran

Spring 2024

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Related Work</b>	<b>5</b>
<b>3</b>	<b>Data Sources</b>	<b>6</b>
<b>4</b>	<b>Data Collection</b>	<b>7</b>
<b>5</b>	<b>Model Implementation</b>	<b>12</b>
<b>6</b>	<b>Results</b>	<b>17</b>
<b>7</b>	<b>Limitations</b>	<b>20</b>
<b>8</b>	<b>Conclusion</b>	<b>22</b>
<b>9</b>	<b>Acknowledgements</b>	<b>22</b>

# **Abstract**

Deforestation in the Amazon rainforest has the potential to have devastating effects on ecosystems on both a local and global scale, making it one of the most environmentally threatening phenomena occurring today. In order to minimize deforestation in the Amazon and its consequences, it is helpful to analyze its occurrence using machine learning architectures such as the U-Net. The U-Net is a type of Fully Convolutional Network that has shown significant capability in performing semantic segmentation. It is built upon a symmetric series of downsampling and upsampling layers that propagate feature information into higher spatial resolutions, allowing for the precise identification of features on the pixel scale. Such an architecture is well-suited for identifying features in satellite imagery. In this thesis, we construct and train a U-Net to identify deforested areas in satellite imagery of the Amazon through semantic segmentation.

# 1 Introduction

Forests are a key component in our world's ecosystem. Releasing oxygen, absorbing carbon dioxide, regulating climate, and providing habitat for countless species are just a few of the indispensable services that forests provide. Specifically, the tropical rainforest within the Amazon biome plays an especially vital role. As of 2020, 6.7 million square kilometers of the 40.6 million square kilometers of global forest existed within the Amazon region alone, comprising over 16.5% of the global forest area [1, 2]. The Amazon rainforest provides habitat for more than 10% of all known wildlife species [3], absorbs a quarter of all carbon dioxide absorbed by land globally [4], and acts as a necessary component of global moisture transport systems that could be permanently damaged by a sufficient amount of degradation towards the region [5].

Despite its importance, the Amazon rainforest is being depleted rapidly. 750 thousand square kilometers of the Amazon rainforest were destroyed between 1978 and 2017 [2] and the region is continuing to lose approximately 34 thousand square kilometers of forest area per year, leading it to be considered the most threatened ecology region globally in terms of forest area cleared per year [6]. The main driver behind deforestation in the Amazon is cattle ranching, as farmers clear forests to produce beef. Other key drivers include large-scale and small-scale agriculture (specifically of soy and oil palm), mining and logging operations, and infrastructure development [2].

Brazil, which contains 60% of the entire Amazon rainforest, has made considerable efforts to reduce rates of deforestation in the Amazon. Organizations such as the Program for Deforestation Monitoring (PRODES), an initiative by the Brazilian National Institute for Space Research launched in 1988, monitor rates of deforestation by processing remote sensing imagery, thereby encouraging and evaluating policies to protect the Amazon rainforest [7]. Specifically, the 2004 Action Plan for Prevention and Control of Deforestation has been considerably effective as it contributed to an 84% reduction in deforestation rates between 2004 and 2012 [7, 8]. However, between 2012 and 2021, there was a 185% increase in deforestation [7], which could be attributed to changes made in 2012 to the

Brazilian Forest Code and reductions to the Brazilian Ministry of Environment's capabilities [8]. Between 2021 and 2023, there was a 31% decrease in deforestation rates [7], perhaps signifying another downward trend (see Figure 1).

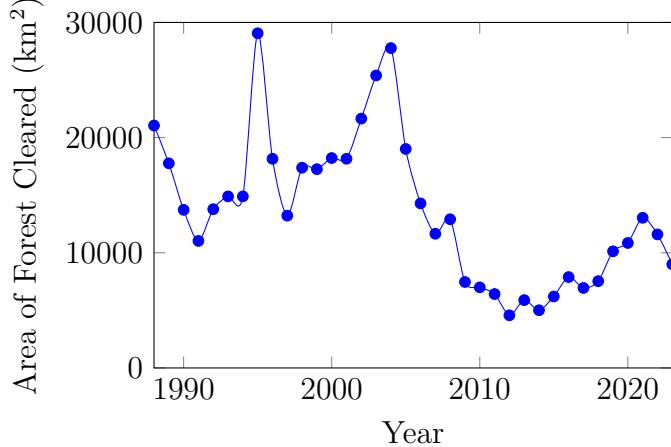


Figure 1: Forest Area Cleared per Year in Brazilian Legal Amazon (retrieved from PRODES [7]).

In the following sections, we propose and implement a U-Net model to identify deforested regions in optical sensor data captured over the Brazilian Legal Amazon, a 5.2 million square kilometer region containing the entirety of the Amazon rainforest located in Brazil [2]. Figure 2 shows a map of the Brazilian Legal Amazon.

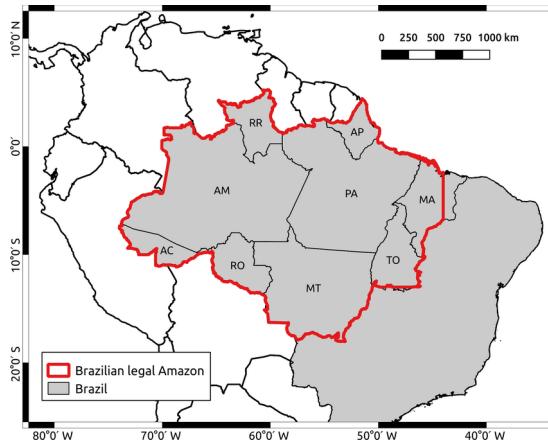


Figure 2: “Map of the Brazilian Legal Amazon and its nine federal states: Acre (AC), Amapá (AP), Amazonas (AM), Maranhão (MA), Mato Grosso (MT), Pará (PA), Rondônia (RO), Roraima (RR), and Tocantins (TO)” [9].

