FOREX Publication
Open Access | Rapid and quality publishing

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 10, Issue 2 | Pages 405-410 | e-ISSN: 2347-470X

Automatic Framework for Vegetable Classification using Transfer-Learning

Harendra Singh[1], Roop Singh[2], Parul Goel[3], Anil Singh[4] and Naveen Sharma[5]

[1,3]*Assistant Professor, G. L. Bajaj Institute of Technology and Management, Greater Noida, India*
[2,5]*Biomedical Applications, CSIR-Central Scientific Instrumentations Organization, Sector-30C, Chandigarh-160030, India*
[4]*Associate Professor, Arni University, Kathgarh Indora Himachal Pradesh, India*

***Correspondence:** Roop Singh, E-mail: roopsolanki@gmail.com

ABSTRACT- Globally, fresh vegetables are a crucial part of our lives and they provide most of the vitamins, minerals, and proteins, in short, every nutrition that a growing body need. They vary in colors like; red, green, and yellow but as our ancestors say that green vegetables are a must for every age. To identify the fresh vegetable that makes our body healthy and notion positive the proposed automatic multi-class vegetable classifier is used. In this paper, a framework based on a deep learning approach has been proposed for multi-class vegetable classification from scratch. The accuracy of the proposed model is further increased using the transfer-learning concept (DenseNet201). The whole process is divided into four modules; data collection and pre-processing, data splitting, CNN model training, and testing, and performance improvement using a pre-trained DenseNet201 network. Data augmentation and data shuffling are used to free from lack of data availability during the training phase of the model. The proposed framework is more efficient and can predict the type of vegetables comparatively in less computational time (2 to 3 minutes) with an 'Accuracy' of 98.58%, 'Sensitivity' of 98.23%, and 'Specificity' of 94.25%.

Keywords: Convolution Neural Network (CNN), Data Augmentation and shuffling, Transfer learning (DenseNet201).

ARTICLE INFORMATION

Author(s): Harendra Singh, Roop Singh, Parul Goel, Anil Singh and Naveen Sharma
Received: 28/04/2022; **Accepted**: 25/06/2022; **Published**: 30/06/2022;
e-ISSN: 2347-470X;
Paper Id: IJEER22EDI404;
Citation: 10.37391/IJEER.100257
Webpage-link:
https://ijeer.forexjournal.co.in/archive/volume-10/ijeer-100257.html

crossref member
CROSSREF.ORG
THE CITATION LINKING BACKBONE

Publisher's Note: FOREX Publication stays neutral with regard to Jurisdictional claims in Published maps and institutional affiliations.

1. INTRODUCTION

Vegetables are eaten all around the globe yet many are unaware of how different they all are. Nowadays, identifying, picking, and storing fresh vegetables become more difficult and also required more labour. The proposed vegetable classifier is an automatic model which helps human beings correctly classify fresh vegetables with less computational time. In this research, a convolution neural network for vegetable classification has been developed from scratch [1]. In this work, 2300 vegetable images of fifteen different types of vegetables are collected through a webcam using an open CV library. To improve the extracted feature's quality these collected images are pre-processed using different pre-processing operators and then resized to feed to the proposed convolution model. To improve the proposed model's accuracy, dropout layers are also used during the CNN model training phase. The accuracy of the proposed model is 98.58%. To improve the model performance, the concept of transfer-learning with the 'DenseNet201' model is used by freezing all layers except the dense layer. The pre-trained model is trained with a collected dataset with five different data augmented operators. The performance of the proposed approach is quite good as compared to conventional CNN.

2. LITERATURE REVIEW

Sakai et al [2] proposed a vegetable category recognition system using a deep neural network. The proposed model, trained with eight different types of vegetable images and can classify the unseen vegetable image with an accuracy of 99.14%.

Zeng [3] has proposed a convolution neural network-based classification and object region drawing model for vegetables and fruits with image saliency. The proposed model is trained with different types of vegetable and fruit images and the overall accuracy of the model is 95.6%.

