agriculture

MDPI

Article

Classification of Cassava Leaf Disease Based on a Non-Balanced Dataset Using Transformer-Embedded ResNet

Yiwei Zhong, Baojin Huang and Chaowei Tang *

School of Microelectronics and Communication Engineering, Chongqing University, Chongqing 400044, China
* Correspondence: cwtang@cqu.edu.cn

Abstract: Cassava is a typical staple food in the tropics, and cassava leaf disease can cause massive yield reductions in cassava, resulting in substantial economic losses and a lack of staple foods. However, the existing convolutional neural network (CNN) for cassava leaf disease classification is easily affected by environmental background noise, which makes the CNN unable to extract robust features of cassava leaf disease. To solve the above problems, this paper introduces a transformer structure into the cassava leaf disease classification task for the first time and proposes a transformer-embedded ResNet (T-RNet) model, which enhances the focus on the target region by modeling global information and suppressing the interference of background noise. In addition, a novel loss function called focal angular margin penalty softmax loss (FAMP-Softmax) is proposed, which can guide the model to learn strict classification boundaries while fighting the unbalanced nature of the cassava leaf disease dataset. Compared to the Xception, VGG16 Inception-v3, ResNet-50, and DenseNet121 models, the proposed method achieves performance improvements of 3.05%, 2.62%, 3.13%, 2.12%, and 2.62% in recognition accuracy, respectively. Meanwhile, the extracted feature maps are visualized and analyzed by gradient-weighted class activation map (Grad_CAM) and 2D T-SNE, which provides interpretability for the final classification results. Extensive experimental results demonstrate that the method proposed in this paper can extract robust features from complex non-balanced disease datasets and effectively carry out the classification of cassava leaf disease.

Keywords: cassava diseases; intelligent agricultural engineering; convolutional neural network; focal angular margin penalty softmax loss (FAMP-Softmax); transformer-embedded ResNet (T-RNet); unbalanced image samples

Citation: Zhong, Y.; Huang, B.; Tang, C. Classification of Cassava Leaf Disease Based on a Non-Balanced Dataset Using Transformer-Embedded ResNet. *Agriculture* **2022**, *12*, 1360. https://doi.org/10.3390/agriculture12091360

Academic Editors: Xiuliang Jin, Hao Yang, Zhenhai Li, Changping Huang and Dameng Yin

Received: 8 August 2022
Accepted: 28 August 2022
Published: 1 September 2022

1. Introduction

The root system of cassava grows on many continents, including Africa, Asia, and South America [1] because of its ability to adapt to harsh soil conditions and complex climates. In Africa, cassava is grown in large quantities for food supply and economic consumption [2]. In addition, due to its rich protein and starch contents, cassava is widely used for starch processing, paper production, feed production, and edible use [3]. Due to its high economic and nutritional value, the utilization and development of cassava have become a focus of contemporary tropical agricultural science and technology. However, cassava leaf disease, such as cassava green mite (CGM), cassava bacteria blight (CBB), cassava brown streak disease (CBSD), cassava mosaic disease (CMD), cassava American latent leaf disease (CALD), cassava brown streak Uganda disease (CBSUD), and cassava Colombian symptomless disease (CCSD), have seriously affected cassava production and caused substantial economic losses [4]. Manual diagnosis of cassava leaf disease is a costly and inefficient method of diagnosis. Therefore, to reduce the expenses of farmers and increase cassava production, there is a need to find more accurate methods for the classification of cassava leaf disease.

Techniques based on image processing can be effective for the timely diagnosis of plant diseases. Among them, Smith et al. [5] developed an algorithm based on image processing

that automatically identifies visual symptoms of plant diseases. This method analyzes color images and extracts image features from disease sites to identify disease-causing factors in plants. A. Meunkaewjinda et al. [6] also proposed an intelligent system based on color image blending to classify scab, rust, and disease-free grape leaves. However, when the images are affected by light and angle, it is not easy to extract the high-level semantic information of the images by traditional methods, and the classification task will face a considerable challenge.

Furthermore, a lot of methods combining computer vision, image processing, and machine learning have been proposed by researchers in plant leaves disease classification. Bracino et al. [7] used graph cut segmentation aided by lazy snapping process to separate background from the apple leaf images, and it extracted a total of 12 color and texture features from which only 3 features are selected using Neighborhood Component Analysis (NCA). Pravin Kumar et al. [8] separated background and foreground by Gaussian Mixture Model (GMM), which was used to model each pixel in the frame into Gaussian distribution. Diseased regions are segmented using Particle Swarm Optimization (PSO) based Fuzzy C-means algorithm. They proposed Multi-Kernel Parallel Support Vector Machine (MK-PSVM) as a classifier. Emrullah Acar et al. [9] extracted cassava leaves multi-feature textures using local binary pattern (LBP), histogram of oriented gradients (HOG), texture energy map (TEM), Gabor Wavelet Transform (GWT), and gray level co-occurrence matrix (GLCM). Then, the obtained feature vectors from relevant feature extraction algorithms were put through a distributed structural-based *k*-NN classifier. However, as is known to all that in the field of image recognition, extracting image features is the most critical part of the pattern recognition system. The quality of feature extraction directly affects the final recognition rate of the system. Most of the machine learning methods deal with data in shallow structure. These structural models have only one or two layers of nonlinear feature transformation at most, the features acquired from shallow structure are designed by hand. It is difficult to make use of the advantages of big data by relying on the prior knowledge and parameter adjustment experience of designers.

In recent years, besides these classical machine learning methods, deep learning techniques based on CNN have occupied an important position in plant disease classification, which provides a new approach to the development of disease classification for automated classification systems. This approach uses multilayer neural networks and optimization algorithms to train data to perform classification or regression tasks, among others, and many research works applied CNN techniques to cassava leaf disease classification [10–13]. For example, Ramcharan et al. [11] employed the inception-v3 model for transfer learning to identify three diseases and two instances of pest damage on a dataset of cassava disease images taken in Tanzanian fields, where support vector machine (SVM) and K-nearest neighbor algorithm (KNN) were used to evaluate the accuracy of the system for classification. The detection results showed that the proposed method has good overall performance in terms of the classification accuracy and the confusion matrix. However, its performance and efficiency are low when dealing with unbalanced samples. Isaman Sangbamrung et al. [13] creatively used the view of detection to localize disease images. By proposing a multi-model cascade approach, high accuracy detection of CBSD diseases is achieved. However, the primary deficiency is that the use of detection will increase the workload of manual annotation and annotation error rate, and it is only used in cases of apparent and bounded category features, and cannot effectively annotate other categories. Without specific image feature patterns, it cannot effectively annotate other categories, such as CSBD, CMD, and other diseases. In addition, this approach mainly focuses on classifying CBSD diseases and lacks the ability to classify other disease categories, it is difficult to determine the generalization and robustness of the model. G. Sambasivam et al. [12] used the synthetic minority oversampling technique (SMOT) and focal loss function to train CNNs for cassava disease classification. However, the author did not mention the effects of SMOT and focal loss on the model performance, and the structure of the model proposed in the article is relatively simple and challenging to apply to the complex cassava leaf disease dataset.

Aiming at the plant leaves disease classification task, there are a number of works achieving state-of-the-art performance, such as Inception-v3 [14], VGG [15], MobileNetV2 [16,17], and DenseNet-121 [18,19]. However, for some cassava leaf diseases, such as CGM, CMD, and CBB, leaves suffering from these diseases have similar symptoms and it is difficult to distinguish disease category from image textures, furthermore many image samples contain environmental background noise. Most of these regions include plants or leaves unrelated to specific classes, this makes it difficult for convolutional neural network-based models to learn accurate and discriminative features. These models do not effectively use global information to focus on important target regions. In addition, due to the large imbalance in the samples of multiple disease categories in the dataset, it will further increase the difficulty of model training, resulting in overfitting of the model to categories with a large number of samples.

