Strawberry Disease Detection through an Advanced Squeeze-and-Excitation Deep Learning Model

Jiayi Wu, Vahid Abolghasemi, *Senior Member, IEEE*, Mohammad Hossein Anisi, *Senior Member, IEEE*, Usman Dar, Andrey Ivanov, and Chris Newenham

***Abstract*—In this paper, an innovative deep learning-driven framework, adapted for the identification of diseases in strawberry plants, is proposed. Our approach encompasses a comprehensive embedded electronic system, incorporating sensor data acquisition and image capturing from the plants. These images are seamlessly transmitted to the cloud through a dedicated gateway for subsequent analysis. The research introduces a novel model, ResNet9-SE, a modified ResNet architecture featuring two Squeeze-and-Excitation (SE) blocks strategically positioned within the network to enhance performance. The key advantage gained is achieving fewer parameters and occupying less memory while preserving a high diagnosis accuracy. The proposed model is evaluated using in-house collected data and a publicly available dataset. The experimental outcomes demonstrate the exceptional classification accuracy of the ResNet9-SE model (99.7%), coupled with significantly reduced computation costs, affirming its suitability for deployment in embedded systems.**

***Index Terms*—Plant disease detection, Deep learning, Internet of Things, Crop monitoring, Computer vision.**

I. Introduction

PLANT diseases have become a major concern in today's forestry development. The impact of different diseases is all-encompassing, affecting plants externally and internally, from the top to the bottom, spanning from flowers and fruits down to the root system. This not only affects the normal growth of plants but can also cause a reduction in the yield and quality of agricultural products and, in serious cases, food safety problems [1]. Therefore, rapid identification and diagnosis of plant diseases can reduce the economic losses caused by plant diseases to the agricultural industry in the shortest possible time. Plant disease identification is a technique for processing, analysing, and understanding plant image datasets to identify potential kinds of disease objects. Traditional techniques are mainly performed manually by growers, which is a labour-intensive and time-consuming task. In large-scale farming systems, it is difficult to make an accurate estimate of the infected areas and the severity [2]. Today, there are many different types of plant diseases in different stages of growth and several growing areas, which makes it difficult for laymen to accurately identify the types of disease in a short period and on a large scale. Besides, manual identification has the disadvantages of slow identification speed and low accuracy which poses a major challenge in containing the outbreak of diseases in agriculture.

In particular, the impact of disease on strawberry yield has become more serious due to its soft and delicate nature. Some major strawberry plant diseases include Powdery Mildew, Botrytis cinerea, Anthracnose, and Angular Leaf Spot. Diseases affecting strawberry crops have multifaceted impacts, ranging from diminished yields and compromised fruit quality to economic losses for farmers. The need for chemical interventions to manage diseases contributes to increased production costs and may disrupt sustainable agricultural practices. Rapid disease spread poses contamination risks, while climate change vulnerabilities further stress plant resilience. To this end, integrating smart data collection and transmission with artificial intelligence (AI) for early disease detection in plants can mitigate these challenges. This could offer the potential to swiftly identify and manage diseases, minimising the reliance on chemical interventions. This approach not only enhances sustainability by reducing environmental impact but also contributes to improved fruit quality.

With the continuous development of deep learning on the one hand, and increasing the computation power on another hand, many researchers have started to study plant disease identification based on deep learning with either sensors or image data (or both). Using computer vision technology to identify plant disease areas and species can effectively reduce time costs and improve the efficiency of agricultural production [3]. Furthermore, with the advances in the Internet of Things (IoT) technology, effective and continuous monitoring of various systems has become easier and more accessible. This has led to greater autonomy of systems in practice. The solutions that IoT offers are complemented by machine learning (ML) and computer vision-based techniques to improve classification and detection performance.

A. Our Contributions

There is limited research on the utilization of reliable IoT systems for agricultural data collection and processing. Furthermore, there is a lack of focused research on early onset detection for strawberry plants. In our previous work, we demonstrated the development of a prototype convolutional neural network capable of classifying three types of strawberry images: healthy, powdery mildew-affected, and leaf scorch-affected. We obtained an average accuracy of 95.48% [4].

Jiayi Wu is with the Xidian University, No. 2 South Taibai Road, Xi'an, Shaanxi, 710071, email: jiayiwu@stu.xidian.edu.cn

Vahid Abolghasemi and Mohammad Hossein Anisi are with the School of Computer Science and Electronic Engineering, University of Essex, UK, emails: {m.anisi, v.abolgahsemi}@essex.ac.uk

Usman Dar, Andrey Ivanov, and Chris Newenham, are with Wilkin & Sons, Tiptree, UK, emails: {usman,ai,cwm}@tiptree.com

This research was funded by a Knowledge Transfer Partnership (KTP) from Innovate UK (Partnership No: 12298), between Wilkin & Sons Ltd and the University of Essex.

Furthermore, we have established an advanced test-bed in Wilkin & Sons farm, dedicated to sensors and images' data collection from the strawberry farms [5].

In this paper, three major contributions are introduced. First, we demonstrate the details of a fully scalable and automated system (including sensors, cameras, connectivity, etc.) that can be utilised in agricultural applications. Our hardware design can be easily adapted and extended for data collecting data from an operational farm. Using this system, we can have valuable sensory and imagery data for future analysis. Second, we provide a comprehensive overview of the state-of-the-art in plant disease detection (with a focus on Strawberry plants) in both hardware design and machine learning approaches. Third, we propose a novel deep learning architecture benefiting from a unique feature, i.e., a Squeeze-and-Excitation (SE) block to reduce the number of model parameters and hence occupying less memory leading to better adaptation to hardware implementation. We also introduce a "variable learning rate" in our model which leads to higher accuracy compared to a fixed rate. Our results exhibit robust performance, showcasing the potential for automated, real-time disease identification.

