Enhanced Agricultural Monitoring through Anomaly Detection in Farmland Images

Vital Ahishakiye
MSECE
Carnegie Mellon University-Africa
vahishak@andrew.cmu.edu

Vincent Nshimiyimana
MSECE
Carnegie Mellon University-Africa
vnshimi2@andrew.cmu.edu

Joseph Nkubito
MSIT
Carnegie Mellon University-Africa
jnkubito@andrew.cmu.edu

Shami Sion
MSIT
Carnegie Mellon University-Africa
ssion@andrew.cmu.edu

***Abstract*— Machine learning has undergone remarkable advancements since its inception in 1959, witnessing exceptional utilization across diverse domains in the 21st century. However, its integration with agriculture has lagged compared to fields like transportation, Entertainment, and education. Given agriculture's crucial role in sustaining livelihoods, addressing recurring yield production hindrances such as weed clusters and standing water is imperative. Previous efforts have explored integrating computer vision and deep learning into agriculture. For instance, Agriculture Vision curated a dataset comprising 21,061 images from 3,432 farmlands in the USA. These images, featuring RGB and NIR color channels, were utilized to detect six anomalies using a pretrained ImageNet model with ResNet 101 as a backbone. In this work, we process, train, evaluate, and test these images using a U-Net model with combined RGB and NIR channels. Our project aims predict anomalies images considering a light, fast and efficient model. Agriculture Vision achieved an Intersection over Union (IoU) of 43.66% in 2020, and Hyunseong led the leaderboard in 2021 with 63.9% IoU in the same annual challenge. Our experiments yielded a promising IoU of 31.99%, utilizing only one-third of the training dataset on U-net model.**

***Keywords*—RGB (Red, Green, Blue), NIR (Near Infrared Images)**

I. Introduction

The Agriculture Vision dataset has unlocked numerous opportunities for deep learning enthusiasts aiming to contribute to the application of computer vision in agriculture. The shared objective of this dataset was to provide a comprehensive array of images to facilitate enhanced learning by models. While some approaches, such as combining hybrid CNN-LSTM architectures [1] or assembling multiple CNN models [2], have shown promising results, they didn't show Intersection over Union (IoU) scores for the entire model framework. Additionally, these frameworks tend to be complex and computationally intensive. A lightweight and efficient CNN model presents a viable solution for monitoring farmland conditions to aid decision-making processes in a shorter time frame and with fewer computational resources. However, existing approaches, such as those employed by Agriculture Vision [3], still present challenges. For instance, training each model for 25,000 iterations with a batch size of 40 on four RTX 2080Ti GPUs remains resource-intensive and may not be feasible for all practitioners in the field.

II. Literature review

Several studies have explored approaches using the Agriculture Vision dataset, with some focusing on methods that did not incorporate NIR/RGB fusion. In contrast, Agriculture Vision made minor modifications to existing DeepLabV3 and DeepLabV3+ architectures to accommodate NRGB images [4]. Notably, in scenes where RGB and NIR data fusion was crucial, methodologies from scene classification were adopted [5]. These approaches aimed to address data loss in images and mitigate discrepancies between RGB and NIR images over long ranges. LRINet, [6] for example, successfully produced natural-looking color images with clear details, effectively handling the differences between RGB and NIR inputs. Additionally, Perceptual Image Fusion [7] techniques were employed to enhance image visibility, providing valuable insights into data processing methodologies for achieving desired outcomes.

III. Methodology

With the primary objective of developing a lightweight, fast, and efficient model, our approach began with the processing of images by combining Near-Infrared (NIR) and RGB images. Subsequently, we explored the utilization of the U-Net architecture. U-Net was selected due to its capability to employ a novel loss weighting scheme for each pixel, assigning higher weights at the borders of segmented objects. Renowned for its flexibility and optimized modular design, U-Net aligns closely with our main objective of efficiency [8] [9]. We then proceeded with hyperparameter tuning to optimize the model's performance and achieve the anticipated results.

A. Image processing

To prepare the dataset for model training and evaluation, a series of data processing steps were undertaken. The dataset consists of three subsets: Train, Validation, and Test. Each subset includes directories containing RGB images, Near-Infrared (NIR) images, masks, boundary masks, and labels for training and validation. RGB images were loaded in color, while NIR images, masks, and boundary masks were loaded as grayscale images. Upon loading, each image underwent normalization to ensure consistent pixel intensity ranges. Semantic segmentation masks and boundary masks were applied to the combined RGB-NIR images using bitwise operations. This process facilitated the extraction of relevant features from the images while preserving spatial information outlined by the masks. RGB and NIR images were then combined into a single 4-channel image, allowing the model to leverage both spectral bands for improved feature extraction. This combined image served as the input to the subsequent model architecture. Processed images were saved to specified output directories for both the training and validation datasets. Each processed image was assigned a unique filename corresponding to its original input, ensuring traceability and data integrity.