Therefore, to solve the problems mentioned above, this study first introduces the transformer into the cassava disease classification task to learn more robust disease features by fusing local and global feature information. Because it can achieve long-range information modeling and compensate for the deficiencies of CNNs, the model considers typical local features of the disease and observes the information of adjacent or even global image regions. Therefore, this study proposes T-RNet based on the transformer to prevent the model from learning local features with background interference, which leads to the model overfitting the local features and ignoring the critical target regions. In addition, the dataset used in this paper has the problem of an unbalanced number of samples of disease categories, which is not conducive to the feature learning of disease categories with few samples. In order to better learn the feature differences between different disease categories, inspired by the focal loss [20] and ArcFace loss [21], this paper proposes a fusion loss function named focal angular margin penalty softmax loss (FAMP-Softmax). Because of the inherited properties of focal loss, this loss function can help to distinguish between complicated and simple samples, while also reduce the sample imbalance problem. At the same time, using angle space instead of the Euclidean distance can further increase the feature distance between disease categories and increase the clustering within classes. Experimental results demonstrate that the FAMP-Softmax loss function can help the model to combat the imbalanced sample problem.

The main contributions are as follows:

- In the cassava leaf disease detection task, transformer structure is introduced for the first time to pay attention to the global information and prevent the model from overfitting to the local background noise regions, and a new convolution network model (T-RNet) integrating the advantages of ResNet and transformer is proposed, which can extract more discriminative features. Experimental results for performance tests show that the proposed model has better classification performance than the popular commonly used CNN model;
- A new loss function (FAMP-softmax) is proposed to solve the problem of class non-balance in cassava leaf disease datasets. According to the accuracy and F1-score, the FAMP-Softmax performance is better than cross-entropy and focal loss function based on Resnet-50 and T-RNet;
- The interpretability of CNN feature extraction method for cassava leaf disease classification is discussed by using Grad_CAM attentional map and T-SNE visualization technology.

2. Related Work

2.1. Attention Mechanism and Transformer

The attention mechanism, a vital component to improve the performance of CNNs, is a simulation of biological attention mechanisms that enables CNNs to model the relationships of all input elements. Attention mechanisms can enhance information-rich features and suppress information-irrelevant features. Currently, many networks have achieved strong results based on the attention mechanism. Among them, Hu [22] used a fully connected layer to model feature-channel relationships; Wang et al. [23] used a

one-dimensional convolution instead of a fully connected layer to reduce the parameters of the model; and Woo et al. [24] used a combination of spatial and channel attention mechanisms to further improve the performance of the model. In 2017, Google proposed a transformer network model based on a self-attention mechanism. Instead of using a long short-term memory (LSTM) model structure [25], the entire model is formed by the self-attention module based on an encoder and decoder architecture. In natural language processing (NLP) tasks, the transformer's network performance far exceeds that of LSTM. Afterward, many models began to use similar architectures and modules for NLP tasks, such as GPT [26]. The transformer's success in NLP tasks led researchers to employ it in computer vision tasks to compensate for the shortcomings of CNNs, such as BoTNet [27], DETR [28], VideoBERT [29], and VILBERT [30]. Inspired by this, this study introduces the transformer module into the cassava leaf disease classification task, which enables the model to focus on the interrelationships among information types in different regions to capture useful feature information and to enable the network model to learn discriminative and robust features.

2.2. Classification Loss Function

The classification loss function is essential for guiding deep convolutional neural networks (DCNNs) toward better training. Most classification tasks are based on DCNNs and supervised by classification loss functions. The softmax loss function is considered as the classical classification loss function and has been applied in many fields. However, the softmax loss function has several shortcomings. Most notably, the classification boundaries is not strict enough [21]. Many researchers believe that softmax loss is effective in optimizing interclass differences, but it cannot reduce intraclass differences [21,31–33]. In recent years, there have been many efforts to improve the softmax loss function for classification. A-Softmax [33] normalizes the weights to make DCNN focus on the angle information. L-Softmax [31] applies a multiplicative penalty factor to the angle between weights and features. Wang et al. [32] applied some penalties to the cosine value to enhance the intraclass compactness. However, Deng et al. [21] argued that a direct penalty on angles would yield a better performance in the classification task, and they proposed an improved softmax called Arcface loss [21]. The related improved softmax loss function focuses on producing an acceptable classification margin by penalizing the cosine values or angles.

The improved softmax loss can increase the category spacing very well. However, it still does not achieve satisfactory results for the data imbalance problem. This is because it does not address the problem that the model will tend to predict the categories in which the sample occurs more frequently [20]. Currently, in the field of deep learning, methods for dealing with imbalanced samples can be broadly classified into two categories: balanced sampling methods and weight adjustment methods. Hermans et al. [34] first proposed a PK sampling strategy for the human reidentification task to solve the imbalanced sample problem. Performing sampling strategies in classification tasks will undoubtedly increase time and space resources. Focal loss was first proposed for unbalanced target detection, which was based on a weight assignment strategy to address the problem with unbalanced datasets. Currently, it has been widely used in related classification tasks.

Focal loss solves problems with unbalanced datasets, but it does not produce stricter classification boundaries. The improved softmax loss function considers classification boundaries, but it is not suitable for handling balanced datasets. Therefore, there is an urgent need to explore a loss function that can handle unbalanced datasets and optimize stricter classification boundaries. This study propose a novel loss function, focal angular margin penalty softmax loss (FAMP-Softmax), by combining two types of classification loss functions to meet our task challenge.

3. Proposed Methods

This section will first present the overall architecture of the proposed method of this study, then present the specific details of the proposed network model, and finally present

the details of the ideas and principles of the proposed loss function. The overall framework of the proposed method of this study is shown in Figure 1.