Li et al [4] proposed a transfer learning-based, vegetable recognition, and classification model. This model can categorize as well as identify the different kinds of vegetables with an accuracy of 96.50% after adding four batch normalization and four fully connected layers in the VGG-16 after each convolution layer and in the last.

Kumar et. Al. [5] proposed an approach a deep learning model for the nutrient shortage of apple fruit. This model's ability to 'classify' and 'recognize' any form of deficit present in apple fruit is 94.24 percent accurate. With a camera, the author records data on roughly 1000 photos of apples in real time, 600 of which have four sorts of deficiencies—boron, calcium, iron, and manganese—and 400 of which are healthy.

Kumar et. al.[6] suggested a citrus fruit classification model according to its color and 'gray-level co-occurrence matrix' (GLCM) features using image processing. The proposed automatic model has been used to identify as well as classify a bulk amount of citrus fruit according to quality measures in minimum time and also provides secondary advice to the storekeeper regarding fruit storing time.

Khan and Debnath [7] proposed a fruit classification model using object detection and recognition technique. The develop proposed CNN concept of overlapping of multiple fruit images dataset is used and gives better accuracy of 98.75%.

Chithra and Henila [8] have developed a fruit classification model using otsu thresholding. The collected RGB images (apple and banana), were first converted into hues images to reduce the model complexity and then classified as banana or apple with the accuracy of 100% using a 'support vector machine' (SVM) classifier.

M. Khatun and N. A. Turzo [9] proposed a fruit classification model using a convolution neural network (CNN) approach. The 80% dataset (180-images) is used to train the model with some augmented operations. The model performance at 25 epochs and 32 as batch size is 98% in terms of model accuracy. Muresan and Oltean [10] developed a fruit recognition model using a deep learning approach. The proposed model is trained by 40,000 images of apple with different augmented operations after performing image normalization, therefore the computational time and model complexity is reduced. During the model evaluation, 100% were noted as training and 98.23% as validation accuracy.

3. CONVOLUTION NEURAL THEORY

A "Convolution neural network" (CNN) and an "Artificial neural network" are the two networks that make up the "Deep learning" model (DNN). The deep learning models are mostly used to classify the different types of images, object localization, and object segmentation. A labelled dataset means the training dataset has both features as well as a corresponding label to train any machine learning model. In the deep learning model, the CNN part of deep learning models extract the required features of an input image and is directly fed to the 'ANN' network for the classification of the images [11]. The CNN has basic four stages; convolution, relu, pooling, and flattening.

3.1 Convolution Layer

The feature extraction is done by the convolution layer and has three main parameters. These parameters are; 'filter size', 'stride', and 'padding'. The filter size represents the initial weights and these weights are further modified until the required features are not extracted. The movement of the filter over the input image during convolution operation is decided by stride and padding is used to maintain the constant size of an image during feature extraction [12]. The new size of the convoluted image after convolution operation is calculated by using given *equations (1), (2), and (3)* as:

$$y_h = \frac{x_h + 2*\text{Padding} - \text{filter size}}{\text{stride}} + 1 \quad (1)$$

$$y_w = \frac{x_w + 2*\text{Padding} - \text{filter size}}{\text{stride}} + 1 \quad (2)$$

$$y_c = k \quad (3)$$

Where:

x_h, x_w, x_c: Length, width, and depth (input image)

y_h, y_w, y_c: Length, width, and depth (convoluted image)

K: number of filters or initial weights

The basic convolution operation on an image with a filter is shown below in *Figure 1*.

