**Frontiers in Computing and Intelligent Systems**
ISSN: 2832-6024 | Vol. 4, No. 2, 2023

# Weed Recognition Method based on Hybrid CNN-Transformer Model

**Jun Zhang ***

College of Software Engineering, South China Agriculture University, Guangzhou, Guangdong, 510642, China
*** Corresponding author Email:** opip47@163.com

**Abstract:** As an important task in precision agriculture, weed recognition plays a crucial role in crop management and yield increase. However, achieving high accuracy and efficiency at the same time remains a challenge. To address the balance between accuracy and timeliness in weed recognition, this paper proposes a hybrid CNN-Transformer model for weed recognition. The model uses a combination of convolutional neural network (CNN) and Transformer structures for feature extraction and classification, taking into account both global and local information. In addition, the proposed Transformer Block incorporates the SDTA (Segmentation Depth Transpose Attention) mechanism to improve timeliness. Furthermore, this paper improves the original ViT model to enhance its accuracy. Experimental results on the Deep Weeds dataset by Olsen et al. show that the proposed hybrid model outperforms the original Vision Transformer model in weed recognition accuracy (89.43% vs. 96.08%). This research provides an effective solution for weed recognition using a hybrid model, with high practical value and application prospects.

**Keywords:** Weed Recognition; Vision Transformer; Hybrid CNN-Transformer Model; SDTA.

## 1. Introduction

With the rapid development of global agriculture, weed problem has become one of the urgent issues to be solved in agricultural production[1]. Weeds grow profusely and compete with crops for nutrients, water and sunlight, which not only severely affects the growth of crops, but also significantly reduces crop yields[2]. In order to achieve the goal of weeding, people began to spray large amounts of herbicides for weeding, but this method is prone to environmental pollution and hazards. Therefore, in order to improve agricultural production efficiency, reduce environmental pollution and hazards, an accurate and efficient method to identify and classify weeds is needed[3].

Weed identification and management has become a frontier and hotspot issue in the current research field of weed control, and the development of image processing technology and artificial intelligence technology provides new ideas and possibilities for solving this problem[4]. The current weed recognition technology can be primarily divided into two categories in academic research: machine learning methods based on traditional manual feature extraction, and deep learning models based on automatic feature extraction. There algorithms not only have the ability to automatically extract features from weed images, but also achieve higher recognition accuracy and real-time performance.

In current weed identification technology, the application of machine learning and deep learning models based on traditional manual feature extraction is becoming more and more popular. These algorithms can not only automatically extract features from weed images, but also have higher recognition accuracy and real-time performance[5].

Method based on manual feature extractor and classifier: This method usually requires manual design and extraction of features related to weeds, and uses classifiers for classification, such as Support Vector Machine (SVM), Random Forest, etc. [4].Cristóbal[6] et al. used Hough transform and SVM classifier to separate weeds in corn fields, achieving good results. Lottes[7] et al. proposed a weed species detection method based on dense stereo vision, comparing the depth value with manually selected weed height lines to identify the presence and species of weeds. Zhang[7] et al. preprocessed weed images using morphological filtering and threshold segmentation, extracted a series of features, and trained a random forest classifier to classify different types of weeds. However, this method has some drawbacks. Firstly, this method requires manual design and extraction of features related to weeds, which may require expert experience and sample data, and important information may be lost in the feature extraction process. Secondly, the performance and accuracy of the classifier mainly depend on the selected features. If the features are not representative, the recognition effect and accuracy may be affected. Finally, compared with using deep learning methods for weed identification, this method requires more manual operations and time-consuming[3].