The paper is organised as follows. The next section reviews the related works. In Section III, the proposed system in both the data collection and machine learning stages is described. Section IV is devoted to experimental results and data analysis. Finally, Section V, concludes the paper.

II. Related works

Existing studies have investigated the use of ML algorithms for the early detection of various plant diseases, and notable efforts have been directed towards addressing strawberry-specific issues. Machine learning models, particularly convolutional neural networks (CNNs) and support vector machines (SVMs) have demonstrated significant promise in accurately identifying and classifying different strawberry diseases.

Researchers in [6] presented an approach using deep learning techniques to detect powdery mildew disease on strawberry leaves. Due to dealing with a small dataset, they have applied data augmentation for around 1400 healthy and infected leaf images to prevent overfitting. Their study suggests that SqueezeNet would be the most suitable model when considering the memory requirements for hardware deployment. Additionally, Jiang et al. [7] explored the integration of spectral imaging and machine learning for early disease diagnosis in strawberry plants. By extracting intricate features from hyperspectral data, their model achieved high accuracy in distinguishing between healthy and infected plants, providing valuable insights into the spectral signatures associated with specific diseases. The drawback of this system is the need for expensive cameras for imaging. Irfan Abbas et al. [8] used a deep learning model to identify strawberry fungal leaf blight in strawberry fields in real-time. By training and testing four convolutional neural network (SqueezeNet, EfficientNet-B3, VGG-16 and AlexNet) models, they evaluated health and leaf scorch. Their study showed that the model EfficientNet-B3 achieved the highest classification accuracy, 80% and 86% for the initial and severe stages of the disease, respectively. The model proposed by Tariqul Islam M. et al. [9] was designed to diagnose the disease leaf of grapes and strawberries. They used a CNN model to train the dataset where an accuracy rate of 93.63% was obtained.

A self-constructed SPIKE dataset from images of relevant complex wheat fields was used in an object detection method based on identifying diseased plants (or parts affected by diseases or pathogens) proposed by Hasan, M. M. et al. [10]. The model used was an R-CNN architecture that generated four different models, four different datasets of training and test images based on four different datasets to capture plant diseases at different growth stages with an accuracy of 93.4%. Toda Y et al. [11] employed the YOLOV3 - DenseNet algorithm for direct object detection, focusing on disease object detection concerning growing apple leaves, with an accuracy of 95.75%. And using human intervention to validate the authenticity of the model and the training dataset, a CNN trained using publicly available plant disease image datasets, various neuron and layer visualisation methods were applied. Zhang, S. et al. [12] developed a method, called GPDCNN, for multiclass classification detection of cucumber images, i.e., using different stages of the plant for possible disease detection where the accuracy of 94.65% was achieved.

In another study, Hari et al. [13] used the PDDNN algorithm for the detection of various plant disease images, using TensorFlow as the framework, with an accuracy of 86%. As a comparison, Picon et al. [14] also published a paper using the RESNET-MC1 algorithm for the detection of various plant disease images using TensorFlow and Keras as the framework of choice, with an accuracy of 98%. Howlader et al. [15] employed the AlexNet algorithm to detect plant diseases on guava leaves with an accuracy of 98.74%. Nagasubramanian et al. [16] used the 3D-CNN algorithm to detect plant diseases in soybean using a binary classification method, i.e., only diseased or healthy, without distinguishing between specific growth regions and growth stages, with an accuracy of 95.73%. Arunangshu Pal [17] proposed an Agricultural Inspection (AgriDet) framework, which combines the traditional Inception-Visual Geometry Group network (INC-VGGN) and the Kohonen-based deep learning network to detect plant diseases and classify the severity of diseased plants where the performance of the statistical analysis is validated to demonstrate the effectiveness of the results in terms of accuracy, specificity, and sensitivity.

In the article by Amal Mathew et al. [18], the SVM classifier was replaced with a voting classifier to classify the data into multiple classes. The accuracy of voting and SVM classifiers are compared. The results show that the accuracy of the proposed method is improved by 10%. Punam Bedi et al. [19] proposed a hybrid system based on convolutional auto-encoder (CAE) and CNN that can achieve automatic detection of plant diseases. In the experiment, CAE is used to compress the parameters required for training, and the parameters required for the hybrid model are reduced. The proposed hybrid model used only 9914 training parameters. The experiment uses a public dataset called PlantVillage to obtain leaf images of peach plants with the training and testing accuracies reported at 99.35% and 98.38%, respectively. Abdalla et al. [20] used the VGG16 Encoder algorithm to detect binary segmentation of 400 oilseed images in two different environments with an accuracy of 96%.

Some studies also address image segmentation along (or before) disease detection. Lin et al. [21] used the U-Net segmentation algorithm to segment cucumber leaves with an accuracy of 96.08%. Wiesner-Hanks et al. [22] implemented a binary segmentation task to identify maize diseases using the ResNet - Crowdsourced algorithm for binary segmentation, which divides the image into homogeneous regions according to defined criteria and generates a binary image of the plant disease with the highest accuracy rate, i.e. 99.79%. In a recent study, Li et al. [23] addressed the detection of powdery mildew on strawberry leaves based on DAC-YOLOv4 model. They compared several models and also deployed their algorithm on the Jetson Xavier NX and Jetson Nano hardware to meet the real-time detection requirements. Their experimental results indicated that the DAC-YOLOv4 can provide an acceptable performance in strawberry leaf detection on the embedded platform. In [24], a mango fruit grading system using a computer vision approach was proposed. The proposed method classified the fruit images using an AlexNet–spatial pyramid pooling network (SPP-Net) with a segmentation algorithm based on a Mask region-based convolutional neural network (R-CNN).