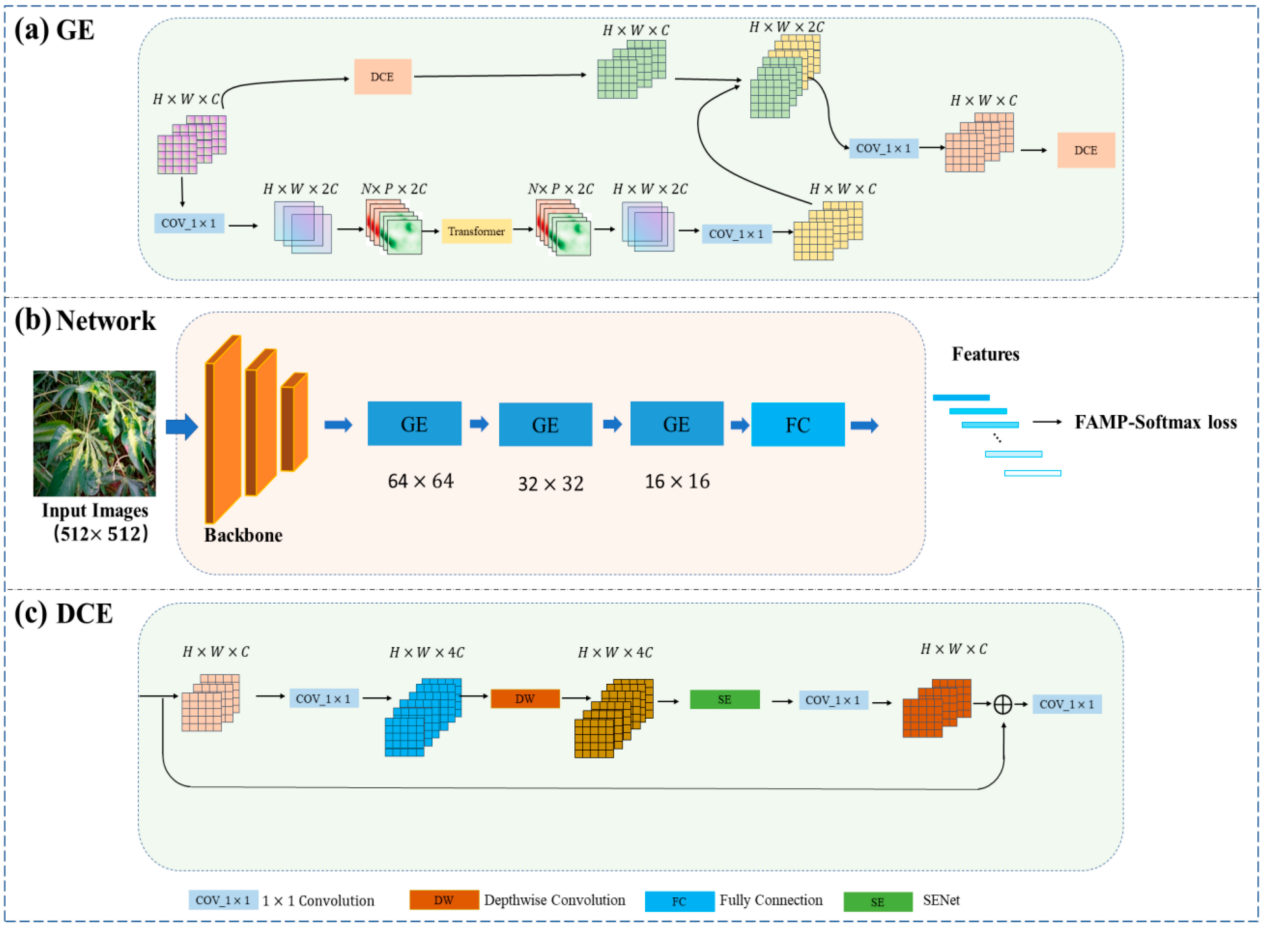


Figure 1. The overall architecture of our method. The images are sequentially processed by the backbone (first two stages of ResNet18) and GE, and the extracted features are then fed into the fully connected layer and our proposed loss function FAMP-Softmax loss to obtain the final classification results. 512 × 512 denotes the size of the input images, and 64 × 64, 32 × 32, and 16 × 16 denote the size of the feature map.

3.1. The General Architecture of T-RNet

Similar characteristics exist between different disease categories, such as leaf wilt and shrinkage symptoms in both CGM and CBB, white or white patchy symptoms in CMD and CGM, and yellow patches in CBSD and CMD. Additionally, environmental background noise is present in a large number of samples in the dataset, which makes it difficult for the model to accomplish correct classification. The current mainstream approaches are based on CNNs, such as VGG16 [15], ResNet-50 [35], DenseNet121 [18,19,36], Xception [37], and Inception-v3 [14]. The results achieved by these networks for this tricky task are not satisfactory because CNNs have a strong ability to extract local information. However, we believe that too much local feature information is likely to mislead the model learning, and there is background noise in the image, which can cause the model to learn the wrong information and ignore the important region information. To change the model's excessive focus on the wrong local region perspective, this study proposes guiding the model to learn a more robust feature representation from a more global region feature, thus requiring the model to display long-range global information. To solve this problem, this study proposes a novel network model called T-RNet to increase the model's ability to focus on global information. GE (global information extraction module) is mainly responsible for

extracting global information and channel information to enhance the feature representation and to improve the classification effect of the model. Specifically, the transformer is introduced into the GE module to help the model to extract the long-range information and to combine it with the CNNs to extract local features to obtain a more powerful feature representation. The DCE module introduces the channel attention mechanism to obtain better channel feature information and uses deep separable convolution to further reduce the model parameters, which is beneficial for edge deployment. In the proposed whole network architecture of this study, the images are first fed into the backbone (ResNet18) and pretrained on ImageNet with layer 3 and layer 4 removed. Then, the feature vectors learned through GE and the fully connected layer are sent to our proposed loss function to obtain classification results.

The structure of the GE module is shown in Figure 1a, which aims to extract local and global information with a convolution and a transformer. For a given input tensor $X \in \mathbb{R}^{H\times W\times C}$, GE applies a 1×1 convolution to generate $X_K \in \mathbb{R}^{H\times W\times 2C}$. The 1×1 convolution projects the tensor to a high-dimensional space to obtain high-dimensional channel information. Then, it changes X_K into N nonoverlapping tensor blocks $X_G \in \mathbb{R}^{N\times P\times C}$ for transformer module learning by dimensionality change. Here, $P = w \times h$ and h, w are the width and height of the feature map of each tensor block, respectively. $N = \frac{H\times W}{P}$ is the number of tensor blocks. To learn the relationship between N nonoverlapping tensor blocks, each tensor block is encoded and sent to the transformer module to obtain the output $X_O \in \mathbb{R}^{N\times P\times C}$.

$$X_O = Transformer(X_G) \tag{1}$$

The transformer structure is shown in Figure 2, which consists of a norm layer, a multiheaded attention (MHA) layer, and a feedforward transmission layer. The norm layer mainly performs the normalization operation on the input feature codes, and the MHA layer mainly learns the weight assignment of different dimensions of the feature codes to obtain the interrelationships between different codes. Finally, the feedforward transmission layer performs feature mapping, while preserving the feature vector dimension alignment. Because it has a self-attentive structure, it is commonly used in vision tasks to learn different image block feature encodings to obtain attentional information about the whole image [38]. Once the information encoding is obtained, this study uses a symmetric 1×1 convolutional structure to obtain the feature tensor in the same dimension as the input. However, unlike convolution, GE transforms the feature map into different tensor blocks and goes through the transformer to learn the relationships between the different tensor blocks to obtain more detailed information about the feature map. In this case, the size of the feature map is usually extremely small, and it is not too costly in terms of model computation.

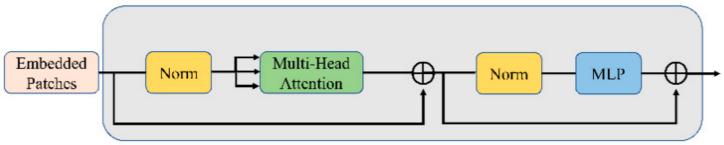


Figure 2. The structure of the transformer module.

Meanwhile, to make full use of the information of the feature maps, the feature maps containing the information of original features and channel features are obtained by the DCE convolution module for the above branch. Finally, the feature maps of different branches are stitched together in the channel dimension by a 1×1 convolution operation to interact with different information. The GE module combines the convolution and transformer to obtain a better feature representation, which is beneficial for easier model convergence.

The DCE module is shown in Figure 1c, which introduces deep separable convolution (DW) and squeeze-extraction (SE) blocks to extract features, while reducing the number of model parameters compared to traditional convolution. The DCE module is

used to obtain richer channel and semantic information to complement the global information. Deep separable convolution is an important module in many CNNs models [16]. Standard convolution inputs an $h_i \times w_i \times d$ tensor X_d and applies a convolution kernel $K \in \mathbb{R}^{k \times k \times d \times l}$ to obtain the output $h_o \times w_o \times l$ tensor X_l. The overall computational cost is $h_i \times w_i \times d \times d \times k \times k$. Using deep separable convolution instead of standard convolution, the computational cost can be reduced to $h_i \times w_i \times d(k \times k + l)$. Experience shows that the performance can often reach the same level as that of the standard convolution. Therefore, applying the deep separable convolution to the DCE module can greatly reduce the computational consumption of the model and facilitate the deployment of the model.

Given an input tensor $X \in \mathbb{R}^{H \times W \times C}$, the DCE module first performs a dimensionality change by a 1×1 convolution and then obtains a new tensor $X_1 \in \mathbb{R}^{H \times W \times 4C}$ by a deep separable convolution when there is no information interaction for each channel of the tensor. Meanwhile, the DCE module uses the squeeze-extraction network (SENet) module to interact with the channel features and obtains a new channel relationship feature tensor $X_2 \in \mathbb{R}^{H \times W \times 4C}$ of the same size as the input. It mainly learns the weight of the channel features through the fully connected layer and then multiplies this weight with the feature tensor to filter out the important channel information. The extra branch is used in combination with the idea of ResNet residuals to add the input, which can further help the module achieve the utilization of multidimensional features and resolve the gradient disappearance problem. The final result $X_o \in \mathbb{R}^{H \times W \times 4C}$ is then obtained by performing a 1×1 convolution to obtain a better nonlinear representation. In addition, the 1×1 convolution can also control the size of the feature map by downsampling as needed. This helps the subsequent module output the recognition result.