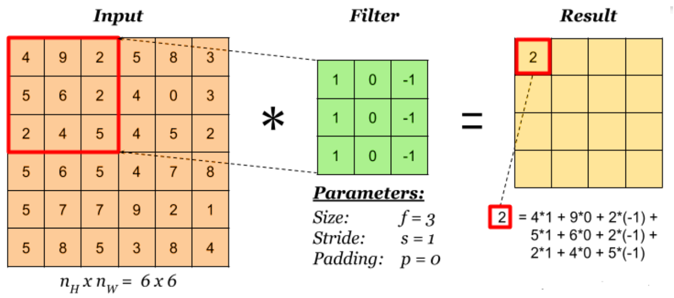

Figure 1: Convolution Operation

In *Figure 1*, the extracted features after performing the convolution in between an input image of size 6*6 and a filter of size 3*3 with valid padding and stridge =1 are shown.

3.2 Relu Layer

The 'rectified linear unit function' (Relu) is used to add the non-linearity in the input image, therefore most important features are extracted by CNN. The relu operated image becomes a binary image, i.e. all available negative pixels are converted to '0' therefore the image becomes darker. The relu activation function $p(x)$ is represented by Eq. (4)

$$p(x) = \begin{cases} 0 & if\ x < 0 \\ 1 & if\ x \geq 0 \end{cases} \quad (4)$$

3.3 Pool Layer

The 'pooling layer' is used for dimension reduction; therefore, computational time and model complexity is reduced without affecting the most important features. As compared to average pooling, max-pooling is used so that model complexity is reduced. The maximum value of the image pixel is considered rather than the average of all pixels, in average pooling.

The max-pooling and average pooling operations on an image are shown below in *Figure 2* and *Figure 3*.

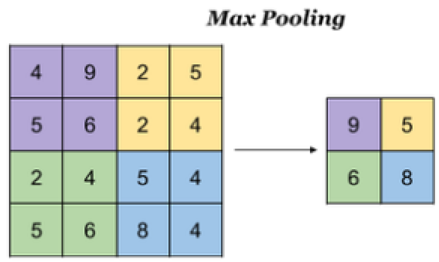

Figure 2: Max pooling Operation

FOREX Publication
Open Access | Rapid and quality publishing

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 10, Issue 2 | Pages 405-410 | e-ISSN: 2347-470X

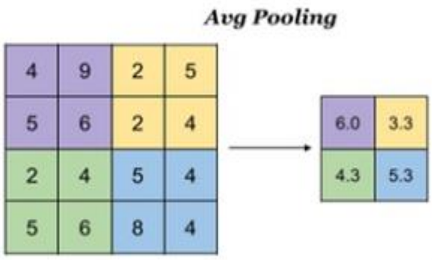

Figure 3: Average pooling Operation

In *Figure 2* and *Figure 3*, Maxpooling and average pooling operations on a binary image are shown. In max-pooling, pixels 9, 5, 6, and 8 of the input images are considered but the average of all pixels like; 6.0, 3.3, 4.3, and 5.3 are considered for an average type of pooling. The maximum or average value of pixels represents the most important feature which is used to classify the type of an image. The computational time and complexity of the CNN model with max-pooling are less as compared to average pooling.

4. BLOCK DIAGRAM OF MULTICLASS VEGETABLE IMAGE CLASSIFIER USING CNN

In this framework, a brand-new "convolution neural network" (CNN) is suggested for the fifteen types of vegetable image classification. The block diagram of the suggested framework of the CNN model is shown in *Figure 4*. The images of fifteen types of different vegetables are collected through a webcam using an open CV library. After performing some preprocessing steps the collected dataset is split into the ratio of 80:20 as train-test sets. Data scrambling and augmentation are employed to avoid the lack of availability of datasets during the training phase. The developed model is tested using the remaining 20% dataset and corresponding scores are measured that evaluate the performance of the multi-class vegetable image classifier.

