Tech Science Press

Fruits and Vegetables Freshness Categorization Using Deep Learning

Labiba Gillani Fahad[1], Syed Fahad Tahir[2,*], Usama Rasheed[1], Hafsa Saqib[1], Mehdi Hassan[2] and Hani Alquhayz[3]

[1]Department of Computer Science, National University of Computer and Emerging Sciences, Islamabad, 44000, Pakistan
[2]Department of Computer Science, Air University, Islamabad, 44000, Pakistan
[3]Department of Computer Science and Information, College of Science in Zulfi, Majmaah University, Al-Majmaah, 11952, Kingdom of Saudi Arabia
*Corresponding Author: Syed Fahad Tahir. Email: fahad.tahir@mail.au.edu.pk

Abstract: The nutritional value of perishable food items, such as fruits and vegetables, depends on their freshness levels. The existing approaches solve a binary class problem by classifying a known fruit\vegetable class into fresh or rotten only. We propose an automated fruits and vegetables categorization approach that first recognizes the class of object in an image and then categorizes that fruit or vegetable into one of the three categories: pure-fresh, medium-fresh, and rotten. We gathered a dataset comprising of 60K images of 11 fruits and vegetables, each is further divided into three categories of freshness, using hand-held cameras. The recognition and categorization of fruits and vegetables are performed through two deep learning models: Visual Geometry Group (VGG-16) and You Only Look Once (YOLO), and their results are compared. VGG-16 classifies fruits and vegetables and categorizes their freshness, while YOLO also localizes them within the image. Furthermore, we have developed an android based application that takes the image of the fruit or vegetable as input and returns its class label and its freshness degree. A comprehensive experimental evaluation of proposed approach demonstrates that the proposed approach can achieve a high accuracy and F1score on gathered *FruitVeg Freshness* dataset. The dataset is publicly available for further evaluation by the research community.

Keywords: Fruits and vegetables classification; degree of freshness; deep learning; object detection model; VGG-16; YOLO-v5

1 Introduction

Appearance, such as color, texture, size, and shape, can be used to find the category of freshness of fruits and vegetables [1,2]. The freshness can be categorized into pure-fresh, medium-fresh, and rotten. An automated system to identify these categories can be useful for individual consumers as well as for fresh items trading industry [3,4]. However, high intra-class variations make it a challenging problem since the fruits in the same freshness category can have different appearances [5]. Approaches

exist that focus on the categorization of a single fruit or vegetable into pure-fresh or rotten only [6], while to the best of our knowledge, none of the approaches identifies the medium fresh category. Identification of medium-fresh category is useful in reducing the wastage of fruits and vegetables. However, differentiating the medium-fresh from the pure-fresh and rotten categories is the most challenging part because of the fine-grained differences between them. Moreover, by assuming that the class of the fruit or vegetable is already known, the existing approaches solve a binary problem of fresh or rotten, while the similarity among different fruit and vegetables does not remain a challenge [2,3,5,6]. However, this interclass variation becomes significant if the class of fruit or vegetable is unknown. For example: a green banana and a cucumber can have similar shape, while a green lemon and a green orange can have same appearance [7].

Unlike the existing approaches solving a binary class problem of fresh and rotten category of a known fruit\vegetable, the proposed approach performs two tasks: first it detects and assigns a label to the fruit or vegetable in an image; second, it categorizes the freshness of that fruit or vegetable as pure-fresh, medium-fresh, or rotten. In this regard, our contributions are three folds:

- We propose FruitVeg Freshness dataset of 60K images of 11 fruits and vegetables, each with three categories of freshness.
- For recognition of a fruit or vegetable and to categories its three freshness levels, we formulate a multi-class classification problem.
- We develop an android application for a real time demonstration of our system.

Our self-collected FruitVeg Freshness dataset contains 60K images of eleven fruits and vegetables, each of which is further categorized based on freshness. In the proposed approach, we first detect and localize the object such as fruit or vegetable in the image. Next, the category of freshness from the three is identified. Thus, we solve a multi-class problem with 33 classes by exploiting two Deep learning models VGG-16 and YOLO-v5 and compare their results. VGG-16 classifies fruits and vegetables and categorizes their freshness, while YOLO, in addition to the classification and categorization, also localizes the objects within the images. We achieve the accuracy of 82% and 84% on the proposed dataset, respectively.

The rest of the paper is organized as follows: In Section 2, related work on fruits and vegetables classification is discussed. In Section 3, we present the proposed FruitVeg Freshness dataset. Next, In Section 4, the proposed fruits and vegetables freshness approach identification approach is discussed. The experimental analysis is presented in Section 5. Finally, Section 6 draws conclusions.

2 Related Work

Fruit texture, color, and shape are three key visual features for assessing fruit quality [2]. Different machine learning techniques including Support Vector Machines (SVM), Multi-Layer Perceptron (MLP) and K Nearest Neighbor (KNN) classifiers are used to identify the healthy apples from the defective ones. Among the three classification techniques, SVM shows better performance. In visual object recognition, deep learning is widely employed [8,9]. YOLO is faster compared to the previous methods, achieving a real-time image processing speed of 20 frames per second [10]. A classification approach using image saliency and Convolutional Neural Network (CNN) differentiate fruits and vegetables [11]. The deep neural network, VGG, is used for recognition, demonstrating that CNN can reach a high accuracy as they go deep [12–14]. In contrast to [11], a shallow network is used for feature extraction [15], with four convolutional and pooling layers followed by two fully connected layers. However, the background of the input image is simple, where all the fruit items are placed on

a white background. Another quality grading approach exploits the visual characteristics to build a computer vision-based system for the classification of tomato as either defective or healthy [5]. An autonomous approach to rank the olive fruit batches uses discrete wavelet transform and textural data [16]. Another approach exploits deep learning for recognition of raspberry using a 9-layered CNN with 3 convolutional and pooling layers, one input and one output layer [3].

In a hybrid deep learning approach for fruit classification, first, hand crafted features are extracted. In the next step, it uses convolution autoencoder to pre-train the images and then applies an attention based DenseNet to process the images [17]. A tomato disease detection and classification approach uses transfer learning [6,18]. Multiple deep learning models are used to classify nine types of diseases in leaves including healthy tomato leaves. Among all, DenseNet Xception shows better performance. CNN is used to learn the inter-class variations between different types of fruits [19]. Deep learning techniques are used to identify leaf stress issues on 33 types of crops related to fruits, vegetables, and other plants [20]. Different machine and deep learning techniques including SVM, KNN, Naïve Bayes (NB) and CNN have been used in the recognition of insect pests for wheat, rice and soybean crops using publicly available datasets [21]. Among all the techniques, CNN shows the best performance. Computer vision and Machine learning based systems have been previously applied to the field of vegetables and fruits classification [22], using texture and color features to identify and describe the damaged areas on fruits [23]. Moreover, color co-occurrence has been used to classify diseased and normal leaves in citrus fruit [24,25]. In [26], local damage to strawberries is detected using local key point detectors, such as Scale Invariant Feature Transform (SIFT) and Oriented Fast and Rotated BRIEF (ORB). Moreover, CNN is also used to identify the freshness of fresh-cut iceberg lettuce [27].