## 2 Related Work

Among the most notable attempts to quantify forested regions is the PRODES initiative. To identify deforested regions, PRODES uses trained specialists to manually identify deforested areas within satellite imagery. After applying a contrast operation to the imagery, they identify deforested polygons greater than 0.0625 square kilometers and compare them to regions identified in previous years to calculate changes in forest coverage. In cases in which cloud coverage prevents a direct calculation of change in forestry, they instead use a calculation that is negatively weighted by the number of successive years in which the region is unobservable [10]. Clearly, this method of classification is highly dependent on humans and requires a considerable amount of labor.

Various attempts to automate the classification of satellite imagery have been completed as well. According to a 2014 study by Yu et al., maximum likelihood classifiers (a statistical Bayesian decision model [11]) have been commonly used in remote sensing classification even though other classification models are found to be more accurate, which they attributed to its prevalence in image-processing software packages [12]. Historical attempts to implement more advanced supervised models have often been made through support vector machines [13] and random forest classifiers [14]. However, as deep learning models have been further studied and developed, they have been found to be more accurate than these traditional machine learning techniques when the training dataset is sufficiently large [15].

In particular, Convolutional Neural Networks (CNNs) have been shown to be especially promising in image classification as they use successive convolution and pooling layers to identify relevant information [16]. While CNNs are only able to detect the probability of the occurrence of an object in an image, the Fully Convolutional Network (FCN), initially proposed by Long et al. in 2014 [17], is able to detect where an object may appear in an image pixel-based segmentation [18]. A further advancement is the U-Net, a specific architecture of the FCN proposed by Ronneberger et al. in 2015 [19], which propagates segmentation information in higher spatial resolutions through concatenation

between respective upsampling and downsampling layers [18].

An attempt to identify deforested regions in the Legal Amazon using a U-Net was implemented by de Bem et al. in 2020, which used 200x200 images of optical and non-optical bands with 30-meter resolution from Landsat 8 as training inputs and masks derived from PRODES data as ground-truth outputs. This implementation used 3,376 images for training and 1,688 images for testing [15]. Another attempt was implemented by Bragagnolo et al. in 2021, which used 512x512 images of optical bands with 10-meter resolution from Sentinel-2 imagery as training inputs and masks obtained through the GRASS-GIS 7.6.1 software as ground-truth outputs. This implementation used 45 images for training, 15 images for testing, and 15 images for validation[18].

### 3 Data Sources

To obtain images of the Brazilian Legal Amazon for training our model, we relied upon the Landsat 8 Level 2 satellite. As a joint collaboration between NASA and the United States Geological Survey (USGS), the Landsat 8 satellite was launched in 2013 and has since been in a sun-synchronous orbit at an altitude of 705 kilometers. The satellite captures data in 185 kilometer swaths, which then get segmented into 185 kilometer by 180 kilometer scenes in accordance with the Second Worldwide Reference System. Landsat 8 repeats each path every 16 days, resulting in each scene being captured a maximum of either 22 or 23 times per year. Although landsat 8 captures a variety of bands, we chose to train our model solely on the optical bands to reduce the size of collected data and to allow for analysis of strictly RGB images. Specifically, we focused on band 2 (0.452-0.512  $\mu\text{m}$ ), band 3 (0.533-0.590  $\mu\text{m}$ ), and band 4 (0.636-0.673  $\mu\text{m}$ ), which represent blue, green, and red surface reflectance, respectively [20, 21].

To obtain labeled ground-truth images of the Brazilian Legal Amazon for training our model, we relied upon the Global Forest Change (GFC) dataset. Using Landsat imagery with a 30-meter spatial resolution, a University of Maryland research group led

by Dr. Matthew Hansen developed a dataset tracking change in global forest coverage using a supervised classification approach [22]. The GFC dataset maps the global extent of canopy coverage in the year 2000, where canopy coverage is defined as the existence of vegetation taller than 5 meters in height; areas where loss occurred within the continued study period of 2000-2022, where loss is defined as a change from a forest to a non-forest state; and areas where gain occurred (and remained) during the initial study period of 2000-2012, where gain is defined as a change from a non-forest to a forest state. The data for the extent of canopy coverage in the year 2000 is available as a band containing pixels in the range [0, 100], while the data for the loss within 2000-2022 and the gain within 2000-2012 are available as bitmasks. Additionally, a band pertaining to the year in which forest loss occurred (values in the range [0, 22]) for each pixel is provided, however no such band exists for forest gain [23]. For both the Landsat and GFC datasets, we chose to limit our selection of images to those captured in 2022.

## 4 Data Collection

We utilized Google Earth Engine (GEE) to obtain Landsat and GFC images. GEE provides access to a variety of geospatial datasets, including the entire Landsat and GFC archives, which can be accessed through either their application programming interface (API) or an online code editor. Each geospatial image is composed of a set of bands that can refer to different captured wavelengths or classifications, and each band is ingested into GEE in its original projection, bit depth, and resolution. Furthermore, geospatial images from the same source are accessed as ‘collections,’ which can be filtered based on temporal, spatial, and atmospheric constraints [24]. Thus, we chose to access the Landsat and GFC datasets using GEE because of the platform’s accessibility and information-preserving nature.

To collect Landsat images, the SR\_B4, SR\_B3, and SR\_B2 bands (red, green, and blue surface reflectance, respectively) were selected from the Landsat 8 Level 2, Collection

2, Tier 1 GEE dataset. The dataset was then spatially filtered over the Legal Amazon using a shapefile from TerraBrasilis (Figure 3).



Figure 3: Shapefile of Legal Amazon region (retrieved from PRODES [7]).

The datasets were further filtered to include only images from the dry season of 2022 (June 1st-August 1st) and to only include images where clouds obscure less than 10% of the landscape (Figure 4), a threshold which we found to provide an appropriate tradeoff between dataset reduction and content obscuration.



Figure 4: Landsat images with no cloud restriction (left) and 10% maximum cloud coverage (right)

Because pixels in Landsat 8 Level 2 imagery are scaled and offset to bring values to the 16-bit integer range, the original floating point values of the Landsat imagery were obtained by mapping

$$p = (0.0000275 \cdot \hat{p}) - 0.2$$

across each band's pixels where  $p$  represents the original floating point pixel value and  $\hat{p}$  represents the 16-bit pixel value, as specified by the USGS [25].

