CNN and Transformer-Based Segmentation to Classify Deforestation Drivers

Yasmine Mabene
Stanford University
ymabene@stanford.edu

Ayesha Khawaja
Stanford University
akhawaja@stanford.edu

Claire Morton
Stanford University
mortonc@stanford.edu

Abstract

We use deep learning to classify drivers of deforestation in Indonesia. Deforestation depletes carbon sinks, negatively impacts biodiversity, and is a global driver of climate change. Classifying drivers of deforestation automatically from satellite imagery would benefit policymaking and research into forest loss. Previous work has attempted to classify four drivers of deforestation in Indonesia. In this study, we use the same dataset and attempt to classify 12 drivers of deforestation. For this segmentation task, we use a UNet baseline, a modified ResNet-UNet model, and a transformer. We find that our baseline struggles to learn patterns in the data. Our modified ResNet improves performances across all classes, achieving test accuracy of 52%*, but still suffers from issues related to class imbalances. Our transformer model is unsuccessful at distinguishing between classes, likely due to a combination of issues in weighting the classes and a lack of hyperparameter tuning. Overall, this project shows that pretrained ResNet-UNet models can outperform transformer models for image classification problems, especially when there are many small classes in the training data.*

1. Introduction

Deforestation plays a central role in biodiversity loss and greenhouse gas emissions. Understanding the causes of deforestation enables the creation of targeted responses to protect forest habitats, an important step for mitigating climate change. In previous work, researchers have utilized machine learning to determine the extent to which sociodemographic and infrastructure characteristics impact deforestation [3] and to classify deforestation drivers in southeast Asia [16]. However, many of these models do not rely on high resolution images. To address this issue, researchers from the Stanford Machine Learning Group curated an existing dataset of labeled forest loss events in Indonesia [1] and developed a convolutional neural network called ForestNet to classify four broad deforestation drivers [11]. These drivers—plantation, small-holder agriculture, grassland/shrubland, and other—are generalizations of twelve specific deforestation drivers, and since finer-grained identification of deforestation causes enables more focused mitigation strategies, we build on the ForestNet implementation to identify all twelve in satellite images.

We implement two different classification models, a convolutional neural network and a transformer, to predict the drivers of forest loss given satellite images of forest loss events. We additionally compare the performances of the two models.

For our convolutional neural network models, we employ UNets, as described in Ronneberger et al. [17]. UNets are architectures for image segmentation that take in input data and contract it down to important features using convolutional layers. UNets then expand the data back to the final output size. Importantly, UNets are symmetric, which allows the expanding pathway to use information from the contracting pathway (through concatenations of the corresponding cropped feature map to the current upsampling layer) at each layer. UNets are appropriate for this task because their output is per-pixel classifications, so performing both down- and up-sampling allows UNets to extract important image features and retain the input size. Using UNets for semantic segmentation improves performance compared to other model alternatives and also leads to efficient training. Additionally, we employ UNets in combination with transfer learning [2] to further improve model performance as an extension to our baseline approach. To do so, we combine a ResNet backbone with the UNet architecture to take advantage of the information the pre-trained model already contains. In our approach, we input satellite images into our models and output classifications for each pixel in the image.

For our transformer, we implement the SegFormer architecture as described in Xie at al. [24]. SegFormer is a transformer framework pretrained on ImageNet-1K that utilizes a hierarchical encoder, overlapped patch merging, and a decoder of multilayer perceptron layers to achieve high efficiency and performance on semantic segmentation tasks. Hierarchical feature representation allows the transformer to process both coarse features and fine-grained image features; dividing images into overlapping patches allows the model to handle inputs efficiently without losing spatial information or long-range dependencies; the decoder is lightweight and not computationally demanding. SegFormer achieves state-of-the-art performance on common segmentation tasks.

1.1. Related Work

Pre-deep learning approaches to semantic segmentation include strategies like using Random Forests [18], an ensemble decision tree classifier that incorporates randomness, and attribute graph grammars [6], which use inference algorithms to decompose images into small primitives.

However, deep learning methods like CNNs and transformers are better equipped to process hierarchical representations of data, spatial information, and end-to-end learning, which prior methods lacked. Most state-of-the-art performances on public segmentation datasets are attributed to deep learning models [7]. On semantic segmentation tasks, vision transformers can sometimes surpass convolutional and recurrent approaches [13]. Transformer-based pipelines are often simpler yet stronger than CNNs because transformers utilize an attention mechanism to better attend to complex global context, which suits semantic segmentation tasks where multiple objects need to be identified and isolated in a broader image.