An automated approach based on computer vision and deep learning is presented in [28] to detect the different diseases of Guava Fruit. Local Binary Patterns (LBP) are used for feature extraction and Principal Component Analysis (PCA) is used for dimensionality reduction. Different machine learning approaches including SVM, KNN, bagged tree and RUSBoosted tree are used for classification. Among these, SVM shows better performance [28]. An automated plant leaf disease detection approach is discussed in [29], where features are extracted using LBP, segmentation is performed to identify the Region of Interest (ROI), while one class SVM is used for the classification of leaf diseases. In a citrus fruit disease detection approach, ROI is identified using Delta E segmentation, color histogram and textural features are extracted and then SVM is used for classification [7]. The Performance evaluation of five CNN based models including NasNet, Mobile, Xception, DenseNet, Inception ResNetv2 and Inception v3 with transfer learning and data augmentation techniques is discussed in [30] for classification of tomatoes.

An approach to identify the fresh and rotten fruits based on ensemble learning is discussed in [31], that combines two deep convolutional neural networks: Resnet-50 and Resnet-101. The approach first differentiates between the fresh and rotten fruits and then identifies the type of fruit. An automated fruit classification approach presented in [32] exploits deep learning and stacking ensemble methods to identify the fresh and rotten categories of banana and apples. Moreover, it is emphasized that a cost-effective solution based on Respberry Pi module, camera and touchscreen can also be used for the real time monitoring of fruits. Deep features extracted using CNN can be effective in detecting and analyzing the freshness of different varieties of food items [33]. A fruit classification approach is presented in [34], where images obtained from the CMOS sensors and Kaggle fruit360 dataset [30] are used to train a CNN model to identify six categories of fruits including fresh and rotten apples, bananas, and oranges. Moreover, augmentations techniques of scaling, rotation, translation, Gaussian noise addition, brightness variation have also been applied to the obtained images [34].

Hyperspectral imagery and machine learning techniques based on Artificial Neural Network (ANN) and Decision Tree (DT) have been applied to detect the penicillium fungi in citrus fruits [35]. A performance comparison of three deep learning models including Alexnet, Googlenet and Inception V3 is performed for the identification of water stress conditions of three crops maize, okra and soyabean [36]. Among the three models, Googlenet shows the best performance compared to the rest of the models. A fruit ripeness estimation technique first resizes the images to the same size and then perform segmentation to extract the image from the background [37]. In the next step, color features are extracted, and an adaptive neuro-fuzzy inference system is used for identification of different stages of fruit ripeness.

Finally, we compare the existing datasets based on the number of classes, number of images, size of images, and freshness categories per class (Tab. 1). Most of the datasets are fruit-and-vegetable classification datasets only. A few exist for freshness categorization; However, these datasets focus on binary categorization of fresh and rotten only. The medium-freshness category is ignored by all the datasets, Moreover, these datasets have images of only a single fruit or vegetable. It is important to collect a dataset that includes multiple fruits and vegetables and the three categories of pure-fresh, medium-fresh, and rotten for ranking of fruits and vegetables to be used for trade industry and for timely consumption of the fresh items.

Table 1: Existing datasets used in fruit classification, freshness categorization or disease detection

Ref.	Dataset names	Classes	No. of images	Size of image	Categories		
					Pure-fresh	Medium-fresh	Rotten
[7]	Citrus	4	199	256 × 256	✓		
[17]	Fruit26	26	124,212	100 × 100	✓		
[17]	Fruit15	15	44,406	320 × 258	✓		
[30]	Fruit360	1	90,483	100 × 100	✓		
[31]	Papaya	3	300	227 × 227	✓		
[33]	Cherry	3	400	-	✓		
[38]	ImageNet	6	4108	1296 × 964	✓		
[2]	Golden apple	2	120	360 × 360	✓		✓
[28]	Healthy and diseased guava	5	400	520 × 530	✓		✓
[35]	ImageNet & web crawled images	7	19083	224 × 224	✓		✓
[36]	Internal feeding worm	2	8791	120 × 120	✓		✓

(Continued)

Table 1: Continued

Ref.	Dataset names	Classes	No. of images	Size of image	Categories		
					Pure-fresh	Medium-fresh	Rotten
[36]	Egyptian banana species	2	300	120 × 120	√		√
[39]	Fruit images using CMOS sensors combined with Fruit360	6	30846		√		√
Own	FruitVeg freshness	11	60059	1088 × 1088	√	√	√

3 FruitVeg Freshness Dataset

Fig. 1 shows the sample images from the dataset. The similarity among the three categories of freshness is high within the same fruit/vegetable. In addition, different fruits and vegetables may appear similar because of the shape/color, while the same class may appear different in different instances, which makes it a challenging dataset. The FruitVeg Freshness dataset is publicly available for further use by the research community.

Tab. 2 shows the details of the proposed FruitVeg Freshness dataset. The proposed dataset contains a total of 60059 images divided into 11 fruits and vegetables: six fruits (Apple, Banana, Guava, Orange, Lemon and Tomato) and five vegetables (Brinjal, Cucumber, Chili, Pepper and Potato). Each fruit\vegetable is further categorized into three categories based on the freshness. Digital images of size 1088 × 1088 are captured using hand-held mobile cameras and are also gathered using the various sources from internet, such as some samples of pure-fresh category are obtained from Fruit365 [30]. RGB colored images are collected in both dark and bright lighting conditions with variable background complexities such as from plain white background to complex backgrounds with varying color patterns. Size of each image is 1088 × 1088. Images contain both single and multiple objects such as 30% of the dataset consists of single objects, while remaining has multiple objects ranging from 2 to 20. We annotate the whole dataset manually using the MakeSense tool online.