To collect GFC images from 2022, the ‘treecover2000’ (Figure 5), ‘loss’ (Figure 6), and ‘gain’ (Figure 7) bands were selected from GEE’s GFC dataset and spatially filtered over the Legal Amazon.



Figure 5: GFC data [23] showing percent tree coverage in 2000, where lighter areas indicate a high percentage and darker areas indicate a lower percentage.

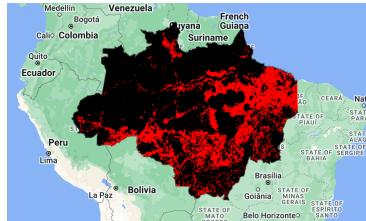


Figure 6: GFC data [23] areas in which tree coverage was lost between 2000-2022 in red.



Figure 7: GFC data [23] areas in which tree coverage was gained between 2000-2012 in turquoise, obtained via GEE.

To create a binary mask from the continuous 'treecover2000' band, we experimented with discretizing the band with different thresholds based on findings by Gaspirini et al. [26] and chose a threshold of 80% based on our judgment of our model's resulting accuracy. This threshold was applied by mapping

$$p_m = \begin{cases} 1 & \text{if } p_b \geq 0.8 \\ 0 & \text{if } p_b < 0.8 \end{cases} \quad (1)$$

across the 'treecover2000' band's pixels, where  $p_b$  represents the binary mask's pixel value and  $p_t$  represents the 'treecover2000' band's pixel value.

The 'gain' and 'loss' bands were considered in the binary mask through GEE by

mapping

$$p_m = \begin{cases} 1 & \text{if } p_g = 1, p_l = 1, y_l \leq 12 \\ 1 & \text{if } p_g = 1, p_l = 0 \\ 0 & \text{if } p_g = 1, p_l = 1, y_l > 12 \\ 0 & \text{if } p_g = 0, p_l = 1 \\ p_m & \text{if } p_l = 0, p_g = 0, \end{cases}$$

across the 'loss' and 'gain' band's pixels, where  $p_m$  represents the binary mask's pixel values,  $p_g$  represents the GFC gain band's pixel values,  $p_l$  represents the GFC loss band's pixel values, and  $y_l \in [0, 22]$  represents the GFC 'lossyear' band's pixel values (i.e., values corresponding to the year between 2000-2022 in which forest loss occurred).

Note that in the first case, forest area was lost and subsequently gained within the study period of 2000-2012. In the second case, forest area was simply gained (with loss possibly occurring before 2000). In the third case, forest area was lost after 2012, but the value of the gain band can not be considered as it only pertains to forest area gained within the study period 2000-2012 [23]. In the fourth case, forest area was simply lost without being gained. In the final case, forest area was neither lost nor gained, so it remains as it was calculated in Equation 1. The result of this operation can be seen in Figure 8.



Figure 8: Mask of tree coverage obtained after considering 'loss' and 'gain' bands using Equation (2), obtained via GEE.

After obtaining the Landsat and GFC datasets, they were reprojected to obtain a 60-meter spatial resolution from the original 30-meter resolution. Next, the specified Amazon region was partitioned into sections of size 30,720 meters by 30,720 meters to obtain image dimensions of 512 pixels by 512 pixels, as seen in Figure 9.

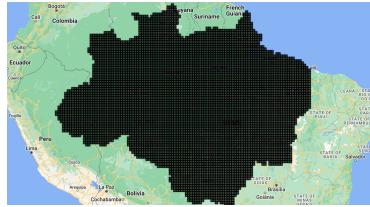
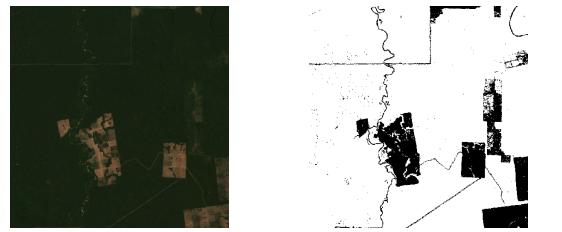


Figure 9: Partition of Legal Amazon region into 30720m by 30720m regions.

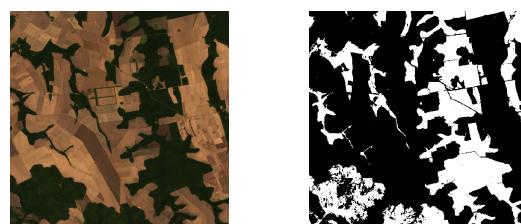
500 sections were then randomly selected and their corresponding Landsat and GFC images were exported via Google Drive. Examples of the resulting 512x512 images of 60-meter resolution can be seen in Figure 10.



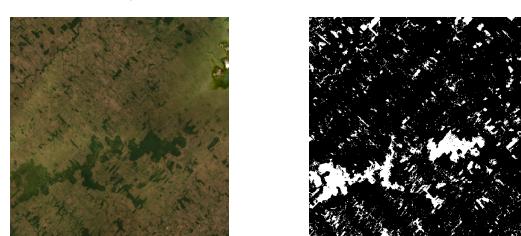
(a): Centered at (-10.06606179903945°, -60.81010568554429°)



(b): Centered at (-6.480889256034368°, -55.2944498410516°)



(c): Centered at (-13.927016498921605°, -55.57023263327493°)



(d): Centered at (-10.893409269578752°, -62.740585231116874°)

Figure 10: Landsat images (left) and GFC images (right), where white pixels signify the presence of trees and white areas signify no presence of trees.

## 5 Model Implementation

We implemented a Fully Convolutional Network in the style of a U-Net using Google’s TensorFlow and Keras libraries in Python. U-Net model architecture relies on an encoder-decoder structure in which inputs are downsampled until reaching a bottleneck layer and subsequently upsampled until the desired output dimension is reached. These models are typically symmetric, with an equivalent number of upsampling and downsampling layers forming a ‘U’ shape. While this encoder-decoder structure mimics many standard implementations of FCN architecture, skip-connection concatenations are added between corresponding downsampling and upsampling layers to improve finer detail recognition along with Rectified Linear Units (ReLU) following each convolution to improve model performance for more complex data [19]. The architecture of the constructed U-Net can be seen in Figure 11.

