Proof of Concept: RGB to Multispectral Reconstruction using Transformer-Based Spectral Conditioning

Earl Ranario, Shreya Kulkarni, and Shreya Maddhali

University of California, Davis

Abstract

RGB imaging is widely used but needs more depth of information than multispectral cameras provide in capturing critical infrared data that can be used to analyze plant health and other factors. Because of how expensive these multispectral cameras are, our motivation in developing this project is to create a cost-effective implementation utilizing artificial intelligence and machine learning. We used a dataset that includes RGB images with a resolution of 750×750 pixels, along with monochrome spectral images in red (660 nm), green (550 nm), red-edge (735 nm), and near-infrared (790 nm), each at a resolution of 416×416 pixels. We found that our method, with and without a signal condition, outperforms other baseline generative methods. Additionally, our method has the potential to generate more than four bands, using a signal as a condition during the decoding process.

1 Introduction, Motivation and Problem

Multispectral cameras have specialized sensors that detect radiation at different wavelengths; they can capture images in multiple electromagnetic spectrum bands. These cameras are beneficial but expensive and inaccessible in medical imaging and agriculture fields. RGB imaging is widely used but needs more depth of information than multispectral cameras provide in capturing critical infrared data that can be used to analyze plant health and other factors. Because of how expensive these multispectral cameras are, our motivation in developing this project is to create a cost-effective implementation utilizing artificial intelligence and machine learning. Extracting information from multispectral images without purchasing the required camera allows us to extract vital details while maintaining cost efficiency. Our approach focuses on generating multispectral images from standard RGB formats, which will then create more accessibility to these valuable insights and potentially transform how this data can be analyzed. We are planning to utilize a U-Net architecture, as well as PyTorch, to generate multispectral images in a simple RGB format.

2 Dataset

The dataset consisted of of aerial agricultural images of a potato field, with manually labeled regions from annotations in a CSV file that indicated healthy and stressed plants; these images were collected by a research team from the University of Idaho [2]. The dataset includes RGB images with a resolution of 750×750 pixels, along with monochrome spectral images in red (660 nm), green (550 nm), red-edge (735 nm), and near-infrared (790 nm), each at a resolution of 416×416 pixels. Annotated bounding boxes for healthy and stressed crop areas are provided in accompanying XML files. Due to differences in the sensor outputs, image alignment was be required to avoid issues during model training. To align the images, we defined a method that used SIFT keypoints and RANSAC to filter out good matches between a base image and image to align, estimate the homography between the source and destination points, and finally warp the image that we wanted to align before returning it. We cropped the images by 80 percent after aligning them to solve the issue of blank space between the crop images and edges of the box messing with the training model. We loaded in the images, split them into training, validation, and testing datasets, sorted them, and then used these datasets to train and test the model.

3 Method

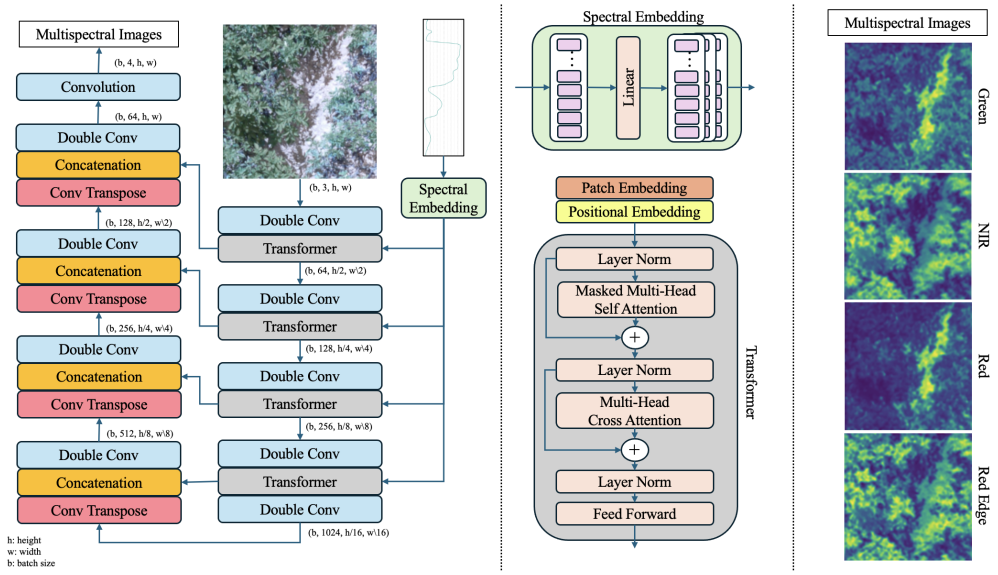

Figure 1: This approach incorporates transformer blocks and a hyperspectral signal, which is embedded and conditioned with the RGB image to highlight important features for spectral image reconstruction. Masked self-attention within the transformer blocks focuses the signal on the object of interest, with skip connections aiding in the upsampling process.