The original ForestNet paper framed the deforestation driver classification task as semantic segmentation to acknowledge that there can be multiple land uses in a single image and to better facilitate high resolution predictions [11]. They still designed their model to predict a single driver per image, but baseline classification models like k-nearest neighbor and logistic regression performed poorly compared to segmentation approaches. This is likely because the per-pixel classification in semantic segmentation allows for finer granularity in analyzing the nuanced land use in each image. ForestNet achieved 75% accuracy on this task, and the model consisted of a convolutional neural network with a Feature Pyramid Network architecture [14] and an EfficientNet-B2 backbone [21], scene data augmentation, and pretraining with a large land cover dataset.

We similarly formulate the task as semantic segmentation and develop a CNN with a UNet architecture [17] and a ResNet backbone [8]. We also finetune a pretrained transformer that uses the SegFormer archiecture [24]. Tzepkenlis et al. found that U-Net has slightly higher accuracy and precision than SegFormer on a land cover classification task for satellite imagery in Greece, but SegFormer has a slighly higher recall. Cleverly, researchers have tried to get the best of both worlds with U-TAE, a transformer integrated within a U-Net-like architecture that performs better on the Greece land classification task than both U-Net and SegFormer [5], [22]. We explore how these results extend to semantic segmentation for the latter two models.

In order to improve model performance across a large number of classes, we implement data augmentation methods evaluated in existing literature. In [19], Shorten summarizes commonly used data augmentation techniques and their considerations. Combinations of flipping, rotating, and cropping images have been found to increase model accuracy on CIFAR-10 by 3.5%. Translations have been shown to make models more robust to positional biases while also preserving the spatial dimensions. Additionally, random erasing, which randomly selects patches of data to mask, improves model performance when there may be occlusion in training data. ForestNet implements random cropping, affine transformations, artificial occlusion, and salt and pepper noise [11]. In our models, we experiment with randomly rotating and flipping images as well as with introducing random changes in brightness and contrast. These modifications may represent the natural variation within satellite imagery.

2. Methods

We implement a UNet CNN and a transformer. We considered implementing a recurrent neural network, but RNNs tend to be less effective than UNet or transformer approaches to semantic segmentation because they process inputs sequentially, rather than taking into account spatial information and long-range dependencies that can help in segmentation tasks [19].

2.1. UNet Architecture

For our convolutional network, we implement two different UNet architectures: a baseline and a modified UNet architecture that utilizes a 152-layer ResNet as the encoder. Our baseline UNet model consists of an encoder with the architecture [conv-relu-conv-relu-pool]x4 followed by [conv-relu]x2, and a decoder with the structure [transposed conv-conv-relu-conv-relu]x4 followed by a final convolutional layer to generate the final output. Our second method replaces the downsampling layers in the original UNet with the pre-trained ResNet model. This approach has been found to outperform traditional UNet models [4].

We train both the baseline and ResNet UNet models using a 58%/17%/24% train/validation/test split to match the ForestNet paper [11]. We use a batch size of 8 for training and a batch size of 1 to evaluate model performance on our validation set. To test our planned approaches and evaluate our baseline methods, we train for 5 epochs. Our loss function evaluates the cross-entropy loss for all pixels in the input image, weighted by the proportion of the class that appears in the training and validation datasets. Cross entropy loss is defined as the following:

$$L = -\sum_{c=1}^{C} w_c y_c \log\left(\frac{\exp(y_c)}{\sum_{i=1}^{C}\exp(x_i)}\right) \tag{1}$$

where x_i refers to the predicted score of the ith class, y_c is the score of the true class, C refers to the number of classes, and w_c is the weight of each class.

While training, we use the masks as the target labels and compute the loss using individual pixels in the entire input image and mask. To generate predictions, we set our predicted labels to be the maximum probability for each class over each pixel. We then obtain the class that appears the most frequently in the polygon region and generate a single driver prediction for the entire forest loss region.

After training our baseline and initial ResNet UNet model, we modify the ResNet model by implementing a new loss function and augmenting the training images. We replace our Cross Entropy Loss implementation with the Dice Loss [20] in order to better address class imbalances. The Dice Loss measures the similarity between the classification of predicted image and the true classification. The loss is defined as the following:

$$\text{Dice Loss} = 1 - \frac{2\sum_{i=1}^{N} p_i y_i}{\sum_{i=1}^{N} p_i + \sum_{i=1}^{N} y_i} \tag{2}$$

where p_i is the predicted class for pixel i, y_i is the true class for pixel i, and N is the total number of pixels in the image.