3.2. Focal Angular Margin Penalty Softmax Loss

The traditional Softmax is defined as follows:

$$softmax = \frac{e^{W_{y_i}^T f_i}}{\sum_{j=1}^{c} e^{W_{y_j}^T f_i}} = \frac{e^{\|W_i\| \ \|f_i\| cos\theta_i}}{\sum_{j=1}^{c} e^{\|W_j\| \ \|f_i\| cos\theta_j}} \quad (2)$$

where $\boldsymbol{f_i}$ is the feature vector belonging to the i category before the last fully connected layer. $\boldsymbol{W_{y_i}^T}$ or $\boldsymbol{W_i}$ is the weight corresponding to the feature vector $\boldsymbol{f_i}$. $cos\theta_i$ is the cosine value, and θ_i is the interval between the weight $\boldsymbol{W_i}$ and the feature $\boldsymbol{f_i}$. $\boldsymbol{W_{y_i}^T f_i}$ is often referred to as the target Logit.

Softmax implements the optimization task on θ and $\boldsymbol{W}$. This optimization direction is somewhat vague and nonstrict for the classification task. If the optimization objective is focused on a specific variable (θ or $\boldsymbol{W}$), then the optimization direction will become more explicit and eventually improve the performance. The main intention of this paper is to obtain tighter classification boundaries. We believe that the accomplishment of this goal depends on intraclass and interclass interactions. Therefore, this study defines two classes of angular kernel functions, including the intraclass kernel function $\psi(\theta_i)$ and the interclass kernel function $\Phi(\theta_j)$. $\psi(\theta_i)$ and $\Phi(\theta_j)$ can be expressed as follows:

$$\psi(\theta_i) = cos(\theta_i + m) \quad (3)$$

$$\Phi(\theta_j) = cos(\theta_j - m) \quad (4)$$

where $m \in [0, \pi]$ denotes the angle margin, and the weights and feature vectors are normalized according to [21,31,33] to optimize the DCNN by cosine similarity. Referring to [21,31],

an adaptive variable $s > 0$ is introduced on $\psi(\theta_i)$ and $\Phi(\theta_j)$, which are automatically learned by the model. Thus, FAMP-Softmax is defined as follows:

$$p_{FAMP-i} = \frac{e^{s \cdot \psi(\theta_i)}}{e^{s \cdot \psi(\theta_i)} + \sum_{j=1, j \neq i}^{c} e^{s \cdot \Phi(\theta_j)}} \tag{5}$$

where s is the scale factor that amplifies the difference in sample distribution. p_{FAMP-i} is the predicted probability value of class i.

This study notes that focal loss is beneficial for learning unbalanced samples and can resolve the problem of an unbalanced dataset. Therefore, FAMP-Softmax is integrated into the focal loss. In this way, both the imbalance problem and the interclass and intraclass problem can be resolved to some extent. In other words, this study focus not only on the inter/intraclass embedding feature space but also on the imbalanced dataset. The final classification loss can be expressed as follows:

$$L_{FAMP} = -\frac{1}{N} \sum {}_{i}^{N} a_t (1 - p_{FAMP-i})^{\gamma} \log(p_{FAMP-i}) \tag{6}$$

$$a_t = \log \frac{\sum_{k=t}^{c} G_k}{\sum_{k \neq t}^{c} G_k} \tag{7}$$

where t denotes the true label of the current sample, k denotes a category, a_t denotes the weight value corresponding to the category to which the current sample belongs, and G_k denotes the number of samples in different categories. In the training process, to avoid the model favoring a certain category, γ = 2 can make the model learn the samples that are not easily classified.

Figure 3 shows the difference between the optimization of softmax and FAMP-Softmax loss in the binary classification case. For traditional softmax, the classification boundaries is B_s, i.e., $W_1 B_S = W_2 B_S$. For FAMP-Softmax, since the weights and feature vectors are normalized, the classification boundaries is determined by $cos(\theta_1 + m) = cos(\theta_2 - m)$ and $cos(\theta_2 + m) = cos(\theta_1 - m)$. The classification boundaries are B_{F1} and B_{F2}. We argue that the FAMP-Softmax loss relies on the intraclass and interclass penalty factors m to produce a classification effect with the margin, which acts to separate the classification boundaries more, thus achieving a better classification effect than the softmax loss. The proposed loss function of this study fully incorporates the advantages of focal loss and inter/intraclass margins, allowing the network to not only learn difficult samples and mitigate the effects of sample imbalance but to also better learn larger interclass distances, which is beneficial for the model to improve in classifying the data of different diseases.

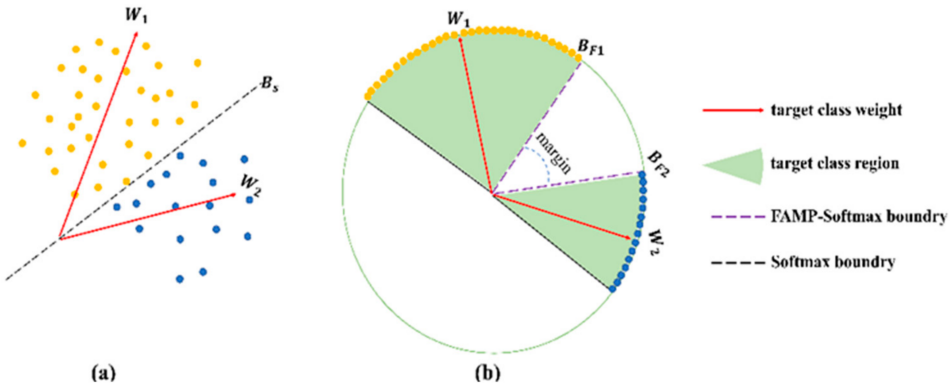


Figure 3. (**a**) Classification boundaries for Softmax (**b**) Classification boundaries for FAMP-Softmax.

4. Experiments and Analyses

Based on the proposed network model and loss function, this study conducts a large number of experiments to evaluate the effectiveness of the proposed method. The details

are divided into the following sections. First, this study introduces the experimental dataset of this paper in Section 4.1. Section 4.2 describes the performance evaluation metrics used in this paper. The experimental configuration and implementation details of this paper are presented in Section 4.3. In Section 4.4, the performance comparison between the proposed network T-RNet and other mainstream models in terms of various performance metrics is presented, while the performance advantage of the proposed loss function over other loss functions is experimentally verified. Finally, in Section 4.5, the effectiveness of the proposed method in classifying various diseases is verified by visualizing the attention graph and t-SNE visualization techniques.

4.1. Datasets

The image dataset used in the experiments is from the 2021 Kaggle competition. The competition provided a cassava leaf disease dataset containing five disease categories with 21,397 annotated images collected during regular surveys in Uganda. Most of the images were crowdsourced from garden photographs taken by farmers and annotated by experts from the National Crop Resources Research Institute (NaCRRI) in collaboration with the Artificial Intelligence Lab at Makerere University in Kampala. The cassava leaf disease dataset consists of images of cassava bacterial blight (CBB), cassava mosaic disease (CMD), cassava brown streak disease (CBSD), cassava green spot (CGM), and healthy individuals. Figure 4 shows four images of diseased and healthy cassava leaves. The correspondence between the label and the symptom images assigned in this dataset is shown in Table 1.

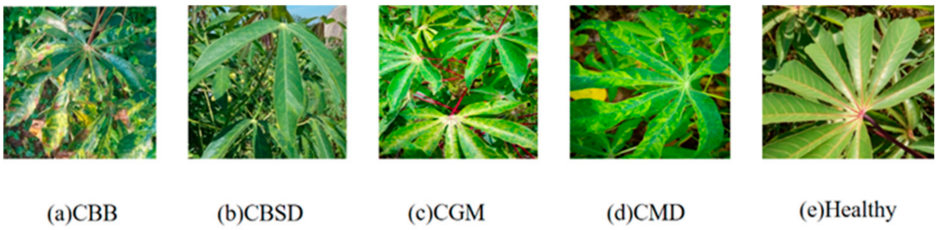


Figure 4. Images of healthy and unhealthy cassava mosaic diseases. (**a**) CBB, (**b**) CBSD, (**c**) CGM, (**d**) CMD, (**e**) healthy. Dataset URL: https://www.kaggle.com/c/cassava-leaf-disease-classification/data (accessed on 1 June 2022).