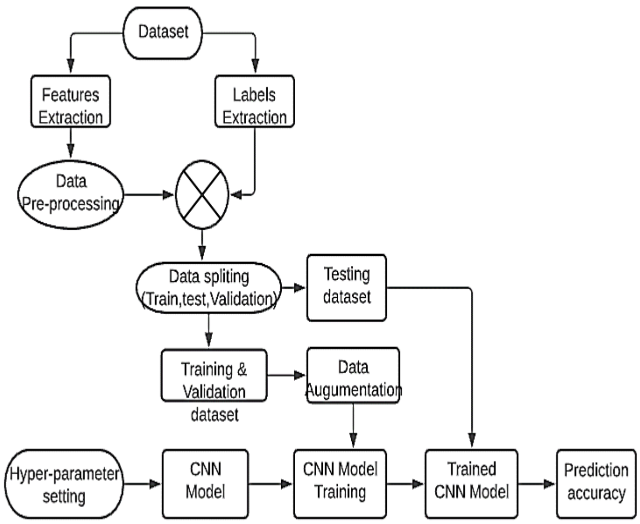

Figure 4: Block diagram of multi-class vegetable image classifier using CNN

The proposed multi-class vegetable image classifier consists of four stages; Dataset collection and pre-processing, Data splitting and argumentation, classifier parameter tunning and training, and proposed classifier model testing.

4.1 Dataset Collection and Pre-processing

2300 images of fifteen different vegetables (bean, bitter gourd, bottle gourd, brinjal, broccoli, cabbage, capsicum, carrot, cauliflower, cucumber, papaya, potato, pumpkin, radish, tomato) are collected through webcam using an open cv library as shown in *Figure 5*. To extract the most efficient features from the collected dataset some preprocessing like; RGB to a grayscale image, quality enhancement, resizing, and histogram equalization are performed. After RGB to grayscale conversion, the vegetable image dataset is normalized to the same size of 150*150*3, therefore the classifier's computational time, as well as classifier complexity of both, are optimally normalized.

Figure 5: Sample of a collected dataset through webcam

4.2 Data Shuffling and Augmentation

The performance of the vegetable image classifier depends on the amount of the collected dataset. Therefore, to create the hues amount of dataset for model training and also to avoid the data shortage, data shuffling and data augmentation are used on the training dataset i.e. 80% of 2300. The five data augmentation operators; image rotation, image zooming, image sharing, and flipping of images in horizontal and vertical directions are applied to the training dataset. The total number of images for model training after data augmentation is 9,200 and the remaining 460 images are used for model evaluation.

4.3 Architecture of Multi-class Vegetable Image Classifier

In this, a multi-class vegetable image classifier using CNN from scratch has been developed. The fifteen types of vegetable images are resized in 64*64*3 after pre-processing of the dataset. The five types of data augmentation operation and data shuffling are processed before the training of the image

FOREX Publication
Open Access | Rapid and quality publishing

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 10, Issue 2 | Pages 405-410 | e-ISSN: 2347-470X

classifier [13]. The vegetable image classifier has three convolution layers with relu as activation function, 62,64, and 128 filters of size; 3*3,3*3,3*3, strides =1, and valid padding respectively. The kernel sizes are the initial weights that are modified using back propagation algorithms until the desired outputs are achieved. After each convolution layer, max-pooling layers of size (2*2) are used so that the most important feature defined as the maximum pixel value is selected out of four.

Three dropout layers in the multi-class vegetable classifier are also used; 20%, 30%, and 40 % respectively to protect the model from overfitting. A flattening layer (n-dimensional array into 1-dimensional) total no. of input parameter or neuron is 685,138 and all are trainable. In the last, two fully connected layers with nodes 128 and 15 are used [14]. A detailed summary of the proposed multi-class vegetable image classifier model is shown below in *Figure 6*.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 62, 62, 32)        896

 max_pooling2d (MaxPooling2D  (None, 31, 31, 32)       0
 )

 dropout (Dropout)           (None, 31, 31, 32)        0

 conv2d_1 (Conv2D)           (None, 29, 29, 64)        18496

 max_pooling2d_1 (MaxPooling  (None, 14, 14, 64)       0
 2D)

 dropout_1 (Dropout)         (None, 14, 14, 64)        0

 conv2d_2 (Conv2D)           (None, 12, 12, 128)       73856

 max_pooling2d_2 (MaxPooling  (None, 6, 6, 128)        0
 2D)