2.2. Transformer Architecture

We implement SegFormer-B0, a 3.7M-parameter transformer-based model first proposed by Xie et al. [24], using NVIDIA's mit-b0 model on Hugging Face. SegFormer-B0 has four transformer encoder blocks and uses a modified version of multi-head self-attention that reshapes the key vector according to a reduction ratio R, hard-coded in the original experiments as 64, 16, 4, and 1 for each respective transformer block, as seen in Figure 1. Equation 3 shows the original multi-head self-attention calculation, where Q, K, and V denote the query, key, and value heads respectively, and each have the dimensions $N \times C$. The modification involves resizing K using equations 4 and 5, which reduces the complexity of the mechanism from $O(N^2)$ to $O(\frac{N^2}{R})$.

$$\text{Attention(Q, K, V)} = \text{softmax}(\frac{QK^T}{\sqrt{d_{head}}})V \tag{3}$$

$$\hat{K} = \text{Reshape}(\frac{N}{R}, C \cdot R)(K) \tag{4}$$

$$K = \text{Linear}(C \cdot R, C)(\hat{K}) \tag{5}$$

SegFormer also foregoes positional encodings for Mix-FFN, a feed-forward network that uses a 3 × 3 convolution and MLP layer to provide positional information. The decoder consists of an MLP layer, upsampling, and two more MLP layers to predict segmentation masks.

The model uses the AdamW optimizer and a polynomial learning rate schedule that begins at 6×10^{-5} and is pretrained on ImageNet-1k. We use a training batch size of 8 and a validation batch size of 1 and finetune for ten epochs using cross entropy loss (see equation 1). To conserve memory, we train on 1,000 examples and evaluate on 300 examples.

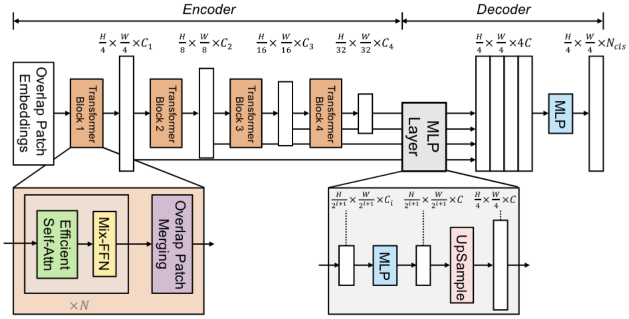
Encoder
Decoder
$\frac{H}{4}\times\frac{W}{4}\times C_1$
$\frac{H}{8}\times\frac{W}{8}\times C_2$
$\frac{H}{16}\times\frac{W}{16}\times C_3$
$\frac{H}{32}\times\frac{W}{32}\times C_4$
$\frac{H}{4}\times\frac{W}{4}\times 4C$
$\frac{H}{4}\times\frac{W}{4}\times N_{cls}$
Overlap Patch Embeddings
Transformer Block 1
Transformer Block 2
Transformer Block 3
Transformer Block 4
MLP Layer
MLP
Efficient Self-Attn
Mix-FFN
Overlap Patch Merging
$\times N$
$\frac{H}{2^{i+1}}\times\frac{W}{2^{i+1}}\times C_i$
$\frac{H}{2^{i+1}}\times\frac{W}{2^{i+1}}\times C$
$\frac{H}{4}\times\frac{W}{4}\times C$
MLP
UpSample

Figure 1. SegFormer architecture. Figure taken from the original paper [24].

3. Data

We use a satellite imagery dataset curated by the Stanford ML group [11] based on a previous study [1]. This dataset includes 2,756 332x332x3 satellite images of forest loss events in Indonesia with labels of the corresponding deforestation drivers. The Stanford ML group generated each image as composites of several satellite images of the same region at different times. We similarly preprocess the data to convert multiple images taken in the same area to a single median composite image to use for the classification task. There are files in the provided dataset that define regions within each image as forest loss regions. We convert each forest loss region into a polygon mask that defines where the forest loss event is located in the image and the corresponding driver that caused that loss. We adjust these labels to include all twelve drivers, rather than the original four merged drivers. For our SegFormer implementation, we use Hugging Face's AutoImageProcessor to normalize and resize our images from 332x332 to 512x512, since the model was pretrained on 512x512 ImageNet images.

We use twelve data labels to segment our images (Figure 2). The original ForestNet paper only classified loss in four broad categories: *plantation, grassland/shrubland, smallholder agriculture,* and *other* [11].