Table 1. Correspondence between disease categories and labels.

Category	Label
CBB (Cassava Bacterial Blight)	0
CBSD (Cassava Brown Streak Disease)	1
CGM (Cassava Green Mottle)	2
CMD (Cassava Mosaic Disease)	3
Healthy leaf	4

The numbers of leaves in each category of cassava leaves in the training and test datasets are shown in Table 2. A large portion of the data contains CMD and CBSD disease images, and only a small portion of the data includes healthy images. There is a category imbalance in this dataset. In addition, as shown in Figure 5, the t-SNE visualization of the dataset reveals that the distribution of different categories of the dataset is scattered and blended together, and the dataset shows a highly nonlinear separable situation, where different colors represent different categories of samples, thus it is challenging for the model to correctly classify the disease.

Table 2. Cassava leaf dataset detail.

	CBB	CBSD	CGM	CMD	Healthy	Total
Training	869	1751	1909	10,527	2061	17,117
Testing	218	438	477	2631	516	4280
Total	1087	2189	2386	13,158	2577	21,397

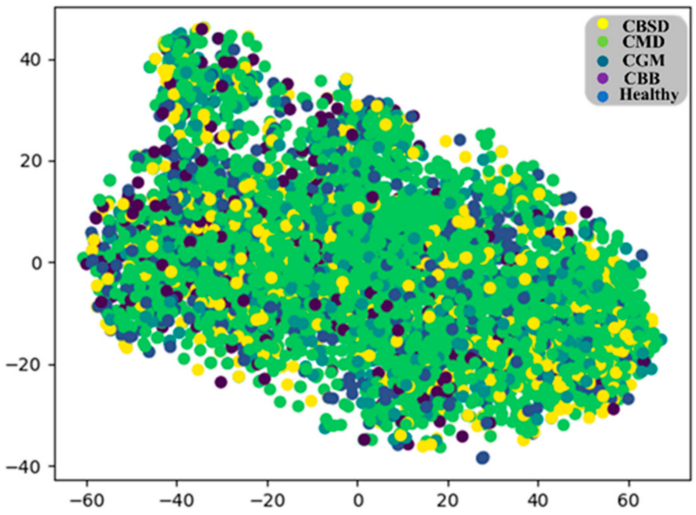


Figure 5. Dataset visualization using t-SNE.

4.2. Evaluation Metrics

This study used the following most common evaluation metrics, including accuracy, precision, recall, and *F*1-score, to evaluate the performance of the proposed model and the current state-of-the-art models on the cassava leaf disease dataset.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{8}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{9}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{10}$$

$$F1\text{-score} = \frac{2 * Precision * Recall}{Precision + Recall} \tag{11}$$

where *TP* denotes true positive, *FP* denotes false positive, *FN* denotes false negative, and *TN* denotes true negative. Precision is a measurement of the true positive value of accurate prediction in relation to the total number of positive prediction observations. Recall is a measure of the number of positive predictions for all positive predictions. The *F*1-score is a metric that balances precision and recall, and these metrics can be specifically classified as macro average and weighted average estimates. In an unbalanced dataset, the weighted average is considered a good metric for precision, recall, and *F*1-score. Macro measures calculate the precision, recall, and *F*1 scores for each category and returns the mean without considering the proportion of each category in the cassava leaf disease dataset, in contrast to the weighted average, which considers the proportion of each category in the dataset.

4.3. Experimental Environment

This study conducted a set of experiments on the Cassava Leaf dataset, which was based on PyCharm 2020, using a server with an Intel Xeon Scalable Silver 4210 CPU @ 2.20 GHz × 8, GeForce GTX 2080TI 11 GB × 4 and 128 GB RAM, a 500 GB SSD (solid-state drive) hard disk, and a 64-bit Ubuntu 16.04 LTS (Xenial Xerus) OS server. Anaconda IDE (Integrated

Development Environment) and all necessary libraries, including Python 3.8, Matplotlib, OpenCV-Python, Scikit-learn, Numpy, and PyTorch-1.7.0, were installed on the server.

The improvement of the model's performance depends mainly on the tuning of the parameters. For optimization, this study used stochastic gradient descent with a hot restart (SGDR) to learn the set of weights and biases that minimize the loss function. To avoid excessive fluctuations in the model parameters, the initial learning rate for all layers is 0.0002, and the optimization strategy controls the variation of the learning rate. All experiments were based on training for 50 epochs, and the batch size was set to 32. In addition, random cropping, random flipping, random brightness contrast, random rotation, random panning, random scaling, random inversion, clipping, and normalization were used for data enhancement during training in this paper to mitigate the overfitting of the model.

4.4. Experimental Verification

On the cassava leaf dataset, this study used the current mainstream models VGG16, ResNet-50, InceptionV3, Xception, and DenseNet121 to construct five different classifiers for training and compared them with our network, T-RNet, with respect to different performance evaluation metrics, such as precision, recall, F1-score, accuracy, and model parameters. These five pretrained classifiers all use weights trained on ImageNet beforehand, which speeds up the convergence of the model and improves the recognition accuracy. In addition, each classifier was trained using the five-fold cross-validation method, and the final results were the average of five tests to ensure the reliability of the results. The performance results are shown in Table 3.

Table 3. Performance comparison of different classification methods on the cassava disease dataset.

	CBB			CBSD			CGM			CMD			Healthy			ACC	Par. *
Model	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**		
Xception	62.9	65.1	63.9	83.6	80.6	82.0	84.5	74.2	79.0	95.1	96.2	95.6	70.8	76.0	73.3	88.07	21
VGG16	66.2	64.7	65.4	80.8	81.9	81.4	82.2	79.4	80.8	95.1	96.5	95.8	75.1	71.5	73.3	88.50	134
Inception-v3	62.3	66.0	64.1	86.7	78.7	82.5	81.6	77.3	79.4	94.8	95.8	95.3	71.8	75.0	73.4	87.99	22
ResNet-50	70.8	60.5	65.3	86.9	80.3	83.5	85.8	74.0	79.5	93.5	97.6	95.4	75.0	77.7	76.3	88.93	25
DenseNet-121	65.1	70.1	67.5	**88.0**	75.3	81.1	81.0	80.7	80.8	94.5	96.7	95.6	74.6	72.8	73.7	88.50	27.2
T-RNet	68.3	69.2	68.8	87.4	81.1	84.1	86.1	80.7	83.3	95.6	**97.8**	**96.7**	79.9	79.6	79.7	90.63	**14**
T-RNet *	**72.5**	**74.8**	**73.6**	84.8	**85.6**	**85.2**	**88.6**	**82.8**	**85.6**	**96.3**	97.0	**96.7**	**80.0**	**80.5**	**80.2**	**91.12**	**14**

P: Precision (%); R:Recall (%); F:F1-score (%); ACC: Accuracy (%); T-RNet * indicates the combination of T-RNet and FAMP-Softmax loss; Par. * indicates the number of model parameters (in millions); bold: the highest values.

In Table 3, it can be seen that the accuracy, recall, and F1-scores of all models with respect to CBB categories are generally lower in the test set because the number of CBB categories is the lowest among all categories. In contrast, the number of CMD categories is more than ten times greater; thus, the models learned the features of CMD to a large extent in training, which is why the accuracy and recall F1-scores of all of the models were high with respect to CMD categories. Nevertheless, T-RNet still achieved the highest F1-score value among all models in CBB category classification. It also achieved relatively close accuracy and recall scores without large performance fluctuations similar to other models, which indicates that T-RNet can learn more robust features in categories with sparse sample sizes. Once the amount of data increases, T-RNet will likely achieve higher scores as well. In addition, T-RNet far outperformed other models in terms of accuracy, recall and F1-scores for the health category. Because other models were more likely to classify health categories as disease categories, these models were somewhat confused by local information similar to disease features and did not learn more robust category features.

