European Food Research and Technology (2023) 249:1833–1843
https://doi.org/10.1007/s00217-023-04258-1

ORIGINAL PAPER

A comparative study of vision transformers and convolutional neural networks: sugarcane leaf diseases identification

Süleyman Öğrekçi[1,2,3] · Yavuz Ünal[1,2,3] · Muhammet Nuri Dudak[1,2,3]

Received: 11 December 2022 / Revised: 31 March 2023 / Accepted: 1 April 2023 / Published online: 21 April 2023

Abstract

Diseases in agricultural products cause significant decrease on harvest efficiency and economic values of the products, early detection of diseases can prevent this loss. The development of artificial intelligence has brought its use in the field of agriculture. These practices have facilitated the work of farmers and increased productivity. Sugarcane is one of the agricultural crops with high economic value, and in this study, diseases in sugarcane leaves were classified by using deep learning methods. The dataset we use contains a total of 2521 images and there are 5 classes; healthy, mosaic disease, redrot disease, rust disease and yellow leaf disease. DenseNet121, one of the convolutional neural network (CNN) models, is applied to this dataset first, followed by the Vision Transformers (ViT) model, and finally the ViT + CNN combination is applied, and the results are compared. As a result of the observations, it is understood that the precisions of 92.87%, 93.34%, and 87.37%, respectively, were obtained.

Keywords Sugacane diseases · Deep learning · Image processing · Vision transformers

Introduction

Agriculture is an important food and economic resource for mankind and the economy of many countries still mainly relies on agriculture [1]. It is well known that diseases in agricultural products significantly affect food supply all over the world. Sugarcane is one of the important agricultural products all over the world and it is indispensable for sugar and ethanol production. Sugar is one of the most important nutrients produced and consumed every year, millions of tons. About 80% of sugar production in the world is obtained from sugarcane [2]. Diseases that occur in sugarcane affect mainly farmers and sugar production; it is clear that early detection of diseases can significantly reduce harvest loss [3]. The use of artificial intelligence-based systems in disease detection has become quite common in recent years. In many developing countries, farmers' knowledge of diseases is still at low levels, traditionally, the diagnosis of this disease is made with the observation of farmers and this situation brings many difficulties with it. The use of deep learning-based systems facilitates overcoming this problem [4, 5]. In this study, a sugarcane leaf image dataset consisting of 2521 images is classified by deep learning models.

Considering the studies done in this field; Daphal et al. [6] applied VGG16 and ResNet, one of the transfer learning algorithms, to a sugarcane leaf image dataset consisting of 1470 images and obtained 83% and 91% classification accuracy, respectively. Militante et al. [7] used a dataset consisting of 13,842 sugarcane images in 7 different classes and achieved 95% classification accuracy in their analysis with a CNN architecture. Miltante and Gerardo [3] applied three different CNN-based models, StridedNet, LeNet and VGGNet, to 14,725 sugarcane images in their study and they obtained 90.10%, 93.65% and 95.40% classification accuracy, respectively. Murugeswari et al., in their study [1], developed an Android application that detects sugarcane diseases using the Faster Region-based Convolutional Neural Network on a dataset consisting of 1500 sugarcane images. Alencastre-Miranda et al. in their study [5], detected damaged and healthy leaves by using transfer learning models AlexNet, VGG-19, GoogleNet and Resnet101.

✉ Süleyman Öğrekçi
suleyman.ogrekci@amasya.edu.tr

1 Department of Mathematics, Amasya University, Amasya, Turkey

2 Department of Computer Engineering, Amasya University, Amasya, Turkey

3 Department of Computer Engineering, Harran University, Şanlıurfa, Turkey

Hernandez et al. [4], classified the diseases in sugarcane leaves with the transfer learning algorithms InceptionV4, VGG16, ResnetV2-152, AlexNet on a dataset consisting of 16,800 sugarcane leaf images and they obtained 99.61%, 98.88%, 99.23% and 94.24% classification accuracy. In their study, Saavedra-Burbano and Marin-Hurtado [8] used the pre-trained networks InceptionV3 and InceptionResNetV2, which are transfer learning models, on the dataset consisting of 820 sugarcane leaf images. Malik et al. [9] detected the infected region in Sugarcane crops using YOLO Faster R-CNN object detection algorithms and achieved a classification accuracy of 93.40%. Sharma and Kukreja [10] classified Sugarcane red rot disease in sugarcane products with multilayer perceptron based deep learning in their study, they succeeded in classifying five different levels of Sugarcane red rot disease with a classification accuracy of 99.15%. Chen et al. [11] performed sugarcane stem node detection using YOLO v4, Faster R-CNN, SSD300, RetinaNet, and YOLO3 methods and had the highest success with YOLO v4 at 95.17%. Tamilvizhi et al. [12], utilized quantum behaved particle swarm optimization based deep transfer learning (QBPSO-DTL) model for sugarcane leaf disease detection and performed feature extraction with SqueezeNet model. Grijalva et al. [13], classified the density of sugarcane aphids with pre-trained deep learning models in their study. They used Inception V3 and Xception as models and achieved 86% classification accuracy. The comparison of this study with other studies in the literature is presented in Table 1.

The remainder of the article is organized as follows: the following section contains details on the materials and methods, the subsequent section discusses the results from the study, and the next section elaborates on the discussions and recommendations.

Materials and methods

Dataset

In this study, the dataset used consisted of 2521 sugarcane leaf images[14], which were taken at different sizes, including 1040×493 pixels, 493×1040 RGB, and 622×1280 pixels. The variations in size were due to the images being captured in both horizontal and vertical orientations. The dataset was comprised of images belonging to 5 categories, which included healthy leaves and four different types of sugarcane leaf diseases: redrot, rust, yellow, and mosaic. To enhance the training and validation of the DenseNet121 model on this dataset, various data augmentation methods were applied using the ImageDataGenerator. These methods included rescaling the pixel values to be within the range of 0 and 1, randomly rotating the images by up to 40 degrees, randomly shifting the images horizontally and vertically by up to 20% of their total width and height, applying a shear transformation to distort the images by up to 20%, randomly zooming in or out of the images by up to 20%, randomly flipping the images horizontally, and filling in any empty pixels using the nearest pixel value. These techniques aimed to increase the diversity of the dataset, resulting in a more robust and accurate model for predicting the sugarcane leaf diseases. The number of images belonging to these categories are given in Table 2.