Furthermore, the intersection of IoT devices and ML for continuous monitoring has been a focal point. However, there exist limited works reported on developments of plant disease diagnosis with on-board hardware. Kim et al. [25] proposed a sensor-based system that collected environmental data and employed machine learning algorithms to predict disease outbreaks in strawberry crops. This proactive approach offers the potential for timely interventions and preventive measures. Pankaj et al. [26] proposed an IoT hardware sensor-based Cotton Disease prediction using a convolutional neural network (CNN). Their IoT gadget has different sensors such as temperature, humidity, and PH to collect the data to be used for classification. In another work by Mora et al., [27], a plant disease detection using the Raspberry Pi 4 was implemented. Not many results are reported in this work, however, an accuracy of around 90% was obtained for the prediction of plant disease on a private dataset. A diagnostic system implemented on Raspberry Pi was proposed for scab and leaf disease detection. The authors used a CNN model and four classes of Scab, Black Rot, Cedar Rust, and Healthy were detected. A smart crop growth monitoring using edge artificial intelligence (AI) was developed in [28] where a complex system was implemented to monitor health plants and classify the pest and disease severity. They used cryptographic hardware modules, including RTEA32, RTEA64, XTEA32 and XTEA64, and used the binarized neural network and achieved 76.57% accuracy for disease detection on dragon fruits.

While strides have been made, challenges persist, including the need for diverse datasets representative of geographical and environmental variations. Future research directions may involve the refinement of existing models, the exploration of transfer learning techniques, and the incorporation of explainable AI for transparent decision-making in the agricultural domain. The convergence of machine learning and AI technologies holds immense promise for revolutionizing strawberry disease detection. The amalgamation of sophisticated algorithms, sensor technologies, and data analytics paves the way for sustainable, technology-driven agriculture, ensuring the resilience of strawberry crops against the threat of diseases and contributing to global food security initiatives.

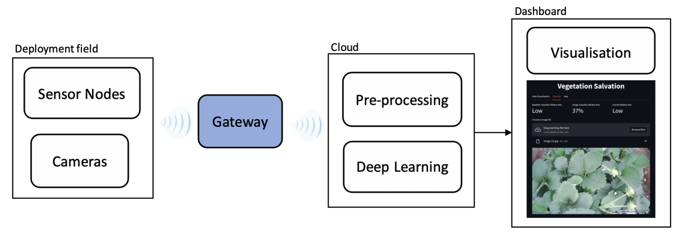


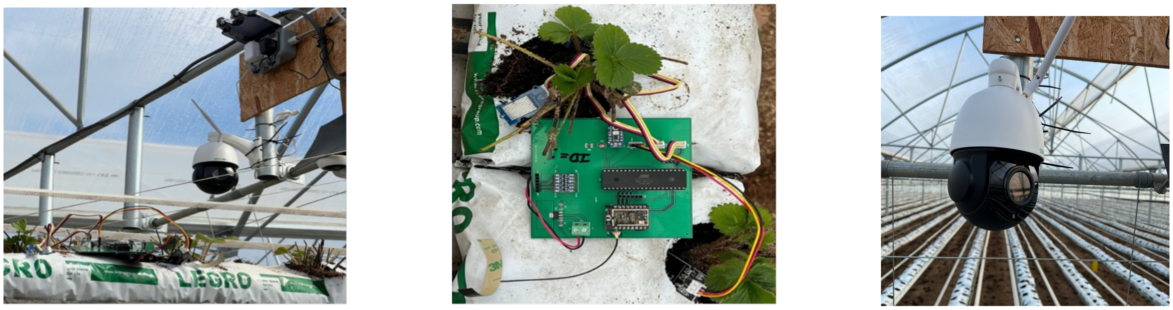
Fig. 1. Data collection, transmission, and analysis layout.

III. Proposed System

In this section, we showcase the details and results of our plant disease detection system. As shown in Figure 1, we develop a network of sensors and cameras that are wirelessly connected to a base station, continuously monitor the conditions of plants, and seamlessly transmit the images and sensors' data. In the following, first, the hardware specifications and design for data/image capturing and communication are described. Then, the results of applying novel deep learning models on both the collected dataset and existing datasets are provided.

A. Data collection

The imaging system is composed of an SVC3 camera that can capture images at 2560×1920 resolution. The camera features $20\times$ optical zoom as well as 255 degrees pan and 120 degrees tilt that enables the capture of high-quality close-up images of the plant matter over a large area. A Raspberry Pi-based camera controller which is deployed on the same Wi-Fi Network as the cameras, requests images from each camera at fixed intervals during the day before uploading them via a Wi-Fi access point. In contrast to the imaging system, the sensor network has been custom-designed to meet the needs of this application. The ATMega644p microcontroller, shown in Figure 2 (a), is responsible for interfacing with 7 sensor modules; temperature, pressure, humidity, ambient light, UV light, soil moisture, and leaf wetness. The microcontroller samples the sensors roughly once every 30 minutes and uses a Semtech SX1262 LoRa transceiver to transmit the data to a Dragino LG01-N LoRa gateway which pushes this data to a privately hosted server that is responsible for parsing the data and storing it in a database hosted on Amazon Web Services. The system is designed to be flexible to the farm conditions through the modular architecture allowing different components (hardware and software) to be easily added, removed, or replaced without disrupting the overall functionality. This facilitates scalability and the addition of



(a) (b) (c)

Fig. 2. The proposed IoT-based plant disease detection system implemented at Wilkin & Sons in Tiptree. (a) View of the entire sensing and imaging system, (b) the processing unit hardware, and (c) the camera installed with a view of the farm.

new sensors or devices to adapt to the specific needs of different farms. Our power-efficient hardware and software components can maximise battery life and minimise energy consumption in remote farm environments where access to power sources may be limited.