To implement data augmentions, we randomly flip each

Driver	Train Count	Val Count
Oil palm plantation	337	124
Timber plantation	231	69
Other large-scale plantations	118	25
Grassland shrubland	143	45
Small-scale agriculture	355	81
Small-scale mixed plantation	119	31
Small-scale oil palm plantation	82	28
Mining	48	21
Fish pond	24	7
Logging	39	5
Secondary forest	71	21
Other	49	16
Total	1616	473

Figure 2. The twelve deforestation drivers that we classify and their frequency in the ForestNet dataset.

training image with probability .5. We modify the brightness of the training images with probability .75 and rotate the images by up the 90 degrees. Depictions of these transformations can be seen in Figure 3.

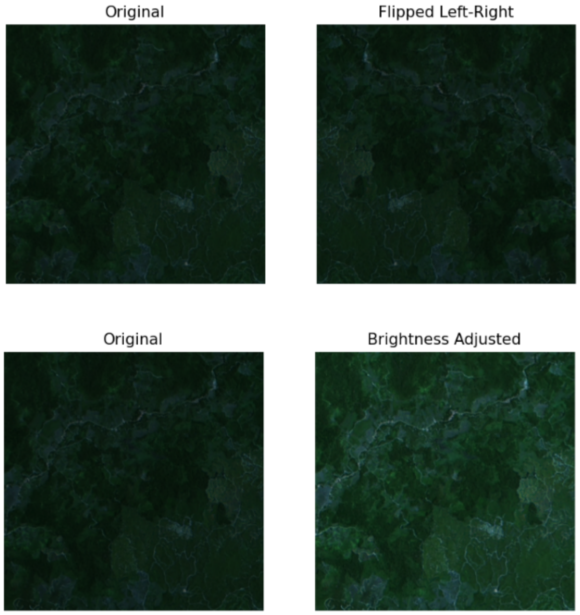

Figure 3. ResNet Image Transformations.

4. Experiments and Discussion

4.1. Baseline UNet Results

The baseline UNet model achieved a peak validation accuracy at 55%. However, throughout training, the loss, training accuracy, and validation accuracy largely remained the same (Figure **??**), which suggests that limited learning was occurring. This model took eight hours to train for five epochs on GPU. By contrast, the ResNet UNet model took half that time on CPU.

The baseline model was very successful with classifying the first driver of deforestation, ”Oil Palm Plantation” (with an accuracy of 99%) and the second driver, ”Timber Plantation” (with an accuracy of 52%). However, the model failed to classify any of the other drivers correctly. This could be because plantation classes are the most represented in our dataset, and the model got stuck predicting labels of $1s$ and $2s$.

Generally, we suspect that the baseline model is too simple to learn to distinguish properly between classes. The low training and validation accuracies indicate underfitting and plateau quickly. Therefore, we increase the model complexity by introducing the ResNet backbone.

When training the combined ResNet and UNet model on the input images and masks, we find that although our loss steadily decreased across the training epochs, the training accuracy did as well. In Figure 4, we see that the decrease in the training loss mimics that of the training accuracy. We hypothesize that this may be a result of inadequate weighting within our loss function. More advanced methods may be needed to address the class imbalances within our data. Additionally, these results indicate a need to alter the loss function to better fit our training objective.

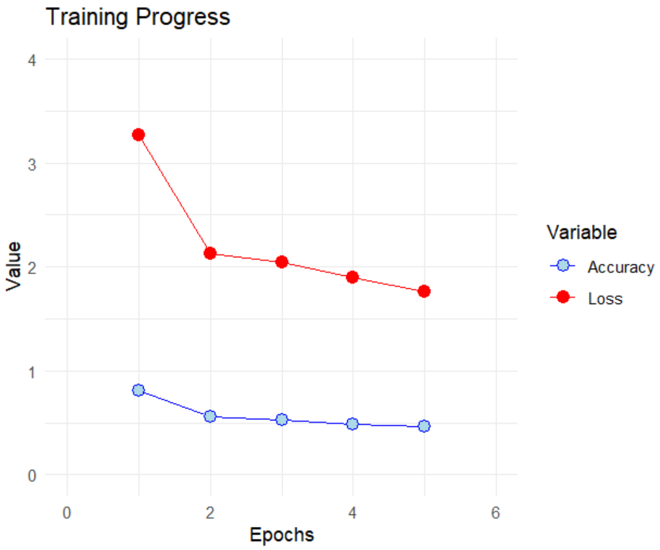

Figure 4. Train accuracy and loss for ResNet-UNet model.