 dropout_2 (Dropout)         (None, 6, 6, 128)         0

 flatten (Flatten)           (None, 4608)              0

 dense (Dense)               (None, 128)               589952

 dense_1 (Dense)             (None, 15)                1935

=================================================================
Total params: 685,135
Trainable params: 685,135
Non-trainable params: 0
```

Figure 6: Summary of proposed CNN model

In *Figure 6*, a detailed summary of the multi-class vegetable image classifier is shown. The total number of parameters to train the classifier is 685,138. These are the most important features which are extracted by the convolution neural network and fed to ANN for vegetable image classification.

5. EXPERIMENTAL RESULTS

The proposed multi-class vegetable image classifier using CNN and DenseNet201 for the classification of different vegetables is implemented by Google Colab with python 3.8. The model is tested using a system that has 8 GB RAM, a 7th generation i5-2.5 GHz processor, and 4 GB of graphics. The accuracy and losses plots for training as well as validation of the proposed multi-class vegetable image classifier are shown in *Figure 7(a)* and *Figure 7(b).*

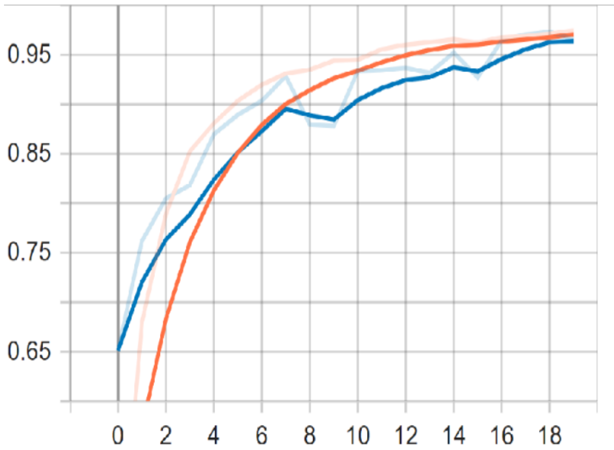

Figure 7 (a): Proposed CNN model's training and validation accuracy

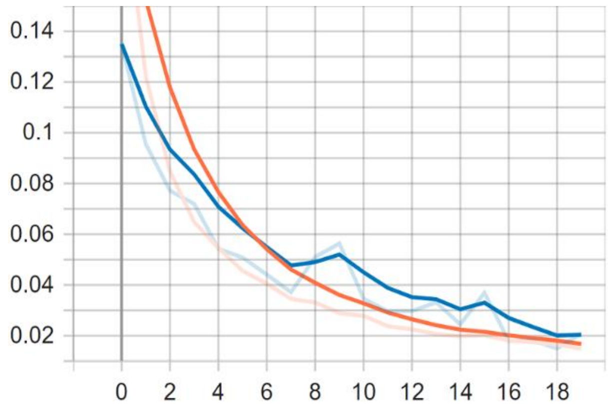

Figure 7 (b): Proposed CNN model's training and validation losses

Figures 7(a) and 7(b) display that the proposed model's training and validation accuracy are 99.80 % and 89.22 % respectively. The model is trained at 16, 20, and 287 as batch size, total no. of epochs, and steps per epoch respectively.

5.1 Confusion Matrix

To evaluate the performance of a multi-class vegetable image classifier, a confusion matrix is used [15], [16]. The proposed model is tested on 460 images of 15 classes. Some performance parameters like; precision, accuracy, sensitivity, specificity, and f-score are determined using the following equations: (5), (6), (7), and (8).

$$\text{Precission} = \frac{\text{TP}}{(\text{TP+FP})} \tag{5}$$

$$\text{Sensitivity} = \frac{\text{TP}}{(\text{TP+FN})} \tag{6}$$

$$\text{Specificity} = \frac{\text{TN}}{(\text{TN+FN})} \tag{7}$$

$$\text{Accuracy} = \frac{\text{TP+TN}}{(\text{Positive+Negative})} \tag{8}$$

$$\text{F} - \text{score} = \frac{(2*\text{TP})}{(2*\text{TP+FP+FN})} \tag{9}$$

The evaluated performance parameters for a multi-class vegetable image classifier are shown below in *Table 1*.

FOREX Publication
Open Access | Rapid and quality publishing

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 10, Issue 2 | Pages 405-410 | e-ISSN: 2347-470X

Table 1: Lists the performance parameter's accuracy metrics

Vegitable Name	TP	TN	FP	FN	Precision	Sensitivity	Specificity	accuracy	F-score