4 Proposed Methodology

Let N × 3 be the number of classes of fruit and vegetable, where N is the number of fruits\vegetables, and each fruit\vegetable has three categories based on the freshness level. Fig. 2 shows the block diagram of the proposed freshness recognition approach. The preprocessing step involves preparing the dataset for the training. It represents object cropping, resizing and annotation using the annotation tool 'makesense.ai'. We exploit and compare two classification models for

recognition and categorization of fruits and vegetables: VGG-16 and YOLO-v5. VGG-16 is a classification model, while YOLO-v5 performs localization and classification in an image.
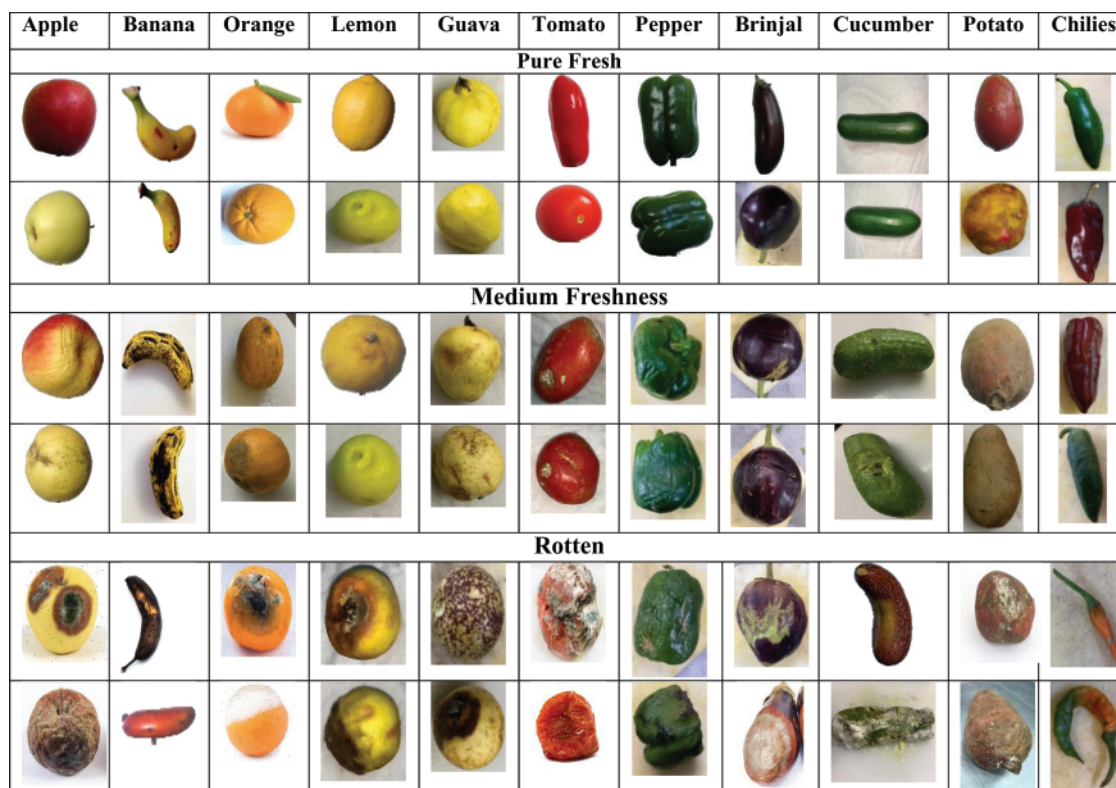

Figure 1: Sample images from fruitveg freshness dataset separated in three categories

Table 2: Dataset containing 60059 images of 11 fruits and vegetables in 3 categories

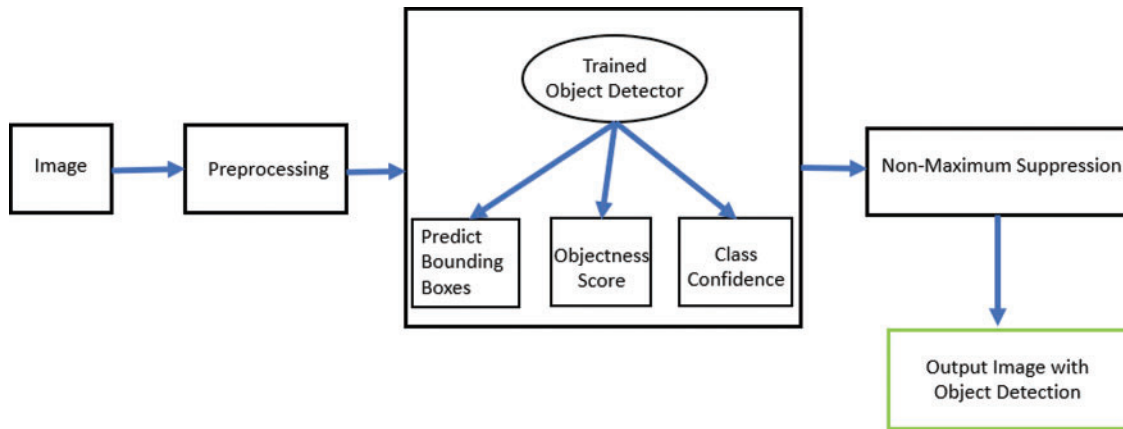| Name	Pure-fresh	Medium-fresh	Rotten
Apple	4082	2975	2785
Banana	2530	899	1999
Orange	2335	1287	1988
Tomato	4261	4715	859
Guava	768	1696	773
Lemon	2281	2881	4033
Brinjal	999	1240	528
Chilies	1083	1392	805
Cucumber	775	857	682
Pepper	2478	927	980
Potato	2115	971	1080
Total	**23,707**	**19,840**	**16,512**

Figure 2: Block diagram of the proposed approach

4.1 VGG-16 Architecture

Fig. 3 shows the architecture of VGG-16 [39,40]. We use softmax activation function in the output layer of model because there are multiple classes available i.e., Pure-fresh, Medium-fresh and Rotten. In the case of softmax, the sum of probabilities of all classes in output is 1, given in Eq. (1).
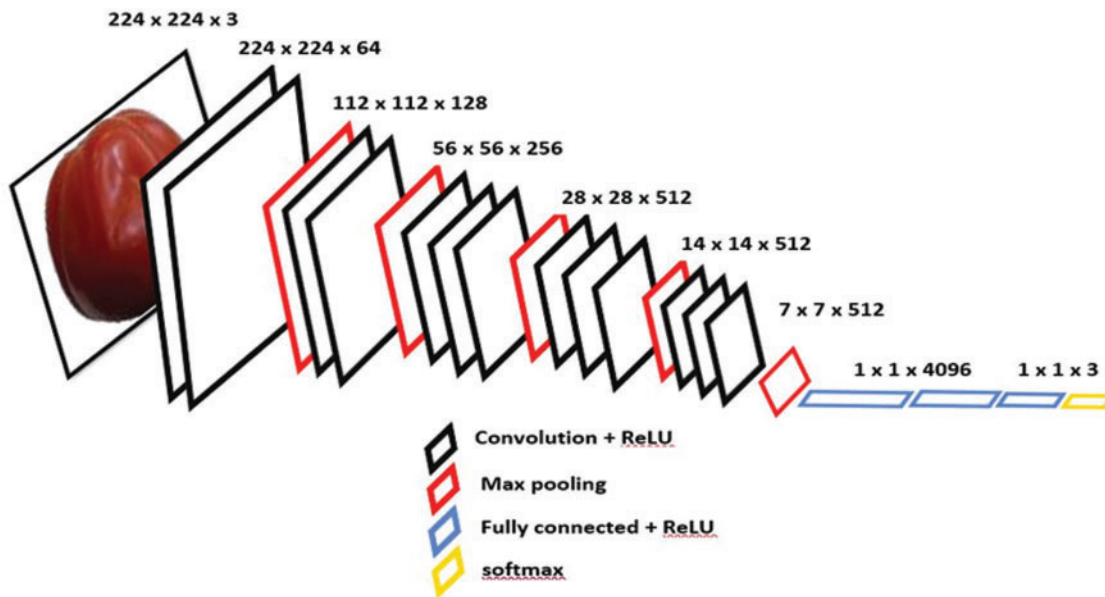