B. Data Analysis

1) ResNet9: ResNet [29], with its ability to train very deep networks effectively, has found applications in various computer vision tasks, including plant disease diagnosis. In this study, we enhance the performance ResNet9 by introducing the Squeeze-and-Excitation (SE) blocks. The key innovation in ResNet is the utilisation of residual blocks, which allow the model to skip connections across layers. Traditional deep networks often suffer from the vanishing gradient problem, making it difficult to train very deep networks. Residual blocks alleviate this issue by introducing shortcut connections or skip connections, which enable the gradient to flow directly through the network. The basic ResNet layers include Convolutional Layers (for initial feature extraction), Residual Blocks (the core building blocks of ResNet, consisting of stacked convolutional layers with skip connections), Pooling Layers (to reduce spatial dimensions), and Fully Connected Layers (often used for classification tasks).

Many variants of ResNet model have been proposed in the literature. Among these, ResNet9 has a relatively shallow version of the ResNet architecture compared to deeper variants like ResNet50 or ResNet101. In scenarios with limited data or computational resources (such as our case), a shallower network may be more practical while still benefiting from the advantages of residual connections. When choosing a neural network architecture for plant disease diagnosis, factors such as the size of the dataset, computational resources, and the specific requirements of the application should be considered. ResNet9 offers a good balance of performance and efficiency, making it a viable option for image classification tasks, including plant disease diagnosis.

As mentioned above, we used the 9-layer structure of the ResNet as a baseline. In this model, each layer feeds into the next layer and directly into the layers about 2-3 hops away. Conventional pre-processing such as image re-sizing was applied to input images where required. The network in this study uses a combination of two convolutional layers and two residual blocks, regularising each layer first, then using ReLU as the activation function and Max Pooling to reduce the size of the data and increase the speed of computation. Furthermore, the Adam Optimiser and the Cross-Entropy loss function were employed in this model. In the last layer, the data is flattened, and linear regression is used to classify the different types of plant diseases.

2) ResNet9-SE: To further enhance the discriminative power of our model, we introduce Squeeze-and-Excitation blocks into the ResNet9 architecture. SE blocks have proven effective in capturing channel-wise dependencies, allowing the network to focus on essential features while suppressing less informative ones. The SE block consists of two key operations: *squeeze* and *excitation*. In the squeeze operation, global average pooling is applied to compress spatial information into channel-wise descriptors. The dimensions of the original feature map are $H \times W \times C$, where H is the height, W is the width, and C is the number of channels of the input data matrix. What Squeeze does is compress $H \times W \times C$ into $1 \times 1 \times C$, which is equivalent to compressing H and W into one dimension. In practice, it is generally implemented using global average pooling. After H and W are compressed into one dimension, this dimensional parameter obtains the global view of the previous $H \times W$, and the sensing area is wider.

The excitation operation involves learning channel-wise attention weights that are then applied to the input features. After obtaining the $1 \times 1 \times C$ representation of Squeeze, add a fully connected layer (FC) to predict the importance of each channel, obtain the importance of different channels, and then apply (stimulate) it to the previous feature. On the corresponding channel of the map, perform subsequent operations. This adaptive recalibration mechanism enables our model to better capture intricate patterns and crucial information relevant to plant disease detection. The SE block contains a global average pooling layer, two fully connected layers, and a Sigmoid activation function. The calculation process is as follows:

1) $F = FC_1(Z)$
2) $F = ReLU(F)$
3) $F' = FC_2(F)$
4) $SE(X) = Sigmoid(F')$

where X is the input of the SE block, Z denotes the output of

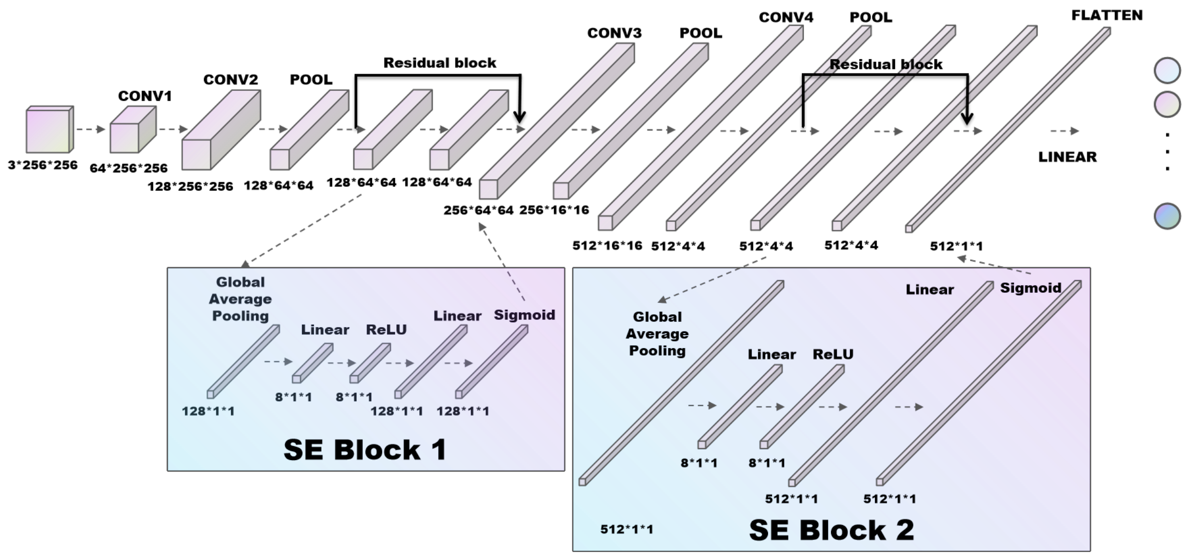


Fig. 3. Structure of the proposed model based on ResNet with SE Blocks.

