Improving Semantic Segmentation through Task Adaptation for UAV Hyperspectral Agricultural Imagery

Mazharul Hossain[a], Aaron Robinson[b], Lan Wang[a], and Chrysanthe Preza[b]

[a]Computer Science Department
[b]Electrical and Computer Engineering Department
The University of Memphis, Memphis, TN, USA

ABSTRACT

Accurate crop mapping—identifying both the location and types of crops—is crucial for effective agricultural planning and informed decision-making. Advances in remote sensing, notably hyperspectral imagery from unmanned aerial vehicles (UAV), greatly enhance the efficiency and accuracy of crop mapping, reducing the reliance on traditional, labor-intensive field surveys. However, applying deep classifiers directly to hyperspectral data can lead to overfitting. Conversely, deep semantic segmentation models may struggle due to limited annotated hyperspectral imagery. To address this problem, we propose enhancing a U-Net-style model—originally trained on RGB imagery—by incorporating task adaptation, a custom loss function, and a spectral attention mechanism to better optimize it for hyperspectral data and improve crop mapping performance. Our proposed segmentation network achieved 76.6% accuracy and a 74.9% Dice score on a UAV-acquired hyperspectral agricultural dataset, which is competitive and well-rounded compared to other state-of-the-art methods while offering significantly lower computational complexity.

Keywords: hyperspectral remote sensing, semantic segmentation, near-infrared NIR, unmanned aerial vehicles UAV, deep learning, crop classification, U-Net

1. INTRODUCTION

Crop mapping identifies the types of crops grown in a specific area and their spatial distribution (as shown in Figure 1), which facilitates important agricultural activities such as assessing plant health and estimating crop production[1,2] and helps farmers and agricultural experts make better decisions. Recently, remote sensing technology is replacing traditional, labor-intensive, on-the-ground surveys for gathering data for crop mapping.[3] In particular, hyperspectral imagery gathered via unmanned aerial vehicles (UAV) provides much more spectral information than traditional RGB imagery, which could lead to more accurate crop mapping. However, using AI to analyze hyperspectral images presents challenges. One straightforward approach to crop mapping is to use a **deep classifier** to process the hyperspectral imagery, but the classifier may overfit due to numerous training parameters and limited data points.[4] Moreover, the inference task on a scene with this approach is slow with a quadratic time complexity as it classifies individual pixels in an image. In contrast, crop mapping using **semantic segmentation** has the advantage of classifying all pixels simultaneously, which is much faster than deep classifiers. However, semantic segmentation requires detailed spatial understanding and training datasets where each pixel is annotated, but there are few annotated hyperspectral agricultural datasets. In this work, we aim to improve the accuracy of semantic segmentation on hyperspectral images for crop mapping, given that computation resources may be limited in an agricultural setting and, therefore, a more efficient approach is generally preferred.

There are several high-performing deep convolutional neural networks for semantic segmentation available such as U-Net,[6] FastFCN,[7] and DeepLab[8] as well as Transformer based models such as Segmenter[9] and ViT (Vision Transformer).[10] Based on the fully convolutional network (FCN), Ronneberger et al.[6] proposed the U-Net architecture for semantic segmentation. It comprises four components: an encoder (contracting path) for feature abstraction, an encoder (bottleneck) for capturing the most critical features and a point of convergence for the

Further author information: (Send correspondence to Mazharul Hossain); E-mail: mhssain9@memphis.edu

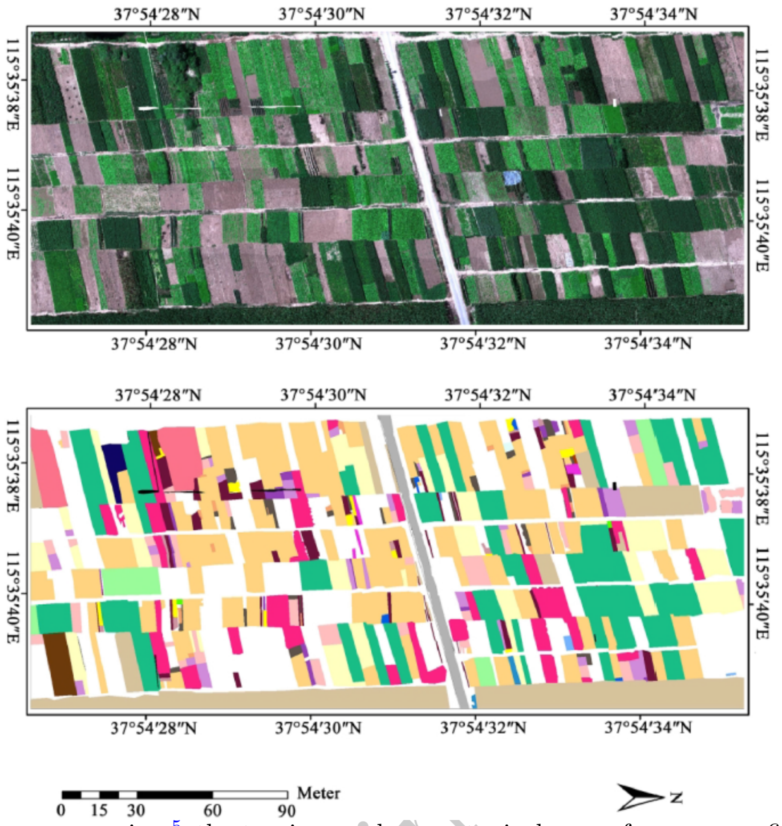

Figure 1: Example of crop mapping:[5] the top image shows a typical scene from a crop field, while the bottom image shows the corresponding crop map where individual crops are specified in different colors.

extracted features, a decoder (up-sampling expansive path) for feature reconstruction, and a final convolutional layer for classification. U-Net is a versatile segmentation network capable of working with different data types, such as medical images, satellite images, and natural scenes. It is also easier to modify its architecture.[11,12] Multiple researchers have included transformers,[5,13] gated attention,[14,15] and attention networks[16,17] in U-Net to improve its performance. We also chose U-Net because it is simpler to prototype and demonstrate proof of concept in this architecture.