Figure 3: VGG-16 CNN architecture

$$softmax = f(v)_i = \frac{e^{v_i}}{\sum_{j=1}^{nClasses} e^{v_j}}, \tag{1}$$

where v is the vector, nClasses = 3 (fresh, medium, rotten). We use the loss function Categorical Cross Entropy (CE) for multi class classification given in Eq. (2).

$$CE \ Loss = - \sum_{i}^{nClasses} t_i log(f(v)_i), \tag{2}$$

where $f(v)_i$ is the softmax output of ith position in vector, t_i is target value, nClasses $= 3$ (Pure-fresh, Medium-fresh, Rotten).

4.2 YOLO-v5 for Detection and Classification

YOLO is an object detection algorithm based on regression [10]. YOLO can simultaneously predict location and classes of many objects present in an image. We use YOLO-v5 as it is fast and accurate as compared to other object detection models. YOLO-v5 [38,41] is a family of four models (small, medium, large, and extra-large). They differ in number of layers and number of weights in the network. We use medium version of YOLO-v5 (Fig. 4), which provides good performance with acceptable computational complexity.

For training, we provide the annotation file containing location of each object in the image along with its label and image. YOLO predicts the class and bounding box specifying object location. Each bounding box can be described using four descriptors: (i) Center of the box (bx, by), (ii) Width (bw), (iii) Height (bh), and (iv) class ID. Along with the bounding box, it also predicts the probability of presence of an object in the bounding box. YOLO-v5 uses Leaky ReLU activation function in hidden layers along with the sigmoid activation function in the final output layer. Another important parameter of the algorithm is its loss function, YOLO simultaneously learns about all the four parameters it predicts as discussed before. YOLO-v5 uses Binary Cross-Entropy with Logits loss function from PyTorch for calculating object score and class probability while the loss function for predicting bounding box is given in Eq. (3).

$$\begin{aligned}
Box \ Loss = &\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} M_{ij}^{object}[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2] \\
&+ \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} M_{ij}^{object} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i}\right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i}\right)^2 \right] \\
&+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} M_{ij}^{object}(C_i - \hat{C}_i)^2 \ + \lambda_{noobject} \sum_{i=0}^{S^2} \sum_{j=0}^{B} M_{ij}^{noobject}(C_i - \hat{C}_i)^2 \\
&+ \sum_{i=0}^{S^2} M_i^{object} \sum_{c \in nClasses} (p_i(c) - \hat{p}_i(c))^2,
\end{aligned} \tag{3}$$

where S is number of cells in the image, B is the number of Bounding boxes predicted in each grid cell, and c represents the class prediction for each grid cell. $p_i(c)$ represents the confidence probability. For any box *j* of cell *i,* x_{ij} and y_{ij} represent the coordinates of the center of the anchor box, h_{ij} gives height, w_{ij} gives width of the box and C_{ij} gives the confidence score. Finally, λ_{coord} and $\lambda_{noobject}$ are the weights to decide the importance of localization *vs.* recognition in the training.

Eq. (3) has five terms summed up to measure the loss. The 1st term penalizes the poor localization of centre of the cells. The 2nd term penalizes those bounding boxes having inaccurate width and height. The presence of square root indicates that errors in smaller bounding boxes are more penalized than

the errors in larger bounding boxes. The 3rd term attempts to equalize confidence score to Intersection-over-Union (IoU) between the prediction and object when there is only a single object. The 4th term minimizes confidence score close to zero when there is no object in the cell. Finally, 5th term represents classification loss. We train the model to detect if a cell contains an object and penalize the model if it predicts an object that does not exist in the cell. We achieve this by using two binary masks (M_{ij}^{object} and $M_{ij}^{noobject}$). M_{ij}^{object} is 1 if there is an object in the cell and 0 for other cells, while $M_{ij}^{noobject}$ is 0 for non-object cells and 0 otherwise.
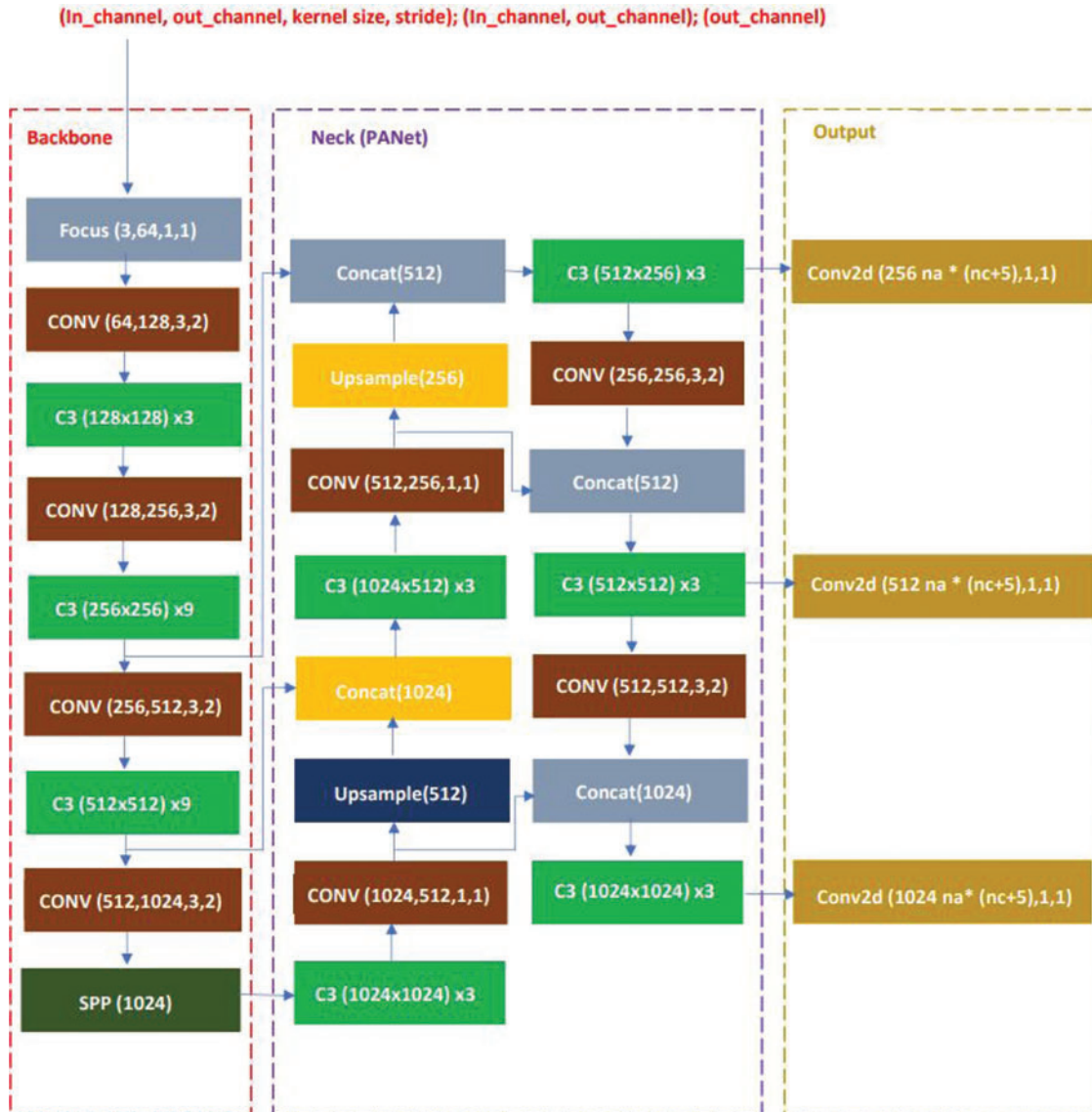