The overall validation accuracy of the ResNet UNet model after training was 46%. While this is lower than the peak accuracy of our baseline, the ResNet UNet model was able to classify more diverse drivers. The proportion of correctly identified labels varied substantially across classes with “Oil Palm Plantation” and “Grassland Shrubland” classes having class recall values of 76.10% and 81.67%. “Timber plantations” were accurately classified 69.56% of time. However, “Small-Scale Agriculture” and “Mining” were only correctly classified around 24% of the time. All other classes were never correctly classified.

When examining the classification results using the four broader deforestation driver categories from [11], we see that the model performs the best on the plantation class, as indicated by the high precision and recall values (Figure 4). The model has the worst performance on the ”Other” category which is one of the least represented classes in the data. The ForestNet model similarly had the highest recall for the plantation class, though it had significantly higher precision and recall values for the “Agriculture” and “Other” classes than our approach, likely due to the more complex architecture and longer training times, as well as their use of covariates [11].

Interestingly, the ResNet UNet model has a high recall value for the grassland class (86.67%) but a low precision (26.53%). This means the model is successful at accurately classifying images that belong to the grassland class but at the same time is wrongly classifying images as “Grassland” that belong to other classes. In Figure 4, we see that the majority of images that are labeled as “Grassland” are actually “Agricultural” data. This may be due to a combination of our model struggling to distinguish between the two classes and because of the over representation of agricultural data.

4.2. Modified ResNet-UNet Results

Our modified ResNet-UNet architecture obtained a test accuracy of 52%, 6% higher than the UNet baseline. While the updated model still struggled to accurately classify deforestation drivers that appeared less frequently in the data, the model did produce predictions for the mining, fishing, and logging classes. These classes previously were never predicted in the baseline model. In Figure 5, we see that the modified ResNet-UNet results it slight reductions in the recall values for the broad plantation and grassland classes. At the same the precision for the two classes increases by approximately 7%. This is most likely because the model is now outputting more predictions belonging to the agriculture and other classes. Thus, of the predictions that belong that are plantation and grassland drivers, a higher proportion of them are correct compared to the baseline RestNet model. Both the precision and recall values from the broad agriculture and ”other” classes have substantial improvements from the baseline model including a 50% increase in the recall value for the ”Other” class. There is a small proportion of the test data and the data overall that belongs to this class. Thus, even small improvements in our model’s ability to predict this driver results in large gains in precision and recall.

Deforestation Drivers	Precision	Recall
Plantation	74.13	88.07
Grassland	26.53	86.67
Agriculture	53.33	22.86
Other	8.57	8.60

Figure 5. Validation set precision and recall on baseline ResNet-UNet model.

In 5, we see that while the majority of broad plantation drivers are classified correctly, the second most likely prediction for this class is agriculture. The probability distribution for the agriculture class is spread widely across several drivers, indicating difficultly in the model for classifying this driver. Although, improvements were made in the modified ResNet UNet for predicting classes that appear less frequently in the data, Figures 6 and 7 reveal there are still many drivers that are never predicted by the model. In future work, we may explore generating more examples from these classes to improve model performance.

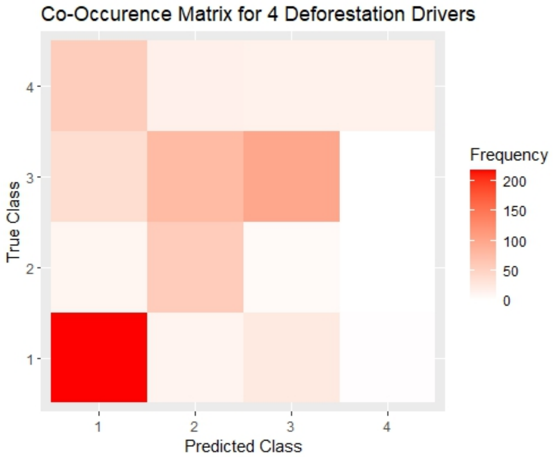

Figure 6. Modified ResNet Co-occurence Matrix with 4 Deforestation Drivers.

Our results emphasize the value of transfer learning and in incorporating pretrained weights into the UNet architecture in order to classify more drivers. We also believe the data augmentation may have made our ResNet UNet more robust and contributed to its improved performance on the smaller classes.