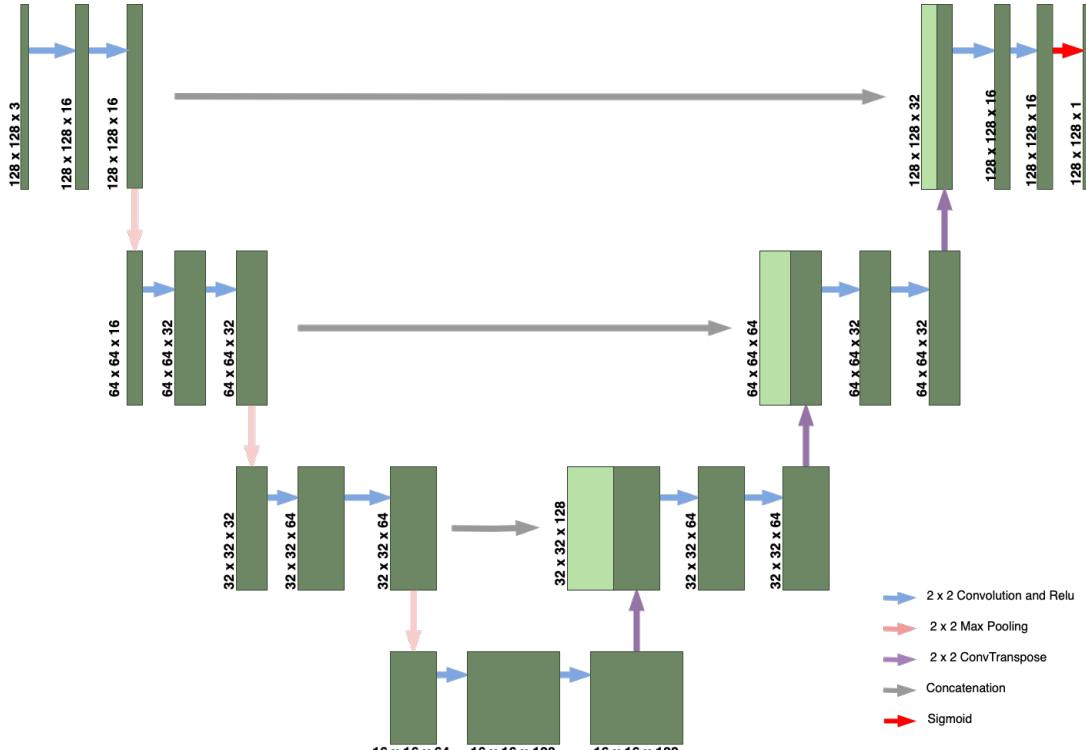


Figure 11: Implemented U-Net architecture. Each green box corresponds to a multi-layer feature map with the dimensions provided, and light green boxes represent concatenated feature maps. Each arrow represents an operation used to generate the subsequent feature map.

The model consists of 4 downsampling units, 4 upsampling units with 3 skip connections, and a final output layer. This depth reduces over-fitting concerns inherent in deeper models while maintaining validation set accuracy and fine feature detection that is not easily achieved in shallower networks.

Each downsampling unit consists of 2 convolution layers which use the operation defined as

$$O_{N_i, C_{out_j}} = \sum_{k=1}^{C_{in}} \omega_{C_{out_j}, k} \star I_{N_i, k},$$

where  $\star$  is the two-dimentional cross-correlation operation,  $O$  is the output image,  $I$  is the input image,  $\omega$  is the kernel,  $N_i$  is the  $i$ th batch out of  $N$  batches,  $C_{out_j}$  is the  $j$ th output channel out of  $C_{out}$  output channels, and  $C_{in}$  is the number of input channels [27].

A kernel size of 2x2 was chosen to improve the detection of fine features despite relatively small input dimensions. These convolution layers use a ReLU function defined as

$$f(x) = \max(0, x).$$

The convolution layers are followed by a max pooling layer, which uses the operation defined as

$$O_{N_i, C_j, h, w} = \max_{m=0, \dots, k_h-1} \left( \max_{n=0, \dots, k_w-1} (I_{N_i, C_j, h+m, w+n}) \right),$$

where  $O$  is the output image,  $I$  is the input image,  $N_i$  is the  $i$ th batch out of  $N$  batches,  $C_j$  is the  $j$ th channel out of  $C$  channels,  $h$  is the output height,  $w$  is the output width,  $k_h$  is the kernel height, and  $k_w$  is the kernel width [28].

The max pooling layer is followed by a batch normalization layer, which uses

$$\mu_c = \frac{1}{N \times H \times W} \sum_{n=1}^N \sum_{h=1}^H \sum_{w=1}^W f_{n,h,w}$$

to calculate the mean, where  $N$  is the number of batches,  $H$  is the height of the images,

$W$  is the width of the images. To calculate the variance, we use

$$\delta_c^2 = \frac{1}{N \times H \times W} \sum_{n=1}^N \sum_{h=1}^H \sum_{w=1}^W (f_{n,h,w} - \mu_c)^2$$

and to calculate normalization, we use

$$\hat{f}_{n,h,w} = \frac{f_{n,h,w} - \mu_c}{\sqrt{\delta_c^2 + \epsilon}},$$

where  $\epsilon$  is a parameter used to ensure stability of division [29]. Batch normalization was chosen as the normalization technique due to its preferable results in previous U-Net implementations [29] and its ability to improve training time and rate of convergence during experimentation.

Each upsampling unit follows a mirrored structure, beginning with an upsampling deconvolution layer (ConvTranspose in TensorFlow), which reverses the downsampling operations made in the encoder portion of the model [30]. This operation is followed by a skip-connection concatenation step, which appends the resulting feature maps from the corresponding downsampling layer to the upsampled feature map from the deconvolution step for further processing. Since concatenation requires at least 2 equivalent dimensions, upsampling rates are correlated with downsampling rates to facilitate skip connection implementation.

The final processing unit consists of 2 convolution layers and a densely connected sigmoid layer, which uses the function defined as

$$f(x) = \frac{1}{1 + e^{-x}}.$$

A binary segmentation mask is obtained by mapping a thresholding technique across all pixels, defined as

$$\hat{p} = \begin{cases} 1 & \text{if } p \geq 0.5 \\ 0 & \text{if } p < 0.5, \end{cases}$$

where  $p \in [0, 1]$  is the pixel value produced by the model and  $\hat{p}$  is the resulting mask's pixel value. An example of the output of this thresholding technique can be seen in Figure 12.

