**Fruits Recognition using Deep Learning Techniques**

Nicholas C. McCormick & Pedram Khannakhjavani

Shiley-Marcos School of Engineering

AAI – 521: Team 6

Dr. Saeed Sardari

December 08, 2022

# Abstract

Computer vision is a mathematical method used for analyzing, processing, and understanding the details of images. The interest in computer vision specifically, fruit and vegetable recognition, is increasing exponentially because it is considered an effective way of reducing the human workload. It can also be helpful for disease detection, fruit harvesting, optimizing point of sale (POS) systems, and educating children/people with learning disabilities. A lot of research has been conducted on fruit recognition but there are still many challenges in current methodologies that needed to be addressed to design an efficient fruit recognition system. Thus, this research focuses on developing a fruit recognition system using a machine learning algorithm. This study performs the classification of cherry, strawberry, and tomato using convolutional neural networks. The methodology includes background exploration, dataset exploration and analysis, model training, and model testing. The model scored an accuracy rate of 71%.

# Introduction

Numerous real-life applications, for example, object recognition, robotics, and autonomous vehicles mimic the functioning of the human brain to understand and analyze images. The food industry heavily relies on object recognition for the classification of freshly produced fruits and vegetables. In the conventional approach, a trained person is required to conduct visual inspection of fruits and vegetables for assessment of product/production quality. There are some human-related constraints in the case of conventional inspection approaches such as the person needing to acquire detailed training on information about the characteristics of fruit and vegetables (Hameed et al., 2018). The conventional approach needs to evolve to have a continual and consistent aspect recognition technique to maintain consistency. With advancements in technology, the agriculture industry is now moving towards mechanized classification for the harvesting of crops, which is based on computer vision (Bhargava and Bansal, 2018). It can be helpful for pest management (Pandey et al., 2013), insect detection (Koumpouros et al., 2004), disease detection (such as fungus) (Gulhane and Gurjar, 2011), fruit harvesting, plant nitrogen estimation (Tewari et al., 2013), and tracking of an object in real-time (Ozyildiz et al., 2002). The fruit and vegetables recognition technology can also be implemented in point of sale systems to make them more efficient, (Dubey and Jalal, 2015) and can also be used as an educational tool for cognitively impaired children in an effort to help them learn (Sahin, 1997).

Recent state-of-the-art techniques for recognizing fruits and vegetables combine machine learning algorithm(s) and feature(s) description techniques. Much of the research carried out on fruit recognition today, still face many challenges in their current methodologies that needed to be addressed to design efficient fruit recognition systems. Thus, For this project, we have implemented a computer vision machine learning application (CVML) that is capable of detecting tomatoes, cherries, and strawberries by primarily utilizing a convolutional neural network. We adhered to the following methodologies while developing our CVML application; in-depth Background exploration of this research and finding new computer vision techniques, detailed exploration of related baseline methodologies, dataset exploration and preprocessing, training and testing the model, and overcome the challenges faced in the previous studies.

# Problem investigation and Background

The background of the problem must first be understood in order to get a good grasp of the problem. Related work with their limitation in stated in table below.

Table 1: Related Methodologies

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr.no** | **Ref.** | **Method/Algo** | **Limitations** |
| 1. | (Vogl et al., 2014) | Formatting of field Code | Fruits recognition was based on color, texture, and shape of the fruit image. The system for more efficient for just cell phones.  . |
| 2. | (Meng and Wang, 2015) | Boundary Curvature | This research lacks detection of overlapping fruits. |
| 3. | (Agushinta et al., 2017) | Augmented Reality | This study was failed to detect fruits without the internet and it was fully dependent on servers. |
| 4. | (Rachmawati et al., 2017) | “Hierarchal Multi-model Classifier” | The hybrid feature approach used in this paper is difficult to implement. |
| 5. | (Buzzelli et al., 2018) | Neural baseline Architecture | This study presents a fruit recognition fridge that is not suitable for a larger area. |
| 6. | (Othman et al., 2016) | RGB sensor and Fuzzy Logic | The research only considers one type of fruit. |
| 7. | (Moallem et al., 2017) | SVM and KNN | The only direction of apple was considered in this research. |