Bean(30)	28	29	1	2	0.93	0.93	0.94	0.95	0.94
Bitter gourd(30)	27	28	2	3	0.90	0.90	0.90	0.91	0.92
Bottle gourd(30)	29	30	0	1	100	0.97	0.97	0.98	0.98
Brinjal(30)	28	29	1	2	0.97	0.93	0.94	0.95	0.94
Broccali(30)	26	24	6	4	0.81	0.87	0.87	0.83	0.84
Cabbage(35)	32	29	6	3	0.84	0.91	0.93	0.93	0.87
Capsicum(35)	29	30	5	6	0.85	0.82	0.83	0.90	0.84
Carrot(30)	25	27	3	5	083	0.83	0.84	0.87	0.89
Califlower(30)	29	28	2	1	0.94	0.97	0.96	0.95	0.95
Cucumber(30)	28	26	4	2	0.88	0.93	0.93	0.90	0.90
Pappaya(30)	29	30	0	1	100	0.97	0.97	0.98	0.98
Potato(30)	28	28	2	2	0.93	0.93	0.93	0.93	0.93
Pumpkin(30)	30	27	3	0	0.90	100	100	0.95	0.95
Radish(30)	25	26	4	5	0.86	0.83	0.84	0.85	0.85
Tomato(30)	27	27	3	3	0.90	0.90	0.90	0.90	0.90

The overall accuracy of the proposed multi-class vegetable image classifier model with a softmax function for fifteen types of vegetable image classification is 98.58 %.

5.2 Area under the ROC Curve (AUC) and Receiver Operating Characteristic (ROC)

The ROC curve measures the trained model performance according to "True positive rate" (TPR) and "False-positive rate" (FPR). To measure the probability-based performance of the model area under the receiver operating characteristic curve is calculated [17]. For a 100% correct prediction type of model area under the curve is unity. The graph between TPR and FPR is shown below in *Figure 8*.

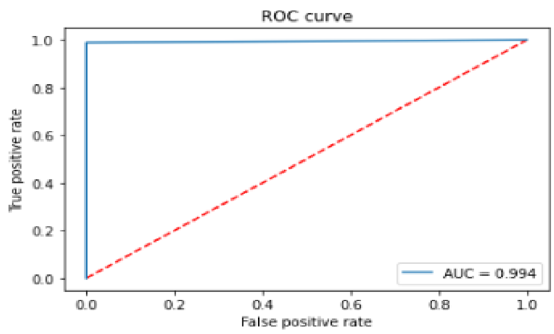

Figure 8: ROC and AUC curve of the proposed model

Figure 8, shows a ROC curve of the proposed multi-class vegetable image classifier. In a good classifier, the area under the curve is unity but the proposed model gives the prediction of about 99.4 for a tested vegetable image.

The predicted results of multi-stage vegetable classifiers with their labels

The proposed trained model is saved as the name 'harendra.h5' and tested with two different vegetable images as shown in *Figure 9 (a). Figure 9 (b)* illustrates the model's expected image with a label after these vegetable images are supplied to it

Figure 9 (a): A vegetable image of broccoli and tomato for model testing

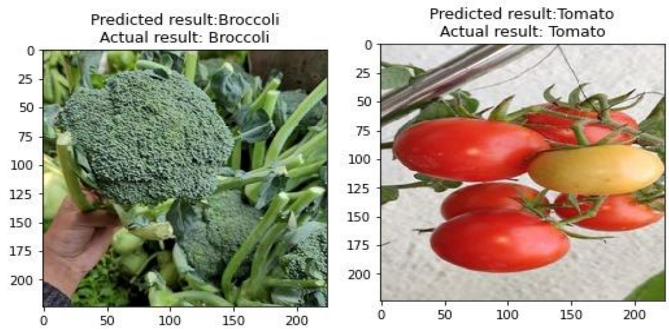

Figure 9 (b): Model predicted vegetable images with labels

Figure 9 (b), shows a model that predicted vegetable images with its level [18]. The proposed transfer learning model correctly predicts the type of vegetables according to the feed input vegetable image with an 'accuracy' of 98.58%, 'sensitivity' of 98.23%, and 94.25% of 'specificity'.

6. CONCLUSION AND FUTURE WORK