Figure 4: YOLO-v5 architecture

YOLO splits the image into cells, typically into a 19×19 grid. Each cell then predicts K number of bounding boxes. During the forward propagation, YOLO determines the probability of a certain

class for each cell given as:

$$score_{c,i} = p_c \times c_i, \tag{4}$$

where p_c is the probability that there is an object of certain class 'c'. The class with the maximum probability is assigned to that grid cell. Similar process is performed for all grid cells present in the image. After predicting the class probabilities, the next step is non-max suppression, it helps the algorithm to get rid of the unnecessary anchor boxes. YOLO finds the Intersection over Union (IoU) for all the bounding boxes with respect to the one with the highest-class probability given in Eq. (5) as:

$$IoU = \frac{B1 \cap B2}{B1 \cup B2}. \tag{5}$$

An object is in a specific cell if the center co-ordinates of the anchor box lie in that cell. Therefore, the center co-ordinates are always calculated relative to the cell, whereas the height and the width are calculated relative to the whole image size. It then rejects any bounding box whose value of IoU is less than a threshold. The above process is repeated to find the bounding box with the next highest-class probabilities until we locate all the objects in the image in separate bounding boxes.

5 Evaluation and Discussion

We evaluate the proposed approach on the challenging Fruit Freshness dataset containing 11 different fruits and vegetables and three categories of freshness for each class. For evaluation of the proposed approach, we split each Fruit and vegetable in the dataset into train and test sets in the ratio 70:30, respectively. Further we develop the Android based application, which gives real time output from the system.

5.1 Production Environment (Android Application and Sockets Based Server in Python)

We have developed an android based mobile application that connects the mobile devices with our trained models loaded on a server for the real time predictions. Following are the functionalities of the application:

- Capture Image using camera/load image from gallery or URL
- Connect to server using socket
- Send image to server
- Receive prediction from server
- Display the prediction

Fig. 5 shows the Architecture diagrams of the system. The input to the application is an image and the output is an image containing bounding boxes drawn around objects, label (classes and freshness category), and confidence score.
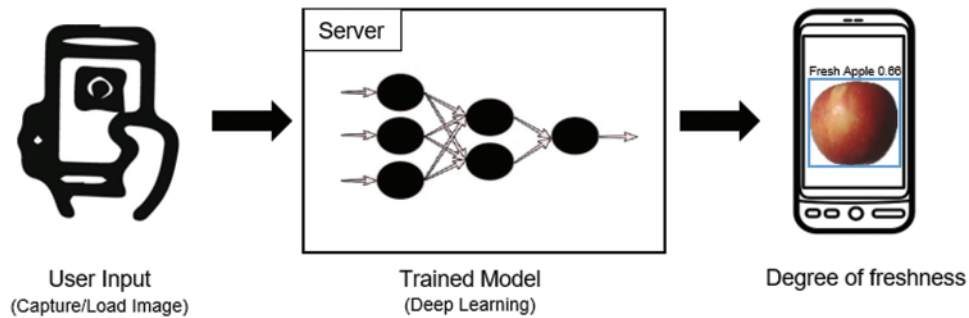
Figure 5: Functionality diagram of the android application

5.2 VGG-16

VGG-16 is trained for 50 epochs with a batch size of 32. The technique of early stopping is applied where if the loss is not reduced consecutively for 4 epochs, then training is stopped to avoid the overfitting of the model (Fig. 6). In classification of 33 classes, VGG-16 achieves an average test accuracy of 82.2%.
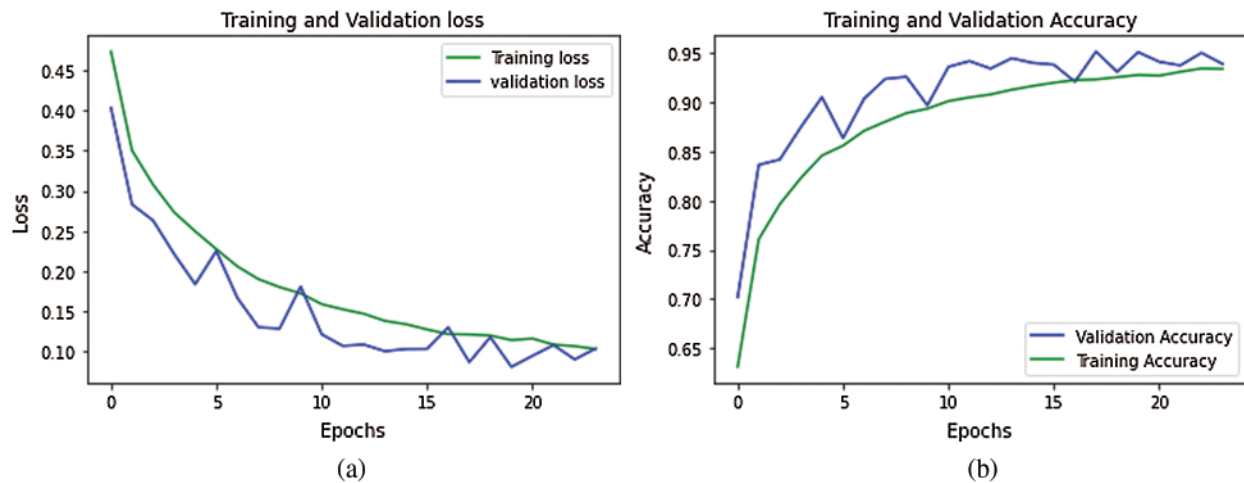

