FRUIT CLASSIFICATION USING NEURAL NETWORK-BASED TECHNIQUES

Course: CN7023 - Artificial Intelligence and Machine Vision

University of East London

Tutor: Shaheen Khatoon

Date Submitted: 09 May, 2024

Submitted by

Dhruvi Vekariya - 2590401

TABLE OF CONTENTS

S.No.	Name of Contents		Page No.
1	**Introduction**		
2	**Simulations**		
	2.1	Description of dataset including the sample images	4
	2.2	Process for encoding the dataset to use the images as input to the neural network	6
	2.3	The network architecture that is used to train, validate, and test the network	6
		2.3.1 Implement of GoogLeNet	7
		2.3.2 Train Network	9
3	**Results Obtained**		
4	**Critical Analysis of Results**		11
	4.1	Information about the results by making changes to the simulations	11
	4.2	Detailed analysis and discussion of the results obtained	12
5	**Conclusion**		12
6	**MATLAB Certificates**		13
	6.1	Course 1: MATLAB Onramp	13
	6.2	Course 2: Machine Learning Onramp	14
	6.3	Course 3: Deep Learning Onramp	15
	6.4	Course 4: Image Processing Onramp	16
7	**References**		17

FIGURES LIST

S.No.	Name of Figures
1	Visualization of sample images
2	An interactive visualization of the network architecture and detailed information about the network layers (Image input layer)
3	An interactive visualization of the network architecture and detailed information about the network layers (Image output layer)
4	The detail results of GoogLeNet
5	Confusion Matrix

1. INTRODUCTION

The primary objective of this study is to support the development and evaluation of deep learning models, particularly neural networks algorithms utilized for the precise classification of natural products and vegetables based on their visual characteristics. Classification of a variety of fruits is an important framework in the agriculture field for import and export. The fruits are used in a range of businesses and this aims to facilitate research in automated quality control in the food industry, agricultural produce sorting, and other real-world applications by providing a standardized collection of diverse images. The purpose of this report is to enable the development and provide a generalized and all-embracing assessment of the neural network-based methodology for accurate classification of fruits and vegetables based on visual characteristics. The research is initiated by providing an overview of the problem and objectives.

The report also recommended the network architecture, along with the procedures for training and validation, alongside the utilized learning algorithm. Subsequent sections of the paper present the obtained results, encompassing test set accuracy, accuracy curves for training, validation, and testing, and a confusion matrix for visualizing classification performance. Then, the results are outlined clearly, accompanied by descriptions of implemented methods and potential areas that could be worked upon to further improve the results.

2. SIMULATIONS

2.1 Description of dataset including the sample images

The dataset contains 22,495 images of fruits and vegetables, with 33 distinct classes representing various types of produce. The dataset is separated into two sets. To begin with is the training set containing 16,854 pictures and moment is the test set containing 5,641 images, each image describing a single fruit or vegetable. In this dataset the images are standardized to a size of 100x100 pixels, facilitating uniformity in data representation across different classes. In the training set, images are organized into subfolders corresponding to each fruit or vegetable class and also many images are rotated to augment the dataset and enhance model generalization by exposing it to variations in orientation. The testing set consists of images located in a separate folder. These images serve as unseen data for evaluating the performance of trained models.

This dataset gives a comprehensive collection of pictures representing a various range of fruits and vegetables, enabling researchers and practitioners to develop and evaluate neural network algorithms for automated classification tasks. The availability of a large number of images per class ensures that models can learn to distinguish between different types of produce effectively. Moreover, the standardized image size and naming conventions streamline data preprocessing and model development workflows, facilitating reproducibility and comparability across experiments.

Figure 1: Visualization of sample images

2.2 Process for encoding the dataset to use the images as input to the neural network

To ensure the dataset is properly prepared for input into the neural network, several preprocessing steps are conducted. The image dataset is used in training and validating a neural network model. Initially, created an image datastore named `imds`, that source images from the 'archive' directory with included subfolders, and each folder names are utilized as labels for the images. Next, counts the number of images in each class and displays the result into the distribution of images across different categories. Following this, the images were resized to a fixed dimension of 224 x 224 pixels to ensure consistency in image sizes. The dataset is then split into training and validation sets, with 70% of the images allocated to training and the remaining 30% to validation.

Lastly, a choice of sample images from the training set is shown for visual assessment, helping in understanding the nature and characteristics of the dataset. These preprocessing steps are significant for organizing, reviewing, and partitioning the data before training and validating the neural network model, ensuring its readiness for subsequent machine learning tasks.

2.3 The network architecture that is used to train, validate, and test the network

The network architecture used for the fruit classification task is based on the GoogLeNet convolutional neural network (CNN) architecture. This architecture is characterized by parallel processing of image features at diverse scales. The architecture includes multiple convolutional layers, pooling layers, and fully connected layers, with auxiliary classifiers at intermediate layers to aid in training. Once training is complete, the network's performance is evaluated on a separate test set to assess its accuracy and performance on new, unseen data. The learning algorithm used, such as SGD or Adam, determines how the network's parameters are updated during training to minimize the loss function and improve performance.

