PFDI: A Precise Fruit disease Identification Model based on Context Data Fusion with Faster-CNN in Edge Computing Environment

Poonam Dhiman
ambala cantt
Poongodi M (dr.m.poongodi@gmail.com)
Hamad Bin Khalifa University College of Science and Engineering https://orcid.org/0000-0001-6801-6138
Umesh Kumar Lilhore
KIET Group of Institutions: Krishna Institute of Engineering & Technology
Salman A. AlQahtani
King Saud University College of Engineering
Amandeep Kaur
Chitkara University
Celestine Iwendi
University of Bolton
Deema mohammed alsekait
Princess Nourah bint Abdulrahman University
Kaamran Raahemifar
Penn State University: The Pennsylvania State University

Research Article

Keywords: Data Fusion, Deep learning, Pruning, Disease, Sparsity, Quantization, edge computing, citrus fruit

Posted Date: January 17th, 2023

DOI: https://doi.org/10.21203/rs.3.rs-2416757/v1

Version of Record: A version of this preprint was published at EURASIP Journal on Advances in Signal Processing on June 22nd, 2023. See the published version at https://doi.org/10.1186/s13634-023-01025-

y.

Abstract

Fruits have a significant impact on everyday living i.e., citrus fruits. Numerous fruits have a solid nutritious value and are packed with multivitamins and trace components. Citrus fruits are delicate, so they are susceptible to many diseases and infections. Many researchers have suggested various deep learning and machine learning based fruit disease detection and classification models. In this research we are presenting precise fruit disease identification (PFDI) model based on context data fusion with Faster-CNN in edge computing environment. The goal is to develop a precise, efficient, and trustable fruit disease detection model, which is a critical component of an autonomous food production in robotic edge platform. This research examines and explores four different diseases of citrus fruits using CNN deep learning models to be adopted as edge computing solution. Identification of citrus diseases namely cankers black spot, greening, scab, melanose, and healthy citrus fruits are implemented using the proposed sequential model without pruning, with pruning having different sparsity level followed by post quantization. Through transfer learning method, we optimize this model for the assignment of fruit disease detection employing visuals from two patterns: Near-infrared (NIFR) and RGB. For integrating multi-model (NIFR and RGB) facts, early and late data fusion techniques are evaluated. The accuracy obtained from the proposed model for the canker disease is 97%,scab 95%, melanose 99% ,Greening 97%,Black spot 97% and for healthy 97%. In this paper we compared and evaluated the results of proposed model with the sparsity levels of 50–80%, 60–90%, 70–90%, 80–90% pruning and also obtained the results of post-quantization on each level. The results show that the size of the model with 60–90% pruning can be counteracted to the 47.64 of the baseline model without significant loss of accuracy. Moreover, post quantization can further reduces the of 60–90% pruning from 28.16 to 8.72. In addition to enhanced precision, the above initiative is much faster to implement for new fruits diseases because it needs bounding box annotation (BBA) instead of pixel-level annotation (PLA).

1. Introduction

Fruit products ought to be the establishment of a sound eating regimen. Citrus fruits are the signature product in agriculture and nearly everybody consuming them consistently. From last decade's research results demonstrated the criticality of the fruit product quality and its impact on human well beings [1]. The citrus fruits include lemon, orange, grapefruit, and tangerine assortments. There is a prerequisite for high return in horticultural enterprises; better-quality yield of fruit product is significant that lead to the requirement of automatic system in diseases detection process of diseases present in citrus fruit [2].

Utilizing modern innovative solutions, it is possible to achieve economic progress in agriculture. The use of pesticides by farmers to prevent and treat different diseases and improve crop productivity is widespread. Diseases in fruit crops are a significant cause of problems including reduced productivity and financial hardship for farmers. Therefore, the fundamental necessity in the agricultural field is early illness identification. Disease diagnosis is crucial for the efficient farming framework. From past decay, post-harvesting of citrus, one of the key challenges is the recognition of exterior defects [3].

Citrus species and cultivars are extremely unpredictable in color and texture that causes difficulties to establish an unsupervised system capable of identifying the apparent diseases. Secondly, edge computing adheres suitable with low computational efficiency and is well adapted to infield estimation that allows for real-time response and enables data or results transfer over narrowband networks. As all agricultural field instruments have minimal resources, using underlying DL models at infield is seen as a major obstacle. Identifying diseases accurately is crucial for using the right treatments, so diminution or eradication of the infestation or disease in the plantation, which may occasionally need specialized expertise or prior knowledge from the farmer [4].

Additionally, some detection procedures necessitate collecting samples and sending them to a lab for evaluation, taking much more average time, making the farming vulnerable to illness, and opening the door for disease to spread to other areas. Alternatively, in an effort to provide precise and dependable ways for disease diagnosis, researchers have started looking into novel techniques based on artificial intelligence and mobile devices. Furthermore, strategies utilizing machine learning and artificial intelligence have started to be deployed because the majority of illnesses significantly alter the visual appearance of leaves, fruits, and plant stems [5, 6, 42–48].

The paper contributed the efficiency to the fruit disease detection i.e., citrus fruit, by using Faster-CNN model and data fusion technique for edge computing environment. To execute the network infrastructure, diverse edge node specifications are employed. The novelty of our paper involves the optimization of network size using magnitude based pruning and post- quantization process. In this research Citrus fruits data with different diseases collected from different public sources are sent to the edge network for classification purposes.

Complex deep learning networks are unable to do the complex calculations necessary for disease identification on low resource devices. Extremely competent devices of the edge network contribute in obtaining the disease detection and classification and also edge clusters as complex computations by offloading deep learning contexts to the edge network. Also network is pruned and quantized for optimization of network size so that it can be further used by low resource devices by the farmers to detect the different classes of citrus fruit disease during post harvesting.

Below is a summary of our work's key contribution:

1. We are designing a Faster-CNN model that has already been pre-trained on a sizable image database, like ImageNet, to develop a powerful fruit disease detection model that can be quickly learned with a limited number of images.
2. We present multi-model data fusion perspectives that merge data from NIFR and RGB images, resulting in state-of-the-art recognition accuracy.
3. The proposed model is trained for 2 hours over real-time K-60 GPU.
4. The proposed model is the first utilization of data fusion using faster CNN and combining NIFR-RGB pattern images for fruit disease detection. Later it utilizes a bounding box prediction to generate the more precision results.

5. We conduct in-depth performance reviews of citric fruit data from an online UCI dataset utilizing performance measurement metrics, such as accuracy curves, precision, and the F1 score.

The complete article is organized as follows: Section 2 covers the related work, section 3 covers the materials and methods, and section 4 covers the experimental analysis and result discussion and section 5 covers conclusion and future scope.

2. Related Work

The scientific community pays close attention to Multi-Access Edge Computing because of its scientific, technical, and business ramifications. The arguments around edge computing are particularly consolidated by the ETSI guideline integration. Nevertheless, the majority of the current MEC research applications are insufficient, which hinders or invalidates their adoption. Understanding a variety of experimental prototypes, implementations, and deployments is crucial to closing this gap. The Early deployments can demonstrate the capabilities, constraints, linked technologies, and development tools for multi edge computing adoption [7].

A lot of IoT data is transmitted to the cloud under the cloud-based infrastructure for decision-making, investigation, and data handling. It might not only result in a high demand on the cloud but also in excessive network latency, which would potentially negate the advantages of cloud computing. Researchers are searching for new decentralized computing models for the IoT to address these difficulties. Edge computing is one such model that is gaining popularity among academic and industrial researchers. The primary principle of edge computing is to move data processing from distant cloud servers to near-edge devices. The development of more scalable, low-latency IoT systems is promising [8].

A segmentation approach was implemented by using a threshold method based on PCA and B-spline lighting correction analysis (B-spline) to get an appropriate methodology of identifying the defects on the citrus peel. Machine vision systems execute the conceptual and algorithmic estimation through which valuable data about the image or object can be automatically retrieved and examined from the acquired images [9]. Images can be incarcerated using various sensors like non-destructive, non-invasive sensors for evaluating agricultural products. Multispectral acquisition system usually acquired 2-D data at a time and scanned across the 3D. Multispectral scanners and imagers are divided into different categories based on the filtering technique like optical filters and electronically tunable filters [10].

The CCD camera mounted with a light source using the gray card is the most commonly used image acquisition system [11]. In past years, many image fusion techniques like feature level, pixel level, and symbolic level have been developed. The researchers have used image fusion for detecting orange fruits (visible-thermal images) [12]. Near-infrared spectrometer and spectral acquisition exhibit two different NIR devices i.e. micro-NIR FT-spectrometer. The comparison between this two micro-NIR and FT-spectrometer was conducted using a linear regression model [13]. Disease recognition and detection can be achieved at a reduced cost using image processing. Orange fruit images are provided as input in the proposed work; color threshold segmentation is applied to the images. Eight characteristics that are unique to orange fruit are extracted in feature extraction.