Most agricultural segmentation solutions use RGB imagery[15] as most off-the-shelf UAVs come with an RGB camera. However, RGB images miss out on additional features a plant emits in near-infrared (NIR),[18] which can help differentiate crops. Compared to RGB and satellite images, hyperspectral (HS) imagery gathered via UAVs strikes a remarkable balance by providing good spatial resolution and rich spectral data.[5,19] HS Image Semantic Segmentation (SS) utilizes spectral and spatial information to solve this semantic segmentation problem. However, we cannot directly use most of these AI models as they are designed for regular RGB images and pre-trained with RGB images, which have only three color channels, while hyperspectral images contain many more. Moreover, if we want to design and train a model from scratch, we need a large HS image dataset, which is hard to obtain. Thus, these deep neural networks (DNN) may not achieve their full potential.[20] This problem is of significant importance in the field of agricultural technology. In response, we propose tapping into task adaptation's[21] potential through transfer learning[22,23] and fine-tuning. Through this technique, an AI model trained on a large RGB dataset is adapted for a new type of task and data (HS images) and solves this problem for HS crop mapping.

We apply transfer learning to TransUNet,[13] which contains a ViT model pre-trained with RGB images. Transfer learning is possible here as many DNNs trained on natural images tend to learn similar Gabor filters and color blobs in the first layer. These features are generic and valuable across various datasets and tasks.[24]

Hyperspectral images require adjustments to the initial input layers of the model to accommodate them and fine-tune the input while freezing others. However, this can limit performance because middle layers expect RGB features. Alternatively, end-to-end fine-tuning of deep learning (DL) models may compromise pre-trained model's general knowledge. Instead of designing a new model, we want to investigate if a dynamically adapted pre-trained model can overcome these challenges. Thus, we kept adjustments as minimal as possible. By altering the input layer with an appropriate number of channels and adjusting only the immediate next layer of the model, we enable it to process hyperspectral images while keeping most of its pre-trained knowledge intact.

The network's hidden layers, originally designed to capture features from three input channels, may now have a similar number of channels as the input HS image. Attempting to extract features from these hundreds of input channels without increasing the shape of these hidden layers can create a bottleneck in processing. However, increasing the shape can, in return, increase the model's complexity and computational cost, along with the loss of more pre-trained knowledge. Thus, we propose a solution using the **channel attention mechanism** to effectively utilize the model's limited number of feature channels. This mechanism helps the model extract crucial spectral information from HS images, helping the model to learn to focus on specific feature channels with vital spectral details, further improving accuracy without requiring excessive computing power.

Our training and development involves cross-domain transfer and adapting to the new task. We modified the input layer to handle different input modalities (HS imagery), and replaced the last classification layer to match the number of target classes. We present the detailed procedure in Section 3.8. By leveraging task adaptation (i.e., transfer learning and fine-tuning), custom loss and a spectral attention mechanism, we successfully transfer the domain of TransUNet from medical to agricultural HS imaging and improve the accuracy of segmenting crops.[25] Our evaluation investigated the model's performance on a UAV hyperspectral agricultural imagery dataset[5] and achieved 76.6% accuracy and a 74.9% dice score, which is competitive and well-rounded compared to other state-of-the-art methods while offering significantly lower computational complexity.

2. BACKGROUND AND RELATED WORK

Semantic Segmentation (SS) is a computer vision technique that classifies each pixel in an image into its semantic categories using a DL (deep learning) algorithm. In applications that require precise object localization and boundary delineation—such as military surveillance—SS identifies and classifies every pixel within an image. It allows us to understand a scene better by distinguishing various objects at a granular level, which is also highly valuable in other fields, including autonomous driving, agricultural monitoring, industrial inspection, and medical imaging. In agriculture, SS assists with crop classification, weed detection, and disease identification. It enables stakeholders to differentiate between various crops and identify unhealthy plants at a finer level.

Recent studies have examined various methods for agricultural image segmentation. For example, Luo et al.[26] reviewed recent advancements in conventional and DL approaches for segmenting agricultural images, focusing on crop analysis and pest identification. Our research builds upon TransUNet,[13] a model initially developed for medical imaging that we modified for agricultural use. TransUNet replaced the encoder bottleneck layer of UNet with ViTs. We specifically improved their encoder contracting path and bottleneck of the network for better gradient flow, ensuring the model learns effectively from hyperspectral images. Compared to TransUNet, Isensee et al.[27] argue for using vanilla U-Net and developed nnU-Net (no new net) for medical imaging. This adaptive framework can automatically configure itself for various datasets and segmentation tasks. We utilized their open-source framework and configurations as the foundation for our model development. HSI-TransUNet[5] provided the UAV-HSI-Crop Dataset, introduced the attention module[15] in TransUNet, and tested it on this new dataset in an agricultural context. Unlike their work, we designed separate attention and residual pathways. In contrast to TransUNet, Liu et al.[15] proposed a multiscale global attention module (MGA) instead of multi-head attention at a single scale within the encoder bottleneck layer of U-Net to perform semantic segmentation of agricultural RGB images.

When there is insufficient data or a change in dataset distribution, transfer learning solves this problem by using a pre-trained model as a starting point and leveraging knowledge from an initial task for performance improvement. Karimi et al.[21] presented a critical assessment of the role of transfer learning in training fully convolutional networks (FCNs) for medical image segmentation and found transfer learning improved results significantly when the segmentation task was more challenging and the target training data was smaller. We adopted transfer learning as our target dataset was small and complex, containing only 363 images for training and features thirty classes. In comparison, the ”Automatic Cardiac Diagnosis Challenge” (ACDC) dataset,[28] which both nnUNet and TransUNet evaluated in their study, contains 100 cardiac MRI recordings for training and has five classes.

Hu et al.[29] first proposed the attention module, calling it the squeeze and excitation block. Wang et al.[30] later updated it with an adaptive bottleneck in their Channel Attention Module (CAM). We also utilized an adaptive bottleneck in the squeeze and excitation block to improve spectral information extraction, allowing for better differentiation of objects in our proposed model.