2.3.1 Implement of GoogleNet:

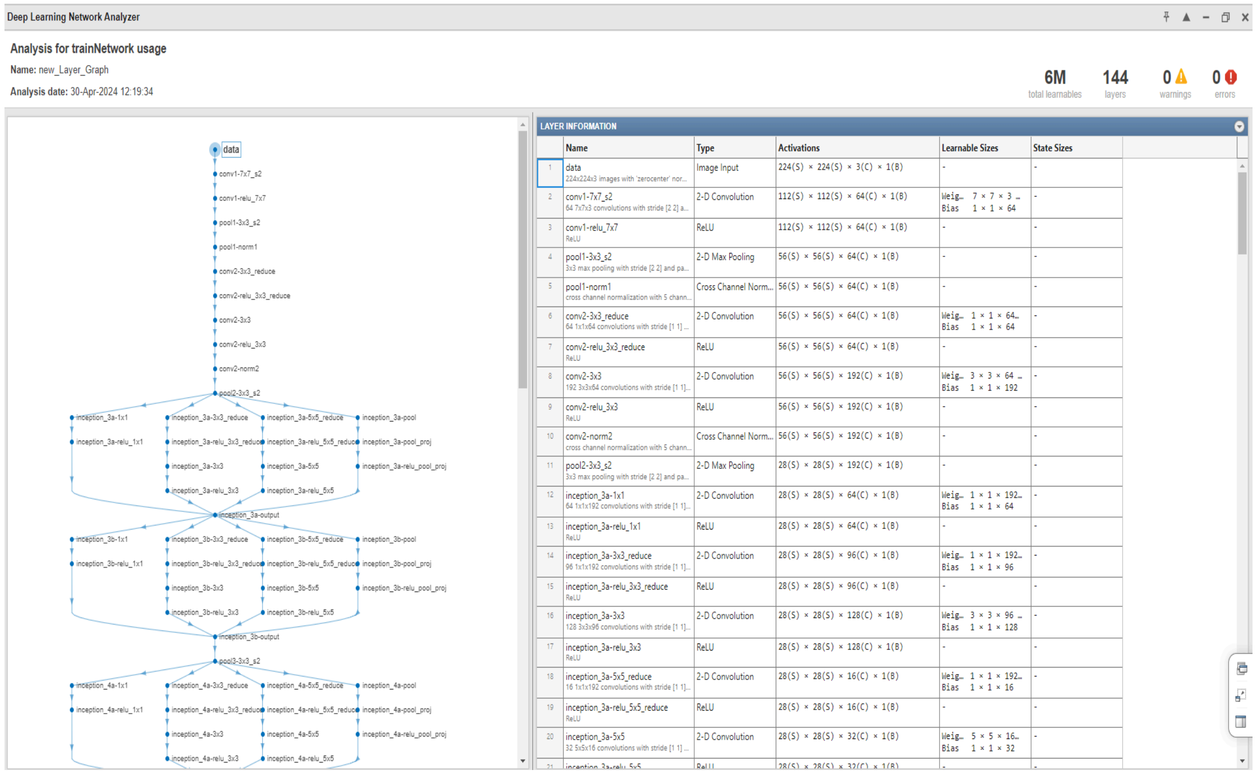

Figure: 2 An interactive visualization of the network architecture and detailed information about the network layers (Image input layer)

In Figure:2, the first layer describes the image input layer that requires input images of size 224-by-224-by-3, where 3 is the number of color channels. In the first convolution layer (conv1) of GoogLeNet, the filter size is typically set to 7x7 pixels, which means each filter in conv1 is a square matrix with dimensions of 7 pixels by 7 pixels. The number of filters commonly ranges from 64 to 192 filters. These filters serve as feature detectors, each looking for different patterns or features within the input image. The stride for this layer refers to the number of pixels by which the filter slides over the input image, and is usually set to 2 pixels. This means that the filter moves 2 pixels at a time horizontally and vertically across the input image.

The size of the output image from conv1 can be calculated using a formula that takes into account the input size is 100x100 pixels, and the filter size is 7x7 pixels with a stride of 2 and padding, the output size can be determined using the provided formula. Furthermore, the number of parameters in conv1, including weights and biases, can be calculated based on the filter size, number of input channels (assuming an RGB image with 3 channels), and the number of filters. This formula yields the total number of parameters required to define the convolutional layer, representing the trainable parameters that the network learns during the training process.