Method based on deep learning: This is a method that uses deep learning technology for object extraction, feature extraction, and classification[8]. Due to the characteristics of weed data, the dataset needs to be preprocessed to remove the background from the image and only retain the weed body to improve the performance of the subsequent model[9]. Common preprocessing methods include the supergreen algorithm[10] and Otsu threshold segmentation algorithm[10]. In terms of feature extraction, Convolutional Neural Network (CNN) is one of the most widely used models. It can automatically learn the most representative features from images and generate a set of high-level abstract feature representations[10]. Based on these features, various classifiers can be used for classification, such as Support Vector Machine (SVM), Random Forest, etc. In recent years, many new deep learning models and methods have emerged. Shah[11] et al. proposed an image classification algorithm mainly targeting pests in soybeans, which was trained and classified using ResNet-50. Zhan[12] et al. used high-resolution remote sensing images as input and achieved

automatic segmentation of different crop types by training an FCN model with dilated convolution kernels. Zhu[13] et al. introduced a self-attention mechanism based on the improved U-Net model and achieved automatic segmentation of different crop types through training. These models have outstanding performance in image segmentation and detection tasks and also have potential in weed identification in the agricultural field. In summary, deep learning models have the advantages of strong adaptive ability and high recognition rate, but they require a large amount of training data and high computing resources.

However, weed identification technology still has some limitations and deficiencies. Firstly, weeds of different species or growth stages have different shapes and features, and weeds are very similar to crops, making the recognition task more complex, which puts higher demands on the recognition algorithm[3]. Secondly, given that most of the existing popular methods pursue higher accuracy, they cannot maintain faster inference speed, which leads to poor performance in deployment on resource-constrained edge devices[14]. Finally, currently, most of the existing weed identification algorithms are based on CNN for feature extraction. The dense computation and parameter sharing of CNN make its ability to extract local semantic information quite strong, but in real-world scenarios, not only local information needs to be focused on, but also global semantic information needs to be paid attention to[15].

Therefore, in this paper, a weed identification method combining CNN and Transformer hybrid model will be proposed to solve the above problems. This method first uses continuous convolutional layers for local feature extraction, then uses Transformer Block for global feature extraction, and finally combines the features of both to conduct image classification. Among them, the Transformer Block is based on the EdgeNeXt architecture, introducing SDTA (Segmentation Depth Transpose Attention), and using the "Patchify" strategy in the input section, followed by continuous stacking of three convolutional encoders to extract local features. At the same time, this paper will optimize it according to the characteristics of practical application scenarios and datasets, and use real data for testing to verify the accuracy and precision of this method in weed identification, providing scientific support for achieving more accurate and efficient weed identification.

# 2. Experimental Materials

## 2.1. Dataset


**Fig 1.** Images of different weed categories.

The weed dataset used in this project comes from the DeepWeeds weed dataset publicly available by Olsen[16] et al. The dataset consists of 9 categories, including 8 weed categories and 1 negative class, covering some common weed species such as Chineseapple, Siam weed, Parkinsonia, etc. The dataset contains images captured in natural environments with various lighting and angles. The dataset contains a total of 17,509 images, which have been divided into 14,036 training set images and 3,473 testing set images in this study. Sample images from the dataset are shown in Fig.1.

## 2.2. Data Preprocessing

ViT model was trained on the ImageNet dataset, so its input image size is 224×224. In order to adapt the DeepWeeds dataset to the ViT network, improve computational efficiency, and make the model easier to train and optimize, all images were resized to 224×224 and the data was normalized the data to [0,1] in this study to avoid numerical overflow while maintaining numerical stability.

# 3. Experimental Procedure

The weed recognition network proposed in this paper is mainly composed of a CNN-Transformer hybrid model, which includes StemConv module, ConvBlock(CNN) module, Transformer encoder module, and pooling fully connected classification module.

## 3.1. Overall Model Architecture

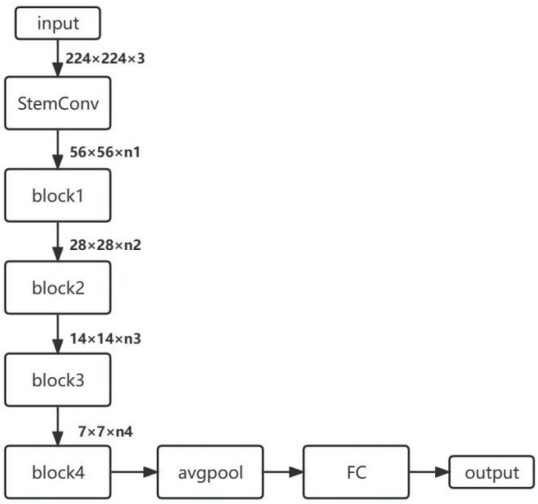


**Fig 2.** Overall architecture diagram of the hybrid model

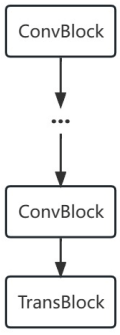


**Fig 3.** The specific structure of block i (1~4)

The overall structure of the CNN-Transformer hybrid model proposed in this study is shown in Fig.2. Firstly, the input image is passed into the StemConv layer (which