Accordingly, there are 522 healthy, 514 rust, 518 redrot, 505 yellow and 462 sugarcane leaf images in the mosaic class. Sample images of the data set are shown in Fig. 1.

CNNs and DenseNet121

CNNs find widespread use especially in the field of image processing after the success of the deep learning model named AlexNet developed by Krizhevsky et al. [15] in 2012. CNNs are deep neural networks that contain special layers called Convolutional layer and Pooling layer. In summary, a CNN is a deep neural network made up of convolutional layers (filters), pooling (downsampling) layers, and fully connected (FC) layers. A CNN can contain many convolutional layers, depending on the complexity of the model [16] (Fig. 2).

Table 2 Classes in the dataset

Disease	Healthy	Rust	Red dot	Yellow	Mosaic
Samples count	522	514	518	505	462

Table 1 Classification of some sugarcane products with different artificial intelligence methods

No	Section	Data pieces	Class	Method	Accuracy (%)	References
1	Sugarcane leaf	1470	5	VGG16, Resnet	83.91	[6]
2	Sugarcane leaf	13,842	7	CNN	95	[7]
3	Sugarcane leaf	14,725	7	StridedNet, LeNet, VGGNet	90.10, 93.65, 95.40	[3]
4	Sugarcane leaf	16,800	6	InceptionV4, VGG16, ResnetV2-152, AlexNet	98.88, 99.61, 99.32, 94.24	[4]
5	Sugarcane leaf	820	2	Scratch network, InceptionV3 and InceptionResNetV2	91.72, 92.07, 87.70	[8]
6	Sugarcane leaf	120	2	Deep Stacked autoencoder (DSAE)	93.75	[12]
7	Sugarcane leaf	2521	5	DenseNet121, ViT, ViT + DenseNet	91.98, 92.85, 88.28	Our approach

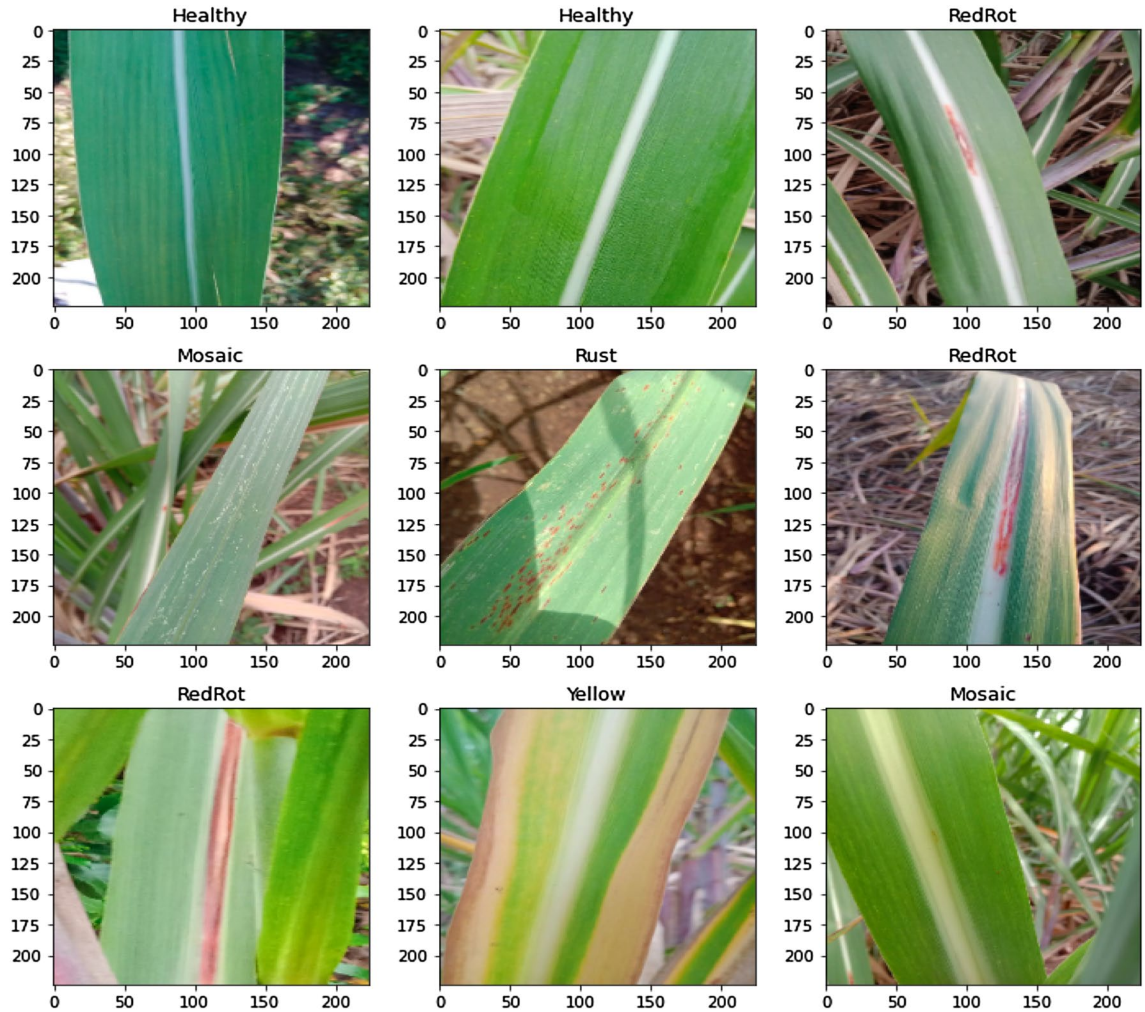

Fig. 1 Example of some diseases of sugarcane: mosaic, rust, redrot, yellow, and healthy