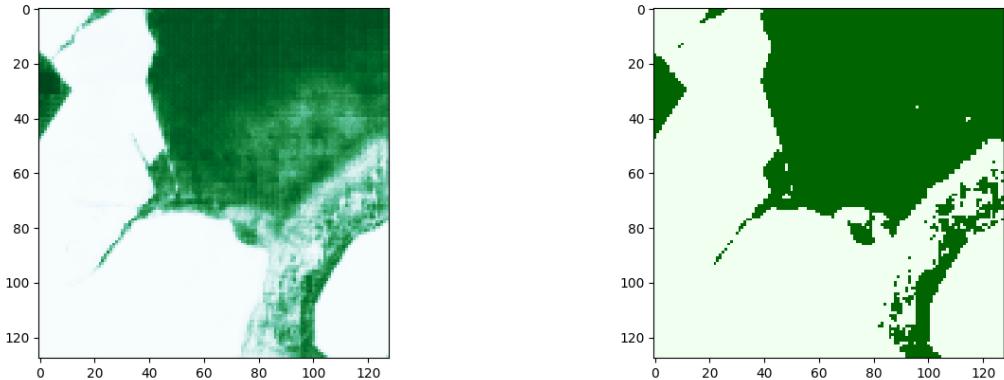


Figure 12: Model output before thresholding (left) and after thresholding (right).

Datasets were created by processing the images generated by GEE into NumPy arrays using Python's Tiff file library. The data was then partitioned into images of dimensions  $128 \times 128 \times 3$  using the Patchify library to obtain a dataset of 8,000 total images. This corrected any export dimension inconsistencies resulting from Earth engine as well as reduced image dimensionality in order to save training time and preserve model depth. Additionally, the patching operation facilitates easy reassembly of input data for visualizing and interpreting larger areas for analysis. Each resulting input image was normalized to  $[0,1]$  for each RGB channel to limit the risk of numerical errors during training. The resulting image tensors were then split into training, test, and validation sets using Sci-Kit Learn's train-test-split function before model fitting began.

To train our model, we used hyperparameters shown in the following table:

Hyperparameters	Values
Optimizer	Adam
Epochs	100
Loss	Binary Cross-Entropy
Batch Size	32
Learning Rate	0.0001
Train Size	6400 images
Test Size	800 images
Validation Size	800 images

Binary Cross-Entropy was chosen as the loss function due to the nature of the classification task and its successful application to general image segmentation problems [31].

The operation for Binary Cross-Entropy is defined as

$$-\frac{1}{N} \sum_{i=1}^N y_i \cdot \log(p(y_i)) + (1 - y_i) \cdot \log(1 - p(y_i)),$$

where  $y$  is the classification label,  $p(y)$  is the predicted probability of the given point being positive, and  $N$  is the total number of data points [32].

The Adam optimizer was chosen because of its efficiency in terms of both computational time and memory [33]. Final Model training was performed on ASU's Sol Supercomputer. The full implementation of our model can be found on [GitHub](#).

## 6 Results

After training our model using the aforementioned hyperparameters, we obtained the following results from our training dataset:

Metrics	Values
True Positives	6,876,792
True Negatives	5,227,887
False Positives	560,515
False Negatives	442,006
Accuracy	0.923
Precision	0.924
Recall	0.939
F1 Score	0.932
Jaccard Score	0.872
Avg. Runtime per Epoch	88 seconds

True positives and true negatives denote the number of positive and negative pixel images that were correctly classified. False positives and false negatives denote the number of pixels that were incorrectly classified as positive and negative. Accuracy is calculated as

$$A = \frac{T_p + T_n}{T_p + T_n + F_p + F_n},$$

where  $T_p$  represents true positives,  $T_n$  represents true negatives,  $F_p$  represents false positives, and  $F_n$  represents false negatives. Precision is calculated as

$$P = \frac{T_p}{T_p + F_p},$$

recall score is calculated as

$$R = \frac{T_p}{T_p + F_n},$$

and the F1 score is calculated as

$$F1 = \frac{2PR}{P + R}.$$

The Jaccard score is calculated as

$$J(Y_t, Y_p) = \frac{|Y_t \cap Y_p|}{|Y_t \cup Y_p|}$$

where  $Y_t$  represents the set of ground-truth pixels and  $Y_p$  represents the set of predicted pixels.

The loss and accuracy per epoch of our training and validation datasets can be seen in Figure 13.

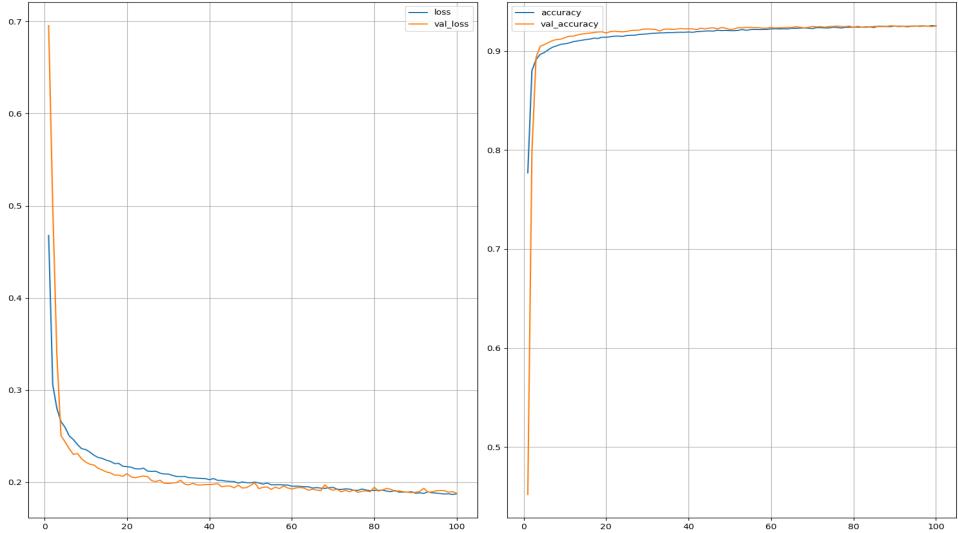


Figure 13: Graph of loss (left) and accuracy (right) per epoch, where the blue line represents our training dataset and the orange line represents our validation dataset.