In contrast, the T-RNet model can better suppress this interference. The model does not overfit disease-like local features because this study introduces the transformer structure into the GE module to help to model the long-range information and combine it with the CNNs to extract local features to obtain a more powerful feature representation compared with other studies. The transformer-based architecture helps convolution increase the global

perspective of the model, while extracting disease features, forcing the model to learn more information and integrated feature representations. In addition, in Figure 6, it can be found that T-RNet achieved an overall average classification accuracy of 90.63%, which was much higher than other models, indicating that T-RNet effectively enhanced the feature representation and improved the classification accuracy. Notably, this study innovatively proposed a novel loss function called FAMP-Softmax, when T-RNet was trained with a FAMP-Softmax loss, the F1-scores of all five categories improved. In particular, the score of the CBB category, which had the smallest sample size, improved by 4.8% because the loss function introduced category number weights, which would suppress the influence of the CMD category on the feature learning of other categories to some extent. Meanwhile, the marginal-based idea requires the model to map different category features into a feature space with more explicit category feature intervals, and the model generates more substantial losses due to classification errors, which fully illustrates that the FAMP-Softmax loss can concentrate on learning fewer sample categories and resolve the sample imbalance problem. In addition, the F1-score achieves the best performance on other categories, indicating that FAMP-Softmax loss can help the model learn feature spaces with tighter classification boundaries. Additionally, the model parameters of this study are the lowest in the test set because the DCE module introduces the channel attention mechanism to obtain better channel feature information and uses deep separable convolution to further reduce the model parameters, which is beneficial for edge deployment.

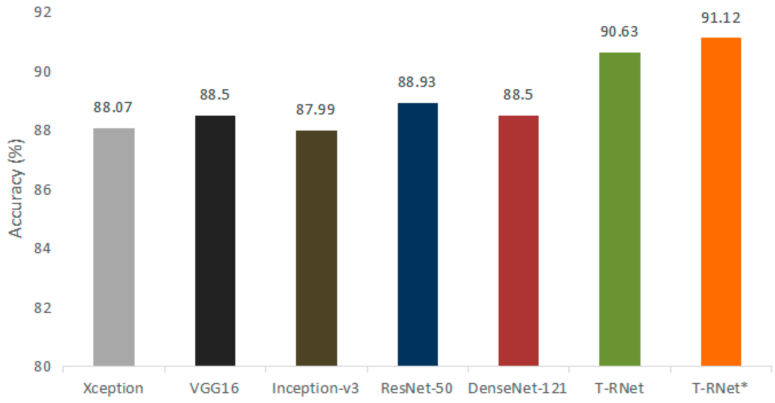
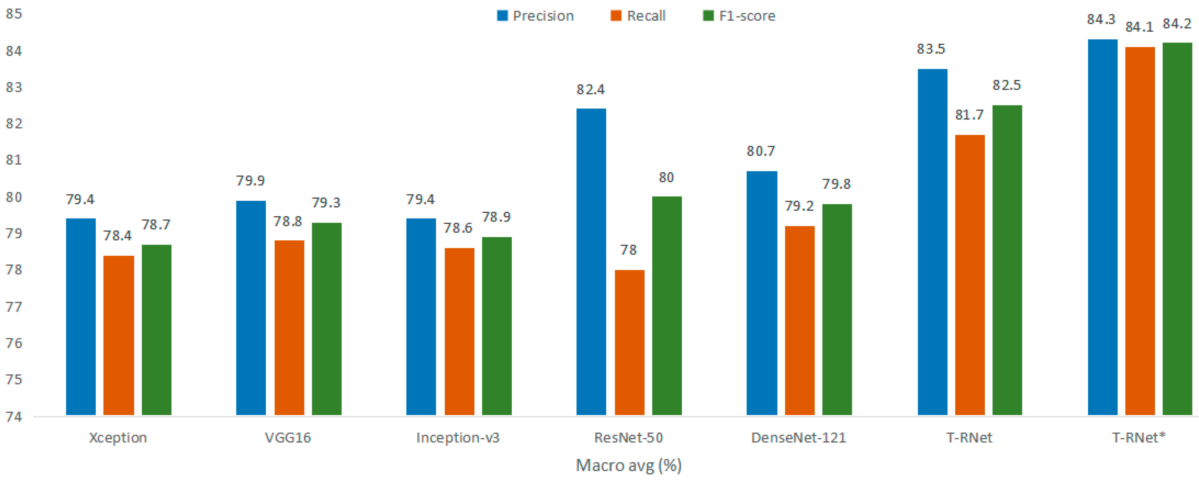


Figure 6. Accuracy comparison of different classification methods on the cassava leaf disease dataset. (T-RNet * indicates the combination of T-RNet and FAMP-Softmax loss).

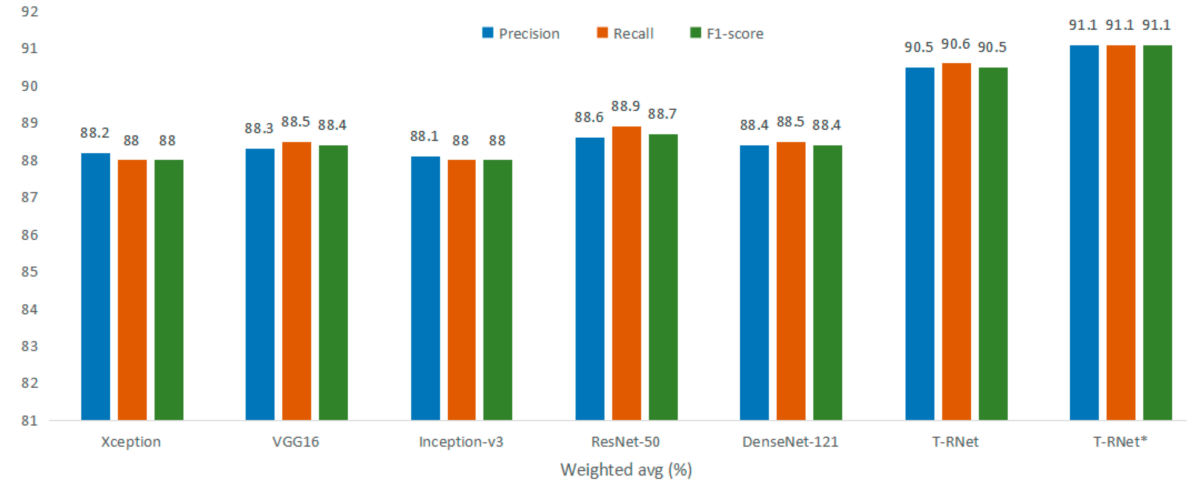
Because of the sample imbalance problem in the cassava leaf dataset, this study further estimated the results by macro averages and weighted averages, whose performance results for all models are shown in Table 4 and Figure 7. The existing mainstream models, such as Xception, VGG16, Inception-v3, ResNet-50, and DenseNet-121, all achieved relatively high-quality results. However, T-RNet achieved higher performances on both macro averages and weighted averages, which further validates the effectiveness of our proposed network T-RNet for the cassava disease classification task, and based on T-RNet and FAMP-Softmax loss, the proposed method in this paper achieved the best results in terms of both metrics and performance.

Table 4. Performances of different classification methods in terms of precision (%), recall (%), and F1-score (%). (Macro: Macro avg; Weighted: Weighted avg; T-RNet * indicates the combination of T-RNet and FAMP-Softmax loss; bold: the highest values).