the first pooling layer, F is the output of the fully connected layer (which is updated by ReLU function), and F' is the output of the second fully connected layer which is inputted into Sigmoid function to yield the output result of the whole SE block. The mathematical equation for SE is as follows:

$$SE = \sigma(W_2\gamma(W_1 z + b_1) + b_2) \tag{1}$$

where σ is the Sigmoid, γ denotes the ReLU function, and $\{W_1 \in R^{\frac{C}{1}xC}, b_1\}$ and $\{W_2 \in R^{\frac{C}{1}xC}, b_2\}$ are the weights connecting two fully connected layers. Inside the SE, the computational cost and capacity are controlled by a reduction ratio r. The input X and the scalar SE are multiplied channel-by-channel to obtain the final output:

$$X_{SE} = X \otimes SE \tag{2}$$

where $\otimes$ denotes the channel-wise multiplication. The SE structure will retain useful features and reduce unnecessary features to boost the network's performance.

Figure 3 shows the proposed structure of the ResNet9-SE model. To appropriately integrate the SE block into the ResNet structure, we embed two SE blocks at the end of each residual block. For each SE block, global average pooling is first used to implement the squeeze operation, and then the fully connected layer, linear rectification function, and activation function are used to implement the excitation operation. The output of the SE block is combined with the output of the residual block through element-wise multiplication, enabling dynamic adjustment of features.

IV. Experimental Results

A. Experimental setup and performance metrics

To provide a comprehensive evaluation and the generalisability power of the proposed system, we use a widely used public dataset, i.e., "PlantVillage" as well as data collected by ourselves at Wilkin & Sons in Tiptree [21] (see Figure 1). The original PlantVillage dataset consists of about 87,000 RGB images of healthy and diseased crop leaves which are categorized into 38 different classes of 14 unique plants, including strawberry. In this study, we used the strawberry images from PlantVillage dataset, which consists of over 800 healthy and unhealthy (leaf scorch) images. The other private dataset, called the "strawberry dataset", contains healthy strawberries and 2 types of diseases of strawberry, including Strawberry Leaf scorch and Strawberry Mildew. Each type contains about 2000 pictures of strawberries. Some sample images of healthy and diseased plants are shown in Figure 4.

(a) (b) (c)

Fig. 4. Sample images of strawberry plant for (a) healthy and (b) leaf scorch, and (c) Powdery Mildew.

For proper model evaluation, a 10-fold cross-validation was considered on both datasets for the evaluation of different algorithms. The ratio for train and test subsets was chosen to be 80% and 20%, respectively. Also, to evaluate the performance of the proposed algorithm for disease detection three well-known performance metrics were used. The accuracy, defined as the ratio of correctly identified samples to the total samples, serves as a comprehensive metric for assessing the overall performance of the algorithm. To quantify the accuracy of an algorithm in detecting corn diseases, the following equation was employed:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

where TP is the True Positive (i.e., samples that have the disease and were correctly identified as such), and TN is the True Negative (i.e., samples that do not have the disease and were correctly identified as such). Also, FP is the False Positive (i.e., samples that do not have the disease but were wrongly identified as having it), and FN is the False Negative (i.e., samples that have the disease but were wrongly identified as not having it).

We also used another metric, called *Recall*, to measure the capability of the proposed models to correctly images of strawberry plants that are infected with a specific disease:

$$Recall = \frac{TP}{TP + TN} \tag{4}$$

The final performance metric is *F1-Score*, which represents the harmonic mean of precision and sensitivity score which is a commonly used performance evaluation metric in machine learning approaches, particularly in binary classification tasks:

$$F1Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{5}$$

In (5), *Precision* is a metric that measures how often the model correctly predicts the positive class and is defined via:

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

The machine used for conducting the experiments was equipped with the NVIDIA RTX 3070 GPU which has 5888 CUDA cores and 8 GB of VRAM. The GPU was interfaced with an Intel i5-8400 @ 2.8 GHz CPU with access to 16 GB of RAM.

B. Pre-processing

Before training the model, the images were transformed and augmented to prepare the data for the input layer of the networks and to reduce biases. The first step was to perform a normalization operation on the images. This was done by calculating the mean and standard deviation of the entire dataset, individually across each of the RGB channels. The resulting values were then used to normalize the images to account for the variations in the way the images were captured. The second step was to resize the images to one standard size that the input to the classifier accepts. This is done to account for the various image sizes in the dataset. The consequence of this approach is that every image that is passed through the classifier during evaluation and deployment will also have to be resized. The dimensions chosen were 224×224 pixels as this is a standard used in many widely used architectures, allowing for a convenient way to swap architectures if required. Additionally, this allowed for better direct comparisons between architectures as the dimensionality of the input was kept consistent throughout the comparisons.

The data was augmented by introducing random vertical and horizontal and vertical flips as well as random 30-degree rotations. This allowed for biases relating to image orientation to be reduced significantly. This step was crucial to improving system generalizability as most images of the plants were taken with the stem appearing on the lower half of the image. In practice, this is not always the case, so it was important to ensure that the networks did not make the wrong association between the orientation of the plants and the disease they intended to classify.

TABLE I
FINETUNING VS FINAL LAYER TRAINING STATISTICS

Model	Run-Time with Fine-Tune (s)	Run-Time without Fine-Tune (s)	Accuracy (%)
Resnet50	11842	13021	92.85
Resnet100	11835	16415	75.86
GoogleNet	11135	12055	86.50
AlexNet	10698	10853	73.58

C. Classification results

Before demonstrating the results obtained using the proposed ResNet9-SE model, we demonstrate and compare the performance fo state-of-the-art algorithms using transfer learning (i.e. pre-trained models on the ImageNet database). The aim is to show why we have selected the ResNet model as our baseline and to modify it to receive the best performance. While not all the images in the ImageNet dataset were relevant to the aims and objectives of this study, further training the models on the strawberry dataset yielded two important benefits. First, the model's generalisability was far greater as it was able to leverage the knowledge gained from the large ImageNet dataset to reduce biases. Second, the losses were lower, and the models converged much faster since the weights and parameters were already well-optimized for image classification problems.