The similar attempt to identify deforestation in the Amazon region using a U-Net by Bragagnolo et al. yielded a precision of 0.936, recall of 0.968, and F1 score of 0.951 [18]. The similar attempt by de Bem et al. yielded a precision of 0.918, recall of 0.951, and F1 score of 0.934 [15]. While these implementations offer improvements in key model metrics, our use of substantially lower-resolution satellite imagery reduces prediction time and our omission of non-optical satellite bands allows for prediction on readily accessible RGB image data with limited sacrifice in model performance.

Examples of our segmentation maps and the corresponding maps of false positives

and false negatives can be seen in Figure 15.

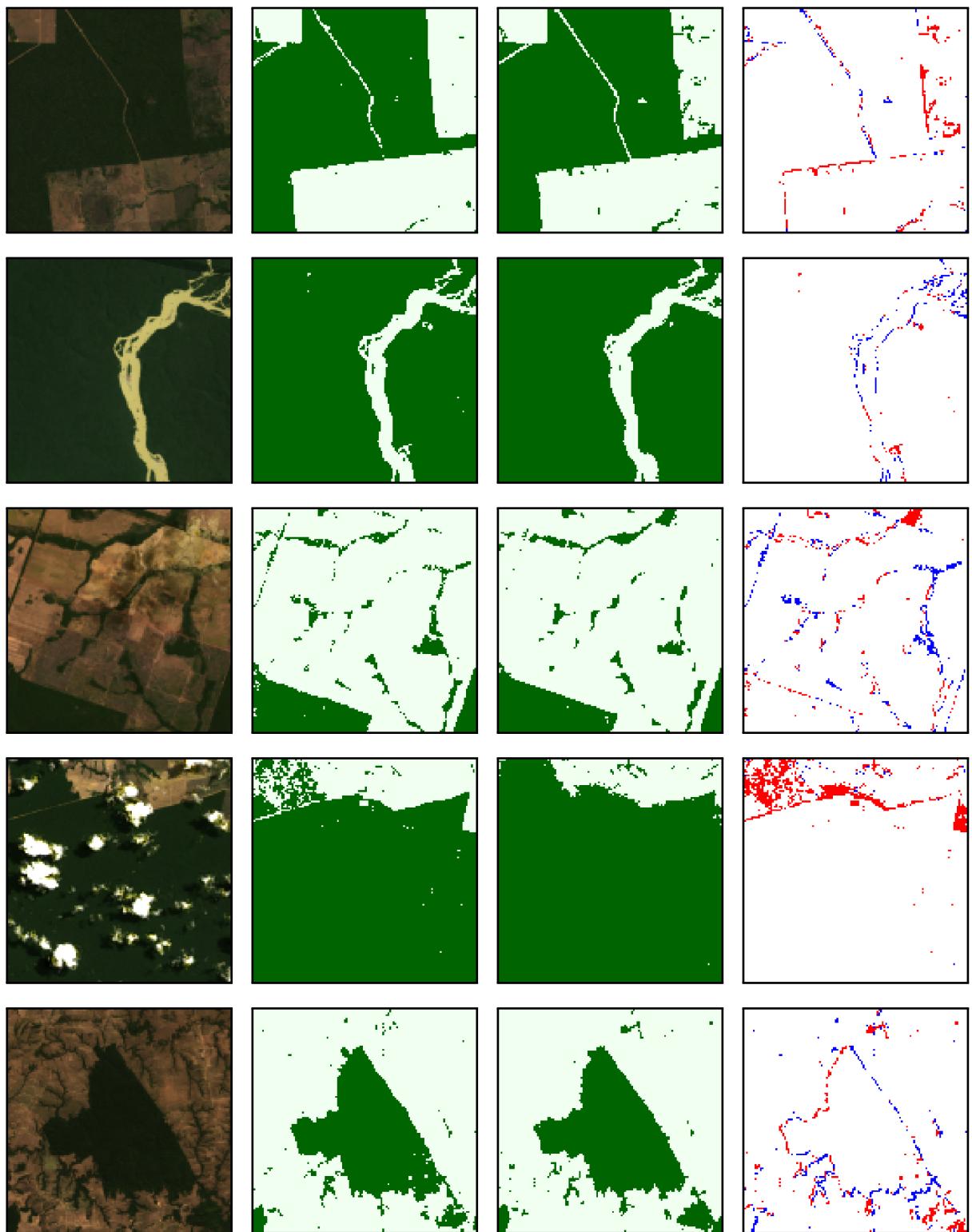


Figure 14: 128x128 Landsat images (leftmost), GFC data (center-left), predictions (center-right), and maps of false positives and false negatives, where red indicates false positives and blue indicates false negatives.

## 7 Limitations

The performance of a machine learning model is often limited by the quality of the dataset used for training. For this image segmentation task, pixel-wise classification data is not widely available due to the labor-intensive nature of satellite imagery annotation and the relative infancy of forest cover image segmentation research [34]. The GFC dataset used in this paper was generated using a supervised learning approach, meaning some deviation from ground-truth is necessarily present in our training data. Possible examples of such deviations can be seen in Figure 15. Other limitations of the GFC dataset include a lack of provided methodology provided by Hansen et al. in the development of their model and the absence of data pertaining to gains in tree coverage after 2012 [22]. Such limitations may have negatively impacted model training.