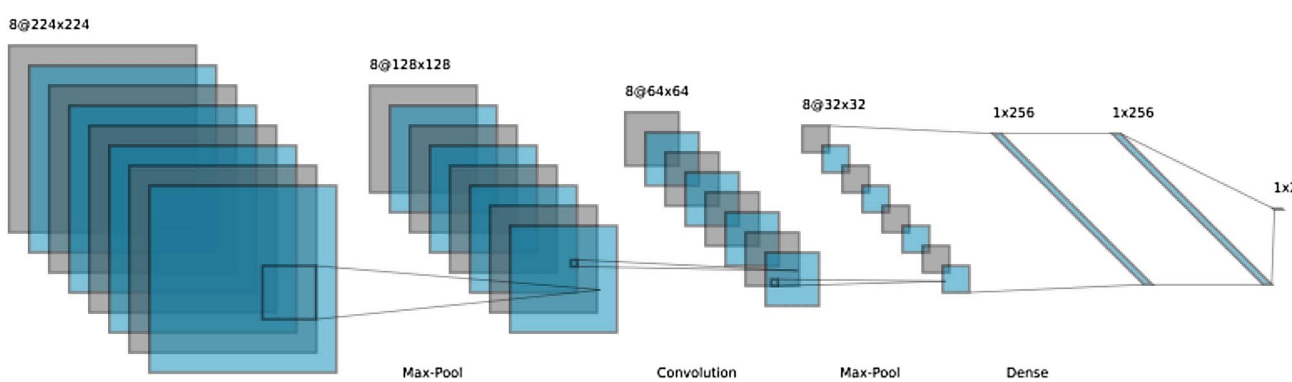

Fig. 2 Typical architecture of a CNN model for image classification

DenseNet [17] was presented as an architecture that can resolve the vanishing gradient problem and greatly reduce the number of parameters. Transition layers and dense blocks are both parts of the DenseNet design. Each layer is connected to the other forward layers by the dense block. The transition layer, which comes after the dense block, is what's employed to cut down on feature maps and create a model that's so small. DenseNet121 is made up of a

7×7 convolutional layer as the first layer, 116 dense block convolutional layers, three transitional layer convolutional layers, and an output layer that is fully connected [18]. The DenseNet concept was developed by Huang et al. [19] and improves on ResNet by introducing dense connections, signifying that each layer is connected to all other layers. This highly linked architecture naturally makes sure that every layer receives network parameters from levels above it and transmits its feature map to layers below it. The ability to reuse features while reducing the overall number of parameters is another important advantage of this structure.

Transfer learning

All sectors lacking access to big labeled datasets are paying close attention to the transfer learning method of deep learning. The most popular transfer learning design is called fine-tuning, in which the network is first pre-trained on a larger dataset, some of the layers are then taken into consideration for the target network from the pre-trained model, and either all or some of the layers are fine-tuned to obtain the desired network. A transfer learning study shows that a decrease in generalization, one of the challenges in transfer learning, can be caused by an increase in source and target differences. While layer transformation adjustments may improve performance, adding more layers to a network serves no purpose [20].

Vision transformers

This model was first revealed in 2021. The original purpose of transformers was natural language processing (NLP). A pure transformer-based design known as vision transformer (ViT) was shown by Dosovitskiy et al. [17] to be capable of competitive performance on a number of vision classification tasks. An increased amount of study has been done on modifying transformers for vision tasks as a result of the ViT design. There has been a general tendency of integrating hierarchical representations and limited region of interests from CNNs into ViTs, or extending CNNs with Transformers in a hybrid method. However, due to the initial tokenization phase and the lack of pooling operations, the ViT, unlike CNNs, only processes the input picture on a single scale [21, 22].

Without employing convolution layers, the ViT models utilize a transformer architecture for sequences of image patches. Image patches are simply split into small parts of the entire image. Transformers creates low dimensional embeddings from these patches [23]. Patches of a sample image from our dataset are visualized in the Fig. 3.

One of the fundamental components of transformers is self-attention. It's a tool that helps to the network for emphasizing the hierarchies and alignments found in input data by quantifying paired entity interactions [22]. Figure 4 shows the result of visualizing the attention maps for a test sample image from the dataset we use in this work.

Evaluation metrics

The evaluation criteria used to determine the effectiveness of the method used are given below:

True Positive (TP): the number of times the model predicts the positive class correctly.

True Negative (TN): the number of times the model accurately predicts the negative class.

False Positive(FP): number of incorrectly predicted positive class of the model.

False Negative (FN): the number of incorrectly predicted negative class of the model.

Accuracy: the ratio of correctly classified samples to all data in the dataset. The formula is given below:

$$Accuracy = (TP + TN)/(TN + TP + FN + FP)$$

Precision: it is defined as the ratio between the correctly predicted samples and the total number of samples for a given class, and is calculated as follows:

Fig. 3 A test sample and its image patches generated by the ViT model

Fig. 4 A test sample and its attention map generated by the ViT model

$$Precision = TP/(TP + FP)$$

Recall: what proportion of the total positives are anticipated to be positive. It is equivalent to TPR (true positive rate):

$$Recall = TP/(TP + FN)$$

F1 score: the Harmonic Mean between precision and recall is known as the F-measure or F1 Score. F1 Score has a range of [0, 1]. It reveals the precision and durability of your classifier (the proportion of correctly classified cases) (it does not miss a significant number of instances [24]).

$$F1 - Score = (2 * Precision * Recall)/(Precision + Recall)$$

Matthews correlation coefficient (MCC) is a measure of the quality of binary (two-class) classifications. It is commonly used in machine learning and bioinformatics to evaluate the performance of classification algorithms, especially when the classes are imbalanced.