includes four convolutional layers) to perform preliminary feature extraction and downsampling. This mainly aims to reduce network computation and improve time efficiency, as high-resolution images require high-performance devices for training a network using Transformer to extract features. Additionally, this step transforms the data to a form suitable for subsequent neural network processing. Secondly, the preprocessed data is fed into a four-stage network structure, where each block receives the output of the previous block as its input and performs further feature extraction and transformation. Specifically, this includes n ConvBlocks at the beginning and 1 TransBlock at the end, as shown in Fig.3, with the number of ConvBlocks adjusted based on pretraining (in this study, the number of ConvBlocks for the four blocks were 3, 4, 10, and 3 respectively). Finally, the model performs final classification by reducing the number of features through a pooling layer, taking the average value at each feature position in the feature map and mapping it to the number of output categories through a fully connected layer to display the predicted results of the input image.

## 3.2. ConvBlock

The ConvBlock is designed to extract high-level features from input data in a deep learning model, improving its generalization performance and enabling it to better complete various tasks. Through operations such as convolutional layers, pooling layers, batch normalization layers, and nonlinear activation functions, the input data is gradually abstracted and compressed, enhancing the model's performance and generalization abilities. In this paper, we continuously trained the model and designed the convolutional block shown in Fig.4. When the input is passed into the ConvBlock, the data is first grouped locally through Patch Embedding for subsequent convolutional operations. Then, 3x3 and 1x1 sliding windows are used to perform convolutional operations on the input, which are activated through BN batch standardization and GELU functions. Subsequently, pooling operations are used to reduce the size of the feature map, followed by batch normalization and finally a fully connected layer that abstracts the high-level features of the data for downstream tasks.

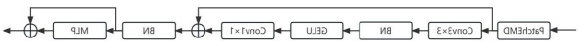


**Fig 4.** The structure of ConvBlock

## 3.3. TransBlock

In the TransBlock module of the neural network structure proposed in this paper, the EdgeNeXt [15] architecture is used to segment and extract features from input data. The EdgeNeXt model is an efficient and deployable deep learning architecture that can be applied to mobile visual tasks. The model consists of two main components: the Convolution Encoder for extracting image features, and the STDA Encoder for combining time and space information to achieve more accurate object tracking.

The specific structure of EdgeNeXt can be seen in Fig.5, which follows the standard "four-stage" pyramid-style design specification and includes two core modules: the convolutional encoder and the STDA encoder. At the beginning of the overall structure, a "Patchify" strategy similar to ViT and SwinTransformer is used, employing non-overlapping convolutions of size 4x4 to achieve better pooling effects, resulting in an output size of H/4xW/4xC1. Next, three stacked convolutional encoders of kernel size 3x3 are used to extract local features without changing the feature map size. In Stage 2, down-sampling is achieved through convolution with a stride of (2,2), after which two consecutive encoder layers with kernel size 5x5 are stacked and position encoding is added via element-wise addition before entering the STDA module. In Stages 3 and 4, summing convolutional encoding layers are used, and different kernel sizes are applied to implement an adaptive kernel size mechanism.

This design was adopted to increase the CNN's local receptive field and improve model performance while avoiding the expensive computational cost associated with directly using larger kernels. Therefore, the pyramid-style design is a reasonable approach. Furthermore, adding position encoding only once in the four stages reduces the impact on detection, segmentation, and other tasks while improving model inference speed.