The deep learning method is then used to identify the diseases of the orange fruit at the initial level of development [14]. The paper proposed the machine vision system that detects deformities in the orange fruits and grading the different defects that occurred on the orange fruit surface. The signs of the fault mark reflect the nature of the disease and propose the best approach to coping with the disease [15]. DE color difference is used for the segmentation to get the desired area affected by the disease. Furthermore, color histograms and textural features were taken for the classification of the disease [16].

Commercial value decreases in lemon and orange fruit that are infected by physiological decays. The image processing technique is proposed by the researcher to identify the physiological disorder on X-ray Images of orange and lemon fruit. KNN techniques were used for the automatic classification of the image set [17]. To investigate the relationship between quality attributes and granulation disorder occurrences partial least square regression method was used by the authors [18].

Hybridized methods using AlexNet and random forest techniques were implemented for the classification of different diseases present in citrus fruits [19].For successful disease detection, a modern AlexNet architecture premised on deep learning was used. This approach consisted of four major processes: pre-processing, segmentation, feature extraction, and classification. Firstly, pre-processing was applied to boost the input object's attributes. The images were then segmented by applying Otsu technique. The Alex-Net model was first introduced as an extractor function. For the classification of citrus diseases, a random forest (RF) classifier was finally used [20]. PCA and neural net was employed for feature selection and ten variables as input of neural network were used to detect shelf lives [21].

In terms of citrus plant disease detection, the efficiency of ML i.e. SVM, RF, SGD & DL is evaluated and compared. There in case of disease detection, the precision of the disease classification showed that DL methods perform much better than those of ML methods. The inhibitor of 1-methylcyclopropene (1-MCP) on citrus fruit post-harvest green mould was examined using vivo analyses that revealed that 1-MCP substantially reduced the occurrence of green mould and degradation in citrus fruit post-harvest and hindered mycelia growth and P. digitatum fungal growth [22]. Table 1 represents the comparison of various existing model for fruit disease detection and classification methods.

Table 1
Comparison of various existing model for fruit disease detection

Reference	Method	Plant Type	Edge Computing	Bounding Box Predication	Transfer Learning	Accuracy %
[1]	Machine Learning with ANN	Plant Village	No	No	No	94.25
[2]	PCA	Apple Leaves	Yes	No	No	95.24
[3]	Random Forest	Cassava Leaves	No	No	No	93.56
[4]	CNN model	Fruit classification	No	Yes	Yes	91.24
[5]	Deep Learning	Apple Leave	Yes	No	Yes	90.75
[6]	SVM with ANN	Cassava Leaves	Yes	Yes	No	94.56
[7]	Hybrid Random Forest	Cassava Leaves	No	No	No	95.33
Proposed Model	Context Data Fusion with Faster CNN	Citrus Fruit disease	Yes	Yes	Yes	98.96

3. Materials And Methods

This section covers the working of proposed model, algorithm, working, dataset and comparison parameters.

3.1 Citrus fruit disease dataset

Figure 1 shows the samples of images of citrus fruit. As target objects, healthy fruit and fruit exhibiting five kinds of typical blemishes like HLB, black-spot, melanose, canker, and scab were extracted.

Samples of each peel state are represented by Fig. 1. (Fig. 1A) represents the healthy peel condition of the citrus sample. (Fig. 1B)HLB (CandidatusLiberibacterasiaticus) decreases the size of citrus fruits and does not color the fruit properly i.e. it remains green and with a bent central core, the fruit is skewed, inducing fruit deformity and cracking (Fig. 1B) [23]. There may be botched seeds in the affected fruit and they have a salty, bitter taste. As several fruits fall prematurely from infected trees, HLB also decreases citrus yields. Black spots (Fig. 1C) in fruits with a diameter of 0.12 to 0.4 are small, circular, and dangerous spots [21]. The presence of Citrus Black Spot symptoms can differ and cause aesthetic abnormalities on the fruit crust. The usual lesion, identified as a hard spot, begins with tiny red brick spots with black borders that grow in size and produce tissue necrosis at the center of the lesion [24].

Melanose (Fig. 1D) is precipitated by Diaporthecitri and is distinguished by dispersed raised blotches of brown to black color. The disease produces a gradual scar in the fruit which is barely capable of affecting the total yield of fruit production but causes noticeable blemishes which reduce the viability of the fruit destined for the producing market [25]. The crop fungal infections that are commonly present in different regions producing citrus cultivars are Citrus scab (Fig. 1E). Generally, the scab incidence seems to be more severe in-plane regions with frequent wetting compared to tropical areas. Scab damages can be identified less than a week after getting an infection of the fruit [26]. The pathogens often occur with a gritty and irregular appearance as tiny dots. The fruit spot diameter of Canker (Fig. 1F) is around 1–10 mm and is covered by water-dipped and yellow curve-like blemishes [27]. Table 2 represents the number of the image samples for each type of disease and its proportion available in the dataset.

Table 2
Count and Proportion of Citrus Samples

Disease Type	Number	Proportion
Canker	370	14.4
Scab	409	16
Melanose	280	10.9
Greening	440	17.2
Black Spot	390	15.2
Healthy	667	26

3.2 Proposed PFDI Model

The proposed PFDI model is based on Context data fusion with Faster-CNN in edge computing environment.

3.2.1 Edge Computing Platform

Encouraging the remarkable resource consumption and delay in the processing time of the complex deep learning mechanism there is a much need of optimized mechanism for many applications. To detect or classify diseases present in citrus fruits in multi-classification manner, it's essential to advance the existing deep learning framework employed in the automatic disease detection system. All image samples collected from the various online platforms, while other services such as pre-processing and computations in the edge nodes make use of different clusters [28].

Figure 2 shows the working of Edge computing architecture of proposed model. The complete framework consists of four major modules, each corresponding to specific tasks. First module is consisting of collector nodes. Web services or edge computing are deployed second module, third module consist of prediction process and model pruning with quantization and user applications are deployed in module 4. The third module serves as a bridge between platform-based local and remote functions. The data collection is carried out locally from public domain, however after it has been set off, edge services are incorporated for processing and computational task [29].

Different clusters were created analogously to provide broader coverage of potential application scenarios. The platform's applications and services are accessible to users through module four, which is the last module. The layer is therefore in charge of giving the user full access to all services offered by the edge computing platform. This module offers access to the computer vision application via virtual network computing, evaluating and analyzing the outcome. The public dataset consist of citrus fruit diseases of different types is send to the local collector node. The detection layer contains a deep CNN model with magnitude pruning and quantization, an internet connection is established. The remote access of deep model output is provided by the visualization layer to [30]. Comparing the edge learning servers to the most advanced cloud computing architecture, the edge learning servers reduce the workload of the network infrastructure. The image samples can be pre-processed to eliminate some challenges due to low contrast, such as light effects and flickering, etc. The preprocessing phase serves a critical role as minimal disparity images decrease the lesion segmentation's accuracy in the field of image processing [31].

We need to normalize to the equivalent image dimension in order to standardize the training data because all of the images in the training data are sparse and have different dimension sizes. Hence, the size of 256 X 256 has been used for all images in the training set. The deep learning based technique can also be employed by the edge learning servers for data augmentation to obtain new data without any labeling costs. As well, the data augmentation would be used to enhance the capabilities of the designed Faster-CNN model [32]. Therefore, 8 standard were included to supplement our training set and test set i.e. horizontal flip, vertical flip, brightness, rescaling, shear, zca-whitening, rotation, height, and width shift [33]. Initially, fruit samples collected were 2556 including six classes which reached to 20,448 after applying the eight function of data augmentation. Table 3 shows the samples of citrus fruits obtained after performing different data augmentation operations on the dataset.

As the edge servers are near to the client devices, the communication latency between the client devices and the edge servers is substantially lower than that of the cloud server. Model used for training. To offer end users consistent service, the cloud could be installed on edge servers, and image data can be continually uploaded to the cloud in order to modify the system. The classification process of citrus fruit diseases is briefly explained in Fig. 3. The major steps are (a) image pre-processing, (b) data augmentation (c) Model implementation and training (d) Pruning (e) Quantization (d) Performance evaluation. As seen in Fig. 3, each phase is made up of a series of steps. The detail of each process is given below.