MCC takes into account four values from a confusion matrix: true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). It is defined as:

$$MCC = (TP * TN - FP * FN)/sqrt((TP + FP) * (TP + FN) * (TN + FP) * (TN + FN))$$

CNN approach

In the CNN experiments of this study, DenseNet121 CNN architecture was preferred. A global average pooling layer was added on top of the convolutional layers of this model, and then a fully connected layer as a classifier. The model architecture used is shown in Fig. 5.

The model has been fine-tuned over 50 epochs, the parameters of the train in the fine tuning stage are as follows. First of all, the images in the dataset are reduced to 224 × 224 size and 1615 images are reserved for training, 400 images are reserved validation and 506 images are reserved for testing. During the training and validation phases, data augmentation methods were applied while the images were fed to the DenseNet121 model. A variable learning rate strategy was adopted during the training phase, which lasted a total of 50 epochs: initially the learning rate parameter was set to 0.00001, and it is gradually increased at the end of each epoch to 0.00005 at the end of the 5th epoch. After this epoch, learning rate were gradually increased in the remaining 45 epochs and the learning rate parameter was set to 0.00001 again at the end of the 50th epoch. The accuracy and loss values observed during the training and validation stages are summarized in the graphics below (Fig. 6).

After the training and validation phases were completed, the fine-tuned CNN was used on the test dataset and its performance was evaluated. The results show that the precision value of the test phase is 0.9287, the recall value is 0.9216, and the f1-score value is 0.9209. Details of the test results are given in the table below (Table 4).

Results

In this study, three different approaches were adopted for disease detection from sugarcane leaves and the test results of each approach were compared. The experiments of each approach, which we will give details below, were performed on virtual servers equipped with Nvidia Tesla T4 GPU. Additionally, the results are presented in Table 3 the comparison of the performance of the various approaches.

Table 3 Confusion matrix of all models

Method (DenseNet121 / ViT / ViT+DenseNet121)		Predicted Class: Healthy	Mosaic	Red Rot	Rust	Yellow
Actual Class	Healthy	102 / 99 / 98	2 / 6 / 5	0 / 0 / 0	0 / 0 / 0	1 / 0 / 2
	Mosaic	0 / 1 / 0	92 / 91 / 86	0 / 0 / 1	0 / 0 / 2	1 / 1 / 4
	Red Rot	0 / 0 / 1	1 / 0 / 1	101 / 103 / 99	1 / 1 / 2	1 / 0 / 2
	Rust	0 / 1 / 3	1 / 1 / 0	9 / 8 / 19	88 / 87 / 79	5 / 6 / 2
	Yellow	0 / 0 / 0	0 / 1 / 1	17 / 10 / 29	1 / 0 / 2	83 / 90 / 69