# EDA

For the exploratory data analysis, we look at the image of each fruits category and concluded that there are the following issues in the images:

* Zoomed in images of fruits
* Zoomed out images of fruits
* Images with only fruits
* Fruits images with backgrounds containing other objects
* Combinations of multiple fruits in a single images

# Data pre-processing

After gaining a strong understanding of the issues in the dataset, we load the dataset and tried to pre-process the data with different techniques. The images are first loaded into one of each category and labelled as follows:

* Cherry as label 0
* Strawberry as label 1
* Tomato as label 2

Next, we finalized the 27000 pixels for each image so we can get the exact photo of the fruit and can ignore the background. This essentially reshaped the images into 300 pixels (width) by 300 pixels (height).

# Methodology

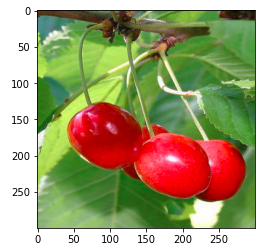
The methodology of this research is divided into different steps. These steps are as following:

* Dataset Exploration
* Dataset Preprocessing
* Simple Model
* Building CNN model
* Tuning and Training model
* Investigation of loss functions
* Applied different activation functions and optimizer

## Dataset Exploration

We used the Kaggle (Fruits 360) dataset. The dataset Fruits360 contains 55244 RGB images consisting of 100x100 pixels each. The dataset will be divided into three classes: tomato, cherry, and strawberry. 4500 images will be evenly divided amongst the three classes; 1500 each. All 4500 images will be converted to 300x300 as stated above. The images of the dataset include some noisy images that needed to be carefully handled.

In the First step, the images of the dataset are explored. There are a total of 3 types of fruits that are displayed. The following images show the image of cherry, strawberry, and tomato.

After that, the total count of the dataset is calculated. The following piece of code calculates the count of the dataset which is 4500.

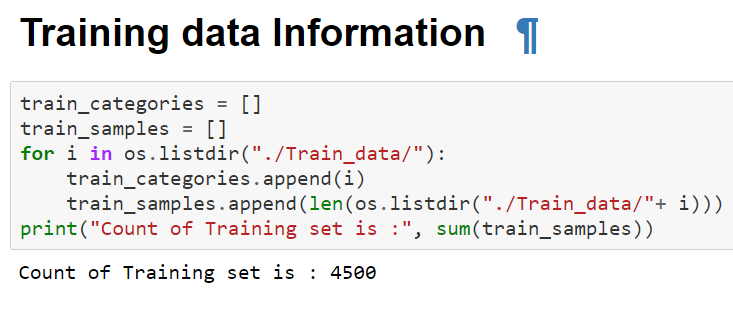


Figure 1: Training Data Information

After calculating the total count of the dataset, the frequency distribution is displayed. The dataset contains three classes. The x-axis represented classes of the dataset and the y-axis represents the number of image counts. Each class contains 1500 images.

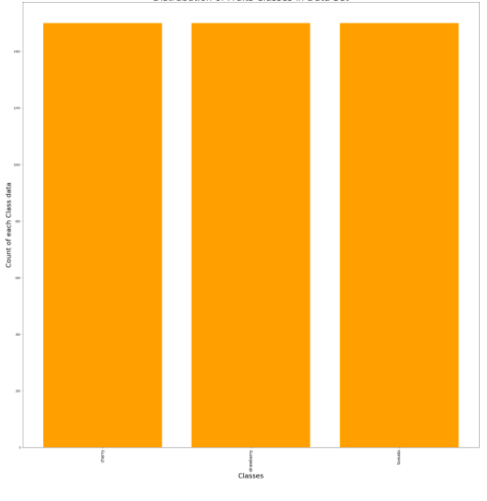


Figure 2: Distribution of Classes in Dataset

## Dataset Preprocessing

The second step of the methodology includes preprocessing of the dataset. First, data is extracted from folders. These are our three folders i.e. “cherry”, “Strawberry", and “tomato”. After extraction of folders, the dataset images larger than the size of 27000 are discarded. After that resizing is performed by giving each image the same dimension and size i.e. 300 x 300. Each image is saved into an array of numbers. Each class has assigned a label. 0 is assigned to cherry, 1 is assigned to strawberry, and 3 is assigned to tomato.