3.2.2 Magnitude based Pruning with Polynomial Decay Based Sparsity on Faster-CNN Model

By using deep learning model have a huge number of trainable weights, where learning can indeed be obtained. Contribution of all the weights and feature vectors to model performance is not equivalent. In this paper proposed model is pruned using magnitude-based pruning which removes insignificant weights after each epoch. This approach compares the exact size of the weight with a certain threshold value δ. Input vector x and weight vector w within the neuron is multiplied [34]. If the weights in the vector are set to zero, the outcome will always be zero that defines in Eq. 1. This, in effect, ensures that the neuron no longer contributes to model performance. Table 4 represents the configuration of the proposed pruned sequential Convolutional neural network model.

$$(w_t) = \{w_t : if\,|w_t| > \delta 0 : if\,|w_t| < \delta$$

1

Table 4
Configuration of Proposed Pruned CNN Model

Layer (type)	Output Shape	Param #
Prune._.low._.magnitude._.conv2d	(None, 256, 256, 32)	1762
Prune_low_magnitude_activation	(None, 256, 256, 32)	1
Prune._.low._.magnitude._.conv.2d._1	(None, 256, 256, 32)	36930
Prune._.low._.magnitude._.activation	(None, 256, 256, 32)	1
Prune._.low._.magnitude._.max _ pool	(None, 256, 256, 32)	1
Prune._.low._.magnitude._.dropout	(None, 256, 256, 32)	1
Prune._.low._.magnitude._conv2d_2	(None, 256, 256, 32)	73794
Prune._.low._.magnitude._.activation	(None, 256, 256, 32)	1
Prune._.low._.magnitude._.con2d._.3	(None, 256, 256, 32)	73794
Prune._.low._.magnitude._.activation	(None, 256, 256, 32)	12
Prune._.low_.magnitude._.max._.pool	(None, 256, 256, 32)	1
Prune._.low_magnitude_dropout	(None, 256, 256, 32)	1
Prune_low_magnitude_flatten	(None,28224)	1
Prune_low_magnitude_dense	(None,512)	28901890
Prune_low_magnitude_activation	(None,512)	1
Prune_low_magnitude_dropout	(None,512)	1
Prune_low_magnitude_dense_1	(None,5)	5127
Prune_low_magnitude_activation	(None,5)	1

At every step, all the connections of the network are iteratively pruned by setting up the levels of step sparsity L using polynomial decay function.Step sparsity L level is represented by the Eq. (2)

$$L = s_t + (s_i - s_t)\left(1 - \frac{j-i}{i-e}\right)$$

2

Where targeted sparsity level is represented by s_t; initial sparsity level is s_i, initial iteration is i, e is the end iteration, and the current iteration is represented by j. The threshold of significant connections is calculated using the sparsity level step. The threshold value is the total weights multiplied by the value at the position step sparsity level. When the total absolute values every weight filters is below the threshold limit then all the mask values belonging to its weight are adjusted to 0 as shown in Eq. 2. Then the dot product sets the pruned weights as zero. Using polynomial decay-based sparsity, more or less sparsity can be used with increasing or decreasing speed, as training progresses [35]. Initially, we set the model to be 50–80% sparse, increasingly getting sparser to eventually 80% and 90%. We begin at 0 and end at end step. Finally, the prune_low_magnitude functionality which generates the prunable model will be executed from our initial baseline Faster-CNN model and the defined pruning_params. The requirements to digitize the system, some numbers are allocated to various pieces of hardware. The requirements the maximum number is offered with good resources.

3.2.3 Edge Learning Platform

Figure 4 shows the edge learning frame work used by the proposed technique.

For every six classes including scab, HLB, melanose, black spot, canker, and healthy samples of citrus fruits, the training, validation, and test sample comprise of randomized input samples data with 8:2 ratios. For training the model one set s used as training and validation another as the test set to evaluate the model training. Furthermore, an optimizer (SGD) controls the gradient steps for every dimension of the loss feature with 0.9 momentum magnitude. The model training is done on the image samples using stochastic gradient descent having 0.0001 as the learning rate, batch dimension is 64, and epoch range is 20. Furthermore, output layer soft-max having categorical cross-entropy loss is used [36].

3.2.4 Quantization Process

Quantization lowers the model's representation that enables faster processing and smaller memory space. By ensuring that all weights and activations are quantized latency, processing, and power consumption can be boosted [37]. Quantization may be done during training that is called pre-quantization and post-quantization is done after training. In pre-quantization, the fixed point training approach is used for the direct training of the floating-point model. Whereas, Post quantization is used for the conversion of fixed-point representation from a pre-trained floating-point model and inferences can be taken from the fixed-point computation. In this paper, post-training quantization was adopted after pruning the proposed model. Fixed-point weights quantization has been applied that compressed the weights matrix from 32-bit to 8-bit floating-point values. Furthermore, Flat Buffer protocols are used as the basis for this transformation, which helps in bypassing a lot of the traditional expensive file parsing and un-parsing which leads to slower execution.

3.2.5 Context Data Fusion

Multiple data sources are combined through the procedure of "data fusion," which results in statistical information which is more reliable, precise, and beneficial than the whole the data supplied by any one source alone. In the proposed model the data a combination of early and late data fusion is applied. The classification data from the two components, colour as well as NIR visual imagery, is combined in late fusion [38]. Each multi-model M generates initiatives for the (NM, R) terrain. Such region key points are coupled to create a singular set of (Ns*Pr = M*NM, Pr), region propositions, which combines the two modules. The pr^{th} proposed terrain of the M^{th}, model type is then set a score (ScoreM, Pr). Eq. 3 shows the score model formulation.

$$\mathrm{ScoreM} = \sum\nolimits_{M=1}^{NM} (\mathrm{ScoreM}, \mathrm{Pr})$$

3

Similar in the early fusion data process the input layer of faster-CNN has number of Channels 4 (one NIR and three RGB).

3.2.6 Transfer Learning

It takes a lot of time and materials to train Faster-CNNs, particularly for networks with many layers. We employed two transfer-learning strategies known as "fixed feature extraction process" and "fine-tuning process" to prevent having to retrain a whole network on citrus fruit. The simpler option is a fixed feature extraction method that further utilizes a channel that has been received training on a previously untrained category or entity to identify it or categories it. To identify fruits in this instance, we utilized weight that was received training and managed to learn on citrus fruits. The trained weights of the network that use the same architectural features are transferred directly to the main channel we just want prepare with updated information through fine tuning. In order to prepare the network for the detection of citrus fruits, that have fewer training sets, we utilized the network's weights that had been trained on fruit including its base [37].

3.3 Performance measuring parameters

The proposed model PFDI and existing model are compared based on performance measuring parameters precision (PC), Recall (RC), F1 score (FS) and accuracy (Acy) [39].

$$PC = \left[\frac{Tp}{Tp + FP}\right]$$

4

$$RC = \left[\frac{Tp}{Tp + FN}\right]$$

5

$$FS = \left[\frac{(2 * PC * RC)}{PC + RC}\right]$$

6

$$Acy = \left[\frac{Tp + TN}{Tp + FP + FN + TN}\right]$$

7

Where TP: True positive, FP: False positive, TN: True Negative, FN: False Negative

4. Experimental Results And Discussion

The model's efficiency is assessed and calculated using the function of 80 – 20 cross-validation. Performance of the classification model was evaluated using cross-entropy loss function. Adam has been chosen for the pruning model, while SGD optimizer has also been adopted for the baseline model to enhance the cross-entropy parameter. The outcome of the proposed model on the citrus fruit’s image dataset is captured and represented in Table 5 as the Confusion matrix. F-measure specificity, recall, and Precision, calculated for each class disease are encapsulated in Table 6.

Table 5
Classification Result of Proposed Deep Model on a Prepared Dataset

Disease	**Predicted Class**						**Accuracy**
	Canker	Scab	Melanose	Greening	Blackspot	Healthy	
Canker	40	3	0	4	1	0	97%
Scab	0	8	1	0	0	0	95%
Melanose	1	0	4	0	0	0	99%
Greening	0	0	0	3	2	0	97%
Blackspot	0	5	1	0	76	0	97%
Healthy	0	0	0	3	6	152	97%

Canker performs the best out of the five diseases, with precision of 97.5%, recall of 83.3%, specificity of 99.6%, and F-measure of 98.9%. Comparing the model with the other classes, citrus scab had the lowest accuracy (95%). Because the signs generated by many diseases might be quite indistinguishable and can act synergistically resulting in low accuracy rate. At some point, the signs of a black spot and a scab may resemble with one another which leads to a misdiagnosis. The encouraging and supporting outcomes of the model in identification of disorders from citrus fruits samples shows that DL approaches has a significant role in disease detection and classification to come. Few limitations of the research can get the better analysis which might be possible if more data can be gathered.

Table 6
Class-wise Precision, Recall, Specificity, and F-measure of the Proposed Model

Class	Precision	Recall	Specificity	F-measure
Canker	97.5%	83.3%	99.6%	89.8%
Scab	89%	92.7%	96%	90.8%
Melanose	66.7%	80%	33.3%	66.9%
Greening	30%	60%	97.7%	40%
Blackspot	80%	88.9	97.3%	84.2%
Healthy	100%	94.4%	100%	97%

Once pruning finishes, the effectiveness of the model is evaluated by measuring how much performance and model size has changed, compared to before pruning and quantization [40]. The baseline model is compared with the pruning as well as a post-quantization model at different sparsity levels i.e. the starting level of sparsity begins with 50 percent and sparsity ends at 80 percent. Likewise, results were compared with 60–90%, 70–90%, and 80–90% sparsity levels using evaluation parameters like accuracy, loss, precision, recall, and size are represented in Table 7.