Figure 6: VGG-16 (a) Training loss; and (b) Accuracy graphs

5.3 YOLO-v5

We train YOLO-v5 to locate and classify fruit and vegetable, and to categorize their freshness. For training of YOLO-v5, we use Google Colab, since it provides virtual machines with state-of-the-art GPUs such as Tesla T4 and Tesla K80 with 16 GB of cache and 16 GB of RAM, for a 12-h session per day. Moreover 20 GB of Secondary storage is required for storing and loading of data. The network is trained for 200 epochs with a batch size of 8 on our FruitVeg Freshness dataset. Fig. 7 shows the performance of the model on training and validation data. First three columns represent the Bounding-box loss, Object loss, and classification loss in (top row) training and (bottom row) validation data. We can see that the precision and recall are also improved in the validation data with the number of epochs in the training. Since YOLO also provides the location of the object in the form of a bounding box, which can be validated using IoU. The last two graphs in the bottom row

show validation results of the mean average precision (maP) with 0.5 threshold as well as with varying thresholds ranging from 0.5 to 0.95 for IoU. It can be seen that the validation results become stable till the 200 epochs.
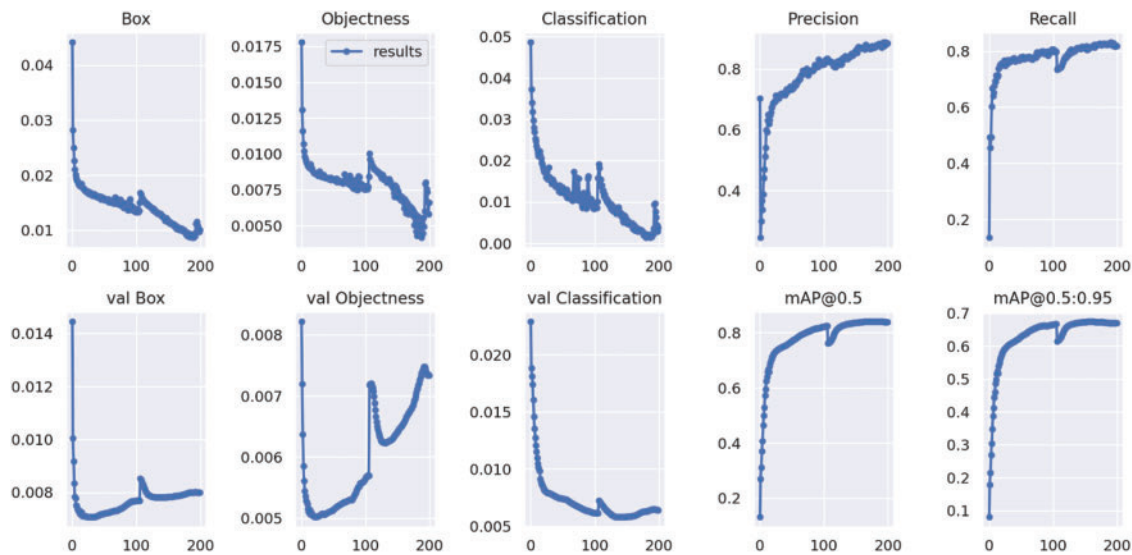

Figure 7: YOLO-v5: (top row) training, and (bottom row) validation graphs

Fig. 8 shows the confusion matrix for classification of 33 classes using YOLO-v5 on test data. Predicted classes are on vertical axis, while ground truth is along horizontal axis. It can be observed that most of the classes are correctly classified by YOLO. However, the error is due to detection of background as an object or an object if detected as background. In addition, some fruits because of their inter-class similarity results in classification confusion. For example, Apple categories are confused with Guava. Also, in a few cases, it is observed that the freshness categories within the same class are also confused because of high intra-class similarities.

Tab. 3 shows the detailed detection and classification results of each fruit and vegetable in FruitVeg Freshness dataset, where P is the Precision, R is the Recall. Also mAP at 0.5 IoU threshold and by varying threshold from 0.5 to 0.95. In addition, we measure the F1 score across all classes, which is considered a better measure compared to the accuracy. The variations in the appearance and structure of the three categories (pure-fresh, medium-fresh, and rotten) of each fruit and vegetable result in adding more complexity to the problem. Overall F1 score achieved by the system is 85%. However, individual classes range between 70% and 99% of F1score. Banana has the least F1 scores because of the less Recall. The reason of low recall is the high number of false negatives, i.e., Many of banana classes (pure-fresh, medium-fresh, and rotten) are classified as chilies or cucumber because of high inter-class similarities. It can be observed from the results that the deep learning model YOLO5 is able to learn these fine-grained differences those are even difficult to recognize visually in correct classification of fruits of vegetables into the defined categories.
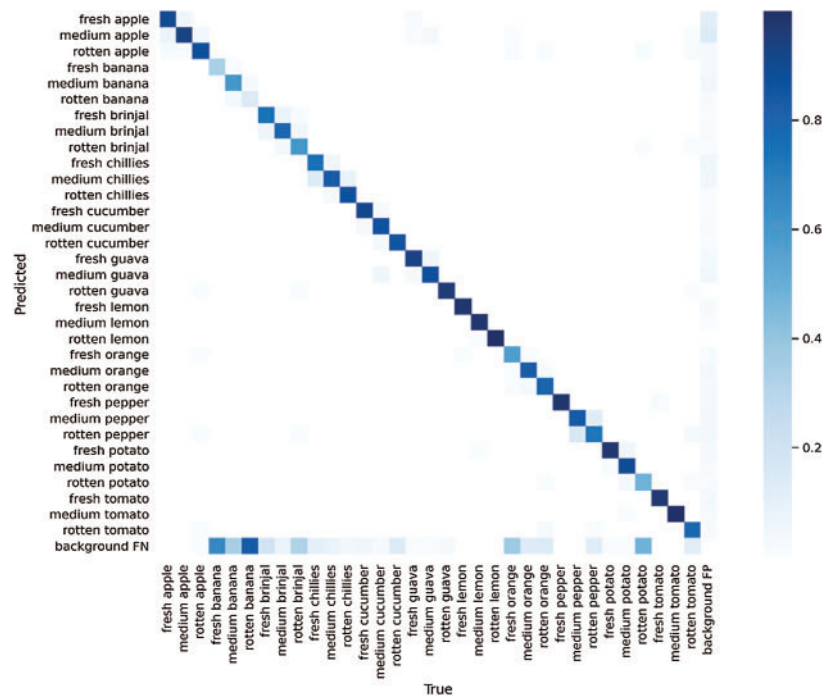
Figure 8: Confusion matrix for YOLO-v5 graphs. The color intensity indicates the probability of each element in a cell