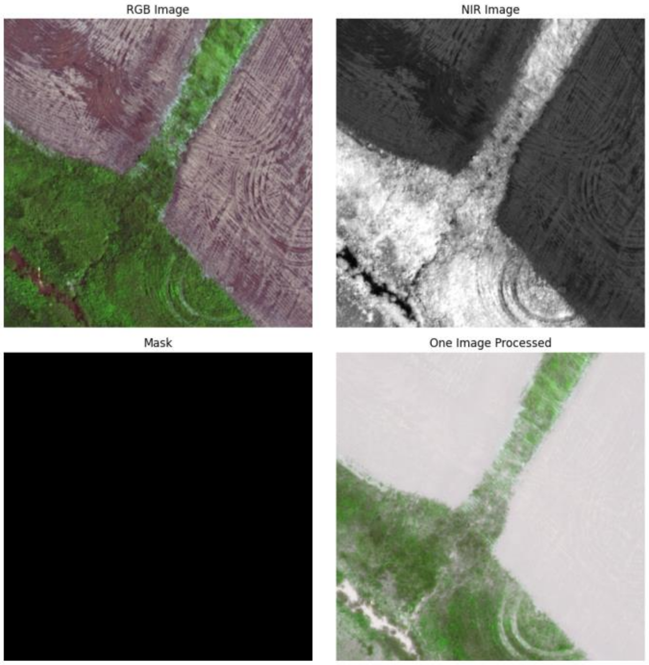


Figure 1 A sample of a processed image, Image with heading (One Image Processed) in a 4channel image.

B. Model Architecture

The U-Net neural network architecture utilized in this model takes a 4-channel input with dimensions (512, 512, 4), representing a combined RGB-NIR image. The architecture consists of three main parts: Encoder Blocks, Bridge, and Decoder Blocks. In the Encoder Blocks, four convolutional blocks (down_block) are employed and a dropout of 30%, each followed by max-pooling layers. These blocks comprise two convolutional layers with batch normalization and ReLU activation functions, progressively increasing the number of filters to capture hierarchical features. The Bridge connects the encoder and decoder, consisting of two convolutional layers with 1024 filters each, followed by batch normalization and ReLU activation. This bridge layer helps retain high-level spatial information crucial for accurate segmentation. The Decoder Blocks mirror the encoder blocks, with four upsampling blocks (up_block) performing upsampling followed by concatenation with the corresponding feature map from the encoder. This enables the model to recover spatial information lost during downsampling. Each decoder block includes two convolutional layers with batch normalization and ReLU activation. The final layer of the model is a 1x1 convolutional layer with softmax activation, producing an output tensor with dimensions (512, 512, 7), representing the segmentation mask for each of the seven target classes. The model is compiled using the categorical cross-entropy loss function and the Adam optimizer.

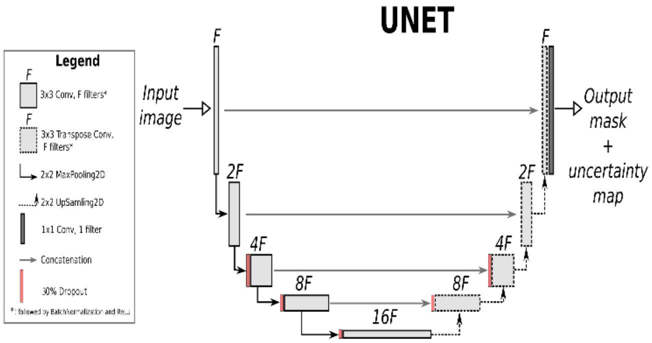
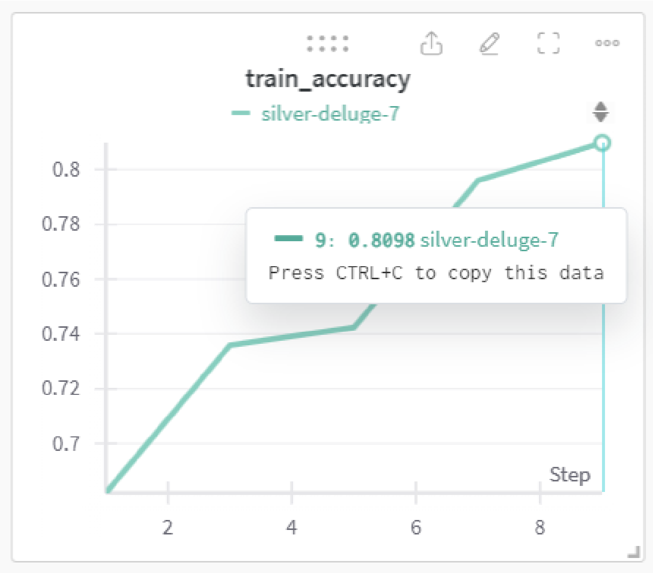