3. METHODOLOGY

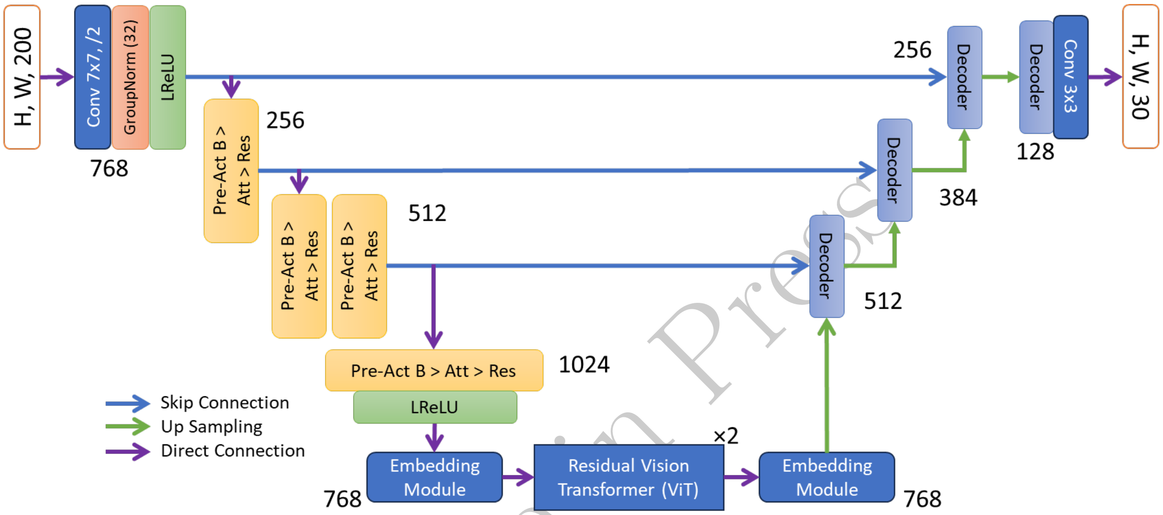

Figure 2: Outline of our proposed segmentation network.

Given an HS agricultural image $x \in R^{H \times W \times C}$ with the spatial resolution of $H \times W$ and C number of channels. We aim to predict the corresponding pixel-wise label map with size $H \times W$ and class values between 0 to $N - 1$. Here, N is the number of classes. The most common way is to directly train a CNN (e.g., U-Net) to convert the images into high-level abstract feature representations, which are then up-sampled for feature reconstruction in the expansive path to the full spatial resolution for classification. Here, we followed the design of TransUNet.[31] Our approach diverges from theirs by thoroughly exploring the attention mechanisms in the contraction path. Initially designed for three input channels, the network's hidden layers may now have to process hundreds of input channels, potentially creating a bottleneck. The spectral channel attention is crucial to effectively utilize the model's limited number of feature channels. This attention mechanism prioritizes vital features while suppressing less relevant ones, improving model performance. Figure 2 outlines the design of our proposed segmentation network.

3.1 Using Residual Blocks from ResNetv2 in Contraction Path

We start our contracting path with the first layer from ResNet architecture, which uses a single 7x7 convolution with stride 2 (Figure 2), optimally reducing computation for the subsequent layers. It combines the advantages of three 3x3 convolutions while maintaining the same receptive field and applying stride to downsample the image. This dual function makes it more computationally efficient.

Additionally, we have updated the input channels of this first layer to accommodate images with more than three channels. We used Eq. 1 to increase the output channels from 64 to 768 adaptively. The original 64 channels were designed to extract features from just three input channels, so it is logical to raise the number of output channels in response to the increase in input channels from three to two hundred.

$$channel_{out} = 3 * 2^{\lceil \log_2 (channel_{in}) \rceil} \tag{1}$$

The later units of TransUNet are Pre-Activation Bottleneck encoders from ResNet models pre-trained on ImageNet-21k and fine-tuned on ImageNet datasets.[32] The original ResNet-50 had a depth of four with [3,4,6,3] blocks, each reducing the image's resolution by a factor of two. In the TransUNet variant, the pre-trained ResNet had a depth of three with [3,4,9] units per block, and only the first unit of each block had a 3x3 convolution layer with a stride of 2.

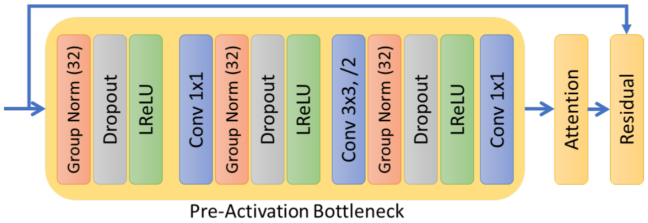

Figure 3: ResNetV2 block with Pre-Activation Bottleneck, Attention mechanism, and Residual block.

ResNet-V2[33] concentrates on making the second non-linearity as identity mapping. Thus, the gradient calculated in the output layer can easily reach the initial layers without changing the network. We modified the original Pre-Activation Bottleneck encoders with this design, as shown in Figure 3. Our contracting path for feature abstraction has [1,2,1] units per block. Similar to the previous design, we used a 3x3 convolution layer with a stride 2 to downsample in the first unit of each block. We employed the Leaky ReLU (LReLU) as the activation function. To mitigate overfitting, we implemented an adaptive dropout technique. Equation 2 shows the way we computed the dropout probability ($drop_{prob}$) for a given number of input feature channels (c_{in}).

$$drop_{prob} = \max(0.1, \min(c_{in}/2560, 0.25)) \tag{2}$$

We used the Max Pooling instead of convolution with strides in the skip connection of the residual pathway to match the resolution of the residuals and a 1x1 convolution layer to match the number of channels. Max Pooling resolves gradient propagation issues in DNNs and introduces non-linearity to the network.[34]

3.2 Using Attention in Residual Blocks of ResNetv2

Hu et al.[29] first proposed the attention module, calling it squeeze and excitation block in their newly proposed SENet. CNNs use their convolutional filters to capture spatial and temporal information from images. CNNs perform image classification by looking for low-level features such as edges and curves and then building up to more abstract concepts through convolutional layers. The computer uses low-level features obtained at the initial levels to generate high-level features to identify the object. In a typical CNN architecture, the network weights each channel equally when creating the output feature maps. Squeeze-and-excitation blocks add parameters to each channel of a convolutional block so that the network can adaptively adjust the weighting of each feature map and recalibrate it. We added channel attention to our ResNetv2 encoder, similar to SENet. However, we updated its fixed reduction ratio of 16 in the bottleneck with an adaptive reduction ratio[30] as mentioned in Eq. 3. Here, C_{in} is the number of channels coming into the attention block. It preserves more information with fewer features and discards more when there is an increased number of input features.