Table 3: Dataset summary for YOLO-v5 model

Class	Prec.	Recall	mAP@.5	mAP@.5:.95	F1 Score
All	0.885	0.819	0.84	0.671	0.856
Fresh apple	0.85	0.982	0.968	0.805	0.911
Medium apple	0.886	0.895	0.946	0.747	0.890
Rotten apple	0.879	0.937	0.929	0.824	0.907
Fresh banana	0.902	0.323	0.394	0.196	0.475
Medium banana	0.769	0.587	0.61	0.363	0.666
Rotten banana	0.82	0.142	0.228	0.127	0.242
Fresh brinjal	0.908	0.716	0.788	0.593	0.801
Medium brinjal	0.835	0.843	0.815	0.571	0.839
Rotten brinjal	0.879	0.545	0.725	0.509	0.673
Fresh chilies	0.746	0.829	0.748	0.447	0.785
Medium chilies	0.846	0.773	0.884	0.62	0.808
Rotten chilies	0.87	0.914	0.93	0.68	0.891
Fresh cucumber	0.951	0.919	0.951	0.744	0.935
Medium cucumber	0.849	0.961	0.929	0.703	0.902
Rotten cucumber	0.974	0.829	0.857	0.584	0.896
Fresh guava	0.915	0.904	0.966	0.854	0.909
Medium guava	0.856	0.965	0.955	0.769	0.907

(Continued)

Table 3: Continued

Class	Prec.	Recall	mAP@.5	mAP@.5:.95	F1 Score
Rotten guava	0.982	0.917	0.962	0.838	0.948
Fresh lemon	0.958	0.999	0.984	0.862	0.978
Medium lemon	0.977	1	0.988	0.885	0.988
Rotten lemon	1	0.996	0.997	0.914	0.997
Fresh orange	0.888	0.594	0.61	0.502	0.711
Medium orange	0.9	0.84	0.87	0.783	0.868
Rotten orange	0.908	0.83	0.849	0.767	0.867
Fresh pepper	0.976	0.992	0.994	0.736	0.983
Medium pepper	0.804	0.804	0.939	0.818	0.804
Rotten pepper	0.779	0.73	0.809	0.733	0.753
Fresh potato	0.955	0.966	0.967	0.856	0.960
Medium potato	0.869	0.977	0.945	0.81	0.919
Rotten potato	0.696	0.42	0.39	0.161	0.523
Fresh tomato	0.946	0.995	0.983	0.761	0.969
Medium tomato	0.981	0.999	0.995	0.877	0.989
Rotten tomato	0.858	0.763	0.841	0.69	0.807

6 Conclusion

We proposed a new dataset with three categories of freshness for 11 fruits and vegetables. We proposed the fruits\vegetables classification and freshness categorizing approach by exploiting two deep learning models i.e., VGG-16 CNN and YOLO-v5. Moreover, an android based application is also developed for the real time classification\categorization of fruits and vegetables from the input image. We achieved mean Average Precision of 84%. These types of automated systems are quite useful in reducing the wastages of fruits and vegetables by early and in time consumption. VGG-16 is easy to setup, however, lacks in classification accuracy. While YOLO-v5 also localizes the object in addition to classification, however its performance is always a tradeoff between speed and accuracy. In the future, we would exploit newer and more efficient models, while increasing the size of dataset by incorporating more classes of fruits and vegetables.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] A. Bhargava and A. Bansal, "Fruits and vegetables quality evaluation using computer vision: A review," *Journal of King Saud University-Computer and Information Sciences*, vol. 33, no. 3, pp. 243–257, 2021.
[2] P. Moallem, A. Serajoddin and H. Pourghassem, "Computer vision-based apple grading for golden delicious apples based on surface features," *Information Processing in Agriculture*, vol. 4, no. 1, pp. 33–40, 2017.

[3]   K. K. J. Prakash, "Spoilage detection in raspberry fruit based on spectral imaging using convolutional neural network," MSc Dissertation, Technological University Dublin, Ireland, 2018.

[4]   R. Pandey, S. Naik and R. Marfatia, "Image processing and machine learning for automated fruit grading system: A technical review," *International Journal of Computer Applications*, vol. 81, no. 16, pp. 29–39, 2013.

[5]   M. P. Arakeri and Lakshmana, "Computer vision based fruit grading system for quality evaluation of tomato in agriculture industry," *Procedia Computer Science*, vol. 79, no. 1, pp. 426–433, 2016. https://cyberleninka.org/article/n/872976.

[6]   H. Hong, J. Lin and F. Huang, "Tomato disease detection and classification by deep learning," in *Proc. of ICBAIE*, Fuzhou, China, pp. 25–29, 2020.

[7]   H. Ali, M. I. Lali, M. Z. Nawaz, M. Sharif and B. A. Saleem, "Symptom based automated detection of citrus diseases using color histogram and textural descriptors," *Computers and Electronics in Agriculture*, vol. 138, no. 1, pp. 92–104, 2017.

[8]   K. Bresilla, G. D. Perulli, A. Boini, B. Morandi, L. C. Grappadelli *et al.,* "Using dl–convolutional neural networks (cnns) for real-time fruit detection in the tree," in *Proc. of ICPA*, Montreal, Quebec, Canada, pp. 510–520, 2018.

[9]   C. Pan and W. Q. Yan, "Object detection based on saturation of visual perception," *Multimedia Tools and Applications*, vol. 79, no. 27–28, pp. 19925–19944, 2020.

[10]   J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. of CVPR*, Las Vegas, USA, pp. 779–784, 2016.

[11]   G. Zeng, "Fruit and vegetables classification system using image saliency and convolutional neural network," in *Proc. of ITOEC*, Chongqing, China, pp. 324–331, 2017.