Figure 2 U-net Model architecture [10]

C. Training

During the training process, the U-Net model is initialized with its pre-saved weights and bias obtained from the best-performing model trained previously. Subsequently, the input batch of RGB-NIR images is propagated through the model to generate predictions, which are compared against the corresponding ground truth masks to compute the loss. Through backpropagation, gradients of the loss with respect to the model parameters are computed, facilitating parameter updates aimed at minimizing the loss function. Following the processing of all batches in the training dataset, the model's performance is assessed on a separate validation dataset to monitor generalization and detect overfitting. Limiting the training dataset to one-third and reducing the size of the test and validation datasets to one-fourth was a strategic decision made in response to the observed performance and computational constraints. The decision was prompted by the model's poor accuracy of 0.14 on the training set and 0.06 on the validation set when trained on the entire dataset. Additionally, the iteration time of approximately 48 minutes per epoch was deemed impractical, total images were 12901 for train and 4431 for validation.



9: 0.8098 silver-deluge-7

Press CTRL+C to copy this data

Figure 3 Training accuracy of U-net model used on a scaled dataset [11].

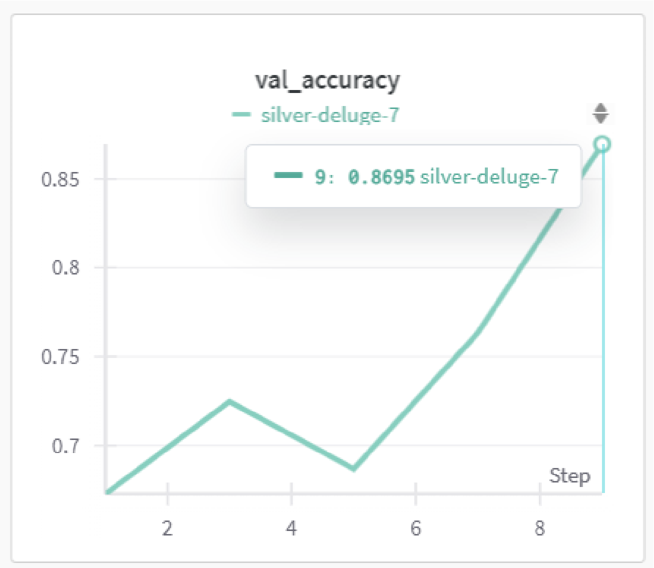
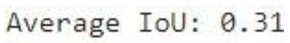


Figure 4 Validation accuracy of U-net model

D. Testing

In the testing phase, our model's efficacy was assessed on a dedicated test dataset comprising RGB, NIR, mask, and boundary images. After preprocessing as it was done on train and test datasets, these images were aggregated into batches for evaluation. Utilizing the best model saved, predictions were generated for each batch in the test dataset. Subsequently, these predictions were compared with the corresponding ground truth masks to compute essential evaluation metrics such as Intersection over Union (IoU) calculated for every sample within the batch and plot three samples, total images were 3729.



```
Average IoU: 0.31
```

Figure 5 The best model has the average IoU.

scores, the insight resulted suggests that approximately one in three of the produced images are accurately predicted by the model.