Before the pre-trained models could be used within the context of this study their architectures had to be adjusted. This meant that for all the models, the fully connected output layer had to be redefined and initialized without the pre-trained weights. This was performed to merely update the weights in the newly redefined final layer while keeping all other weights the same. This would allow for faster, training of the models since the backpropagation would be limited to only the final layer, resulting in fewer calculations needing to be done. The other approach, fine-tuning, would require all the weights to be adjusted at every step during the training which would increase the training time significantly due to the number of calculations required. To determine the feasibility of four models (i.e. Resnet50, Resnet100, GoogleNet, and AlexNet), each architecture was trained with a learning rate of 0.01 and batch size 32 for 100 epochs twice, with and without fine-tuning approach. The times and accuracies of each of the models were observed after the hundred epochs, the results of which are summarized in Table I. As seen from this table, fine-tuning has a positive effect on the running time and most models have achieved acceptable accuracy with ResNet50 being the best.

The optimiser used was Stochastic Gradient Descent (SGD) with momentum. Using SGD with momentum allowed models trained with lower learning rates to carry their "momentum" as they descended into minima. This allowed the models to overcome any shallow local minima that they descended into

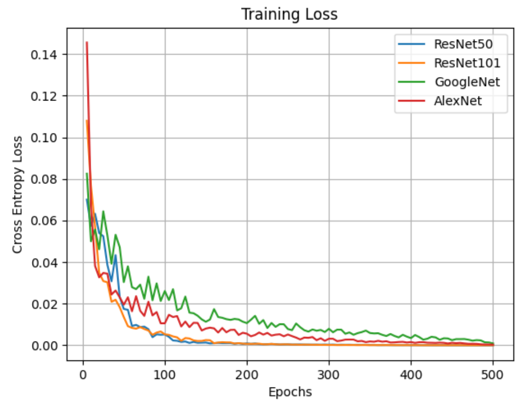
TABLE II
THE PERFORMANCE OF (PRE-TRAINED) DEEP MODELS AFTER FINE-TUNING THE PARAMETERS

Model	Learning Rate	Training loss	Batch size	Accuracy(%)
Resnet50	0.093	3.83e-06	32	89.79
Resnet100	0.025	3.55e-06	32	98.59
GoogleNet	0.045	1072.9e-06	64	85.97
AlexNet	0.086	268.4e-06	128	72.42

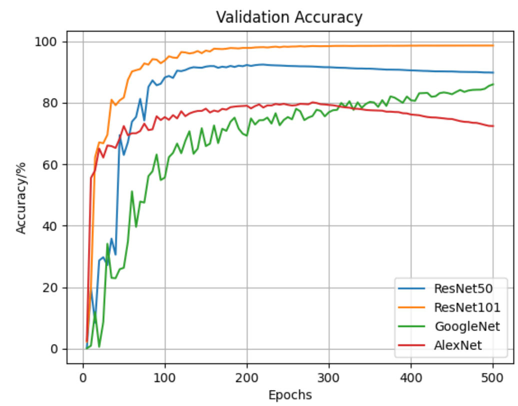
and significantly increased their likelihood of converging at the absolute minimum loss or at least a deeper minima which would provide better classification accuracy.

Both the optimizer (used to adjust parameters and weights during training) and the loss function (used to measure network performance) were kept consistent while training models to obtain the fairest comparison possible. The loss function used was Cross-Entropy loss which provided a robust measurement of the network performance by measuring the difference in probability distributions of the model's predicted output with the correct expected output. This means that it not only measures if the model has made the correct prediction but also accounts for how confident the model was in the prediction it made.

We varied key parameters including the learning rate and the batch size to find the best performance. The batch size was kept as large as possible (while being a multiple of 2) and was restricted by two factors. Also, different learning rates were used for each architecture. These were linearly spaced between 0.001 and 0.1. The aim was to train all models up to 500 epochs; however, some models had their training ended early to minimize time spent on suboptimal training models. The obtained results are given in Table II. Also, the performance of different models in terms of loss and accuracy per epoch is given in Figure 5.



(a)



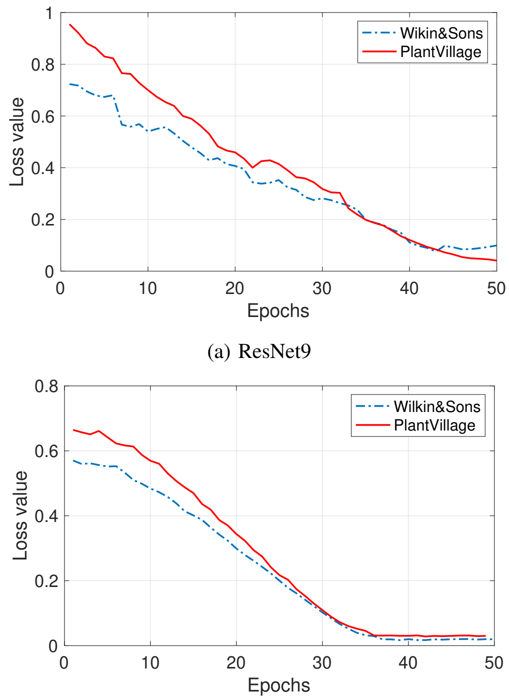
(b)

Fig. 5. Loss function (a) and accuracy (b) performance for pre-trained models.