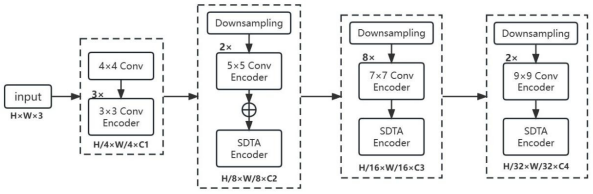


**Fig 5.** The schematic diagram of the EdgeNeXt architecture

## 3.4. SDTA (Segmentation Depthwise Transpose Attention)

When trying to combine the excellent characteristics of Vision Transformer (ViT) and CNN, any combination method inevitably brings a new problem: due to the characteristics of self-attention calculation, the inference speed of the architecture is severely limited. Therefore, in order to make the model balance performance and inference speed, the authors of EdgeNeXt introduce a Segmentation Depthwise Transpose Attention (SDTA[15]) encoder as shown in Fig.6, which achieves efficient integration without increasing additional parameter volume and multiplication-addition operation amount (MAdds). The SDTA encoder consists of two main components: the feature encoding module and the self-attention calculation module.

In the feature encoding module, the input features are first divided into s subsets, each with an equal size, and each subset is encoded by fusing the output features of the previous subset and then passing through a 3x3 depth-wise convolution. The SDTA Encoder uses an adaptive number of subsets to allow flexibility in feature encoding, and the output features of s subsets are concatenated to obtain output features with multiple scales. This module tries to learn an adaptive multi-scale feature representation, encoding different spatial levels on the input image and intuitively encoding the global image representation.

The self-attention calculation module is the second component of the SDTA Encoder, which calculates attention on the input features to obtain important features. In this module, the input features are first transformed into Q, K, and V tensors through three linear layers, and L2 normalization is performed on the Q and K tensors before calculating the cross-covariance attention to stabilize the training. This module uses dot product operation to calculate the dot product between Q T Q and K K T in the channel dimension, resulting in an attention matrix of C x C. The attention matrix is further processed by softmax and dot product with the V tensor to calculate the final attention map. Finally, two 1x1 point-wise

operations are used, with LN and GELU activation layers to produce non-linear features. To avoid complexity problems, this module borrows from the transpose QK attention feature map and uses MSA's dot product operation in the channel dimension to achieve linear complexity.

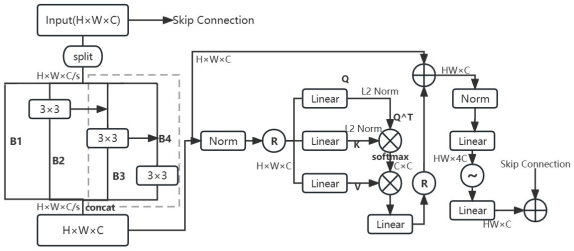


**Fig 6.** The STDA Encoder structure diagram

# 4. Results and Analysis

## 4.1. Experimental Configuration

The experiments were conducted on a Windows 10 operating system, with an AMD Ryzen 7 4800H CPU and an NVIDIA GeForce GTX 1650 GPU. The programming environment used for training was Python 3.7 and the deep learning framework was PyTorch 1.8.0.

## 4.2. Training Settings

The SGD optimizer was selected for model training, with a learning rate initialized to 0.001, momentum of 0.9, weight decay of 5e-5, batch_size of 8, and 100 iterations. Additionally, the cosine annealing function was applied to dynamically adjust the learning rate, with the maximum learning rate set to 0.01.

Specifically, the learning rate corresponding to the current epoch was calculated based on the shape of the cosine function during each iteration, and then updated using the scheduler. At the end of each epoch, the relevant indicators for that epoch, including training loss, training accuracy, validation loss, validation accuracy, and current learning rate, were recorded in TensorBoard for easy observation and comparison of results between different experiments. At the same time, the model parameters for the current epoch were saved to a designated path for future testing or further fine-tuning.

## 4.3. Performance Evaluation Metrics

In this paper, we used five specific metrics to demonstrate the performance of the model: accuracy, precision, recall, F1 score, and confusion matrix. In this paper, assuming TP and TN are the number of correctly identified positive and negative samples, and FP and FN are the number of incorrectly identified positive and negative samples, respectively, then:

Accuracy refers to the proportion of samples correctly classified by the classifier to the total number of samples:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Precision refers to the proportion of actual positive samples among the samples predicted as positive by the classifier:

$$precision = \frac{TP}{TP+FP} \quad (2)$$

Recall refers to the proportion of samples correctly predicted as positive by the classifier among all true positive samples:

$$recall = \frac{TP}{TP+FN} \quad (3)$$

F1 score is a metric that considers both precision and recall, which is the harmonic mean of precision and recall:

$$F1 = \frac{2\times precision\times recall}{precision+recall} \quad (4)$$

The confusion matrix can help gain insight into the model's performance on different categories and provide some reference for subsequent model adjustments.