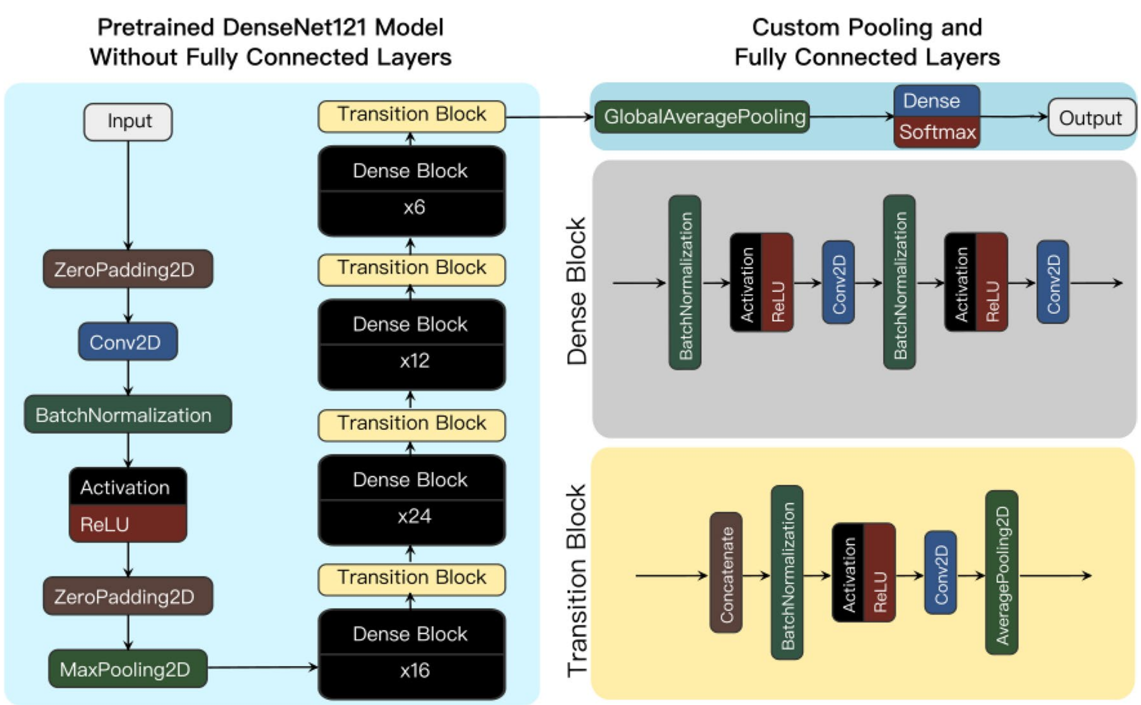

Fig. 5 The architecture of the CNN model used in this study

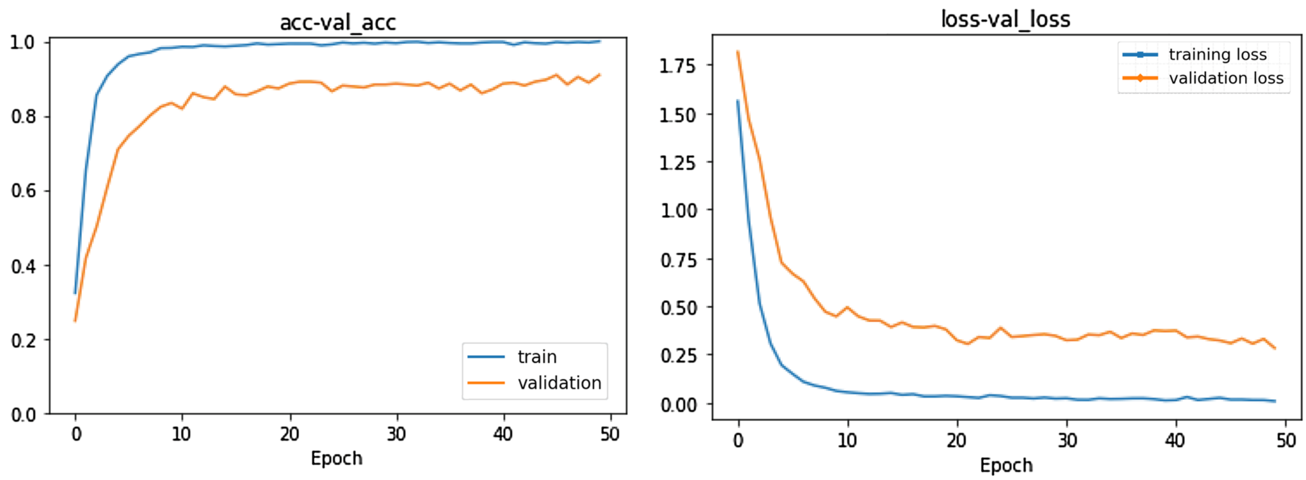

Fig. 6 Graphs showing the changes in the accuracy, validation accuracy, loss and validation loss data observed as a result of the experiment with the DenseNet121 architecture on epoch basis

Table 4 Performance metrics observed as a result of experimentation with DenseNet121 architecture

	Precision	Recall	F1-score	Test samples
Healthy	1.0000	0.9714	0.9855	105
Mosaic	0.9583	0.9892	0.9735	93
Red rot	0.7953	0.9712	0.8745	104
Rust	0.9778	0.8544	0.9119	103
Yellow	0.9121	0.8218	0.8646	101
Weighted average	0.9287	0.9216	0.9209	506
Accuracy			0.9198	

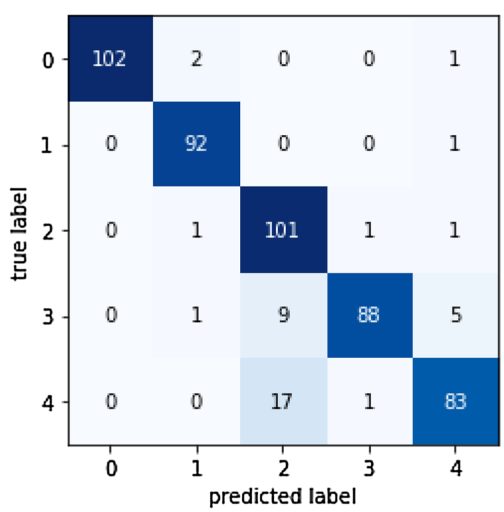

Fig. 7 The confusion matrix of the experiment with DenseNet121 model

The confusion matrix formed as a result of the test phase is given in Fig. 7 below, the accuracy of the above table can be easily observed using this matrix.

ViT approach

In vision transformers (ViT) architecture experiments, the vit_b16 model was preferred, after adding some custom dense layers on the transformer layers of this model, the model was fine-tuned over 50 epochs. First, a batch normalization layer was added on top of the transformer layers, and then three fully connected layers were added consecutively. Unit size parameters of these layers are 256 and activation functions are selected as ReLu. In addition, in one of these dense layers, the L2 regulator was preferred with the kernel, activity and bias regulation parameters $l = 0.006$. After these three dense layers, a dropout layer with a ratio of 0.3 and finally a dense layer with softmax activation were added to the model as a classifier. The structure of the created model is shown in Fig. 8 below.

Other parameters of the training phase with ViT model were chosen in the same way as the DenseNet121 case. The accuracy and loss values resulting from the training and validation stages of this approach are summarized in the graphs given in Fig. 9.

As a result of the experiments performed on the test dataset with this approach, it was observed that the precision value was 0.9334, the recall value was 0.9295, and the f1-score value was 0.9289. The details of the test results can be observed with Table 5.

Confusion matrix of the test dataset resulting from the tests with the fine-tuned ViT model approach is given in