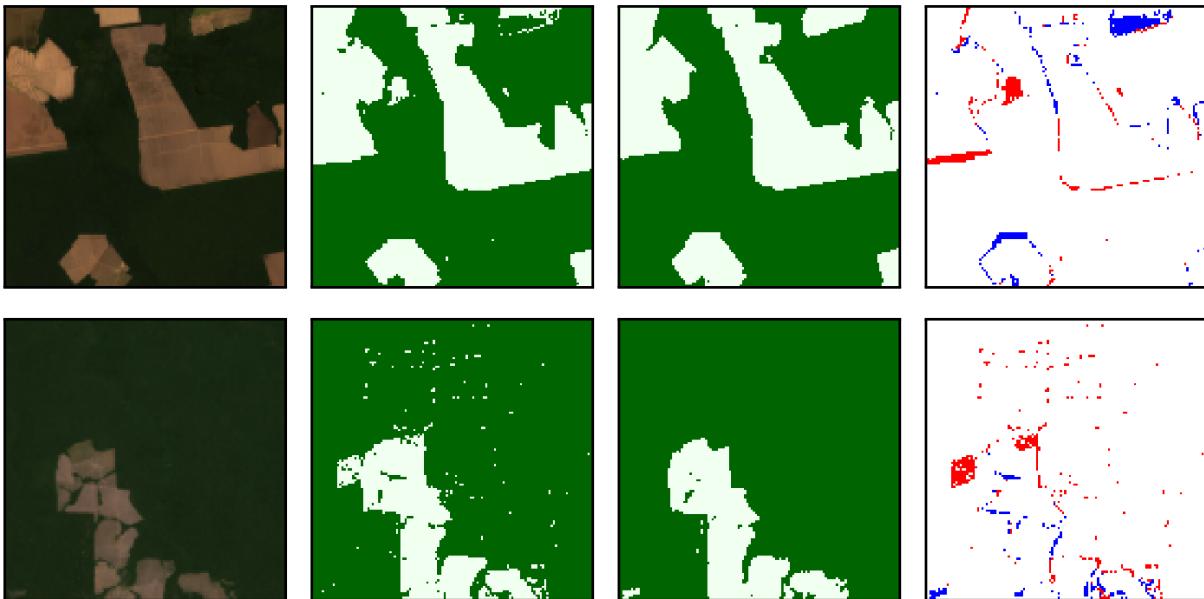


Figure 15: 128x128 Landsat images (leftmost), GFC data (center-left), predictions (center-right), and maps of false positives and false negatives, where red indicates false positives and blue indicates false negatives. It can be observed that GFC data does not seem to perfectly align with Landsat image.

While other datasets may have yielded marginal improvements in model performance, the ease of large-scale collection and correlation to their associated Landsat images made the GFC dataset the most desirable option for this application.

Additionally, the presence of cloud coverage and other atmospheric anomalies can

cause challenges in analyzing forest coverage from Landsat imagery. By requiring cloud coverage in the Landsat images to be less than 10%, we were not able to access satellite imagery of certain regions of the Legal Amazon, which can be seen in Figure 4. In terms of our model performance, it is visually apparent that in some cases, remaining clouds negatively affect the predicted image segmentation mask (Figure 16).

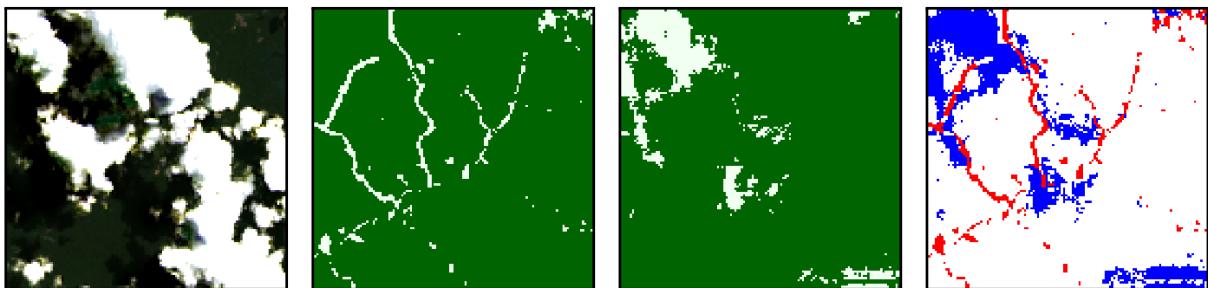


Figure 16: 128x128 Landsat images (leftmost), GFC data (center-left), predictions (center-right), and maps of false positives and false negatives, where red indicates false positives and blue indicates false negatives

Because of the nature of atmospheric weather conditions and satellite paths, some regions that experience high degrees of cloud coverage (many of which represent densely forested areas) may be more challenging to analyze using a Landsat-based approach, specifically in the optical bands. For this reason, it may be advantageous to research approaches that utilize satellite data that penetrates cloud coverage (e.g., SAR).

The presence of non-forest objects in satellite imagery, such as clouds, bodies of water, and settlements, has limited the overall accuracy of our model and could thus warrant the investigation of multi-class classification approaches to identify these features. Such approaches have been implemented by other researchers [35, 36]; however, due to time restrictions and the limited scope of our project, we chose to focus solely on implementing a binary approach.

## 8 Conclusion

Given the importance of the Amazon rainforest within the global ecosystem, efforts to identify deforested regions are necessary to allow for a thorough investigation of the phenomenon and to encourage the implementation of policy to prevent it. The U-Net model was able to effectively identify deforested regions in the Legal Amazon, achieving an accuracy of 0.923, precision of 0.924, recall of 0.939, and F1 score of 0.932. While these performance metrics were slightly lower than comparable implementations, quick prediction time and lower data requirements offer some advantages over other models.

Future work could include implementing a multi-class approach to further segment non-forest objects and applying our model to analyze deforestation through the temporal domain and in forests beyond the Amazon.

## 9 Acknowledgements

This work was supported in part by the Office of the Secretary of Defense Multi-disciplinary University Research Initiative (MURI) under ONR Grant No. N00014-20-1-2595. We would like to thank Dr. Malena Español, our thesis director, and Dr. Douglas Cochran, our second committee member, for their constant support and advice.

**Note:** Dataset collection and preprocessing were performed by Joshua Giel. Model implementation and evaluation were performed by Liam Douglas.