Additionally, our results reveal existing limitations within our approach. Class imbalance remains a challenge to our current architecture. In Figures 8 and 9, we see examples of ResNet UNet predictions next to the true forest loss regions for two drivers. Our model does well in identifying the forest loss region for the Oil Palm Plantation class, but struggles to do so for the mining region. We also recognize that while it is not evaluated, our model makes predictions for the entire image, including pixels outside of the forest loss region. Incorporating additional pretraining stages in our model framework to identify the location of forest loss

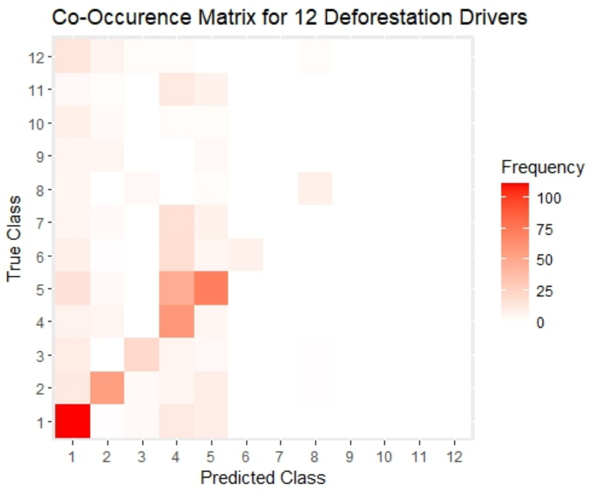

Figure 7. Modified ResNet Co-occurence matrix with 12 Deforestation Drivers.

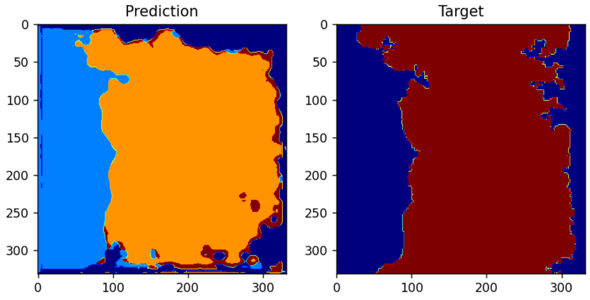

Figure 8. Predicted and Actual Deforestation Loss for Oil Palm Plantation Region.

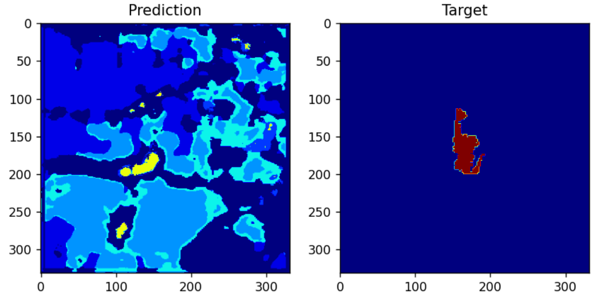

Figure 9. Predicted and Actual Deforestation Loss for Mining Region.

regions prior to classification may also improve model performance and is an avenue of future exploration.

Deforestation Drivers	Precision	Recall
Plantation	67.18	81.41
Grassland	33.14	76.62
Agriculture	68.50	47.00
Other	58.88	15.10

Figure 10. Test set precision and recall on updated ResNet-UNet model.

4.3. SegFormer Results

Our transformer model had trouble classifying the deforestation drivers. When incorporating data augmentation, the model performed better but showed had less changes throughout the training period. The accuracy for our SegFormer peaked during the first epoch at 0.526 and generally decreased after, and the loss decreased until the second epoch, where it hovered around the same values for the rest of training. Meanwhile, when we augmented the data, the accuracy stayed fixed at 0.531 and the loss began at 2.65 and decreased to 2.51 over ten epochs (see Figures 11 and 12). However, SegFormer did use significantly fewer computational resources, and training over ten epochs took less than three hours while training our ResNet UNet over five epochs took over six. We examined the classifications for the transformer trained on the non-augmented data using a t-SNE visualization, and we observed that some classes were generally grouped in the representation space while others were dispersed (see Figure 13). This reflects the poor performance we observed quantitatively.

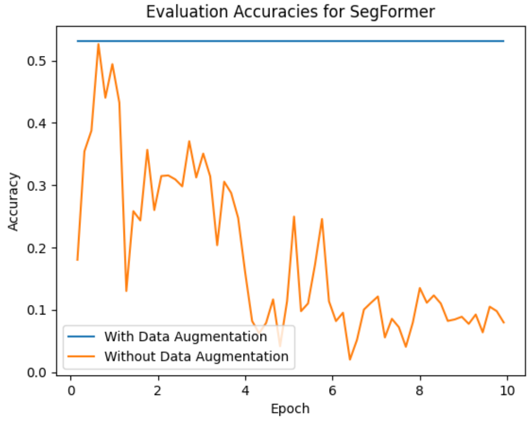

Figure 11. Evaluation accuracy over ten epochs for SegFormer.