## 4.4. Experimental Results and Analysis

To evaluate the accuracy of the model, the CNN-Transformer hybrid model was compared with some classical models such as Vision Transformer, ResNet50, AlexNet, VGG16, etc. in terms of accuracy, as shown in Table 1.

**Table 1.** Comparison of Model Accuracy

| Model | Accuracy/% |
|---|---|
| VGG16 | 86.21 |
| GoogLeNet | 79.23 |
| AlexNet | 80.09 |
| ViT | 89.43 |
| CNN-Transformer | 96.08 |

Based on Table 1, it can be seen that the use of a Transformer network can bring significant performance improvements compared to some classical models obtained by pre-training CNN models. Due to the stronger sequence modeling ability and the ability to capture long-range dependencies, the hybrid model and the original ViT model have higher accuracy than other models. Compared with the original ViT model, the CNN-Transformer hybrid model can better balance local and global features, resulting in further performance improvement.

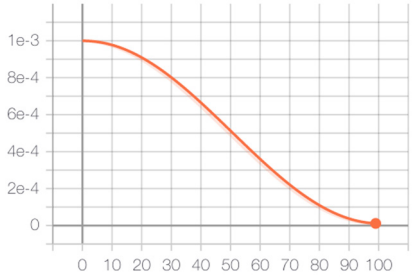


**Fig7.** Learning rate curve

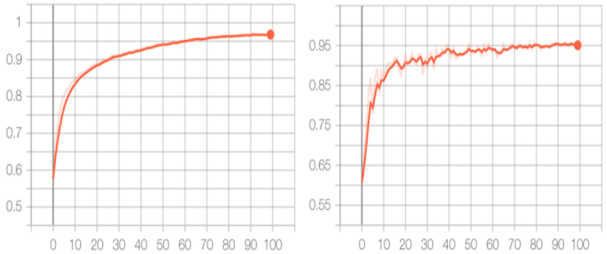


**Fig 8.** accuracy curve on the training set (left) and validation set (right)

To demonstrate the model's fitting ability during the training process and the model's performance on unknown data, the learning rate curve (Fig.7), the accuracy curve on the training set and validation set (Fig.8), and the loss curve (Fig.9) were plotted. Overall, the model showed a good performance. On the accuracy and loss curves of the test set, there were slight fluctuations and instability, which may be

due to the presence of noisy or irregular data in the test set that are different from the data in the training set, including incorrectly labeled data or missing data, but the fluctuation range is still acceptable.