$$reduction\text{-}ratio = 2 * \log_2 C_{in} \tag{3}$$

3.3 Transformer as Encoder Bottleneck

Our Vision Transformer (ViT) as encoder design is identical to the TransUNet.[31] The first embedding module performs tokenization,[35] converting input features coming from the contraction path into the shape of $\frac{H \times W}{P^2}$, where patch size is $P \times P$. Then, it maps the vectorized patches into a latent d_{enc}-dimensional embedding space using a trainable linear projection, where learnable position embeddings are added to the patch embeddings to retain positional information. Each Transformer layer consists of Multi-head Self-Attention (MSA) and Multi-Layer Perceptron (MLP) blocks with layer normalization operation. These ViTs are pre-trained on ImageNet-21k and fine-tuned on ImageNet datasets. The second embedding module converts these tokenized patches back to the shape of input features.

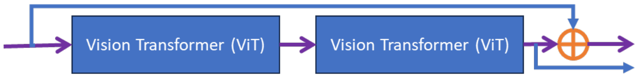

Figure 4: Design of our residual ViT

When we evaluated our model, we found that gradients are not flowing backward properly within the ViT, and 1-layer and 12-layer ViT performance is similar, which Chen et al.[31] also found in their evaluation. Thus, we took the inspiration from HSI-TransUNet,[5] designed our own residual ViT, and added a skip connection to create an alternating residual ViT as seen in Figure 4. We used a 2-layer residual ViT in our network and found no significant improvement in adding more layers.

3.4 Expansion Path

We used a cascaded upsampler (CUP) similar to TransUNet.[13] It comprises a 2× upsampling step and two convolution blocks. Each convolution block has a 3 × 3 convolution layer, a Batch Normalization layer, a Dropout layer, and a ReLU layer. To mitigate overfitting, we employed the same adaptive dropout technique here too. Before the segmentation head, we also reversed the Eq. 1 to compute the number of input channels. Equation 4 shows our updated solution for input channels. The original 16 channels were designed to provide features to the segmentation head to predict four classes. As the number of classes increased to thirty, our last CUP now has 128 channels. We subsequently updated the prior layers to adapt to this change.

$$channel_{in} = 2^{\lceil \log_2(channel_{out} * 3) \rceil} \tag{4}$$

The upsampling step in the original U-Net paper involved the upsampling feature map followed by a 2 × 2 convolution ("up-convolution") that halved the number of feature channels.[6] However, this deconvolution can create checkerboard artifacts.[36] Odena et al.[36] found that nearest-neighbor interpolation provided suitable results. On the other hand, we could utilize operations with learnable weights for upsampling in the expansion path, such as sub-pixel convolutions.[37] Sub-pixel convolution takes in r^2 channels and outputs r channels with double the resolution. It would substantially increase the number of feature channels in hidden layers, creating significant contraction and expansion path changes. Thus, for the sake of simplicity, we continued using the nearest upsampling.

3.5 Dataset

The authors of UAV-HSI-Crop Dataset[5] initially used random sampling in the original dataset, which resulted in one class being absent in the training dataset, some others missing in the test and many others missing in the validation. It posed a challenge in evaluating our model accurately. To overcome this, we performed stratified random sampling on the total dataset with a 70-15-15 split for training, validation, and test datasets. This method ensured that every class was present in all three splits: training, validation, and test, thereby enhancing the efficacy of our model evaluation. However, only one image had okra, and two had bok choy. We kept okra in the training dataset; otherwise, the model would never see okra. We distributed the bok choy between the training and validation datasets.

3.6 Evaluation Metrics

We used the Dice coefficient and Jaccard index to evaluate our model's segmentation performance. The Dice coefficient, also known as the Sorensen index, is calculated between the binary objects in two images, as shown in Eq. 5. Here are some key points to compute and understand dice score:

1. A Dice score of 1 indicates perfect overlap between two sets.
2. If none of the sets are empty, a Dice score of 0 indicates no overlap between the two sets.
3. If one set is empty, the intersection will also be empty, leading to a numerator of 0 and a Dice score of 0.
4. If both sets are empty, the Dice score is 1.

The Jaccard index is very similar to Dice coefficient and usually used for gauging the similarity and diversity of sample sets, as shown in Eq. 6.

$$Dice_{coeff} = \frac{2|A \cap B|}{|A| + |B|} \quad (5) \qquad Jaccard_{index} = \frac{|A \cap B|}{|A \cup B|} \quad (6)$$

Here, A is the first and B is the second set of samples (or binary objects).

We used multiple metrics to evaluate our model's classification performance. Accuracy shows how often a classification machine learning (ML) model is correct overall. Precision indicates how often the model is accurate when predicting the target class. Sensitivity (or recall) measures the model's ability to identify all instances of the target class. Equation 7, 8 and 9 shows how to compute accuracy, precision and recall.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (7)$$

$$Precision = \frac{TP}{TP + FP} \quad (8) \qquad Recall = \frac{TP}{TP + FN} \quad (9)$$

Here, TP refers to true positives correctly identified as positive cases. FP refers to false positives, actual negatives incorrectly identified as positive. FN indicates false negatives, actual positives mistakenly classified as negative. TN stands for true negatives correctly identified as negative cases.

There's often a trade-off between precision and recall. Increasing one can sometimes lead to a decrease in the other. Thus, F1-Score combines precision and recall, providing a balanced view of a model's performance (see Eq. 10).

$$F1_{score} = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (10)$$

Finally, Cohen's kappa (κ) is a statistical measure used to assess inter-rater reliability, especially for qualitative (categorical) items, and it considers the level of agreement that could occur by chance.