5. Conclusion

Ultimately, we found that a UNet network with a ResNet backbone performed better than our SegFormer implementation. The SegFormer struggled to learn how to distinguish between drivers, but it operated faster, in line with existing literature about the superior efficiency of transformers compared to CNNs [13]. Data augmentation, modifying the

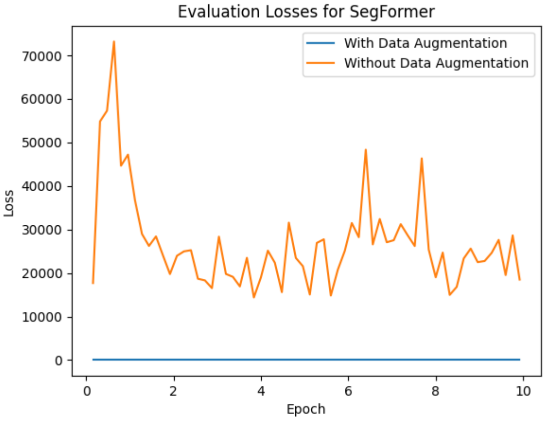

Figure 12. Evaluation loss over ten epochs for SegFormer.

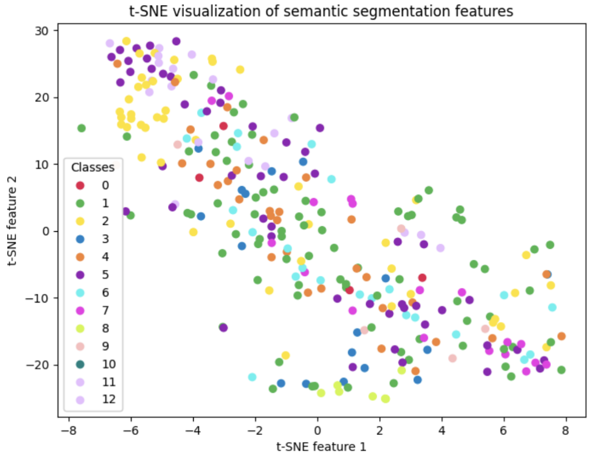

Figure 13. t-SNE visualization for SegFormer trained on non-augmented data.

loss function, and weighting classes improved model performance. Each model we implemented was able to classify only a handful of classes with consistent accuracy.

Given more time and computational power, we would train these models for more epochs and devote more time to tuning hyperparameters such as the learning rate. We also want to explore alternative weighting schemes. When we weighted all classes equally, our models learned that they could achieve high accuracy through never predicting classes that were less common. In response to this, we attempted several reweighting schemes and settled on weighting each class with the normalized inverse of the average proportion of times it appeared in the training and validation datasets. We note that the original ForestNet paper reported that they did not attempt to predict for the full 12-class dataset because of issues with too many small classes in the dataset [11]. We may have seen better model performance with an alternative weighting scheme, such as inverse square root weighting.

We considered several possible extensions to this project. Originally, we planned to use semi-supervised learning to pre-train a U-Net model on a pretext task. We would have used the pretext task of predicting the area of an image that is deforested (predicting the mask). We would pre-train on this pretext task in order to arrive at a neural network with some idea of the characteristics of pixels that mark them as deforested areas. We could check the success of this approach with a t-SNE visualization. Assuming this revealed that the model learned a useful representation space for the images, we would freeze the downsampling layers of the U-Net. Finally, we would train the U-Net on labeled input data. Generally, semi-supervised learning allows classification models to train more accurately than supervised learning models with less input data [12]. This would have been ideal for our setting, in which we were attempting to arrive at a model that performed well on many classes with few examples in our training data. Unfortunately, we were unable to implement this idea due to time constraints but would like to incorporate this method in future work.

References

[1] K. G. Austin, A. Schwantes, Y. Gu, and P. S. Kasibhatla. What causes deforestation in indonesia? *Environmental Research Letters*, 14(2):024007, 2019. 1, 3

[2] S. Bozinovski and A. Fulgosi. The influence of pattern similarity and transfer learning upon training of a base perceptron b2. In *Proceedings of Symposium Informatica*, volume 3, pages 121–126, 1976. 1

[3] R. R. Chowdhury. Driving forces of tropical deforestation: The role of remote sensing and spatial models. *Singapore Journal of Tropical Geography*, 27(1):82–101, 2006. 1

[4] Z. Fan, Y. Liu, M. Xia, J. Hou, F. Yan, and Q. Zang. Resat-unet: a u-shaped network using resnet and attention module for image segmentation of urban buildings. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 16:2094–2111, 2023. 2