Figure 3: Dataset Preprocessing

## Simple Model

In this function crossEntrpyR2, forward, back, updater, and MPL are defined. Cross-Entropy Function is used as a loss function while performing optimization for the neural network. It calculates classification model performance whose output is a probability value between 0 and 1.  The forward pass function calculates the values output layer from input data. It navigates from the first to the last layer. The back pass function counts the weight change by utilizing a “gradient descent algorithm”. It moves from the last layer to the first layer. Multi-Layer Perceptron (MLP) is used to deal with multiple classes.

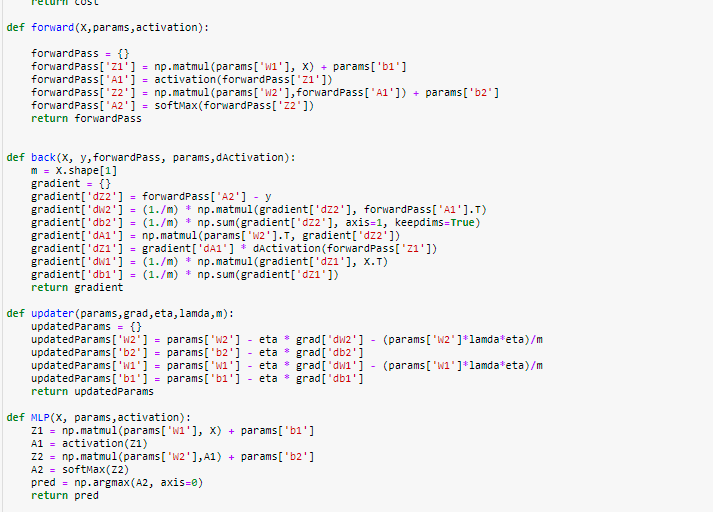


Figure 4: Simple Model

## Building CNN model

The sequential model is implemented in this study. The sequential model allows the handling of neural networks sequentially. There is a total of 4 layers of CNN used. First, ***conv2D*** is applied with 16 filters, 2 kernels size of 2, and input shape with size 300 x 300, where height is 300, and the width is 300. No padding is applied. After that, the ***Activation*** function performs the calculations for the layers. ***MaxPooling2D*** reduces the dimensions in order to extract more interesting features from the images. The conv2D is applied with increasing filter size from 16 to 32, then from 32 to 64 and at last from 64 to 128. After that, dropout is performed on nodes, and flattening is applied to convert matrices into 1d arrays. At the end, the dense function applies gives our 3 filter predictions. 

Figure 5: Build the CNN model

## Tuning and Training model

After Building the CNN model tuning and training of the model are performed. The batch size is set to 32. All the images are grouped into 32 bathes and 1 batch is processed at a time. A total of 30 epochs is used for model training. Categorical Crossentropy is used as the loss function with an adams optimizer because it is effective at solving classification problems that have multiple classes, in this case we have 3.

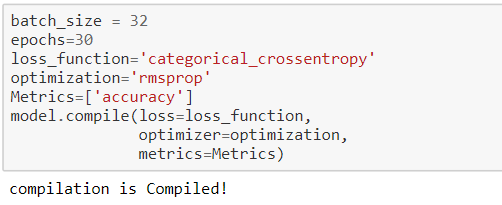


Figure 6: Tuning and Training model

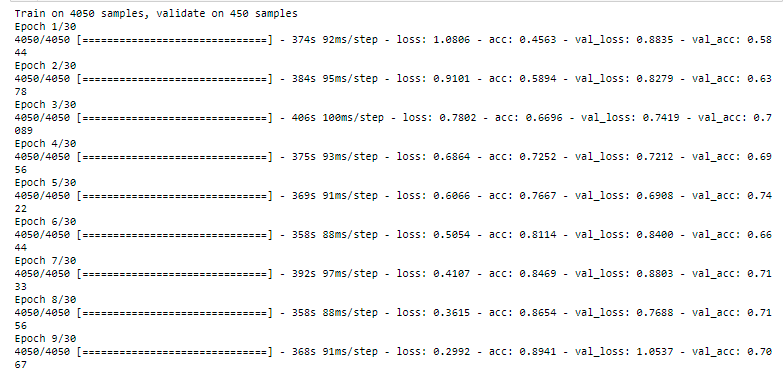
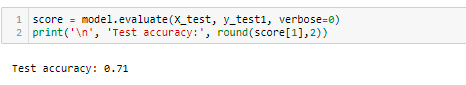