3.7 Loss Function

As shown in Eq. 11, we used a combination of Weighted (w-ce) and Categorical Cross-Entropy (cat-ce) loss. We used the "balanced" heuristic[38] to compute the class weights for the weighted cross-entropy and 0.7 as the β value. Weighted Cross-Entropy ensures that less represented classes will get higher weights when computing the loss. However, we kept the β value high to keep the $loss_{w\text{-}ce}$ low and reduce false positives from these under-represented classes.

$$loss_{comb\text{-}ce} = \beta * loss_{cat\text{-}ce} + (1 - \beta) * loss_{w\text{-}ce} \tag{11}$$

We then combined this cross-entropy loss with Log-Cosh Dice Loss function[39] for our model, as shown in Eq. 12. Our dice loss is defined as $loss_{dice} = \log(\cosh{(1 - Dice_{coeff})})$, and we used 0.4 as the α value. This α value balances performance for segmentation and classification tasks.

$$loss_{total} = \alpha * loss_{dice} + (1 - \alpha) * loss_{comb\text{-}ce} \tag{12}$$

3.8 Training Details

Task adaptation (transfer learning and fine-tuning) in deep semantic segmentation refers to modifying a pre-trained model to work on a different but related task without training from scratch.[21] Our first training step tried cross-domain transfer and label set modification and adaptation. Transfer learning leverages knowledge from a pre-trained model on a related large dataset. We replace the last classification layer to match the number of target classes. We kept our pre-trained ResNetV2 and ViT model frozen and used them as fixed feature extractors; only the up-sampling expansive path for feature reconstruction and the final convolutional layer for classification are trained using the limited NIR-RG image dataset. We chose it due to the small size of our new dataset and the desire to leverage knowledge from a larger dataset. During this step, we used a learning rate of 0.0001. Later, we modified the input layer to handle different input modalities. We trained only the modified contracting path for feature abstraction with a learning rate 0.000 01. Finally, we unfroze all layers of the model, used a very low learning rate of 0.000 001, and fine-tuned it. Fine-tuning trains the pre-trained model on a new dataset to adapt it more closely to the new task. Its effectiveness in suiting the specific needs of the new task makes it more customized than general transfer learning. This iterative approach allowed more adaptation to the new task, resulting in higher accuracy than general transfer learning. We used AdamW[40] as the optimizer in our training. The batch size for our training was 32. It was large enough to suppress noise but not too large to hinder learning from a small dataset. We set the max epoch to 500 with early stopping, and all runs ended before 200 epochs.

4. EVALUATION AND RESULTS

Our evaluation investigated our proposed HRViTUNet (HSI-ResNetV2-ViT-UNet) model's performance on HSI images from UAV-HSI-Crop Dataset.[5] We wrote our code in Python (PyTorch) and ran non-deterministic training four times with the same configuration on a 48GB NVIDIA RTX 6000 Ada Generation GPU with 20 CPU cores and 160GB RAM, ran evaluations on the test dataset, computed the performance metrics per test image, and averaged over all the runs. The estimated total size of our model is 4755.06 MB with 57.70 Million parameters (Params). Params' size is 229.45 MB [*]. Out of 57.70 M Params, we began our training with 26.36 M Params and a forward/backward pass size of 4346.45 MB. The number of floating-point operations (flops) was 2.19 Tflops, and the number of floating-point operations per second (FLOPS) was 37.37 TFLOPS [†]. We gradually unfreeze the whole model during the training. The final number of number of flops was 2.19 Tflops, and the number of FLOPS was 37.39 TFLOPS. During the evaluation, the number of flops was 546.57 Gflops, and the number of FLOPS was 29.63 TFLOPS.

For the qualitative evaluation, the light-blue area in Figure 5(b) represents the ground truth target crop. Figure 5(c) presents results from our runs where our model made some errors by misclassifying the crop as

[*] Computed using: PyTorch Model Size Estimator, Torch Info

[†] Computed using: DeepSpeed

background. More examples from our model is shown in Appendix A. For our comparative evaluation, we trained the TransUNet[13] model using RGB and HS images with a learning rate of 0.001. We updated the data loader and kept all other settings as the default for the TransUNet [‡]. As shown in Figure 5(e), the TransUNet trained with HS images performs qualitatively better than when trained with RGB images. For evaluating the HSI-TransUNet,[5] we modified the code base of TransUNet [‡] to match their paper. However, we could not make it work perfectly and had difficulty reproduce their performance. Here, in the tables, we report what we achieved.

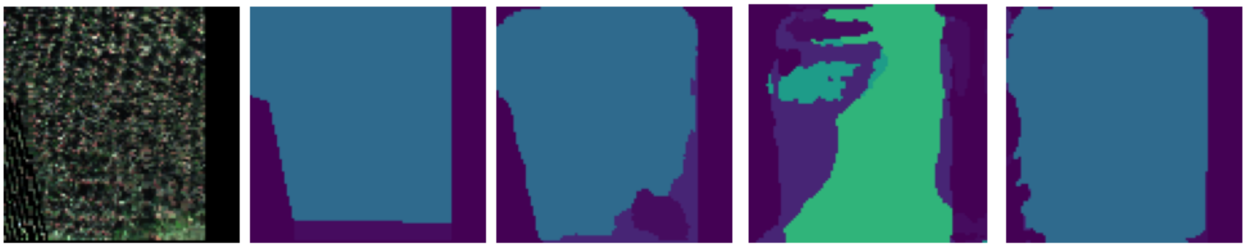

Figure 5: The visualization shows an example of semantic segmentation from our model using an image from UAV-HSI-CropDataset. (**a**) RGB, (**b**) Ground Truth, (**c**) Our HRViTUNet fine-tuned on this agricultural HSI dataset, (**d**) TransUNet fine-tuned on RGB dataset, (**e**) TransUNet fine-tuned on HSI dataset.

We used the Dice and Jaccard coefficients to evaluate the performance of our model's segmentation tasks. Table 1 shows these values.

Table 1: Performance comparison of HRViTUNet using segmentation metrics. Bold marks the better result than other models for that metric.

Model	Dice (mean, median)	Jaccard (mean, median)	Params (M)	Tflops (Tera)
HRViTUNet (Our)	0.749 ± 0.119, 0.749	0.737 ± 0.123, 0.732	**57.70**	2.19
TransUNet (RGB)	**0.779 ± 0.097, 0.794**	**0.771 ± 0.099, 0.785**	105.16	111.58
TransUNet (HSI)	0.766 ± 0.097, 0.760	0.757 ± 0.099, 0.753	105.78	111.65
HSI-TransUNet	0.643 ± 0.133, 0.648	0.631 ± 0.137, 0.633	99.33	**1.18**

To evaluate the performance of our model's classification task, we used Precision, Recall, F1-score, Accuracy, and Cohen-Kappa scores. Table 2 presents these values.