In this paper, an optimal number of layers deep learning model has been developed and which can be used to classify the fifteen types of different vegetable images. In the first stage, after preprocessing the collected dataset is split into train and test sets using a train-test split in the ratio of 80:20. The model has been trained on around 9,200 vegetable photos using five data augmentation operators to avoid the lack of data availability then tested with 460 vegetable images. The accuracy of the multi-class vegetable image classifier is quite good approx. 98.58 % with a softmax activation function and transfer learning model (DenseNet201). The accuracy of the proposed model is further increased using an advanced learning approach.

Table 2, shows a comparative study of the proposed multi-class vegetable image classifier pre-existing approach. The performance of the proposed model in comparison to other available ones, the automatic vegetable classifier is quite good and more effective.

Table 2: **Report comparing the suggested CNN model to earlier similar studies**

Sr. No.	Model	Model Accuracy	Classification Type	Classification Method
1	Sakai et al [2]	99%	Binary	Convolution neural network
2	G. Zeng [3]	95.6%	Multi-class	Convolution neural network
3	Y. Kumar et al. [5]	96.24%	Multi-class	Convolution neural network
4	M. Khatun and N. A. Turzo [7]	98%	For three class	Convolution neural network
7	Li et al [4]	98 %	Binary	VGG-16
8	**Proposed CNN Model**	**98.58%**	**Multi-class (15- class)**	**CNN+DenseNet201**

REFERENCES

[1] K. Hameed, D. Chai, and A. Rassau, "A comprehensive review of fruit and vegetable classification techniques," Image Vis. Comput., vol. 80, no. December, pp. 24–44, 2018, DOI: 10.1016/j.imavis.2018.09.016.

[2] Y. Sakai, T. Oda, M. Ikeda, and L. Barolli, "A Vegetable Category Recognition System Using Deep Neural Network," 2016 10th Int. Conf. on Inno. Mobile and Internet Services in Ubiquitous Comput, pp. 189-192, 2016.

[3] G. Zeng, "Fruit and Vegetables Classification System Using Image Saliency and Convolutional Neural Network," 2017 IEEE 3rd Inf.. Technol. and Mechatro. Eng. Conf. (ITOEC), pp. 613–617, 2017.

[4] Z. Li, F. Li, L. Zhu, and J. Yue, "Vegetable recognition and classification based on improved VGG deep learning network model," Int. J. Comput. Intell. Syst., vol. 13, no. 1, pp. 559–564, 2020, doi: 10.2991/ijcis.d.200425.001.

[5] Y. Kumar, A. K. Dubey, R. R. Arora, and A. Rocha, "Multiclass classification of nutrients deficiency of apple using deep neural network," Neural Comput. Appl., vol. 123456789, 2020, DOI: 10.1007/s00521-020-05310-x.

[6] C. Kumar, S. Chauhan, R. N. Alla, and H. Monica Gurram, "Classifications of citrus fruit using image processing-GLCM parameters," 2015 Int. Conf. Commun. Signal Process. ICCSP 2015, pp. 1743–1747, 2015, DOI: 10.1109/ICCSP.2015.7322820.

[7] R. Khan and R. Debnath, "Multi-Class Fruit Classification Using Efficient Object Detection and Recognition Techniques," Int. J. Image, Graph. Signal Process. vol. 11, no. 8, pp. 1–18, 2019, DOI: 10.5815/ijigsp.2019.08.01.