[12]   F. Valentino, T. W. Cenggoro and B. Pardamean, "A design of deep learning experimentation for fruit freshness detection," in *Proc. of Earth and Environmental Science*, Guiyang, China, pp. 1315–1755, 2020.

[13]   K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," in *Proc. of CVPR*, Las Vegas, USA, pp. 770–778, 2016.

[14]   C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed *et al.,* "Going deeper with convolutions," in *Proc. of CVPR*, Boston, United States, pp. 1–9, 2015.

[15]   H. Mureşan and M. Oltean, "Fruit recognition from images using deep learning," *Acta Universitatis Sapientiae, Informatica*, vol. 10, no. 1, pp. 26–42, 2018.

[16]   A. Nashat and N. M. Hassan, "Automatic segmentation and classification of olive fruits batches based on discrete wavelet transform and visual perceptual texture features," *International Journal of Wavelets, Multiresolution and Information Processing*, vol. 16, no. 1, pp. 185–193, 2017.

[17]   G. Xue, S. Liu and Y. Ma, "A hybrid deep learning-based fruit classification using attention model and convolution autoencoder," *Complex and Intelligent Systems*, vol. 6, no. 3, pp. 1–11, 2020.

[18]   R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. of CVPR*, Ohio, USA, pp. 580–587, 2014.

[19]   M. S. Hossain, M. Al-Hammadi and G. Muhammad, "Automatic fruit classification using deep learning for industrial applications," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 2, pp. 1027–1034, 2019.

[20]   S. K. Noon, M. Amjad, M. A. Qureshi and A. Mannan, "Use of deep learning techniques for identification of plant leaf stresses: A review," *Sustainable Computing: Informatics and Systems*, vol. 28, no. 100443, pp. 1–15, 2020.

[21]   T. Kasinathan, D. Singaraju and S. R. Uyyala, "Insect classification and detection in field crops using modern machine learning techniques," *Information Processing in Agriculture*, vol. 8, no. 3, pp. 446–457, 2021.

[22]   K. Hameed, D. Chai and A. Rassau, "A comprehensive review of fruit and vegetable classification techniques," *Image and Vision Computing*, vol. 80, pp. 24–44, 2018.

[23]   S. Cubero, N. Aleixos, E. Moltó, J. G. Sanchis and J. Blasco, "Advances in machine vision applications for automatic inspection and quality evaluation of fruits and vegetables," *Food Bioprocess Technology*, vol. 4, pp. 487–504, 2011.

[24] R. Pydipati, T. F. Burks and W. S. Lee, "Identification of citrus disease using color texture features and discriminant analysis," *Computers and Electronics in Agriculture*, vol. 52, no. 1, pp. 49–59, 2006.

[25] C. Kumar, S. Chauhan, R. N. Alla and H. M. Gurram, "Classifications of citrus fruit using image processing-GLCM parameters," in *Proc. of ICCSP*, Chengdu, China, pp. 1743–1747, 2015.

[26] O. Mahendra, H. F. Pardede, R. Sustika and R. B. S. Kusumo, "Comparison of features for strawberry grading classification with novel dataset," in *Proc. of IC3INA*, Tangerang, Indonesia, pp. 7–12, 2018.

[27] D. P. Cavallo, M. Cefola, B. Pace, A. F. Logrieco and G. Attolico, "Non-destructive automatic quality evaluation of fresh-cut iceberg lettuce through packaging material," *Journal of Food Engineering*, vol. 223, pp. 46–52, 2018.

[28] O. Almutiry, M. Ayaz, T. Sadad, I. Lali, A. Mahmood *et al.,* "A novel framework for multi-classification of guava disease," *Computers, Materials & Continua*, vol. 69, no. 2, pp. 1915–1926, 2021.

[29] X. E. Pantazi, D. Moshou and A. A. Tamouridou, "Automated leaf disease detection in different crop species through image features analysis and one-class classifiers," *Computers and Electronics in Agriculture*, vol. 156, no. 13, pp. 96–104, 2019.

[30] T. Lu, B. Han, L. Chen, F. Yu and C. Xue, "A generic intelligent tomato classification system for practical applications using DenseNet-201 with transfer learning," *Scientific Reports*, vol. 11, ID. 15824, pp. 1–8, 2021.

[31] J. Kang and J. Gwak, "Ensemble of multi-task deep convolutional neural networks using transfer learning for fruit freshness classification," *Multimedia Tools and Applications*, accepted, pp. 1–23, 2021.

[32] N. Ismail and O. A. Malik, "Real-time visual inspection system for grading fruits using computer vision and deep learning techniques," *Informaton Processing in Agriculture*, accepted, pp. 1–14, 2021.

[33] Y. Liu, H. Pu, D. W. Sun, "Efficeint extraction of deep image features using convolutional neural network for applications in detecting and analysing complex food matrices," *Trends in Food Science and Technology*, vol. 113, pp. 193–201, 2021.

[34] T. Ananthanarayana, R. Ptucha and S. C. Kelly, "Deep learning based fruit freshness classification and detection with cmos image sensors and edge processors," in *Proc. of Int. Symp. on Electronic Imaging*, Burlingam, USA, pp. 172–179, 2020.

[35] J. G. Sanchis, J. D. M. Guerrero, E. S. O. M. M. Sober, R. M. Benedito and J. Blasco, "Detecting rottenness caused by Penicillium genus fungi in citrus fruits using machine learning techniques," *Expert Systems with Applications*, vol. 39, no. 1, pp. 780–785, 2012.

[36] N. Chandel, S. K. Chakraborty, Y. A. Rajwade, K. Dubey, M. K. Tiwari *et al.,* "Identifying crop water stress using deep learning models," *Neural Computing and Applications*, vol. 33, pp. 5353–5367, 2021.

[37] S. Kaur, S. Randhawa and A. Malhi, "An efficient ANFIS based pre-harvest ripeness estimation technique for fruits," *Multimedia Tools and Applications*, vol. 80, pp. 19459–19489, 2021.

[38] R. Joseph and F. Ali, "YOLO9000: Better, faster, stronger," in *Proc. of CVPR*, Sousse, Tunisia, pp. 1–9, 2016.

[39] S. K. Behera, A. K. Rath and P. K. Sethy, "Maturity status classification of papaya fruits based on machine learning and transfer learning approach," *Information Processing in Agriculture*, vol. 8, no. 2, pp. 244–250, 2021.

[40] A. Krizhevsky, L. Sutskever and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2012.

[41] R. Gai, N. Chen and H. Yuan, "A detection algorithm for cherry fruits based on the improved YOLO-v4 model," *Neural Computing and Applications*, accepted, pp. 1–12, 2021.