Table 7
Comparison of Different Parameters before and after compression on Faster-CNN model

Evaluation Parameter	Without Pruning	50–80% Pruning	60–90% Pruning	70–90% Pruning	80–90% Pruning
Accuracy	96.92%	96.85%	96.03%	92.35%	80.36%
Loss	0.1817	0.1328	0.1557%	0.2067%	0.50%
Precision	96.07%	94.85%	96.05%	93.08%	86.41%
Recall	95.5%	94.85%	94.13%	92.35%	69.45%
F-Score	95.78%	94.85%	92.71%	92.71%	77%
Size	53.78 MB	50.91MB	28.16 MB	22.81 MB	17.33 MB
Pruning + post-quantization					
Size	14.11 MB	8.21 MB	8.23MB	6.62MB	4.81MB
Accuracy	89.2%	83.3%	87.2%	78.8%	79.2%

The accuracy graphs and the loss graphs are also plotted for above said, models. It can be observed in loss graph Fig. 3(a) that the training and validation losses percentage of around 0.18 losses is there in the proposed Faster-CNN model without pruning represented in Figs. 5(a) and 5(b) shows training & validation graph which is approaching 96.92 percent.

The loss graph in Fig. 6(a) represents that the training & validation losses is nearly to 0.13 and approaching to zero with model pruning of sparsity range of 50–80% and the training and validation graph which is approaching to 96.03 percent with starting level of sparsity begins with 50 percent and sparsity ends at 80 percent represented in Fig. 6(b).

The loss graph figure VII (a) indicates that both the training and validation loss percentage are nearly to 0.20 with model pruning having sparsity range of 80–90% and the training and validation graph which is approaching to 96.03 percent with starting level of sparsity begins with 80 percent and sparsity ends at 90 percent represented in Fig. 7 (b).

Similarly, the loss graph indicates that the 15 percent of loss is encountered in both the training and validation loss with model pruning having sparsity range of 70–90% as mentioned in Fig. 8 (a) and the training and validation graph which is approaching to 92.35 percent with starting level of sparsity begins with 80 percent and sparsity ends at 90 percent represented in Fig. 8 (b).

The loss graph indicates that the training and validation losses are deviated towards loss percentage of around 0.50 with model pruning having sparsity range of 80–90% as mentioned in Fig. 9 (a) and the training and validation graph which is approaching to 80.36 percent with starting level of sparsity begins with 80 percent and sparsity ends at 90 percent represented in Fig. 9 (b). It is observed that the training and validation accuracy is minimum and loss is maximum in this scenario.

5. Conclusion And Future Work

Effective usages of resources are the primary issues MIC environment due to the devices with limited resources. As the high resource consumption for the process of deep neural network, pruning and quantization turn to the edge network to handle their processing tasks. The paper presented the deep learning inference on low-resource devices, and then outlined key techniques for size compression that results in fast execution of CNN on systems with fewer resources. Edge clusters, which are groups of nodes with suitable performance standards like pre-processing and deep learning techniques arise at the edge networks. The defect identification is done by implementing a Faster-CNN model for the detection of cankers black spot, greening, scab, melanose, and healthy citrus fruits. As there exist an issue of over parameterization of deep neural networks. Pruning techniques can eliminate a large percentage of network parameters while preserving accuracy.

In this paper, deep learning model pruning is implemented on Faster-CNN model using Magnitude-based pruning with polynomial decay based sparsity on Faster-CNN model. The suggested model reduces memory usage and processing time while improving the resource efficiency of the DL techniques. The model attained an average accuracy of 96.92%, and average precision, F-score, recall, and loss are 96.07%, 95.5%, 95.78%, 99.3%, and 0.1817 respectively without using the pruning technique.

In order to assess the model's performance, pruning process followed by quantization to compress the proposed Faster-CNN model that applied on a combined dataset collected from three different resources www, Kaggle, and plant village. Post-quantization followed by pruning is used to compress and deploy the model to a mobile or an embedded application. On the other hand, the results indicate that by pruning 60–90 percent before post quantization of the model with Magnitude-based pruning including polynomial decay-based sparsity, it is able to compress the model by an average factor of 47 without considerable hampering the accuracy. Moreover, pruning followed by fixed-point weights quantization can further compress the size of the proposed model up to 8.23 MB without considerable accuracy loss. The findings are expected to promote the use of pruning with quantization a legitimate a mechanism for compression of deep learning complex models for implementation in scarce resource scenarios. When the edge network contains the components of the same kind, the limitations of the work could be seen. By putting this experiment into practice with real-time farming images, this study can be expanded in the future.

Declarations

Please ensure that the Declarations section contains all of the following sub-sections (all of which should have clear sub-headings):

ETHICS APPROVAL AND CONSENT TO PARTICIPATE

Not Applicable

CONSENT FOR PUBLICATION

Not Applicable

AVAILABILITY OF DATA AND MATERIAL

Data available upon request

COMPETING INTERESTS

The authors declare that they have no competing interests.”

FUNDING

Not Applicable

AUTHORS' CONTRIBUTIONS

Conceptualization by Poonam Dhiman , Umesh Kumar Lilhore and Poongodi M;   Methodology by Salman A. Alqahtani and Umesh Kumar Lilhore; Software by Amandeep Kaur and Deema Mohammed Alsekait; formal analysis by Celestine Iwendi and Kaamran Raahemifar, Investigation by Umesh Kumar Lilhore and Salman A. Alqahtani and Poongodi M; Resources and data collection by Amandeep Kaur and Deema Mohammed Alsekait; Writing by all the authors; Validation by: all the authors. All authors read and approved the final manuscript.

ACKNOWLEDGEMENTS

The authors are grateful to the Deanship of Scientific Research at King Saud University for funding this work through the Vice Deanship of Scientific Research Chairs: Research Chair of New Emerging Technologies and 5G Networks and Beyond.

References

1. Mishra, Satyasis, Tadesse Hailu Ayane, V. Ellappan, Davinder Singh Rathee, and Harish Kalla. "Avocado fruit disease detection and classification using modified SCA–PSO algorithm-based MobileNetV2 convolutional neural network." *Iran Journal of Computer Science* 5, no. 4 (2022): 345-358.
2. Panhwar, Ali Orangzeb, Anwar Ali Sathio, Abdullah Lakhan, Muhammad Umer, Rabia Mushtaque Mithiani, and Sanwali Khan. "Plant Health Detection Enabled CNN Scheme in IoT Network." *International Journal of Computing and Digital Systems* 11, no. 1 (2022): 344-335.

3. Zhang, Wenli, Yuxin Liu, Kaizhen Chen, Huibin Li, Yulin Duan, Wenbin Wu, Yun Shi, and Wei Guo. "Lightweight Fruit-Detection Algorithm for Edge Computing Applications." *Frontiers in Plant Science* (2021): 2158.
4. Khan, Muhammad Attique, Tallha Akram, Muhammad Sharif, Muhammad Awais, Kashif Javed, Hashim Ali, and Tanzila Saba. "CCDF: Automatic system for segmentation and recognition of fruit crops diseases based on correlation coefficient and deep CNN features." *Computers and electronics in agriculture* 155 (2018): 220-236.
5. Cruz, Mateus, Samuel Mafra, Eduardo Teixeira, and Felipe Figueiredo. "Smart Strawberry Farming Using Edge Computing and IoT." *Sensors* 22, no. 15 (2022): 5866.
6. Kamal, Raj, Sadhna Tiwari, Savita Kolhe, and Manojkumar Vilasrao Deshpande. "A design approach for identifying, diagnosing and controlling soybean diseases using CNN based computer vision of the leaves for optimizing the production." In *IOP Conference Series: Materials Science and Engineering*, vol. 1099, no. 1, p. 012037. IOP Publishing, 2021.
7. El-aziz, Abd, A. Atrab, Ashraf Darwish, Diego Oliva, and Aboul Ella Hassanien. "Machine learning for apple fruit diseases classification system." In *The International Conference on Artificial Intelligence and Computer Vision*, pp. 16-25. Springer, Cham, 2020.
8. Hsu, Hsin-Yao, Gautam Srivastava, Hsin-Te Wu, and Mu-Yen Chen. "Remaining useful life prediction based on state assessment using edge computing on deep learning." *Computer Communications* 160 (2020): 91-100.
9. Abirami, S., and M. Thilagavathi. "Classification of fruit diseases using feed forward back propagation neural network." In *2019 International Conference on Communication and Signal Processing (ICCSP)*, pp. 0765-0768. IEEE, 2019.
10. Akram, Tallha, Muhammad Sharif, and Tanzila Saba. "Fruits diseases classification: exploiting a hierarchical framework for deep features fusion and selection." *Multimedia Tools and Applications* 79, no. 35 (2020): 25763-25783.
11. Elangovan, K., and S. Nalini. "Plant disease classification using image segmentation and SVM techniques." *International Journal of Computational Intelligence Research* 13, no. 7 (2017): 1821-1828.
12. Xiao, J.R., Chung, P.C., Wu, H.Y., Phan, Q.H., Yeh, J.L.A. and Hou, M.T.K., 2020. Detection of strawberry diseases using a convolutional neural network. *Plants*, *10*(1), p.31.
13. Hussain, Israr, Jishen Zeng, Xinhong Qin, and Shunquan Tan. "A survey on deep convolutional neural networks for image steganography and steganalysis." *KSII Transactions on Internet and Information Systems (TIIS)* 14, no. 3 (2020): 1228-1248.
14. Khan, Muhammad Attique, Tallha Akram, Muhammad Sharif, Muhammad Awais, Kashif Javed, Hashim Ali, and Tanzila Saba. "CCDF: Automatic system for segmentation and recognition of fruit crops diseases based on correlation coefficient and deep CNN features." *Computers and electronics in agriculture* 155 (2018): 220-236.