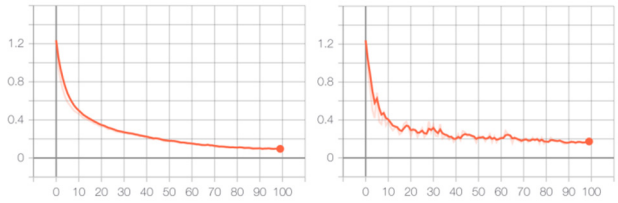

**Fig 9.** loss curve on the training set (left) and validation set (right)

## 4.5. Model Performance Evaluation

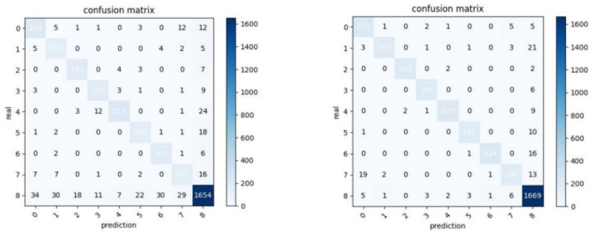


**Fig 10.** Confusion matrix of the original ViT (left) and the hybrid model (right)

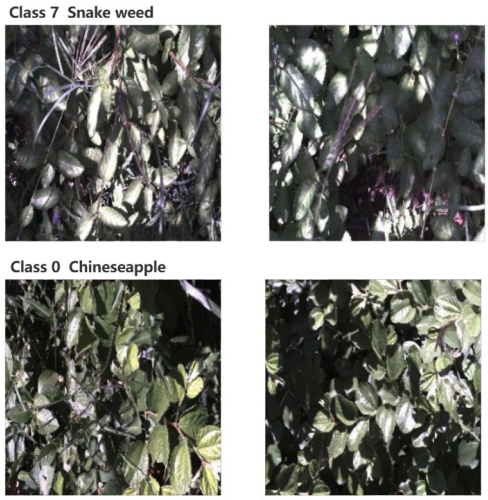


**Fig 11.** Class 7 is recognized as Class 0

The confusion matrix is a tool used to evaluate the performance of classification models. It can help understand the model's performance on different categories, and quantify and visualize its classification performance. Here, a comparison was made between the confusion matrices of the original ViT model and the CNN-Transformer hybrid model as shown in Fig.10. The horizontal axis represents the predicted categories on the test set, while the vertical axis represents the true categories of the test set images. The numbers on the main diagonal represent the number of correct predictions by the model, while the others are the numbers of incorrect predictions. In the original dataset, there are 8 categories of weeds, labeled from 0 to 7, and the negative class is labeled as 8. The sample distribution is uneven, so there are more cases where the eighth category is misclassified as other categories or other categories are misclassified as the eighth category. However, considering that the number of images in the eighth category is almost 8 times that of other categories, this error rate can be accepted. The hybrid model repeatedly misidentified images of category 7 as images of category 0. Based on the analysis of the images as shown in Fig.11, it is speculated in this study that this is due to poor lighting and a similarity in the morphology of the two types of weeds, leading to difficult recognition. Overall, according to the analysis of the confusion matrix, the hybrid model has good performance on this dataset, and has made significant improvements in performance compared to the original ViT model.

# 5. Conclusion

To simultaneously capture local and global information in the feature extraction and classification process of weed recognition, this paper combines the advantages of traditional CNN models and Vision Transformer models to design a more suitable CNN-Transformer hybrid model for weed recognition. Based on a four-stage CNN structure, each stage uses stacked ConvBlocks to perform local feature extraction, and a TransBlock is added at the end of each stage for global feature extraction. In the TransBlock, an STDA attention mechanism is introduced, which efficiently combines the CNN and ViT models to minimize the impact of the slower inference speed caused by the self-attention calculation method. Experimental results show that the hybrid model has high accuracy, precision, recall, and F1 score. This model overcomes the limitations of CNN in modeling global information and does not require the massive computation of the ViT model, making it more suitable for deployment on edge devices for weed recognition.

However, there are still challenges in practical applications, such as the diversity and quantity of weed species increasing the difficulty of model training, requiring more computing resources and memory space to ensure the efficient operation of the model, and different shooting angles and lighting conditions in different environments may also affect the stability and accuracy of the model. Future research can further optimize the structure and parameter settings of the hybrid model, explore more efficient computing methods and algorithms, and combine the model with other weed classifiers and agricultural technologies to improve the efficiency and sustainability of agricultural production.

# References


[1] Haq, M. A. (2022). CNN Based Automated Weed Detection System Using UAV Imagery. Computer Systems Science and Engineering, 42(2), 837-849.

[2] Razfar, N., True, J., Bassiouny, R., Venkatesh, V., & Kashef, R. (2021). Weed detection in soybean crops using custom lightweight deep learning models. Computers and Electronics in Agriculture, 185, 106016.

[3] Wu, Z., Chen, Y., Zhao, B., Kang, X., & Ding, Y. (2021). Review of Weed Detection Methods Based on Computer Vision. Frontiers in Plant Science, 12, 634505.

[4] Su, W.-H. (2021). Advanced Machine Learning in Point Spectroscopy, RGB- and Hyperspectral-Imaging for Automatic Discriminations of Crops and Weeds: A Review. Sensors, 21(14), 4707.