## References

- [1] Food and Agriculture Organization of the United Nations. The State of the World's Forests. <https://www.fao.org/state-of-forests/en/>, 2020.
- [2] B. Wegrowski. Deforestation in the Amazon Rainforest. <https://ballardbrief.byu.edu/issue-briefs/deforestation-in-the-amazon-rainforest#:~:text=The%20Amazon%20is%20the%20largest,Peru%2C%20Suriname%2C%20and%20Venezuela.,> 2019.
- [3] World Wildlife Fund. The Amazon. <https://www.wwf.org.uk/where-we-work/amazon#:~:text=South%20America's%20Amazon%20contains%20nearly,we%20don't%20know%20yet.>, 2023.
- [4] National Geographic. Amazon Deforestation and Climate Change. <https://education.nationalgeographic.org/resource/amazon-deforestation-and-climate-change/>, 2024.
- [5] N. Boers, N. Marwan, and H. Barbosa et al. A deforestation-induced tipping point for the South American monsoon system. *Scientific Reports*, 7(41489), 2017.
- [6] A. Hänggli, S.A. Levy, and D. Armenteras et al. A systematic comparison of deforestation drivers and policy effectiveness across the Amazon biome. *Environmental Research Letters*, 18(7):073001, 2023.
- [7] INPE. PRODES - Amazônia. <http://www.obt.inpe.br/OBT/assuntos/programas/amazonia/prodes>, 2023.
- [8] C.H.L. Silva Junior, A.C.M. Pessôa, and N.S. Carvalho et al. The Brazilian Amazon deforestation rate in 2020 is the greatest of the decade. *Nature Ecology and Evolution*, 5:144–145, 2021.

- [9] F. Müller-Hansen, M. Cardoso, and E. Dalla-Nora et al. A matrix clustering method to explore patterns of land-cover transitions in satellite-derived maps of the brazilian amazon. *Nonlinear Processes in Geophysics*, 24:113–123, 2017.
- [10] C.A. Almeida, L.E.P. Maurano, and D.M. Valeriano. Methodology for Forest Monitoring Used in PRODES and DETER Projects. 2021.
- [11] ArcGIS. How Maximum Likelihood Classification works. <https://desktop.arcgis.com/en/arcmap/latest/tools/spatial-analyst-toolbox/how-maximum-likelihood-classification-works.htm#:~:text=The%20algorithm%20used%20by%20the, Bayes' %20theorem%20of%20decision%20making.>
- [12] L. Yu, L. Liang, and J. Wang et al. Meta-discoveries from a synthesis of satellite-based land-cover mapping research. *International Journal of Remote Sensing*, 35(13):4573–4588, 2014.
- [13] N. Kranjčić, D. Medak, R. Župan, and M. Rezo. Support vector machine accuracy assessment for extracting green urban areas in towns. *Remote Sensing*, 11(6), 2019.
- [14] M. Pal. Random forest classifier for remote sensing classification. *International Journal of Remote Sensing*, 26(1):217–222, 2005.
- [15] P.P. de Bem, O.A. de Carvalho Junior, R. Fontes Guimarães, and R.A. Trancoso Gomes. Change Detection of Deforestation in the Brazilian Amazon Using Landsat Data and Convolutional Neural Networks. *Remote Sensing*, 12(6), 2020.
- [16] M. Wieland, Y. Li, and S. Martinis. Multi-sensor cloud and cloud shadow segmentation with a convolutional neural network. *Remote Sensing of Environment*, 230:111203, 2019.

- [17] J. Long, E. Shelhamer, and T. Darrell. Fully Convolutional Networks for Semantic Segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, 2014.
- [18] L. Bragagnolo, R.V. da Silva, and J.M.V. Grzybowski. Amazon forest cover change mapping based on semantic segmentation by U-Nets. *Ecological Informatics*, 62:101279, 2021.
- [19] O. Ronneberger and P. Fischer and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In N. Navab, J. Hornegger, W.M. Wells, and A.F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing.
- [20] NASA. Landsat 8. <https://landsat.gsfc.nasa.gov/satellites/landsat-8/>.
- [21] D.P. Roy, M.A. Wulder, and T.R. Loveland et al. Landsat-8: Science and product vision for terrestrial global change research. *Remote Sensing of Environment*, 145:154–172, 2014.
- [22] M.C. Hansen, P.V. Potapov, and R. Moore et al. High-Resolution Global Maps of 21st-Century Forest Cover Change. *Science*, 342(6160):850–853, 2013.
- [23] Earth Engine Data Catalog. Hansen Global Forest Change v1.10 (2000-2022).
- [24] R. Lindsey. Drought parches the central Amazon in October 2023. <https://www.climate.gov/news-features/event-tracker/drought-parches-central-amazon-october-2023>, 2023.
- [25] USGS. How do I use a scale factor with Landsat Level-2 science products? <https://www.usgs.gov/faqs/how-do-i-use-a-scale-factor-landsat-level-2-science-products>.

- [26] K.A.C Gasparini, C.H.L Silva Junior, and Y.E Shimabukuro et al. Determining a threshold to delimit the amazonian forests from the tree canopy cover 2000 gfc data. *Sensors*, 19(22), 2019.
- [27] PyTorch. Conv2d. <https://pytorch.org/docs/stable/generated/torch.nn.Conv2d.html>.
- [28] PyTorch. MaxPool2d. <https://pytorch.org/docs/stable/generated/torch.nn.MaxPool2d.html>.
- [29] X.Y. Zhou and G.Z. Yang. Normalization in training u-net for 2-d biomedical semantic segmentation. *IEEE Robotics and Automation Letters*, 4(2):1792–1799, 2019.
- [30] A. Zhang, Z.C. Lipton., and M. Li. et al. Transposed Convolution. In *Dive into Deep Learning*. Cambridge University Press, 2023.
- [31] S. Jadon. A survey of loss functions for semantic segmentation. In *2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*. IEEE, October 2020.
- [32] D. Godoy. Understanding binary cross-entropy / log loss: a visual explanation. <https://towardsdatascience.com/understanding-binary-cross-entropy-log-loss-a-visual-explanation-a3ac6025181a>, 2018.
- [33] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [34] Y. Lu and Y. Huang and S. Sun and et. al. M2fNet: Multi-modal Forest Monitoring Network on Large-scale Virtual Dataset, 2024.
- [35] S. Chandak, V. Chittters, and S. Honnungar. Understanding the Amazon Rainforest from Space using CNNs. 2017.

- [36] A. Loh and K. Soo. Amazing Amazon: Detecting Deforestation in our Largest Rainforest. 2017.