The overall model was motivated by UNet's [1] architecture. UNet is composed of an encoding and decoding stage which first compresses the input image into a high-dimensional feature space: latent space. Then, it follows with a series of decoding stages supported by concatenation from previous encoding layers. This allows for the model to "remember" fine-grained features to aid the reconstruction process. In our approach, as shown in Figure 1, we first decode the input RGB image with double convolutions and transformer [3] blocks. Additionally, we input a hyperspectral signal which goes through a spectral embedding and acts as a condition with the RGB image. This allows the deconstruction process to utilize information from this signal to highlight important features needed for reconstructing the various spectral images. Each transformer block contains a masked self-attention head which forces the signal to attend to only the object of focus. Following that, the resulting attention map computes with the spectral embedding to further highlight important spectral signals for the reconstruction process. Skip connections were included from the transformer to the upsampling process to help remember outputs from the cross-attention.

3.1 Spectral Embedding

The hyperspectral signal which can contain up to 2000 bands, is split into a fixed length of tokens. For instance, one token can contain 40 bands. This would result in each a tokenized embedding of shape length 40 where each token has 50 dimensions. Afterward, this embedding is projected to a higher dimension using a linear layer.

3.2 Spectral Conditioning via Masked Self-Attention and Cross-Attention

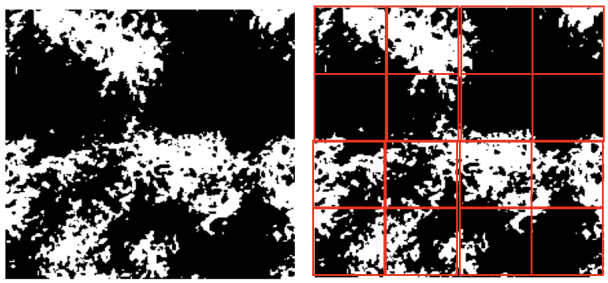

Figure 2: Segmentation process of the RGB input image along with patch split example

A transformer block consists of both self-attention and cross-attention. However, in the self-attention process, a masked parameter is included which was derived from the input RGB image. Masking out specific attention scores allows for the plant to only attend to the input signal during the cross-attention. An HSV segmentation was applied to the RGB image. Then, based on the patch size for the input feature map into the transformer block, an additional patched masked was created to parameterize the attention process, as seen in Figure 3.

4 Results and Discussion

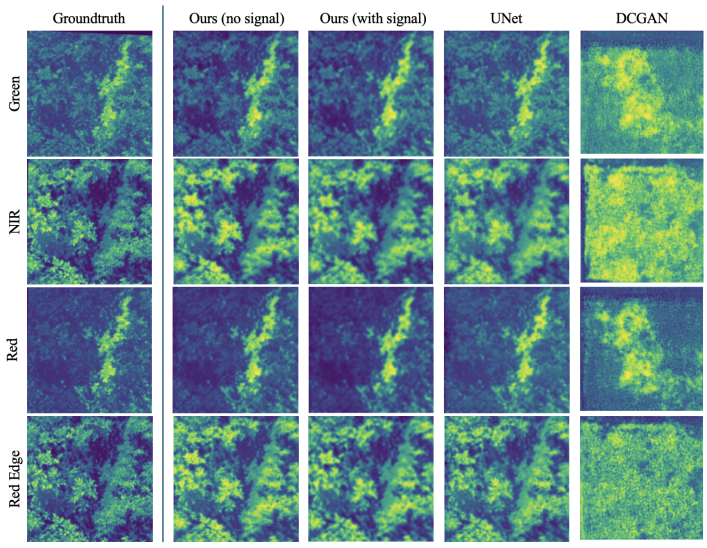

Figure 3: Outputs of each generative method.

As seen in Table 1, our method, with and without the signal, outperforms other baseline generative methods. Given the dataset used, our signal may not have contributed much to the reconstruction process due to the limited groundtruth data available. Essentially, our current setup is only reconstructing NIR images since the other channels are RGB. Therefore, the only spectral image that the model is learning is NIR. A potential improvement is to train the model with the same setup but have a larger groundtruth dataset that contains more than 5 channels the model can learn from. This will allow us to further evaluate our proposed method.

Table 1: Comparison of Methods Across Train, Validation, and Test Sets

	Train			**Validation**			**Test**		
Methods	MSE	SSIM	Euclidean	MSE	SSIM	Euclidean	MSE	SSIM	Euclidean
UNet	0.010	0.677	1.383	0.012	0.676	1.447	0.013	0.642	1.518
VAE	0.038	0.193	5.138	0.037	0.186	5.132	0.039	0.169	5.408
DCGAN	-	0.090	4.176	-	0.085	4.219	-	0.081	4.412
Ours (No Signal)	0.010	0.710	1.368	0.012	0.716	1.408	**0.012**	**0.677**	**1.471**
Ours (With Signal)	0.010	0.652	1.441	0.013	0.658	1.525	**0.012**	**0.640**	**1.557**

Another approach that could be done is to obtain a proximal hyperspectral image dataset where we compress each hyperspectral image into a zero-dimension spectral signal as our condition and optimize on the other channels. However, this can be computationally expensive and may requires some degree of dimensionality reduction on the groundtruth channels.

Contributions:

Earl: proposed idea for project, developed code repository and completed tests/validation

Shreya K: completed datasets portion and added validation method

Shreya M: completed introduction, motivation, problem, method compiled audio/visual presentation

References

[1] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

[2] K. Duellman H. Wang S. Butte, A. Vakanski and A. Mirkouei. Potato crop stress identification in aerial images using deep learning-based object detection, 2021.

[3] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.