Model	Type	Precision	Recall	F1-Score
Xception	Macro	79.4	78.4	78.7
	Weighted	88.2	88.0	88.0
VGG16	Macro	79.9	78.8	79.3
	Weighted	88.3	88.5	88.4
Inception-v3	Macro	79.4	78.6	78.9
	Weighted	88.1	88.0	88.0
ResNet-50	Macro	82.4	78.0	80.0
	Weighted	88.6	88.9	88.7
DenseNet-121	Macro	80.7	79.2	79.8
	Weighted	88.4	88.5	88.4
T-RNet	Macro	83.5	81.7	82.5
	Weighted	90.5	90.6	90.5
T-RNet *	Macro	**84.3**	**84.1**	**84.2**
	Weighted	**91.1**	**91.1**	**91.1**



(**a**)



(**b**)

Figure 7. (**a**) Performances of different classification methods in terms of precision (%), recall(%), and F1-score (%) by Macro averages. (**b**) Performances of different classification methods in terms of precision (%), recall (%), and F1-score (%) by Weighted averages.(T-RNet * indicates the combination of T-RNet and FAMP-Softmax loss).

To further validate the effectiveness of the proposed FAMP-Softmax loss, multiple loss functions were validated on ResNet-50 and T-RNet. As shown in Table 5b and Figure 8b, the T-RNet model trained on FAMP-Softmax loss achieved 91.12% accuracy and a 91.12% F1 score, which exceeded the values of 2.19% and 2.39% of ResNet-50 trained on cross-entropy loss. The FAMP-Softmax loss achieved the best performance compared to models trained with other loss functions, such as cross-entropy and focus loss. The experimental results showed that the loss function of this study can help the proposed model achieve better results on cassava leaf disease data with unbalanced samples. Second, this study also conducted experiments on ResNet-50 to verify the compatibility and generality of the loss function of this study, and the experimental results are shown in Table 5a and Figure 8a. From Figure 8a, it can be seen that the accuracy of ResNet-50 trained on FAMP-Softmax was 89.81%, and the F1 score was 89.76%. Compared with other loss functions, the improvement of the proposed loss function for ResNet-50 model performance is also obvious, indicating that the proposed loss function has model compatibility and generality for cassava leaf disease classification with unbalanced samples, which is also informative for other plant disease classification tasks.

Table 5. Performances of different loss functions on ResNet and T-RNet.

(a) The improvement of accuracy, F1-score (Macro avg), and F1-score (Weighted avg) of ResNet.

Method	Accuracy (%)	▲ (%)	F (Macro avg) (%)	▲ (%)	F (Weighted avg) (%)	▲ (%)
ResNet + Cross	88.93	-	80.01	-	88.73	-
ResNet + Focal	89.30	0.37 ↑	80.83	0.82 ↑	89.34	0.61 ↑
ResNet + FAMP	**89.81**	0.88 ↑	**81.23**	1.22 ↑	**89.76**	1.03 ↑

(b) The improvement of accuracy, F1-score (Macro avg), and F1-score (Weighted avg) of T-RNet.

Method	Accuracy (%)	▲ (%)	F (Macro avg) (%)	▲ (%)	F (Weighted avg) (%)	▲ (%)
T-RNet + Cross	90.63	-	83.18	-	90.59	-
T-RNet + Focal	90.79	0.16 ↑	83.17	0.01 ↓	90.79	0.20 ↑
T-RNet + FAMP	**91.12**	0.49 ↑	**84.26**	1.08 ↑	**91.12**	0.33 ↑

Cross: Cross entropy loss function; Focal: Focal loss function; FAMP: Our proposed loss function; ▲: the exceeded values; bold: the highest values.

Figure 8. (**a**) Performances of different loss functions on ResNet. (**b**) Performances of different loss functions on T-RNet.

4.5. Visualization Analysis

Deep learning has often been referred to as a black box technique for a long time because there is no approachable way to explore its specific internal mechanisms. To

further investigate the specificity of the proposed approach focusing on image regions, this study explains the results of this paper by extracting a gradient-weighted class activation map (Grad_CAM) of the feature maps generated by the visual attention mechanism, since the class activation map is the dominant technology that can currently be used to observe the internal learning of the model [39]. This is displayed in the form of weights, the weights represent the importance of the information in different regions. The higher the weight is, the redder the color of the attention map position it obtains, indicating that the image information in this part of the region wields a more important influence on the class discrimination of the model. The smaller the weight is, the bluer the color of its corresponding attention map location, indicating that the pixel information in these regions has little influence on the model results. This study extracts the attention maps of Grad_CAM produced by various models, such as VGG16, Xception, Inception-v3, ResNet-50, DenseNet121, and T-RNet, for different cassava disease images, and the results are shown in Figure 9. It can be seen in the figure that the model of this study achieves a stable performance on all types of disease images, focusing on the target region with little interference from background noise. However, other models such as Xception and Inception-V3 overfocus on the image background region. In fact, due to the largest number of CMDs, various types of environmental background noise are often distributed in the CMD category images, which considerably negatively impacts the training of the model. In Figure 9, it can be seen that the environmental background noise also makes a substantial contribution to the recognition of other models, which indicates that these models are disturbed by the background noise.

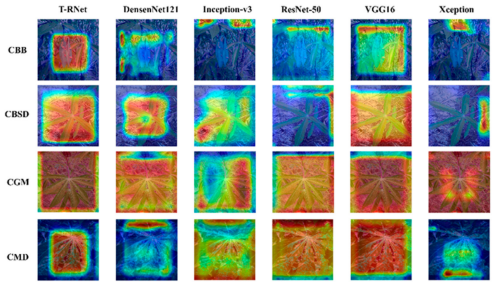


Figure 9. Grad_CAM attention maps for different models in different categories of samples.

Therefore, these models have difficulty learning features that are robust to all types of diseases, since convolution-based networks are lacking in the supervision of global information. Even in this case, the model of this study focuses well on the target region and suppresses background noise interference, which indicates that the model of this study can learn stronger category robust features. In addition, it was found that the distribution of the cassava disease dataset is a highly nonlinear and differentiable case. This study analyzes the spatial distribution of the features learned by the model using the t-SNE technique; t-SNE visualizes the two-dimensional spatial representation of the features generated by the model for the dataset [40]. The visualization results are shown in Figure 10, where different colors indicate different categories of samples. As seen from the figure, there is clear spacing between the feature representations of different categories. In contrast, the distribution of samples in each category is again more concentrated, indicating that the proposed method effectively extracts intercategory discriminative features.

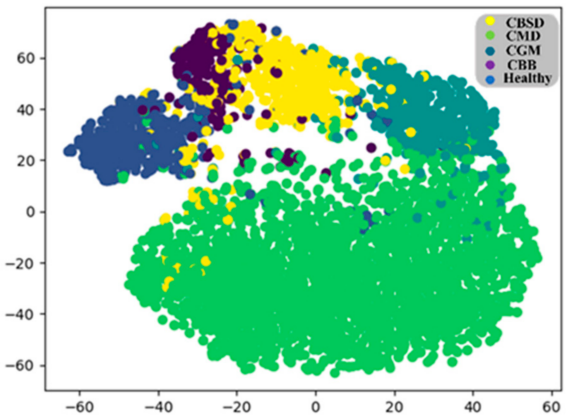


Figure 10. Visualization of the penultimate layer features of the T-RNet model using t-SNE.

5. Conclusions

A method of cassava leaf disease classification based on the T-RNet network model and FAMP-Softmax loss function is proposed in this paper. This study innovatively introduce a transformer structure into the cassava leaf disease classification task for the first time, and a novel GE module is established to fuse CNN and transformer to better learn feature representation. Meanwhile, with the help of deep separable convolution and SENet blocks, the model can obtain more valuable and distinguishable features with a few parameters. This is the first study to validate the classification performance of various models in the cassava leaf disease dataset by experiments, such as VGG16, Inception-v3, Xception, ResNet-50, and DenseNet121, which will also benefit the work of other researchers. Aiming at the problem of class sample imbalance in cassava leaf disease dataset, in order to better learn discriminative class features, this study innovatively proposes a new loss function, FAMP-Softmax, which can not only alleviate the influence of unbalanced sample data but also optimize the interclass and intraclass differences. Extensive experimental results show that the proposed method offers a good extracting ability for global information and can suppress the influence of background noise to some extent, and the model learns more robust class discriminative features. In the evolving field of agricultural disease classification, whether for agricultural deployment or scientific research, T-RNet, with excellent performance and few parameters, will provide a useful reference. Finally, the method presented in this paper is also a good reference for other classification tasks in the field of computer vision.