Table 2: Performance comparison of HRViTUNet using classification metrics. Bold marks the better result than other models for that metric.

Model	Precision	Recall	F1-score	Accuracy	Cohen-Kappa
HRViTUNet (Our)	**0.334 ± 0.166**	0.552 ± 0.201	**0.277 ± 0.159**	**0.766 ± 0.121**	**0.532 ± 0.222**
TransUNet (RGB)	0.315 ± 0.171	0.409 ± 0.238	0.226 ± 0.168	0.606 ± 0.248	0.345 ± 0.247
TransUNet (HSI)	0.315 ± 0.154	0.470 ± 0.231	0.242 ± 0.155	0.668 ± 0.232	0.434 ± 0.270
HSI-TransUNet	0.258 ± 0.141	**0.554 ± 0.216**	0.205 ± 0.140	0.746 ± 0.185	0.521 ± 0.275

Our semantic segmentation model achieved 76.6% accuracy and 74.9% dice score. The confusion matrix shown in Figure 6 demonstrates the model's effectiveness in identifying most objects, primarily because of Weighted Cross-Entropy loss. It is important to note that no examples of bok choy (class 7) or okra (class 29) exist in the test dataset. Our model's precision is highlighted by its zero false-positive for bok choy (class 7). There are some other low-performing classes for the model, along with 100% misclassification for Kohlrabi (class 13) and

[‡] TransUNet

sesame (class 18). Compared to the original HSI-TransUNet, we have fewer classes missing. Weighted Cross-Entropy, while providing overall lower quantitative scores than other loss functions, is a good choice if the goal is to identify the most targets, including underrepresented and hard-to-identify classes. We aimed to balance classification and segmentation accuracy and find a decent semantic segmentation model. Our loss function and task adaptation technique successfully achieved that goal.

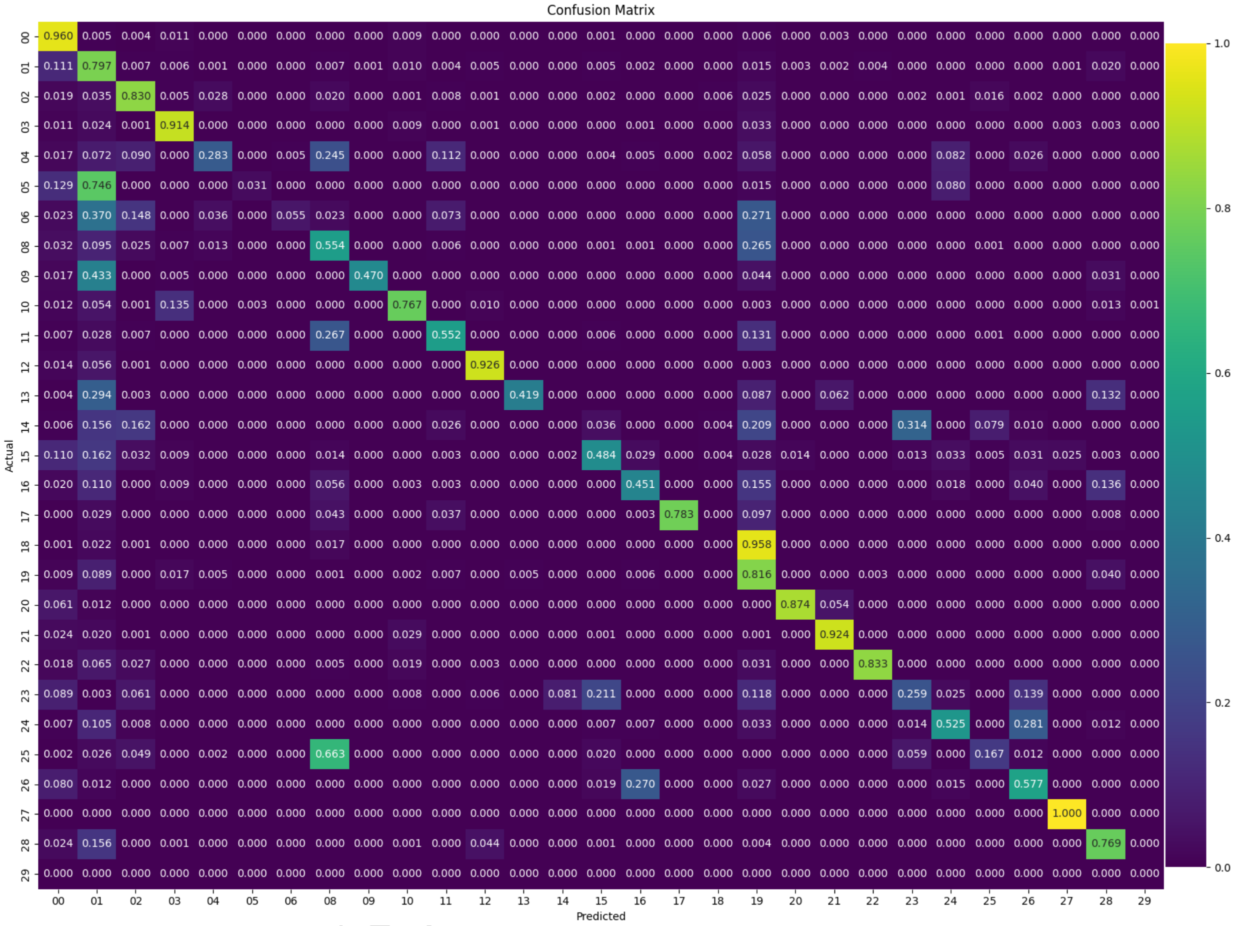

Figure 6: Confusion-Matrix from one of the runs shows low missing classes.