Figure 7: Model training on epochs

The model is saved as model.h5 so that it can be used in the future.

# Results & Discussion

The Trained model is tested on the testing dataset. The test result shows an accuracy rate of up to 71% which is quite low and leaves a lot of room for improvement.



The following confusion matrix shows the results of testing. The x-axis represents prediction labels and the y-axis represents true labels.

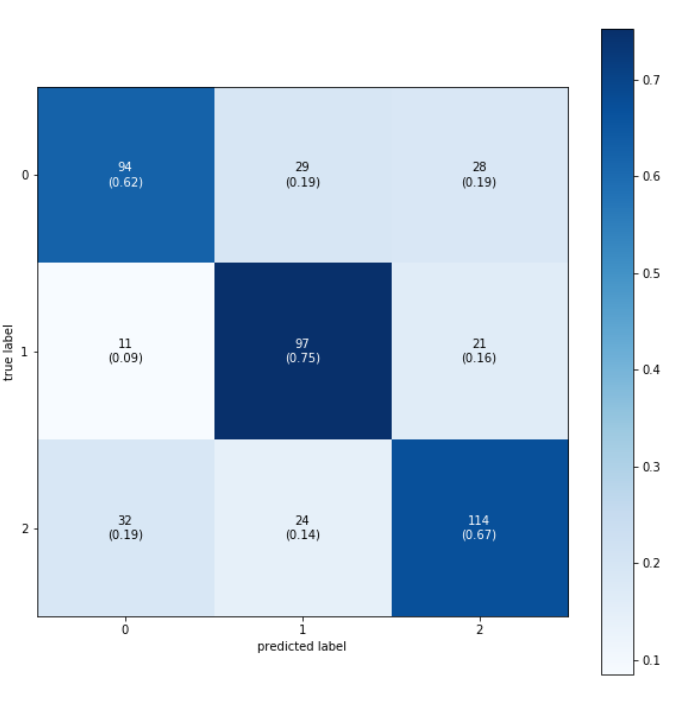


Figure 8: Confusion Matrix

The above figure shows the confusion matrix of the model. 0 represents cherry class, 1 represents strawberry class, and 3 represents tomato class. For cherry class, of 151 examples, 94 are predicted correctly, 29 are predicted strawberry, and 28 are predicted as tomatoes. For strawberry class, of 129 examples, 97 are predicted correctly, 11 are predicted cherry, and 21 are predicted as tomatoes. For tomatoes class, of 170 examples, 114 are predicted correctly, 24 are predicted strawberry, and 32 are predicted as cherry.

The performance of the model was lowest on class cherry with a true positive rate of 0.62. The model performance was high on the tomato class with a true positive rate of 0.71. The overall accuracy of the model was quite low. The code was ran on Jupyter Notebook due to Google Collaboratory not providing enough RAM to complete the necessary calculations. After completing the experiments, we have saved these python files:

* Code.py
* Train.py
* Test.py
* Model.h5
* Traindata

# Conclusions and Future work

The interest in computer vision specifically, fruit and vegetable recognition, is increasing exponentially because it is considered an effective way of reducing the human workload in the agricultural industry. In conclusion our CVML application shows that there is potential for future fruit identification. A fully successful computer vision fruit detection program could reduce the need for manual sorting of fruits, ultimately reducing costs and making harvesting more efficient. It could also potentially lead to helping cognitively impaired children learn more effectively in the future. The methodology used in this project includes background exploration, dataset exploration and analysis, model training, and testing. We achieved the 71% accuracy by splitting the training data further into train set and test set. The MLP model had a low accuracy with 55%, but had a much shorter training time when compared to the CNN model, making it more ideal for a production environment. The CNN model took almost 7 minutes on each epoch and MLP took 3 minutes, making a huge difference. Training time is also important but the primary goal is to get accuracy. The result of the fruit recognition system shows that the performance of CNN was best for the tomatoes class and lowest in the cherry class. There is a need to improve the model performance by considering more features of images. The study aims to consider more features in order to improve more accuracy. However, clearly the research can be carried out on more fruits types and can add other machine learning models to achieve higher accuracies. Transfer learning and ensemble learning could also be used in an effort to achieve higher accuracy. This example application was technically limited by processing power, and time.

**References**

Agushinta, R.D., Medyawati, H., Jatnika, I., others, 2017. A method of cloud and image-based tracking for Indonesia fruit recognition, in: 2017 IEEE 3rd International Conference on Engineering Technologies and Social Sciences (ICETSS). IEEE, pp. 1-5.

Bhargava, A., Bansal, A., 2018. Fruits and vegetables quality evaluation using computer vision: A review. J. King Saud Univ.-Comput. Inf. Sci.

Buzzelli, M., Belotti, F., Schettini, R., 2018. Recognition of Edible Vegetables and Fruits for Smart Home Appliances, in: 2018 IEEE 8th International Conference on Consumer Electronics-Berlin (ICCE-Berlin). IEEE, Pp. 1-4.

Dubey, SR., Jalal, AS., 2015. Application of image processing in fruit and vegetable analysis: a review. J. Intell. Syst. 24, 405 424.

Gulhane, VA., Gurjar, A.A., 201 1. Detection of diseases on cotton leaves and its possible diagnosis, Int, J. Image Process. IJIP 5, 590-598.

Hameed, K., Chai, D., Rassau, A., 2018, A comprehensive review of fruit and vegetable classification techniques. Image Vis. Comput. 80, 24 44. 44,121-131.

Koumpouros, Y, Mahaman, B, Maliappis, M, Passam, H., Sideridis, 2004. Image preprocessing for distance diagnosis in pest management. Comput, Electron. Agric, 41, 121-131.

Meng, J., Wang, S., 2015. The recognition of overlapping apple fruits based on boundary curvature estimation, in: 2015 Sixth International Conference on Intelligent Systems Design and Engineering Applications (ISDEA). IEEE, pp. 874 877,

Moallem, P., Serajoddin, A., Pourghassem, H., 2017. Computer vision-based apple grading for golden delicious apples based on surface features. Inf. Process. Agric. 4, 33 40.

Othman, M., Bakar, M.N.A…, Ahmad, K.A., Razak, TR., others, 2016. Mango Size Classification Using RB Color Sensor and Fuzzy Logic Technique, in: Regional Conference on Science, Technology and Social Sciences (RCSTSS 2014). Springer, pp. 287-296.

Ozyildiz, E., Krahnstöver, N., Sharma, R., 2002. Adaptive texture and color segmentation for tracking moving objects. Pattern Recognition. 35, 2013-2029.

Pandey, R., Naik, S., Marlatia, R., 2013. Image processing and machine learning for automated fruit grading system: a technical review. Int. J. Comput. ppl. 81, 29-39.

Rachmawati, E., Supriana, I., Khodra, M.L.., 2017. Toward a new approach in fruit recognition using hybrid RGBD features and fruit hierarchy property, in: 2017 4th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI): IEEE, pp. 1-6

Sahin, F., 1997. A radial basis function approach to a color image classification problem in a real time industrial application (PhD Thesis). Virginia Tech.

Tewari, V,, Arudra, A .K., Kumar, S.P, Pandey, V. Chandel, N.S., 2013. Estimation of plant nitrogen content using digital image processing. Agric, Eng. Int. CIGR J. 15, 78 86.

Vogl, M., Kim, JAY, Kim, 5.-D. 2014. A fruit recognition method via image conversion optimized through evolution strategy, in; 2014 FEE I 7th International Conference on Computational Science and Engineering, FEE, pp. 1497 1502.

Fruits 360. https://www.kaggle.com/moltean/fruits : Kaggle, 2020.