[5] Tao, T., & Wei, X. (2020). A hybrid CNN-SVM classifier for weed recognition in winter rape field. Precision Agriculture, 21(1), 26-37.

[6] Cristóbal, J., Moreda, G. P., & Muñoz-Rodríguez, M. (2006). Support vector machine classification to localize weeds in images of a maize field. Spanish Journal of Agricultural Research, 4(4), 433-444.

[7] Lottes, P., & Simard, P. (2010). Weed species detection using dense stereo vision. Proceedings of the 17th International Conference on Image Processing (ICIP), 1337-1340. Zhang, M., Liu, H., Dong, T., & Slaughter, D. C. (2015). Automated weed identification using an ensemble of optimized segmentation and classification methods. Computers and Electronics in Agriculture, 116, 225-232.

[8] Hu, K., Wang, Z., Coleman, G., Bender, A., Yao, T., Zeng, S., Song, D., Schumann, A., & Walsh, M. (2020). Deep Learning Techniques for In-Crop Weed Identification: A Review. Computers and Electronics in Agriculture, 178, 105715.

[9] Espejo-Garcia, B., Mylonas, N., Athanasakos, L., Fountas, S., & Vasilakoglou, I. (2019). Towards weeds identification assistance through transfer learning. Computers and Electronics in Agriculture, 162, 183-193.

[10] Wang, Y., Liu, H., Wang, D., & Liu, D. (2020). Image processing in fault identification for power equipment based on improved super green algorithm. Measurement, 154, 107503. Xiao, L., Ouyang, H., & Fan, C. (2019). An improved Otsu method for threshold segmentation based on set mapping and trapezoid region intercept histogram. Optik, 180, 718-726. Jiang, H., Zhang, C., Qiao, Y., Zhang, Z., Zhang, W., & Song, C. (2020). CNN feature based graph convolutional network for weed and crop recognition in smart farming. Computers and Electronics in Agriculture, 178, 105797.

[11] Shah, D., Gupta, R., Patel, K., Jariwala, D., & Kanani, J. (2021). Deep Learning based Pest Classification in Soybean crop using Residual Network-50. 2021 International Conference on Inventive Systems and Control (ICISC), 95-99.

[12] Zhan, Z., Li, Y., Liu, F., Zhang, H., & Xu, L. (2021). Semantic segmentation of agricultural crops using a fully convolutional neural network with dilated convolutions. Remote Sensing, 13(2), 285.

[13] Zhu, B., Zhen, F., Li, S., Hu, J., & Liu, Y. (2021). Crop segmentation with an improved U-Net model based on self-attention mechanism. Remote Sensing, 13(9), 1857. doi: 10.3390/rs13091857.

[14] Li, J., Xia, X., Li, W., Li, H., Wang, X., Xiao, X., Wang, R., Zheng, M., & Pan, X. (2022). Next-ViT: Next Generation Vision Transformer for Efficient Deployment in Realistic Industrial Scenarios. IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR) Workshops.

[15] Maaz, M., Shaker, A., Cholakkal, H., Khan, S., Zamir, S. W., Anwer, R. M., & Khan, F. S. (2023). EdgeNeXt: Efficiently Amalgamated CNN-Transformer Architecture for Mobile Vision Applications. In L. Karlinsky, T. Michaeli, & K. Nishino (Eds.), Computer Vision – ECCV 2022 Workshops. ECCV 2022. Lecture Notes in Computer Science, vol 13807. Springer, Cham.

[16] Olsen, A., Konvalina, D. A., Philippa, B., et al. (2019). DeepWeeds: A Multiclass Weed Species Image Dataset for Deep Learning. Scientific Reports, 9(1), 1-12.

[17] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Houlsby, N. (2021). An Image Is Worth 16x16 Words: Transformers for Image Recognition At Scale. arXiv preprint arXiv:2010.11929.

[18] Wang, X., Chan, K., Gao, Y., & Yang, M. H. (2020). Training attention networks for high-resolution image processing: An empirical study. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 10432-10441.

[19] Kolesnikov, A., Lampert, C. H., & Abdelsalam, M. (2021). Improved Convolutions via Hebbian Block-Sparse Kernel and Convex Optimization. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 8144-8153.