[5] V. S. F. Garnot and L. Landrieu. Panoptic segmentation of satellite image time series with convolutional temporal attention networks. In *International Conference on Computer Vision*, 2021. 2

[6] F. Han and S.-C. Zhu. Bottom-up/top-down image parsing with attribute grammar. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31, 2008. 2

[7] S. Hao, Y. Zhou, and Y. Guo. A brief survey on semantic segmentation with deep learning. *NeuroCcomputing*, 406:302–321, 2020. 2

[8] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 2

[9] J. Howard and S. Gugger. Fastai: a layered api for deep learning. *Information*, 11(2):108, 2020. 9

[10] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in science & engineering*, 9(03):90–95, 2007. 9

[11] J. Irvin, H. Sheng, N. Ramachandran, S. Johnson-Yu, S. Zhou, K. Story, R. Rustowicz, C. Elsworth, K. Austin, and A. Y. Ng. Forestnet: Classifying drivers of deforestation in indonesia using deep learning on satellite imagery. *arXiv preprint arXiv:2011.05479*, 2020. 1, 2, 3, 5, 7

[12] F.-F. Li and E. Adeli. Self-supervised learning, 2024. 7

[13] X. Li, H. Ding, H. Yuan, W. Zhang, J. Pang, G. Cheng, K. Chen, Z. Liu, and C. C. Loy. Transformer-based visual segmentation: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 2, 6

[14] T.-Y. Lin, P. Dollár, R. Girshik, K. He, B. Hariharan, and S. Belongi. Feature pyramid networks for object detection. 2016. 2

[15] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library, 2019. 9

[16] D. R. Richards and D. A. Friess. Rates and drivers of mangrove deforestation in southeast asia, 2000–2012. *Proceedings of the National Academy of Sciences*, 113(2):344–349, 2016. 1

[17] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015. 1, 2

[18] F. Schroff, A. Criminisi, and A. Zisserman. Object class segmentation using random forests. In *Proceedings of the British Machine Vision Conference*, 2008. 2

[19] C. Shorten and T. M. Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019. 2

[20] C. H. Sudre, W. Li, T. Vercauteren, S. Ourselin, and M. Jorge Cardoso. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 14, Proceedings 3*, pages 240–248. Springer, 2017. 3

[21] M. Tan and Q. V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *International Conference on Machine Learning*, 2019. 2

[22] A. Tzepkenlis, K. Marthoglou, and N. Grammalidis. Efficient deep semantic segmentation for land cover classification using sentinel imagery. *Remote Sensing*, 2023. 2

[23] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, and A. M. Rush. Huggingface's transformers: State-of-the-art natural language processing, 2020. 9

[24] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo. Segformer: Simple and efficient design for semantic segmantation with transformers. *35th Conference on Neural Information Processing Systems*, 2021. 1, 2, 3

A. Appendix

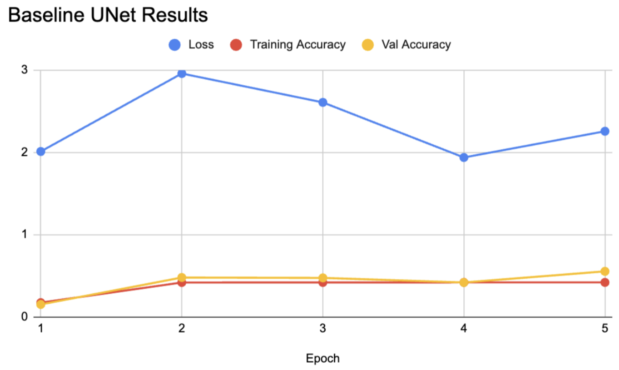

Figure 14. Accuracies and loss for baseline UNet model.

True by Predicted	Plantation	Grassland	Agriculture	Other
Plantation	192	9	16	1
Grassland	2	39	4	0
Agriculture	28	80	32	0
Other	37	19	8	6

Figure 15. Classification counts in validation set for baseline ResNet-UNet model.

B. Contributions Acknowledgements

Claire contributed to writing the paper, visualizing the transformer results, and coding the UNets. Ayesha contributed to implementing the baseline UNet and the SegFormer. Yasmine contributed to constructing the ResNet UNet model and evaluating results. All three members contributed to manuscript preparation.

We made use of several public GitHub repos:

`https://github.com/ngthanhtin/Deforestation_Segmentation/blob/master/inference.py`

`https://stanfordmlgroup.github.io/projects/forestnet/`

We also used the following libraries: matplotlib [10], Hugging Face [23], pytorch [15], and fastai [9].