5. CONCLUSION AND FUTURE WORK

In this paper, we have successfully tackled the task adaptation challenge in the context of agricultural scenarios. We expanded the work of TransUNet, a model known for its promising results in medical imagery, proposed a spectral attention module augmentation, completed cross-domain transfer into agricultural practices, and improved the model's accuracy in segmenting crops. Our transfer learning with label set modification and fine-tuning showed promising results on hyperspectral imagery gathered via unmanned aerial vehicles. When tested on a UAV HS agricultural imagery dataset, our semantic segmentaation model achieved an 76.6% accuracy for the classification task and 74.9% dice score for the segmentation task, paving the way for notable advancements in crop mapping.

We plan to incorporate dimensionality reduction techniques like hyperspectral unmixing to select the most informative channels,[41] ensuring a constant number of channels with the best spectral information for our spectral attention network. Various HS sensors collect HS data differently, and the same sensor may produce a different number of usable channels because of noise, which hinders the way to domain adaptation. This thorough approach will enable us to generalize our model across various datasets and examine the impact of task adaptation on tiny datasets such as Indian Pines,[42] which consists of just one image.

APPENDIX A. QUALITATIVE EVALUATION

Visualizing results to understand a computer vision model's performance is often helpful. Figure 7 shows four more prediction examples. Our model makes more mistakes if the scene is cluttered, whereas simple scenes with clear boundaries are mostly correctly segmented.

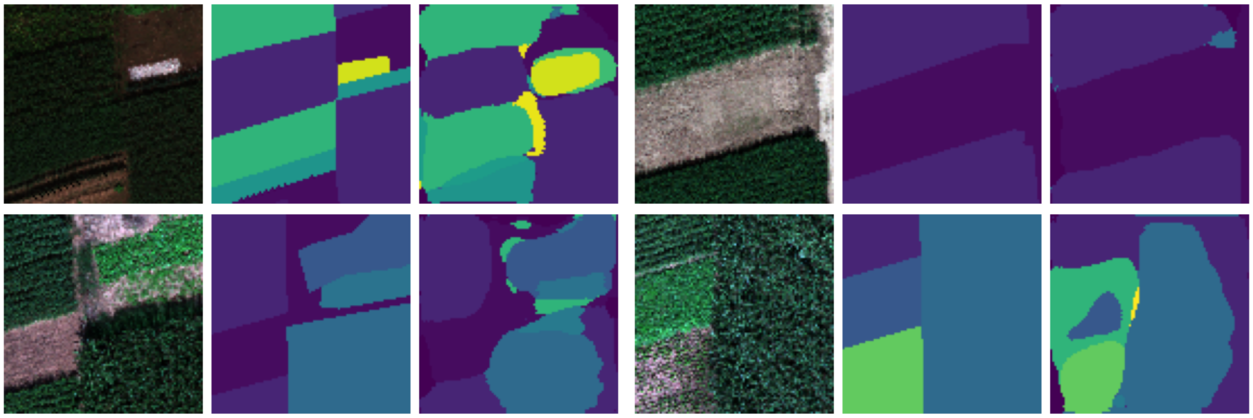

Figure 7: The visualization of our semantic segmentation model's predictions using images from UAV-HSI-CropDataset. In each section: (**a**) RGB, (**b**) Ground Truth, (**c**) Our HRViTUNet's prediction.

ACKNOWLEDGMENTS

This research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-21-2-0294 with Eddie L. Jacobs as PI. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Office or the US Government. The US Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

This material is based upon work supported by the National Science Foundation while Lan Wang was serving at the National Science Foundation. Any opinion, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] EOS, "Crop map." EOS Data Analytics,Inc. (2022). Accessed: Nov 01, 2024.

[2] Becker-Reshef, I., Barker, B., Whitcraft, A., Oliva, P., Mobley, K., Justice, C., and Sahajpal, R., "Crop type maps for operational global agricultural monitoring," *Scientific Data* **10**(1), 172 (2023).

[3] Alami Machichi, M., mansouri, l. E., Imani, Y., Bourja, O., Lahlou, O., Zennayi, Y., Bourzeix, F., Hanadé Houmma, I., and Hadria, R., "Crop mapping using supervised machine learning and deep learning: a systematic literature review," *International Journal of Remote Sensing* **44**(8), 2717–2753 (2023).

[4] Li, R., Zheng, S., Duan, C., Yang, Y., and Wang, X., "Classification of hyperspectral image based on double-branch dual-attention mechanism network," *Remote Sensing* **12**(3), 582 (2020).

[5] Niu, B., Feng, Q., Chen, B., Ou, C., Liu, Y., and Yang, J., "Hsi-transunet: A transformer based semantic segmentation model for crop mapping from uav hyperspectral imagery," *Computers and Electronics in Agriculture* **201**, 107297 (2022).

[6] Ronneberger, O., Fischer, P., and Brox, T., "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*], 234–241, Springer (2015).

[7] Wu, H., Zhang, J., Huang, K., Liang, K., and Yu, Y., "Fastfcn: Rethinking dilated convolution in the backbone for semantic segmentation," *arXiv preprint arXiv:1903.11816* **1** (2019).

[8] Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H., "Encoder-decoder with atrous separable convolution for semantic image segmentation," in [*Proceedings of the European conference on computer vision (ECCV)*], 801–818 (2018).

[9] Strudel, R., Garcia, R., Laptev, I., and Schmid, C., "Segmenter: Transformer for semantic segmentation," in [*Proceedings of the IEEE/CVF international conference on computer vision*], 7262–7272 (2021).

[10] Ranftl, R., Bochkovskiy, A., and Koltun, V., "Vision transformers for dense prediction," in [*Proceedings of the IEEE/CVF international conference on computer vision*], 12179–12188 (2021).

[11] Luo, Z., Yang, W., Yuan, Y., Gou, R., and Li, X., "Semantic segmentation of agricultural images: A survey," *Information Processing in Agriculture* **11**(2), 172–186 (2024).

[12] Tian, S., Dong, Y., Feng, R., Liang, D., and Wang, L., "Mapping mountain glaciers using an improved u-net model with cse," *International Journal of Digital Earth* **15**(1), 463–477 (2022).

[13] Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., Lu, L., Yuille, A. L., and Zhou, Y., "Transunet: Transformers make strong encoders for medical image segmentation," *arXiv preprint arXiv:2102.04306* **1** (2021).

[14] Schlemper, J., Oktay, O., Schaap, M., Heinrich, M., Kainz, B., Glocker, B., and Rueckert, D., "Attention gated networks: Learning to leverage salient regions in medical images," *Medical image analysis* **53**, 197–207 (2019).