The obtained results shown above, imply that ResNet models achieve the best performance among the others. Hence, we considered ResNet as our base architecture to propose our novel approach by adding the SE layers along with our variable learning rate. The aim has been to reduce the model size and also preserve the accuracy so that it can be suitable for deployment on the edge. To measure the performance of our proposed models, here also we have used cross-validation and considered 80% of data for training and 20% for testing. We used batch size 32 and ran the model for 50 epochs, respectively. In this experiment, we trained our models on both PlantVillage and Wilkin & Sons datasets. The training loss behaviour for both models, i.e., ResNet9 and ResNet9-SE are illustrated in Figure 6. It can be observed from this figure that both models achieve a decreasing trend in loss value, but the proposed ResNet9-SE shows a faster reduction with both datasets (Figure 6 (b)). This observation, confirms the effectiveness of adding two new SE blocks to the network architecture.

It is also worthwhile to mention that instead of using a fixed learning rate, we used a variable learning rate which changes after each batch of training. We start with a low learning rate and gradually increase it to a high learning rate in batches over about 30% of the cycles. We then gradually reduce it to a very low value over the remaining cycles, so only the maximum learning rate needs to be set when setting the parameters. The key advantages of this strategy are twofold: initially, gradually lowering the learning rate speeds up model convergence by quickly updating parameters early on and then approaching the optimal value as training progresses. Secondly, this variable learning rate approach prevents the model from getting stuck in local optima. Adjusting the rate during later stages aids in finding the global optimal solution while enhancing model stability to adapt to changing data distributions, reducing oscillation and instability during training. Through following this procedure in our experiments, we found and set the learning rates of 0.04 and 0.01 for ResNet models with the PlantVillage, and Wilkin & Sons datasets, respectively.

The final part of the results is devoted to the analysis of numerical performance and model complexities using the metrics provided at the beginning of this section. In Table III, the total number of parameters, the total size of the model, and the performance metrics are provided. As can be seen from this table, the proposed ResNet9-SE requires a slightly smaller number of parameters and occupies less memory while still achieving a high accuracy. When the SE module was added to the ResNet9 model, the number of parameters and memory footprint were slightly reduced while maintaining high accuracy. This is because the introduction of the SE



(a) ResNet9

(b) ResNet9-SE

Fig. 6. Trend of loss functions for the proposed model with different datasets.

TABLE III
PERFORMANCE EVALUATION AND MODEL COMPLEXITY OF DIFFERENT MODELS

Model	Parameters (M)	Total size (MB)	Accuracy (%)	Recall (%)	F1 Score
ResNet9	6.6	374	98	98	98
ResNet9-SE	6.5	369	97	97	97

module enhances the ability of the neural network to model features and improves the efficiency of parameter usage. First, the parameters in the SE module are globally shared, which means that the number of parameters in the SE module is fixed regardless of the size of the input image. Second, the SE module introduces a channel attention mechanism that allows the network to adaptively learn the importance of each channel and weight the feature map. This helps the network to pay more attention to the information that is useful for the classification task, reducing unnecessary information transfer and computation, and thus reducing parameter redundancy. In addition, the introduction of the SE module enhances the feature modelling capability of the network, thus further reducing the number of parameters and memory footprint. Finally, the SE module also has a regularisation effect, which helps reduce the risk of overfitting and makes the model more generalisable. Taken together, the addition of the SE module not only improves the representation ability and classification accuracy of the network but also effectively reduces the number of parameters and memory occupation, making the network more lightweight and efficient.

V. Conclusion

In this paper, a novel deep learning-based architecture was proposed for the application of strawberry plant disease detection. We designed and implemented a fully embedded electronic system including sensors' data capturing as well as RGB camera image acquisition from strawberry plants. The images were then sent to the cloud through a gateway for analysis. A new modified ResNet model, called ResNet9-SE was proposed and applied to a set of collected data as well as a publicly available dataset. The new model benefits from two SE blocks within the network to improve performance. The obtained experimental results show that the proposed model can achieve very high classification accuracy with less computation costs which shows the effectiveness of deployment on embedded systems. We further aim to expand the number of data collection nodes throughout the farm, enriching the dataset and developing a fusion model to analyse both sensors' data and image data simultaneously to provide an early and accurate prediction of potential diseases. The ultimate aim is to optimise the proposed model and deploy it as a fully operational real-time on-field disease diagnosis system in strawberry farms.

References

[1] I. Navrozidis, T. Alexandridis, A. Dimitrakos, A. Lagopodi, D. Moshou, and G. Zalidis, "Identification of purple spot disease on asparagus crops across spatial and spectral scales," *Computers and Electronics in Agriculture*, vol. 148, pp. 322–329, 2018.

[2] N. ElBeheiry and R. S. Balog, "Technologies driving the shift to smart farming: A review," *IEEE Sensors Journal*, vol. 23, no. 3, pp. 1752–1769, 2023.

[3] R. Rayhana, Z. Ma, Z. Liu, G. Xiao, Y. Ruan, and J. S. Sangha, "A review on plant disease detection using hyperspectral imaging," *IEEE Transactions on AgriFood Electronics*, vol. 1, no. 2, pp. 108–134, 2023.

[4] U. Dar, M. H. Anisi, V. Abolghasemi, C. Newenham, and A. Ivanov, "Visual sensor network based early onset disease detection for strawberry plants," in *2023 IEEE Applied Sensing Conference (APSCON)*, 2023, pp. 1–3.

[5] J. Wu, U. Dar, M. H. Anisi, V. Abolghasemi, C. N. Wilkin, and A. I. Wilkin, "Plant disease detection: Electronic system design empowered with artificial intelligence," in *2023 IEEE Conference on AgriFood Electronics (CAFE)*, 2023, pp. 30–34.