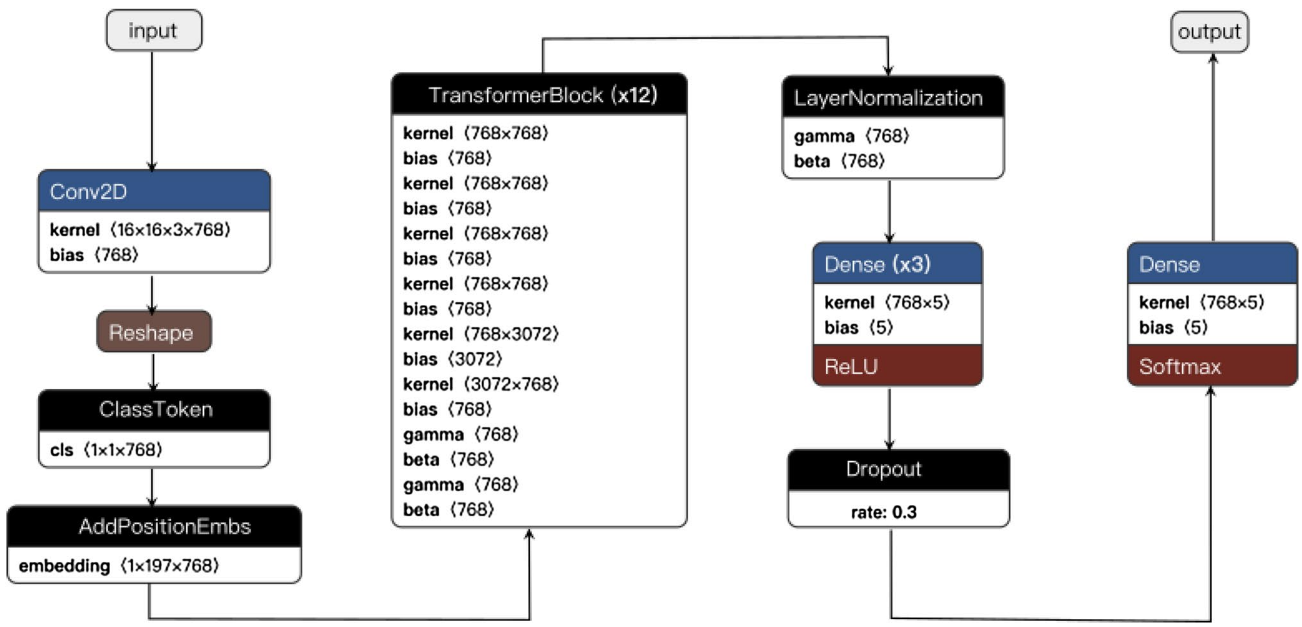

Fig. 8 The architecture of the ViT model used in this work

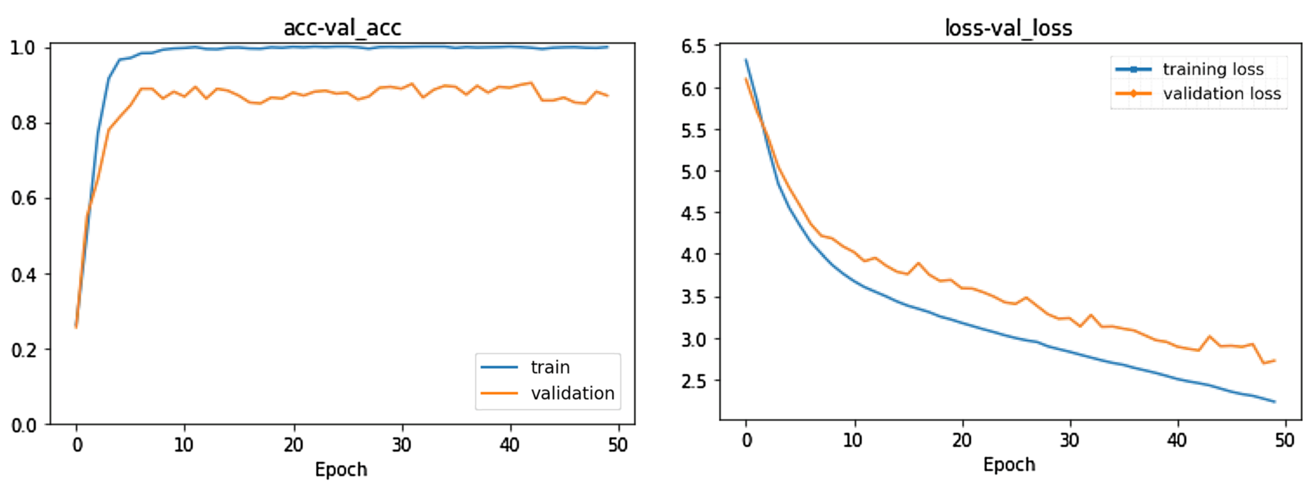

Fig. 9 Graphs showing the changes in the accuracy, validation accuracy, loss and validation loss data observed as a result of the experiment with the ViT architecture on epoch basis

Table 5 Performance metrics observed as a result of experimentation with ViT architecture

	Precision	Recall	F1-score	Test samples
Healthy	0.9802	0.9429	0.9612	105
Mosaic	0.9192	0.9785	0.9479	93
Red rot	0.8512	0.9904	0.9156	104
Rust	0.9886	0.8447	0.9110	103
Yellow	0.9278	0.8911	0.9091	101
Weighted average	0.9334	0.9295	0.9289	506
Accuracy			0.9285	

Fig. 10. It is understood that this approach outperforms the CNN approach.

ViT + CNN combination approach

In this approach, features were taken by applying transformer blocks of ViT structure (without fine tune operation) to the entire dataset and an intermediate dataset was obtained by saving these obtained attention maps as new images. Images of this dataset are in the form of attention maps shown in Fig. 4. Then, the fine tuning process

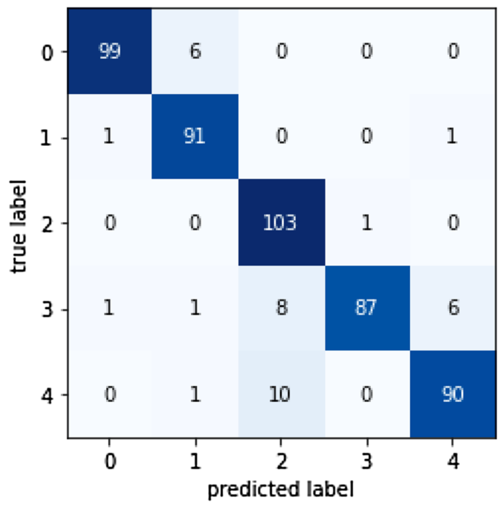

Fig. 10 The confusion matrix of the test experiment with using the ViT model

described in "CNN approach" was repeated using the DenseNet121 CNN architecture on this new dataset. The accuracy and loss values obtained in the training and validation stages with this method are shown in the graphics given in Fig. 11.

The performance of this approach on test data is detailed in the table below (Table 6). As can be seen, the pure ViT approach and the pure CNN approach show higher performance than this method.