[15] Liu, G., Bai, L., Zhao, M., Zang, H., and Zheng, G., "Segmentation of wheat farmland with improved u-net on drone images," *Journal of applied remote sensing* **16**(3), 034511–034511 (2022).

[16] Oktay, O., Schlemper, J., Folgoc, L. L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N. Y., Kainz, B., et al., "Attention u-net: Learning where to look for the pancreas," *arXiv preprint arXiv:1804.03999* **1** (2018).

[17] Fan, Z., Liu, K., Hou, J., Yan, F., and Zang, Q., "Jaunet: A u-shape network with jump attention for semantic segmentation of road scenes," *Applied Sciences* **13**(3), 1493 (2023).

[18] Liu, K.-H., Yang, M.-H., Huang, S.-T., and Lin, C., "Plant species classification based on hyperspectral imaging via a lightweight convolutional neural network model," *Frontiers in Plant Science* **13**, 855660 (2022).

[19] Wan, L., Cen, H., Zhu, J., Zhang, J., Zhu, Y., Sun, D., Du, X., Zhai, L., Weng, H., Li, Y., et al., "Grain yield prediction of rice using multi-temporal uav-based rgb and multispectral images and model transfer–a case study of small farmlands in the south of china," *Agricultural and Forest Meteorology* **291**, 108096 (2020).

[20] Amiri, M., Brooks, R., and Rivaz, H., "Fine tuning u-net for ultrasound image segmentation: which layers?," in [*MICCAI Workshop on Domain Adaptation and Representation Transfer*], 235–242, Springer (2019).

[21] Karimi, D., Warfield, S. K., and Gholipour, A., "Transfer learning in medical image segmentation: New insights from analysis of the dynamics of model parameters and learned representations," *Artificial intelligence in medicine* **116**, 102078 (2021).

[22] Olivas, E. S., Guerrero, J. D. M., Martinez-Sober, M., Magdalena-Benedito, J. R., Serrano, L., et al., [*Handbook of research on machine learning applications and trends: Algorithms, methods, and techniques: Algorithms, methods, and techniques*], IGI global (2009).

[23] Heaton, J., "Ian goodfellow, yoshua bengio, and aaron courville: Deep learning: The mit press, 2016, 800 pp, isbn: 0262035618," *Genetic programming and evolvable machines* **19**(1), 305–307 (2018).

[24] Yosinski, J., Clune, J., Bengio, Y., and Lipson, H., "How transferable are features in deep neural networks?," *Advances in neural information processing systems* **27** (2014).

[25] Li, X., Xiong, H., Wang, H., Rao, Y., Liu, L., Chen, Z., and Huan, J., "Delta: Deep learning transfer using feature map with attention for convolutional networks," *arXiv preprint arXiv:1901.09229* **1** (2019).

[26] Luo, Z., Yang, W., Yuan, Y., Gou, R., and Li, X., "Semantic segmentation of agricultural images: A survey," *Information Processing in Agriculture* **11**(2), 172–186 (2024).

[27] Isensee, F., Jaeger, P. F., Kohl, S. A., Petersen, J., and Maier-Hein, K. H., "nnu-net: a self-configuring method for deep learning-based biomedical image segmentation," *Nature methods* **18**(2), 203–211 (2021).

[28] Bernard, O., Lalande, A., Zotti, C., Cervenansky, F., Yang, X., Heng, P.-A., Cetin, I., Lekadir, K., Camara, O., Ballester, M. A. G., et al., "Deep learning techniques for automatic mri cardiac multi-structures segmentation and diagnosis: is the problem solved?," *IEEE transactions on medical imaging* **37**(11), 2514–2525 (2018).

[29] Hu, J., Shen, L., and Sun, G., "Squeeze-and-excitation networks," in [*Proceedings of the IEEE conference on computer vision and pattern recognition*], 7132–7141 (2018).

[30] Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., and Hu, Q., "Eca-net: Efficient channel attention for deep convolutional neural networks," in [*Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*], 11534–11542 (2020).

[31] Chen, J., Mei, J., Li, X., Lu, Y., Yu, Q., Wei, Q., Luo, X., Xie, Y., Adeli, E., Wang, Y., et al., "Transunet: Rethinking the u-net architecture design for medical image segmentation through the lens of transformers," *Medical Image Analysis* **97**, 103280 (2024).

[32] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L., "Imagenet: A large-scale hierarchical image database," in [*2009 IEEE Conference on Computer Vision and Pattern Recognition*], 248–255 (2009).

[33] He, K., Zhang, X., Ren, S., and Sun, J., "Identity mappings in deep residual networks," in [*Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*], 630–645, Springer (2016).

[34] Sun, S., Pang, J., Shi, J., Yi, S., and Ouyang, W., "Fishnet: A versatile backbone for image, region, and pixel level prediction," *Advances in neural information processing systems* **31** (2018).

[35] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N., "An image is worth 16x16 words: Transformers for image recognition at scale," in [*International Conference on Learning Representations*], (2021).

[36] Odena, A., Dumoulin, V., and Olah, C., "Deconvolution and checkerboard artifacts," *Distill* **1**(10), e3 (2016).

[37] Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A. P., Bishop, R., Rueckert, D., and Wang, Z., "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in [*Proceedings of the IEEE conference on computer vision and pattern recognition*], 1874–1883 (2016).

[38] King, G. and Zeng, L., "Logistic regression in rare events data," *Political analysis* **9**(2), 137–163 (2001).

[39] Jadon, S., "A survey of loss functions for semantic segmentation," in [*2020 IEEE conference on computational intelligence in bioinformatics and computational biology (CIBCB)*], 1–7, IEEE (2020).

[40] Loshchilov, I. and Hutter, F., "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101* **1** (2017).

[41] Du, Q. and Yang, H., "Similarity-based unsupervised band selection for hyperspectral image analysis," *IEEE geoscience and remote sensing letters* **5**(4), 564–568 (2008).

[42] Baumgardner, M. F., Biehl, L. L., and Landgrebe, D. A., "220 band aviris hyperspectral image data set: June 12, 1992 indian pine test site 3," *Purdue University Research Repository* **10**(7), 991 (2015). Accessed: Nov 01, 2024.