Author Contributions: Conceptualization, B.H. and Y.Z.; methodology, Y.Z. and B.H.; software, B.H. and Y.Z.; validation, Y.Z. and B.H.; data collection, B.H.; writing—original draft preparation, Y.Z., B.H. and C.T.; writing—review and editing, Y.Z. and C.T.; visualization, B.H.; supervision, C.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Moses, E.; Asafu-Agyei, J.; Adubofour, K.; Adusei, A. Guide to Identification and Control of Cassava Diseases. 2008. Available online: https://www.isppweb.org/foodsecurity_casava_diseases.asp (accessed on 18 July 2022).
2. Chikoti, P.C.; Mulenga, R.M.; Tembo, M.; Sseruwagi, P. Cassava mosaic disease: A review of a threat to cassava production in Zambia. *J. Plant Pathol.* **2019**, *101*, 467–477. [CrossRef] [PubMed]

3. Ufuan Achidi, A.; Ajayi, O.A.; Bokanga, M.; Maziya-Dixon, B. The use of cassava leaves as food in Africa. *Ecol. Food Nutr.* **2005**, *44*, 423–435. [CrossRef]
4. Oyewola, D.O.; Dada, E.G.; Misra, S.; Damaševičius, R. Detecting cassava mosaic disease using a deep residual convolutional neural network with distinct block processing. *PeerJ Comput. Sci.* **2021**, *7*, e352. [CrossRef] [PubMed]
5. Camargo, A.; Smith, J.S. An image-processing based algorithm to automatically identify plant disease visual symptoms. *Biosyst. Eng.* **2009**, *102*, 9–21. [CrossRef]
6. Meunkaewjinda, A.; Kumsawat, P.; Attakitmongcol, K.; Srikaew, A. Grape leaf disease detection from color imagery using hybrid intelligent system. In Proceedings of the 2008 5th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, Piscataway, NJ, USA, 16–18 October 2008; pp. 513–516.
7. Bracino, A.A.; Concepcion, R.S.; Bedruz, R.A.; Dadios, E.P.; Vicerra, R. Development of a Hybrid Machine Learning Model for Apple (*Malus domestica*) Health Detection and Disease Classification. In Proceedings of the 2020 IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), Manila, Philippines, 3–7 December 2020; pp. 1–6.
8. Pravin Kumar, S.K.; Sumithra, M.G.; Saranya, N. Particle Swarm Optimization(PSO) with fuzzy c means (PSO-FCM)–based segmentation and machine learning classifier for leaf diseases prediction. *Concurr. Comput. Pract. Exp.* **2021**, *33*, e5312.
9. Acar, E.; Ertugrul, O.F.; Aldemir, E. Automatic identification of cassava leaf diseases utilizing morphological hidden patterns and multi-feature textures with a distributed structure-based classification approach. *J. Plant Dis. Prot.* **2022**, *129*, 605–621. [CrossRef]
10. Abayomi-Alli, O.O.; Damaševičius, R.; Misra, S.; Maskeliūnas, R. Cassava disease recognition from low-quality images using enhanced data augmentation model and deep learning. *Expert Syst.* **2021**, *38*, e12746. [CrossRef]
11. Ramcharan, A.; Baranowski, K.; McCloskey, P.; Ahmed, B.; Legg, J.; Hughes, D.P. Deep learning for image-based cassava disease detection. *Front. Plant Sci.* **2017**, *8*, 1852. [CrossRef] [PubMed]
12. Sambasivam, G.; Opiyo, G.D. A predictive machine learning application in agriculture: Cassava disease detection and classification with imbalanced dataset using convolutional neural networks. *Egypt. Inform. J.* **2021**, *22*, 27–34. [CrossRef]
13. Sangbamrung, I.; Praneetpholkrang, P.; Kanjanawattana, S. A novel automatic method for cassava disease classification using deep learning. *J. Adv. Inf. Technol.* **2020**, *11*, 241–248. [CrossRef]
14. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826.
15. Ferentinos, K.P. Deep learning models for plant disease detection and diagnosis. *Comput. Electron. Agric.* **2018**, *145*, 311–318. [CrossRef]
16. Ayu, H.R.; Surtono, A.; Apriyanto, D.K. Deep learning for detection cassava leaf disease. *J. Phys. Conf. Ser.* **2021**, *1751*, 012072. [CrossRef]
17. Bi, C.; Wang, J.; Duan, Y.; Fu, B.; Kang, J.R.; Shi, Y. MobileNet Based Apple Leaf Diseases Identification. *Mobile. Netw. Appl.* **2022**, *27*, 172–180. [CrossRef]
18. Zhong, Y.; Zhao, M. Research on deep learning in apple leaf disease recognition. *Comput. Electron. Agric.* **2020**, *168*, 105146. [CrossRef]
19. Bansal, P.; Kumar, R.; Kumar, S. Disease Detection in Apple Leaves Using Deep Convolutional Neural Network. *Agriculture* **2021**, *11*, 617. [CrossRef]
20. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the 2017 IEEE International conference on Computer Vision, Honolulu, HI, USA, 22–29 October 2017; pp. 2980–2988.
21. Deng, J.; Guo, J.; Xue, N.; Zafeiriou, S. Arcface: Additive angular margin loss for deep face recognition. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4690–4699.
22. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
23. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-net: Efficient channel attention for deep convolutional neural networks. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11534–11542.
24. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional block attention module. In Proceedings of the 15th European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 3–19.
25. Greff, K.; Srivastava, R.K.; Koutník, J.; Steunebrink, B.R.; Schmidhuber, J. LSTM: A search space odyssey. *IEEE Trans. Neural Netw. Learn. Syst.* **2017**, *28*, 2222–2232. [CrossRef] [PubMed]
26. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language models are unsupervised multitask learners. *OpenAI Blog.* **2019**, *1*, 9.
27. Srinivas, A.; Lin, T.Y.; Parmar, N.; Shlens, J.; Abbeel, P.; Vaswani, A. Bottleneck transformers for visual recognition. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 16519–16529.
28. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In Proceedings of the 16th European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 213–229.

29. Sun, C.; Myers, A.; Vondrick, C.; Murphy, K.; Schmid, C. VideoBERT: A joint model for video and language representation learning. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 7464–7473.
30. Lu, J.; Batra, D.; Parikh, D.; Lee, S. ViLBERT: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In Proceedings of the 33rd International Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; pp. 13–23.
31. Liu, W.; Wen, Y.; Yu, Z.; Yang, M. Large-margin softmax loss for convolutional neural networks. In Proceedings of the 33rd International Conference on Machine Learning, New York, NY, USA, 19–24 June 2016; pp. 507–516.
32. Wang, H.; Wang, Y.; Zhou, Z.; Ji, X.; Gong, D.; Zhou, J. CosFace: Large margin cosine loss for deep face recognition. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5265–5274.
33. Wang, F.; Cheng, J.; Liu, W.; Liu, H. Additive margin softmax for face verification. *IEEE Signal Process. Lett.* **2018**, *25*, 926–930. [CrossRef]
34. Hermans, A.; Beyer, L.; Leibe, B. In Defense of the Triplet Loss for Person Reidentification. *arXiv* **2017**, arXiv:1703.07737.
35. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
36. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 22–29 October 2017; pp. 2261–2269.
37. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 22–29 October 2017; pp. 1800–1807.
38. Chen, H.; Wang, Y.; Guo, T.; Xu, C.; Deng, Y.; Liu, Z. Pretrained image processing transformer. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 12299–12310.
39. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In Proceedings of the 2017 IEEE International Conference on Computer Vision, Honolulu, HI, USA, 22–29 October 2017; pp. 618–626.
40. Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.