The confusion matrix resulted from the tests on the test dataset with this method is given in Fig. 12.

Discussion

In this study, the detection of leaf diseases was studied on a dataset with 2521 pictures consisting of leaves of sugar cane plants. The dataset is almost evenly distributed, consisting of one healthy and four diseased groups. Transfer learning methods are adopted for the detection of diseases. First of all, DenseNet121 model, one of the popular CNN models, was used for this purpose, and the test results were observed by fine-tuning the model. Then, the ViT method, which is a relatively new method, was used, and the test results were observed with the fine-tuned ViT architecture. Finally, a combination method consisting of sequential use of ViT and CNN architectures was applied. When the test results were examined, it was concluded that the highest performance was obtained with the ViT model. Comparison matrix is used to compare the performance of different models by using their performance metrics such as precision, recall, F1-score and accuracy. By using the

Table 6 Performance metrics observed as a result of experiment with ViT + DenseNet121 architecture

	Precision	Recall	F1-score	Test samples
Healthy	0.9608	0.9333	0.9469	105
Mosaic	0.9247	0.9247	0.9247	93
Red rot	0.6689	0.9519	0.7857	104
Rust	0.9294	0.7670	0.8404	103
Yellow	0.8846	0.6832	0.7709	101
Weighted average	0.8737	0.8520	0.8518	506
Accuracy			0.8828	

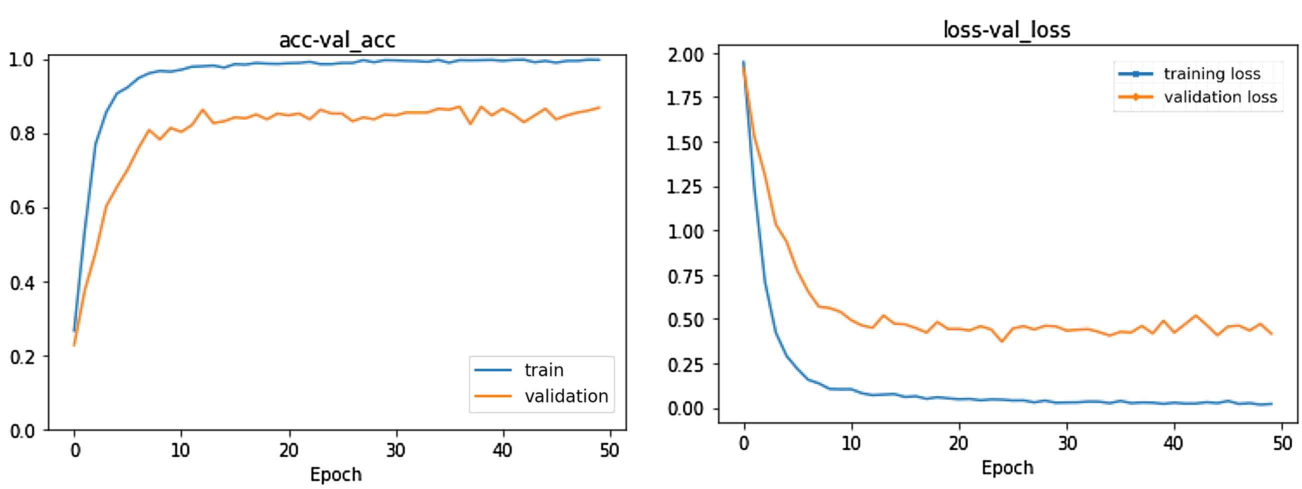

Fig. 11 Graphs showing the changes in the accuracy, validation accuracy, loss and validation loss data observed as a result of the experiment with the ViT + DenseNet121 architecture on epoch basis

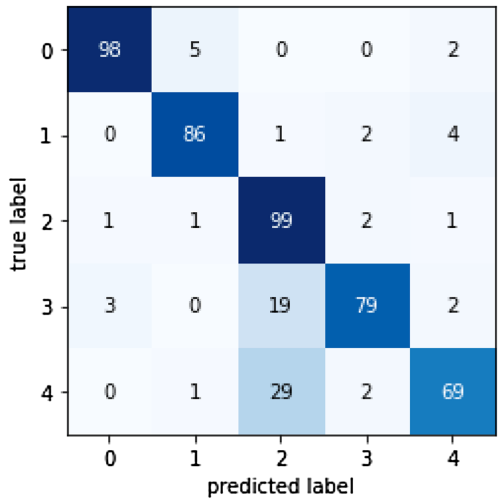

Fig. 12 Confusion matrix formed as a result of testing with the combination of ViT + DenseNet121

Table 7 Performance metrics for each of the three approaches considered in this study

	Precision	Recall	F1-Score	Accuracy	MCC
DenseNet121	0.9287	0.9216	0.9209	0.9198	0.824
ViT	0.9334	0.9295	0.9289	0.9285	0.869
ViT + DenseNet121	0.8737	0.8520	0.8518	0.8828	0.677

given tables, we can create a comparison matrix to easily compare the performance of three different models. The matrix is based on the weighted average values of each performance metric. Table 7 summarizes the performance metrics of these methods.

According to the comparison matrix, ViT has the highest precision, recall, F1-score and accuracy values among the three models. Additionally, ViT also has the highest MCC value, which indicates a better overall performance than the other models. DenseNet121 has MCC value that is also considerably high, however, its F1-score and accuracy values are slightly lower than ViT. ViT + DenseNet121 has the lowest precision, recall, F1-score and MCC values, while its accuracy value is a bit higher than the other two models.

MCC is an important performance metric as it takes into account both true positive and true negative rates, and provides a more balanced assessment of a model's performance. Therefore, the inclusion of MCC values in the comparison matrix allows for a more comprehensive understanding of the models' performances.

Declarations

Conflict of interest The authors have no relevant financial or non-financial interests to disclose. The authors did not receive support from any organization for the submitted work.