15. Kour, Vippon Preet, and Sakshi Arora. "Fruit disease detection using rule-based classification." In *Smart innovations in communication and computational sciences*, pp. 295-312. Springer, Singapore, 2019.
16. Melo, Ramásio Ferreira de, Gustavo Lameirão de Lima, Guilherme Ribeiro Corrêa, Bruno Zatt, Marilton Sanchotene de Aguiar, Gilmar Ribeiro Nachtigall, and Ricardo Matsumura Araújo. "Diagnosis of Apple Fruit Diseases in the Wild with Mask R-CNN." In *Brazilian Conference on Intelligent Systems*, pp. 256-270. Springer, Cham, 2020.
17. Wang, HongJun, Qisong Mou, Youjun Yue, and Hui Zhao. "Research on detection technology of various fruit disease spots based on mask R-CNN." In *2020 IEEE International Conference on Mechatronics and Automation (ICMA)*, pp. 1083-1087. IEEE, 2020.
18. Ranjit, K. N., H. K. Chethan, and C. Naveena. "Identification and classification of fruit diseases." *International Journal of Engineering Research and Application (IJERA)* (2016): 11-14.
19. Alharbi, Asmaa Ghazi, and Muhammad Arif. "Detection and classification of apple diseases using convolutional neural networks." In *2020 2nd International Conference on Computer and Information Sciences (ICCIS)*, pp. 1-6. IEEE, 2020.
20. Shalini, K., Abhishek Kumar Srivastava, Surendra Allam, and Dilip Lilaramani. "Comparative analysis on Deep Convolution Neural Network models using Pytorch and OpenCV DNN frameworks for identifying optimum fruit detection solution on RISC-V architecture." In *2021 IEEE Mysore Sub Section International Conference (MysuruCon)*, pp. 738-743. IEEE, 2021.
21. Lilhore, U.K., Imoize, A.L., Lee, C.C., Simaiya, S., Pani, S.K., Goyal, N., Kumar, A. and Li, C.T., 2022. Enhanced convolutional neural network model for cassava leaf disease identification and classification. *Mathematics*, *10*(4), p.580.
22. Jerlin Rubini, L., and Eswaran Perumal. "Efficient classification of chronic kidney disease by using multi-kernel support vector machine and fruit fly optimization algorithm." *International Journal of Imaging Systems and Technology* 30, no. 3 (2020): 660-673.
23. Jiang, He, Xiaoru Li, and Fatemeh Safara. "IoT-based agriculture: Deep learning in detecting apple fruit diseases." Microprocessors and Microsystems (2021): 104321.
24. Doh, Benjamin, Duo Zhang, Yue Shen, Fida Hussain, Ronky Francis Doh, and Kwaku Ayepah. "Automatic citrus fruit disease detection by phenotyping using machine learning." In 2019 25th International Conference on Automation and Computing (ICAC), pp. 1-5. IEEE, 2019.
25. Chen, Xiao, Guoxiong Zhou, Aibin Chen, Ling Pu, and Wenjie Chen. "The fruit classification algorithm based on the multi-optimization convolutional neural network." Multimedia Tools and Applications 80, no. 7 (2021): 11313-11330.
26. Trivedi, Naresh Kumar, Sarita Simaiya, Umesh Kumar Lilhore, and Sanjeev Kumar Sharma. "COVID-19 Pandemic: Role of Machine Learn-ing & Deep Learning Methods in Diagnosis." Int J Cur Res Rev 13, no. 06 (2021): 150-156.
27. Khan, Muhammad Attique, Tallha Akram, Muhammad Sharif, Majed Alhaisoni, Tanzila Saba, and Nadia Nawaz. "A probabilistic segmentation and entropy-rank correlation-based feature selection

approach for the recognition of fruit diseases." EURASIP Journal on Image and Video Processing 2021, no. 1 (2021): 1-28.

28. Poongodi, M., Mohit Malviya, Chahat Kumar, Mounir Hamdi, V. Vijayakumar, Jamel Nebhen, and Hasan Alyamani. "New York City taxi trip duration prediction using MLP and XGBoost." International Journal of System Assurance Engineering and Management 13, no. 1 (2022): 16-27.
29. Nikhitha, M., S. Roopa Sri, and B. Uma Maheswari. "Fruit recognition and grade of disease detection using inception v3 model." In 2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA), pp. 1040-1043. IEEE, 2019.
30. Lilhore, Umesh Kumar, Sarita Simaiya, Jasminder Kaur Sandhu, Naresh Kumar Trivedi, Atul Garg, and Aditi Moudgil. "Deep Learning-Based Predictive Model for Defect Detection and Classification in Industry 4.0." In 2022 International Conference on Emerging Smart Computing and Informatics (ESCI), pp. 1-5. IEEE, 2022.
31. Poongodi, M., Mounir Hamdi, Mohit Malviya, Ashutosh Sharma, Gaurav Dhiman, and S. Vimal. "Diagnosis and combating COVID-19 using wearable Oura smart ring with deep learning methods." Personal and ubiquitous computing 26, no. 1 (2022): 25-35.
32. Pan, Leiqing, Wei Zhang, Na Zhu, Shubo Mao, and Kang Tu. "Early detection and classification of pathogenic fungal disease in post-harvest strawberry fruit by electronic nose and gas chromatography–mass spectrometry." Food Research International 62 (2014): 162-168.
33. Lilhore, Umesh Kumar, Sarita Simaiya, Himanshu Pandey, Vinay Gautam, Atul Garg, and Pinaki Ghosh. "Breast Cancer Detection in the IoT Cloud-based Healthcare Environment Using Fuzzy Cluster Segmentation and SVM Classifier." In Ambient Communications and Computer Systems, pp. 165-179. Springer, Singapore, 2022.
34. Shahi, Tej Bahadur, Chiranjibi Sitaula, Arjun Neupane, and William Guo. "Fruit classification using attention-based MobileNetV2 for industrial applications." Plos one 17, no. 2 (2022): e0264586.
35. Kumari, Neeraj, and Rajendra Belwal. "Hybridized approach of image segmentation in classification of fruit mango using BPNN and discriminant analyzer." Multimedia Tools and Applications 80, no. 4 (2021): 4943-4973.
36. Ibrahim, Nehad M., Dalia Goda Ibrahim Gabr, Atta-ur Rahman, Sujata Dash, and Anand Nayyar. "A deep learning approach to intelligent fruit identification and family classification." Multimedia Tools and Applications (2022): 1-16.
37. Rauf, Hafiz Tayyab, Basharat Ali Saleem, M. Ikram Ullah Lali, Muhammad Attique Khan, Muhammad Sharif, and Syed Ahmad Chan Bukhari. "A citrus fruits and leaves dataset for detection and classification of citrus diseases through machine learning." Data in brief 26 (2019): 104340.
38. Kim, Changsu, Hyesoo Lee, and Hoekyung Jung. "Fruit tree disease classification system using generative adversarial networks." International Journal of Electrical & Computer Engineering (2088-8708) 11, no. 3 (2021).
39. Ye, Yuanbo, Houkui Zhou, Huimin Yu, Roland Hu, Guangqun Zhang, Junguo Hu, and Tao He. "An Improved EfficientNetV2 Model Based on Visual Attention Mechanism: Application to Identification

of Cassava Disease." Computational Intelligence and Neuroscience 2022 (2022).