[8] P. Chithra and M. Henila, "Fruits Classification Using Image Processing Techniques," Int. J. Comput. Sci. Eng., vol. 6, no. 10, pp. 628–632, 2019, DOI: 10.26438/ijcse/v7si5.131135.

[9] M. Khatun and N. A. Turzo, "Fruits Classification using Convolutional Neural Network," GRD J. for Eng. , vol. 4, no. 8, pp. 118–123, 2020, DOI: 10.26656/fr.2017.4(S6).029.

[10] H. Muresan and M. Oltean, "Fruit recognition from images using deep learning," Acta Univ. Sapientiae, Inform., vol. 10, no. 1, pp. 26–42, 2018, doi: 10.2478/ausi-2018-0002.

[11] Y. D. Zhang, Z. Dong, X. Chen, W. Jia, S. Du, K. Muhammad S. H. Wang, "Image-based fruit category classification by 13-layer deep convolutional neural network and data augmentation," Multimedia Tools and Appl., vol. 78, no. 1, pp. 3613-3632, 2019, Springer, Singapore. https://doi.org/10.1007/s11042-017-5243-3.

[12] R. S. Latha, G.R. Sreekanth, R.C. Suganthe, M. Geetha, N. Swathi, S. Vaishnavi, and P. Sonasri, "Automatic Fruit Detection System using Multilayer Deep Convolution Neural Network,"2021 Int. Conf. on Comput. Commun. and Inform. (ICCCI), 2021, pp. 1-5, DOI: 10.1109/ICCCI50826.2021.9402513.

[13] H. Singh and R.S. Solanki, "Classification & Feature Extraction of Brain Tumor from MRI Images using Modified ANN Approach," Int. J. of Elect. and Electronics Res. (IJEER), vol. 9, no.2, pp.10-15, June 2021. DOI: 10.37391/ijeer.090302.

[14] B. Biswas, S.K. Ghosh, and A. Ghosh, "A Robust Multi-label Fruit Classification Based on Deep Convolution Neural Network," Comput. Intell. In Pattern Recogn. Adv. in Intell. Systems and Comput., vol 999. Springer, Singapore. https://doi.org/10.1007/978-981-13-9042-5_10.

[15] N. A. Muhammad, A. A. Nasir, Z. Ibrahim, and N. Sabri, "Evaluation of CNN, Alexnet and GoogleNet for Fruit Recognition," Indonesian J. of Elect. Eng. and Comput. Sci., vol. 12, no. 2, pp. 1-5, doi:10.11591/IJEECS.V12.I2.PP468-475.

[16] M.S. Mahajan, "Optimization and classification of fruit using machine learning algorithm," Int. J. for Inno. Res. in Sci. & Technol., vol. 3, no. 1, pp. 377-381, 2016.

[17] S. Arivazhagan, S.R. Newlin, S.N. Selva, and L. Ganesan, "Fruit Recognition using Color and Texture Features," J. of Emer. Trends in Comput. and Inform. Sci., vol.1, no. 2, pp. 90-94, 2010.

[18] N. Saranya, K. Srinivasan, P. Kumar, S. K., Rukkumani, and R. Ramya, "Fruit Classification Using Traditional Machine Learning and Deep Learning Approach," Computational Vision and Bio-Inspired Comput., vol.11, no. 08, pp. 79-89,2020.

[19] M. I. Ahmed, S. M. Mamun, and A. U. Z. Asif, "DCNN-Based Vegetable Image Classification Using Transfer Learning: A Comparative Study," 2021 5th Int. Conf. on Comput, Commun. and Signal Proc. (ICCCSP), 2021, pp. 235-243, DOI: 10.1109/ICCCSP52374.2021.9465499.