[6] J. Shin, Y. K. Chang, B. Heung, T. Nguyen-Quang, G. W. Price, and A. Al-Mallahi, "A deep learning approach for rgb image-based powdery mildew disease detection on strawberry leaves," *Computers and Electronics in Agriculture*, vol. 183, p. 106042, 2021.

[7] Q. Jiang, G. Wu, C. Tian, N. Li, H. Yang, Y. Bai, and B. Zhang, "Hyperspectral imaging for early identification of strawberry leaves diseases with machine learning and spectral fingerprint features," *Infrared Physics & Technology*, vol. 118, p. 103898, 2021.

[8] I. Abbas, J. Liu, M. Amin, A. Tariq, and M. H. Tunio, "Strawberry fungal leaf scorch disease identification in real-time strawberry field using deep learning architectures," *Plants*, vol. 10, no. 12, 2021.

[9] M. T. Islam and A. Tusher, *Automatic Detection of Grape, Potato and Strawberry Leaf Diseases Using CNN and Image Processing*, 01 2021, pp. 213–224.

[10] M. M. Hasan, J. P. Chopin, H. Laga, and S. J. Miklavcic, "Detection and analysis of wheat spikes using convolutional neural networks," *Plant Methods*, vol. 14, no. 100, 2018.

[11] Y. Toda and F. Okura, "How convolutional neural networks diagnose plant disease," *Plant Phenomics*, vol. 2019, 2019.

[12] S. Zhang, S. Zhang, C. Zhang, X. Wang, and Y. Shi, "Cucumber leaf disease identification with global pooling dilated convolutional neural network," *Computers and Electronics in Agriculture*, vol. 162, pp. 422–430, 2019.

[13] S. S. Hari, M. Sivakumar, P. Renuga, S. karthikeyan, and S. Suriya, "Detection of plant disease by leaf image using convolutional neural network," in *2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN)*, 2019, pp. 1–5.

[14] A. Picon, M. Seitz, A. Alvarez-Gila, P. Mohnke, A. Ortiz-Barredo, and J. Echazarra, "Crop conditional convolutional neural networks for massive multi-crop plant disease classification over cell phone acquired images taken on real field conditions," *Computers and Electronics in Agriculture*, vol. 167, p. 105093, 2019.

[15] M. R. Howlader, U. Habiba, R. H. Faisal, and M. M. Rahman, "Automatic recognition of guava leaf diseases using deep convolution neural network," in *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, 2019, pp. 1–5.

[16] K. Nagasubramanian, S. Jones, A. K. Singh, S. Sarkar, A. Singh, and B. Ganapathysubramanian, "Plant disease identification using explainable 3d deep learning on hyperspectral images," *Plant Methods*, vol. 15, no. 98, 2019.

[17] A. Pal and V. Kumar, "Agridet: Plant leaf disease severity classification using agriculture detection framework," *Engineering Applications of Artificial Intelligence*, vol. 119, p. 105754, 2023.

[18] A. Mathew, A. Antony, Y. Mahadeshwar, T. Khan, and A. Kulkarni, "Plant disease detection using glcm feature extractor and voting classification approach," *Materials Today: Proceedings*, vol. 58, pp. 407–415, 2022, international Conference on Artificial Intelligence & Energy Systems.

[19] P. Bedi and P. Gole, "Plant disease detection using hybrid model based on convolutional autoencoder and convolutional neural network," *Artificial Intelligence in Agriculture*, vol. 5, pp. 90–101, 2021.

[20] A. Abdalla, H. Cen, L. Wan, R. Rashid, H. Weng, W. Zhou, and Y. He, "Fine-tuning convolutional neural network with transfer learning for semantic segmentation of ground-level oilseed rape images in a field with high weed pressure," *Computers and Electronics in Agriculture*, vol. 167, p. 105091, 2019.

[21] K. Lin, L. Gong, Y. Huang, C. Liu, and J. Pan, "Deep learning-based segmentation and quantification of cucumber powdery mildew using convolutional neural network," *Frontiers in Plant Science*, vol. 10, 2019.

[22] T. Wiesner-Hanks, H. Wu, E. Stewart, C. DeChant, N. Kaczmar, H. Lipson, M. A. Gore, and R. J. Nelson, "Millimeter-level plant disease detection from aerial photographs via deep learning and crowdsourced data," *Frontiers in Plant Science*, vol. 10, 2019.

[23] Y. Li, J. Wang, H. Wu, Y. Yu, H. Sun, and H. Zhang, "Detection of powdery mildew on strawberry leaves based on dac-yolov4 model," *Computers and Electronics in Agriculture*, vol. 202, p. 107418, 2022.

[24] J.-F. Yeh, K.-M. Lin, C.-Y. Lin, and J.-C. Kang, "Intelligent mango fruit grade classification using alexnet-spp with mask r-cnn-based segmentation algorithm," *IEEE Transactions on AgriFood Electronics*, vol. 1, no. 1, pp. 41–49, 2023.

[25] S.-H. Kim, M. hun Lee, and C. Shin, "Iot-based strawberry disease prediction system for smart farming," *Sensors (Basel, Switzerland)*, vol. 18, 2018.

[26] G. Sahitya, C. Kaushik, P. S. Simha, D. Reddy, S. Kumar, and V. Vinay, "Leaf disease detection using raspberry pi," in *2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2022, pp. 1–7.

[27] E. A. H. Mora, V. A. G. Huitrón, A. E. R. Mata, and H. R. Rangel, "Plant disease detection with convolutional neural networks implemented on raspberry pi 4," in *2020 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC)*, vol. 4, 2020, pp. 1–23.

[28] C.-H. Huang, B.-W. Chen, Y.-J. Lin, and J.-X. Zheng, "Smart crop growth monitoring based on system adaptivity and edge ai," *IEEE Access*, vol. 10, pp. 64 114–64 125, 2022.

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.