40. Trivedi, Naresh Kumar, Abhineet Anand, Umesh Kumar Lilhore, and Kalpna Guleria. "Deep Learning Applications on Edge Computing." In Machine Learning for Edge Computing, pp. 143-168. CRC Press.
41. Hua, Haochen, Yutong Li, Tonghe Wang, Nanqing Dong, Wei Li, and Junwei Cao. "Edge Computing with Artificial Intelligence: A Machine Learning Perspective." ACM Computing Surveys (CSUR) (2022).
42. Deng, Fangming, Wei Mao, Ziqi Zeng, Han Zeng, and Baoquan Wei. "Multiple Diseases and Pests Detection Based on Federated Learning and Improved Faster R-CNN." IEEE Transactions on Instrumentation and Measurement 71 (2022): 1-11.
43. Poongodi, M., Ashutosh Sharma, Mounir Hamdi, Ma Maode, and Naveen Chilamkurti. "Smart healthcare in smart cities: wireless patient monitoring system using IoT." The Journal of Supercomputing 77, no. 11 (2021): 12230-12255.
44. Poongodi, M., Bourouis, S., Ahmed, A. N., Vijayaragavan, M., Venkatesan, K. G. S., Alhakami, W., & Hamdi, M. (2022). A Novel Secured Multi-Access Edge Computing based VANET with Neuro fuzzy systems based Blockchain Framework. Computer Communications.
45. Manoharan, P., Walia, R., Iwendi, C., Ahanger, T. A., Suganthi, S. T., Kamruzzaman, M. M., ... & Hamdi, M. (2022). SVM-based generative adverserial networks for federated learning and edge computing attack model and outpoising. Expert Systems, e13072.
46. Ramesh, T. R., Lilhore, U. K., Poongodi, M., Simaiya, S., Kaur, A., & Hamdi, M. (2022). PREDICTIVE ANALYSIS OF HEART DISEASES WITH MACHINE LEARNING APPROACHES. Malaysian Journal of Computer Science, 132-148.
47. Iwendi, C., Srivastava, G., Khan, S., & Maddikunta, P. K. R. (2020). Cyberbullying detection solutions based on deep learning architectures. Multimedia Systems, 1-14.
48. Dhiman, P., Kukreja, V., Manoharan, P., Kaur, A., Kamruzzaman, M. M., Dhaou, I. B., & Iwendi, C. (2022). A Novel Deep Learning Model for Detection of Severity Level of the Disease in Citrus Fruits. Electronics, 11(3), 495.

Table 3

Table 3 is available in the Supplementary Files section.

Figures

(1A) (1B) (1C) (1D) (1E) (1F)

Figure 1

Healthy and infected Fruit images collected from plant village, www and image gallery