Compliance with ethics requirements This article does not contain any studies with human and animal.

References

1. Murugeswari R, Anwar ZS, Dhananjeyan VR, Karthik CN (2022) Automated sugarcane disease detection using faster rcnn with an Android application. 6th International Conference on Trends in Electronics and Informatics (ICOEI). 1:1–7, https://doi.org/10.1109/ICOEI53556.2022.9776685
2. Li X, Li X, Zhang S, Zhang G, Zhang M, Shang H (2022) SLViT: shuffle-convolution-based lightweight Vision transformer for effective diagnosis of sugarcane leaf diseases. J King Saud Univ. https://doi.org/10.1016/j.jksuci.2022.09.013
3. Militante SV, Gerardo BD (2019) Detecting sugarcane diseases through adaptive deep learning models of convolutional neural network. IEEE 6th International Conference on Engineering Technologies and Applied Sciences (ICETAS) 1:1–5, https://doi.org/10.1109/ICETAS48360.2019.9117332
4. Hernandez AA, Bombasi JL, Lagman AC, (2022) Classification of Sugarcane Leaf Disease using Deep Learning Algorithms. IEEE 13th Control and System Graduate Research Colloquium (ICSGRC), 1:47–50, https://doi.org/10.1109/ICSGRC55096.2022.9845137
5. Alencastre-Miranda M, Johnson RM, Krebs HI (2021) Convolutional neural networks and transfer learning for quality inspection of different sugarcane varieties. IEEE Trans Industr Inf 17(2):787–794. https://doi.org/10.1109/TII.2020.2992229
6. Daphal SD, Koli SM (2021) Transfer learning approach to sugarcane foliar disease classification with state-of-the-art sugarcane database. Int Conf Comput Intell Comput Appl (ICCICA) 1:1–4. https://doi.org/10.1109/ICCICA52458.2021.9697312
7. Militante SV, Gerardo BD, Medina RP (2019) Sugarcane disease recognition using deep learning. IEEE Eurasia Conference on IOT, Communication and Engineering (ECICE) 1:575–578, https://doi.org/10.1109/ECICE47484.2019.8942690
8. Saavedra-Burbano RA, Marin-Hurtado JI (2020) Evaluation of deep learning architectures for the detection of rust in sugarcane crops. Virtual Symp Plant Omics Sci (OMICAS) 1:1–5. https://doi.org/10.1109/OMICAS52284.2020.9535653
9. Malik HS et al (2021) Computing disease recognition in sugarcane crop using deep learning. In: Kacprzyk J (ed) Advances in Intelligent Systems and Computing. Springer, Singapore, pp 189–206
10. Sharma R, Kukreja V (2022) segmentation and multi-layer perceptron: an intelligent multi-classification model for sugarcane disease detection. Int Conf Decis Aid Sci Appl (DASA) 1:1265–1269. https://doi.org/10.1109/DASA54658.2022.9765191
11. Chen W, Ju C, Li Y, Hu S, Qiao X (2021) Sugarcane stem node recognition in field by deep learning combining data expansion. Appl Sci 11(18):8663. https://doi.org/10.3390/app11188663
12. Tamilvizhi T, Surendran R, Anbazhagan K, Rajkumar K (2022) Quantum behaved particle swarm optimization-based deep transfer learning model for sugarcane leaf disease detection and classification. Math Probl Eng 2022:3452413. https://doi.org/10.1155/2022/3452413

13. Grijalva I, Spiesman BJ, McCornack B (2023) Image classification of sugarcane aphid density using deep convolutional neural networks. Smart Agric Technol 3:100089. https://doi.org/10.1016/j.atech.2022.100089
14. Daphal SD, Koli, SM (2022), Sugarcane leaf disease dataset, Mendeley Data, V1 https://doi.org/10.17632/9424skmnrk.1
15. Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. Adv Neural Inf Process Syst 25(2):1106–1114
16. Singh D, Taspinar YS, Kursun R, Cinar I, Koklu M, Ozkan IA, Lee H-N (2022) Classification and analysis of pistachio species with pre-trained deep learning models. Electronics 11(7):981. https://doi.org/10.3390/electronics11070981
17. Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S, Uszkoreit J, Houlsby N (2020) An image is worth 16x16 words: transformers for image recognition at scale. arXiv. https://doi.org/10.48550/arXiv.2010.11929
18. Nimsuk N, Paewboontra W (2021) Compact cnn model for classifying rose apple species and detecting their skin defects. 18th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON). 1:136–139, https://doi.org/10.1109/ECTI-CON51831.2021.9454852
19. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ (2017) Densely connected convolutional networks. Proc IEEE Conf Comput Vision Pattern Recogn 1:4700–4708. https://doi.org/10.48550/arXiv.1608.06993
20. Gupta A, Pawade P, Balakrishnan R (2022) Deep residual network and transfer learning-based person re-identification. Intell Syst Appl 16:200137. https://doi.org/10.1016/j.iswa.2022.200137
21. Haurum JB, Madadi M, Escalera S, Moeslund TB (2022) Multiscale hybrid vision transformer and Sinkhorn tokenizer for sewer defect classification. Autom Constr 144:104614. https://doi.org/10.1016/j.autcon.2022.104614
22. Han K et al (2022) A survey on vision transformer. IEEE Trans Pattern Anal Mach Intell. https://doi.org/10.1109/TPAMI.2022.3152247
23. Alhawas N, Tüfekci Z (2022) The identification of red-meat types using the fine-tuned vision transformer and mobilenet models. Eur J Sci Technol 36:237–242
24. Kathamuthu ND, Subramaniam S, Le QH, Muthusamy S, Panchal H, Sundararajan SCM, Alruabie AJ, Zahra MMA (2022) A deep transfer learning-based convolution neural network model for COVID-19 detection using Computed tomography scan images for medical applications. Adv Eng Softw 175:103317. https://doi.org/10.1016/j.advengsoft.2022.103317