IV. Discussion and Future Work

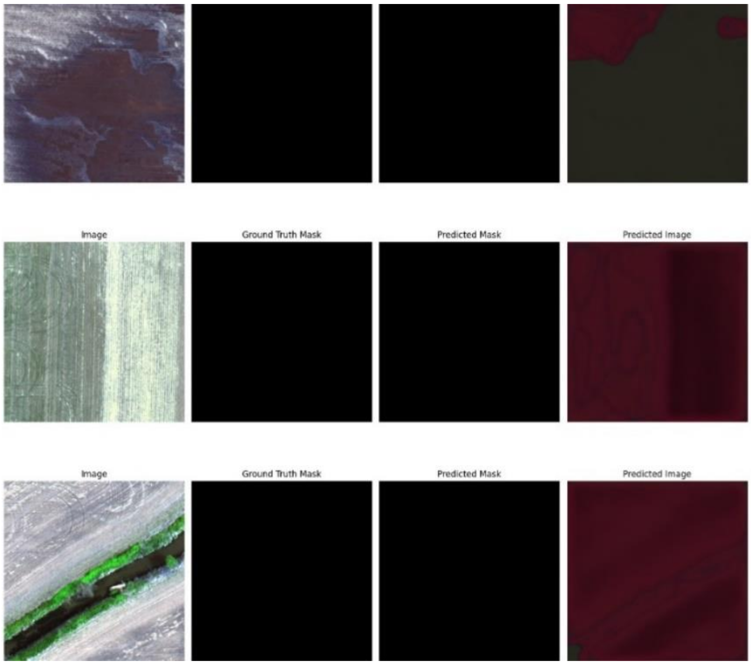
In the process outlined above, we initially attempted training on the full dataset, but encountered poor accuracy metrics. Due to the impractically long duration of each iteration, we strategically downscaled the datasets to a 1:6 ratio, yet still observed poor accuracy. Further adjustments brought the datasets to a 1:4 ratio, with unsuccessful outcomes. Finally, we adopted a different scaling approach, which proved effective. Throughout these iterations, we saved the best-performing models, characterized by lower validation loss, for future use, allowing us to resume training without starting from scratch. Notably, our focus was not on label prediction in this task, but this could be an exciting exploration in future endeavors.

V. Conclusion

Initially, we sought models trained on 4-channel images but found none available. Instead, others commonly modified models trained on 3 channels to accommodate 4 channels at the input. For instance, in Agriculture Vision, minor adjustments were made to existing architectures like DeepLabV3 and DeepLabV3+ by duplicating weights corresponding to the red channel of the pretrained convolution layer, resulting in a convolution layer with four input channels in the backbone. Our focus then shifted to developing the U-Net model trained on 4-channel images. With enough iterations and when used on the full dataset, this model could become a promising baseline for future advancements in farmland anomalies detection.

The notebook that contains the work done can be found here: https://drive.google.com/file/d/1opSRp9lv3FNq9pmxZAmMGgkr0ZiMWm47/view?usp=sharing

Figure 6 Sample images predictions from testing



REFERENCES

[1] 'Full Text PDF'. Accessed: May 02, 2024. [Online]. Available: https://www.researchgate.net/journal/BIO-Web-of-Conferences-2117-4458/publication/377112019_CNN-LSTM_framework_to_automatically_detect_anomalies_in_farmland_using_aerial_images_from_UAVs/links/6595eb1b3c472d2e8eb08a36/CNN-LSTM-framework-to-automatically-detect-anomalies-in-farmland-using-aerial-images-from-UAVs.pdf

[2] 'farmland-anomalies/evaluation at master · amogh7joshi/farmland-anomalies', GitHub. Accessed: May 02, 2024. [Online]. Available: https://github.com/amogh7joshi/farmland-anomalies/tree/master/evaluation

[3] 'Chiu et al. - 2020 - Agriculture-Vision A Large Aerial Image Database .pdf'. Accessed: May 02, 2024. [Online]. Available: https://arxiv.org/pdf/2001.01306

[4] 'Chiu et al. - 2020 - Agriculture-Vision A Large Aerial Image Database .pdf'. Accessed: May 02, 2024. [Online]. Available: https://arxiv.org/pdf/2001.01306

[5] R. Soroush and Y. Baleghi, 'NIR/RGB image fusion for scene classification using deep neural networks', *Vis. Comput.*, vol. 39, no. 7, pp. 2725–2739, Jul. 2023, doi: 10.1007/s00371-022-02488-0.

[6] L. Liu, F. Wang, and C. Jung, 'LRINet: Long-range imaging using multispectral fusion of RGB and NIR images', *Inf. Fusion*, vol. 92, pp. 177–189, Apr. 2023, doi: 10.1016/j.inffus.2022.11.020.

[7] J. Lee, G. Oh, and B. Jeon, 'Perceptual Image Fusion Technique of RGB and NIR Images', in *2019 34th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*, Jun. 2019, pp. 1–4. doi: 10.1109/ITC-CSCC.2019.8793414.

[8] H. Sankesara, 'U-Net', Medium. Accessed: May 02, 2024. [Online]. Available: https://towardsdatascience.com/u-net-b229b32b4a71

[9] W. Baccouch, S. Oueslati, B. Solaiman, and S. Labidi, 'A comparative study of CNN and U-Net performance for automatic segmentation of medical images: application to cardiac MRI', *Procedia Comput. Sci.*, vol. 219, pp. 1089–1096, Jan. 2023, doi: 10.1016/j.procs.2023.01.388.

[10] '42: A blog on A.I.' Accessed: May 02, 2024. [Online]. Available: https://nchlis.github.io/2019_10_30/page.html

[11] 'Weights & Biases', W&B. Accessed: May 03, 2024. [Online]. Available: https://wandb.ai/site