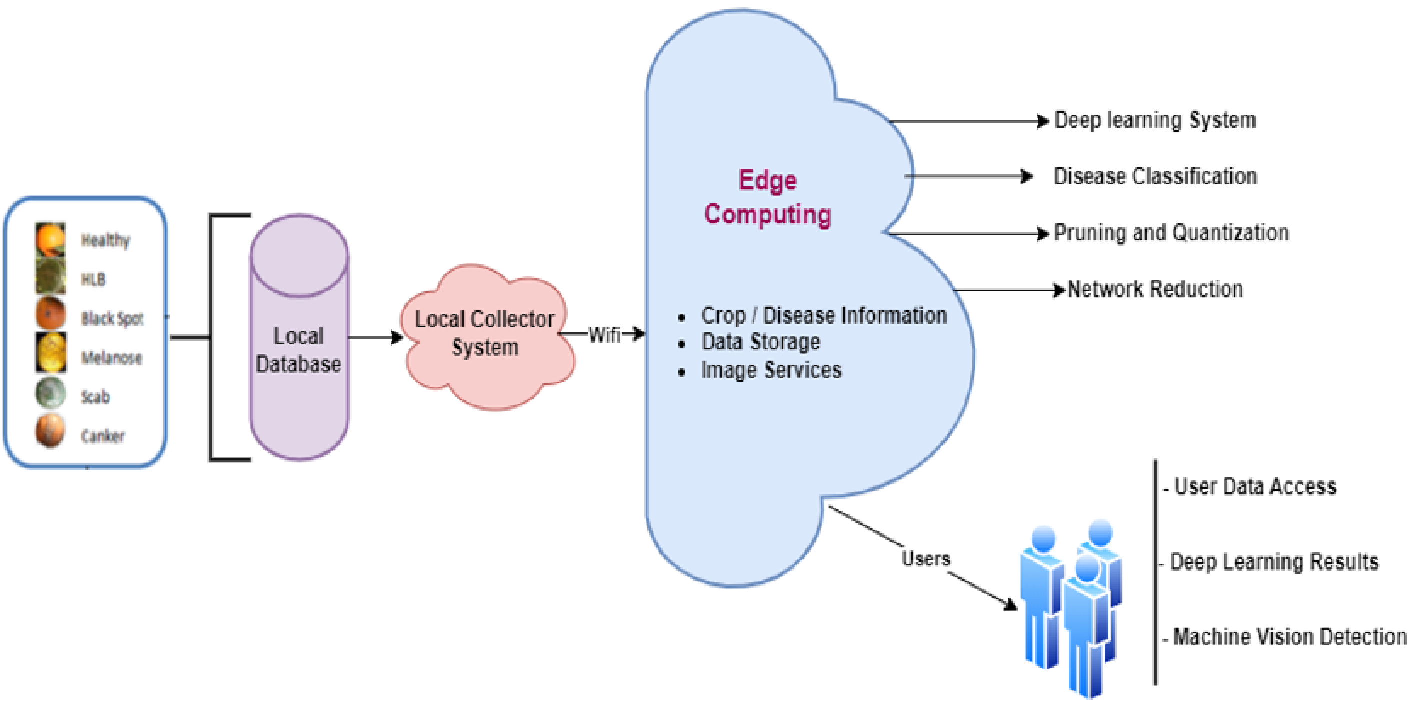


Figure 2

Overall Edge Computing Architecture

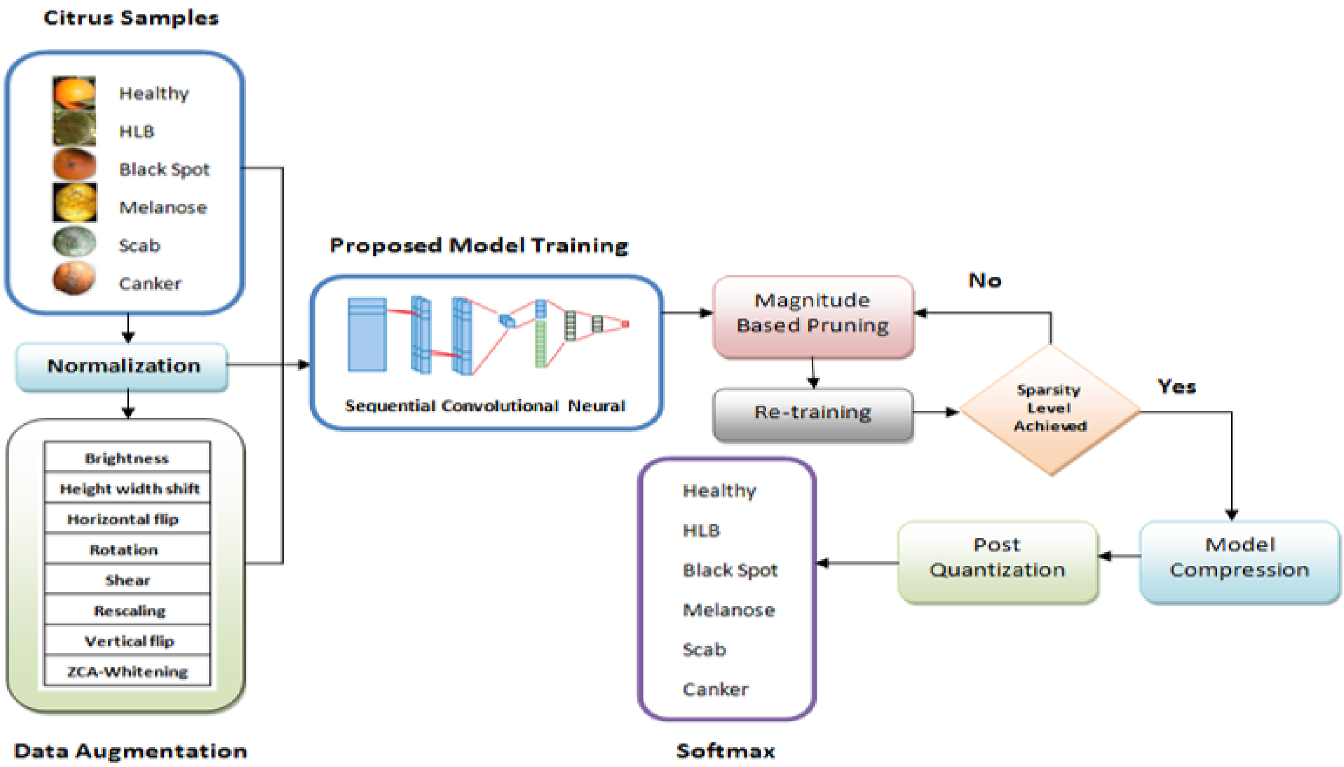


Figure 3

Multi-Classification and Size Reduction Process

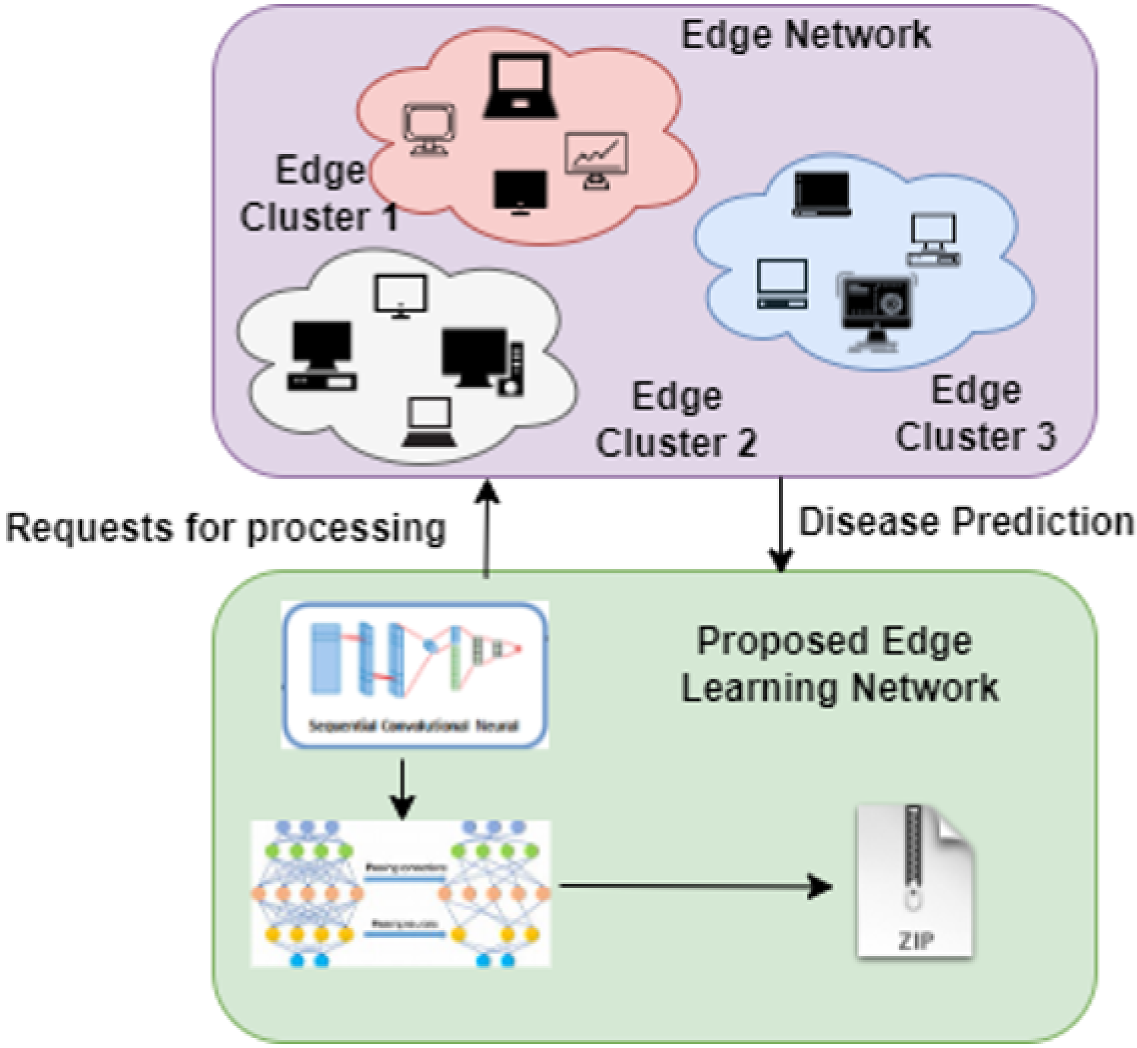
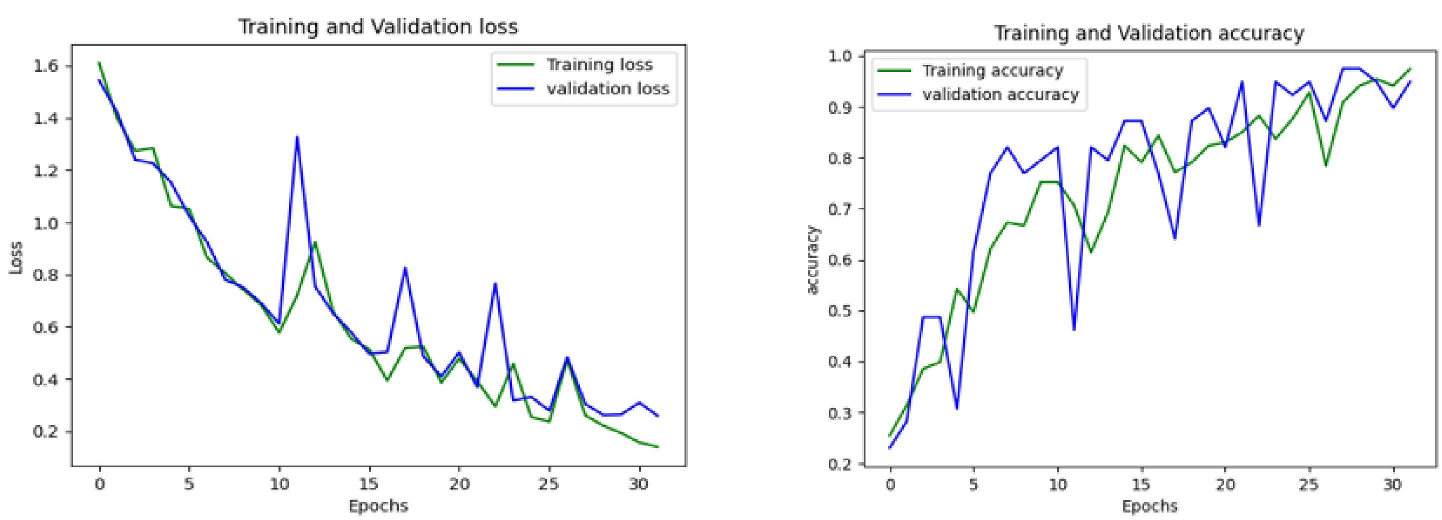


Figure 4

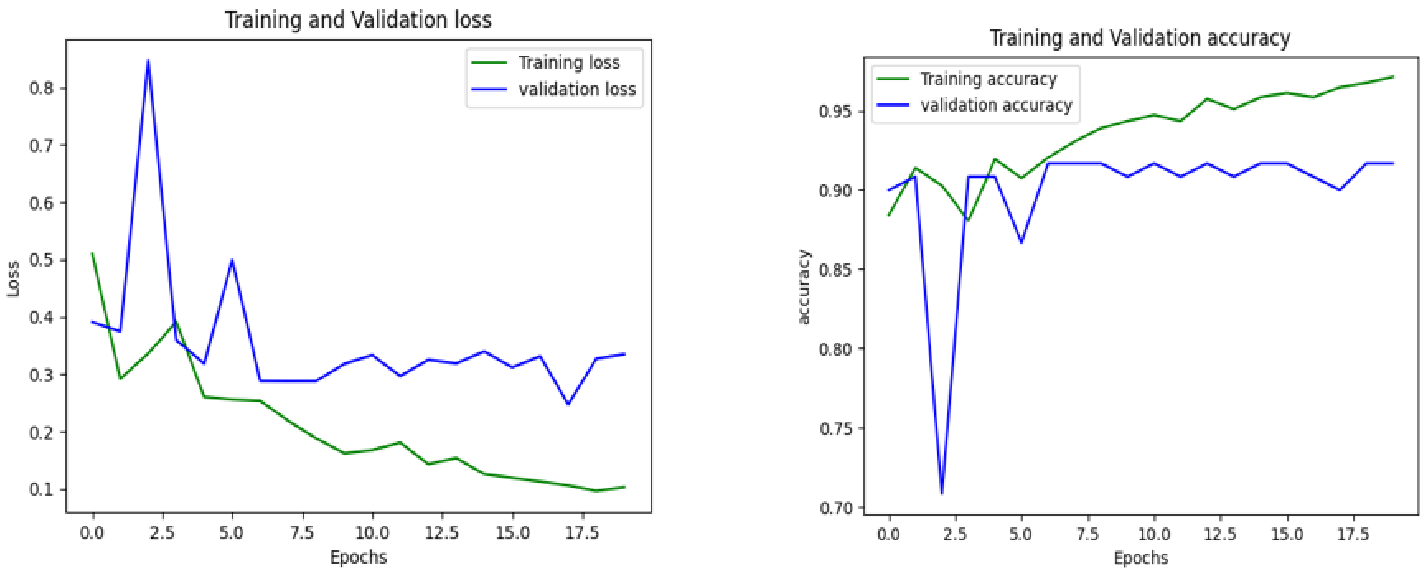
Edge Learning Framework



(a) 5 (b)

Figure 5

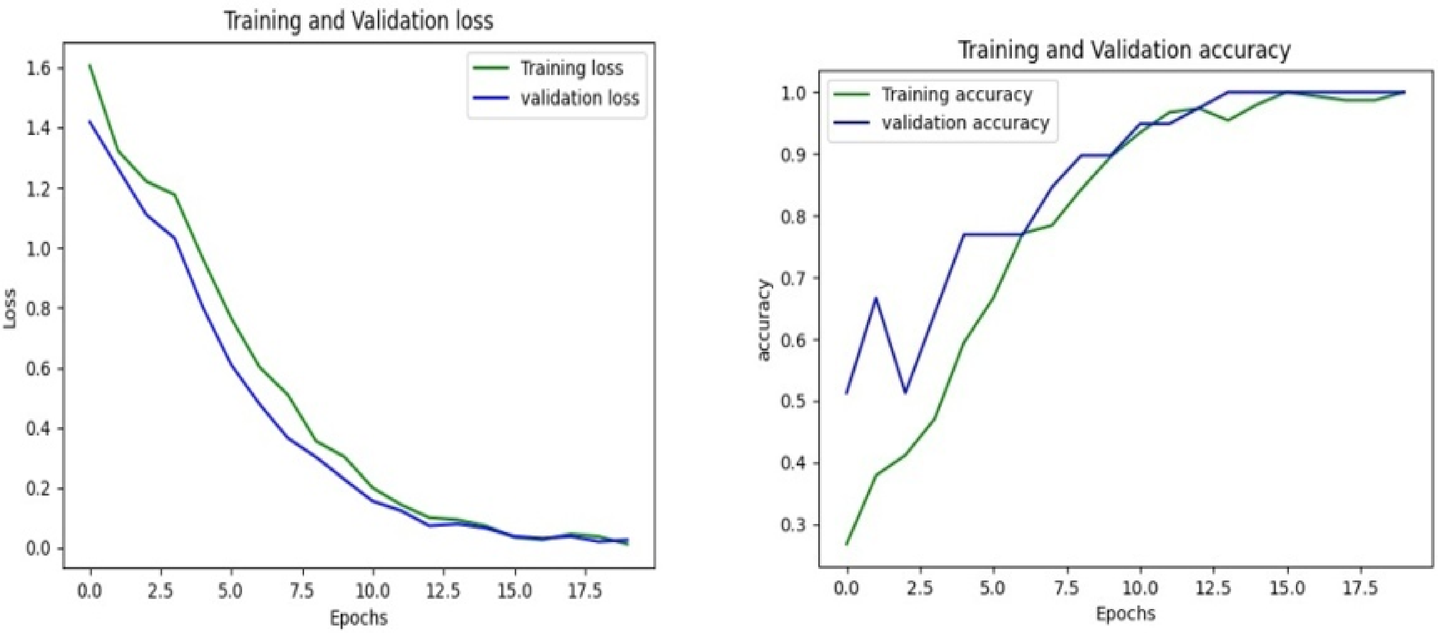
The accuracy graphs and the loss graphs are also plotted for above said, models. It can be observed in loss graph figure 3(a) that the training and validation losses percentage of around 0.18 losses is there in the proposed Faster-CNN model without pruning represented in figures 5(a) and 5(b) shows training & validation graph which is approaching 96.92 percent.



(a) (b)

Figure 6

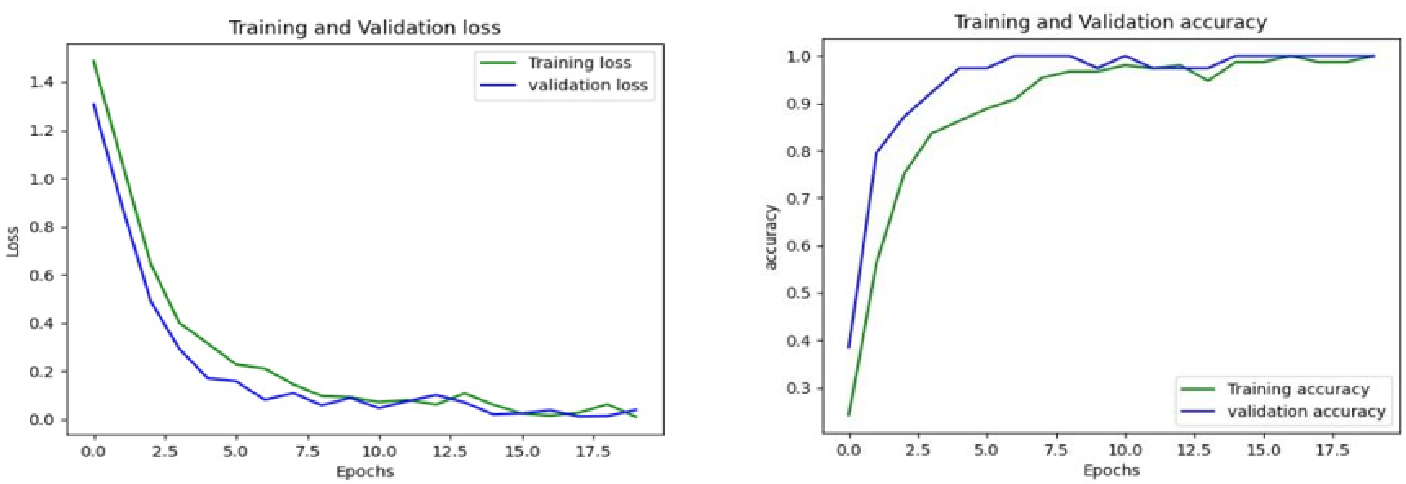
The loss graph in figure 6(a) represents that the training & validation losses is nearly to 0.13 and approaching to zero with model pruning of sparsity range of 50-80% and the training and validation graph which is approaching to 96.03 percent with starting level of sparsity begins with 50 percent and sparsity ends at 80 percent represented in figure 6(b).



(a) (b)

Figure 7

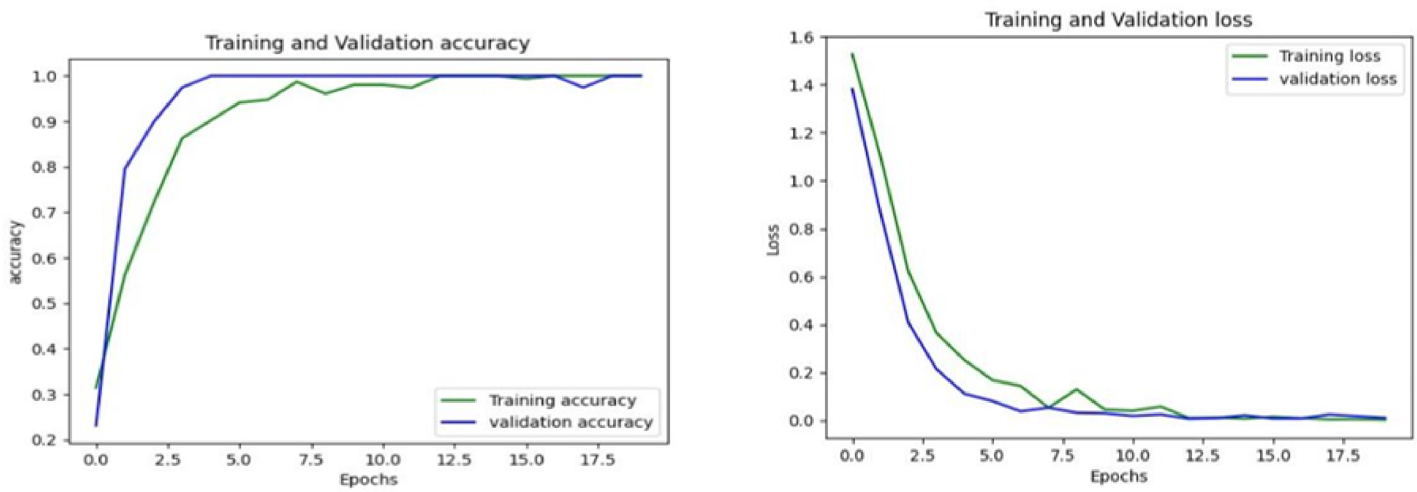
The loss graph figure VII (a) indicates that both the training and validation loss percentage are nearly to 0.20 with model pruning having sparsity range of 80-90% and the training and validation graph which is approaching to 96.03 percent with starting level of sparsity begins with 80 percent and sparsity ends at 90 percent represented in figure 7 (b).



(a) (b)

Figure 8

the loss graph indicates that the 15 percent of loss is encountered in both the training and validation loss with model pruning having sparsity range of 70-90% as mentioned in figure 8 (a) and the training and validation graph which is approaching to 92.35 percent with starting level of sparsity begins with 80 percent and sparsity ends at 90 percent represented in figure 8 (b).



(a) (b)

Figure 9

The loss graph indicates that the training and validation losses are deviated towards loss percentage of around 0.50 with model pruning having sparsity range of 80-90% as mentioned in figure 9 (a) and the training and validation graph which is approaching to 80.36 percent with starting level of sparsity begins with 80 percent and sparsity ends at 90 percent represented in figure 9 (b). It is observed that the training and validation accuracy is minimum and loss is maximum in this scenario.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- Table3.docx