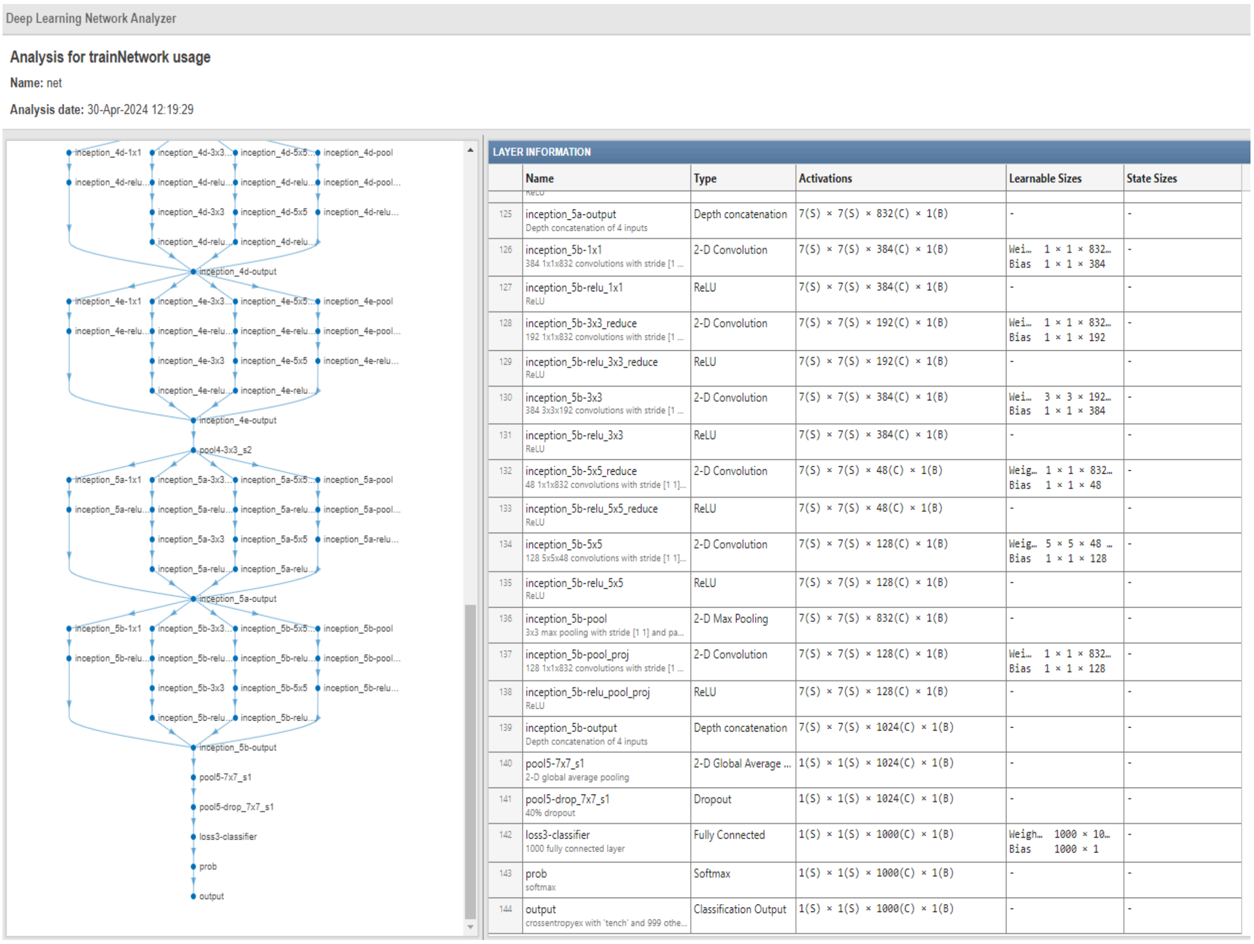

Figure :3 An interactive visualization of the network architecture and detailed information about the network layers (Image output layer)

The convolutional layers of a neural network extract image features from input images, which are then utilized by the last learnable layer and the final classification layer for image classification. In the case of GoogLeNet, these layers are labeled as ‘loss3-classifier’ and ‘output’. They play a crucial role in combining the extracted features to generate class probabilities, compute a loss value, and predict labels for input images.

2.3.2 Train network

To ensure compatibility with the network's input size requirement of 224-by-224-by-3, we employ an augmented image datastore. This facilitates automatic resizing of the training images to meet the specified dimensions. Additionally, we indicate extra expansion operations to enhance the diversity of the training data and mitigate overfitting. Specifically, I arbitrarily flip the preparing images along the vertical axis and apply random horizontal and vertical translations of up to 30 pixels. These augmentation techniques help introduce variability into the training data, making the network more robust to variations in input images and reducing the risk of memorizing specific details from the training set. Ultimately, data augmentation enhances the network's generalization capabilities and improves its performance on unseen data.

To resize validation images without encouraging increase, an augmented image datastore is utilized without specifying additional preprocessing. For transfer learning, keep early layer features from the pretrained network, setting the initial learning rate low to slow learning in these layers. Learning rate factors for fully connected layers were increased in a prior step to expedite learning. This approach ensures fast learning in new layers and slower learning in others. Fewer epochs are needed for transfer learning, where an epoch denotes a full training cycle on the entire dataset. Mini-batch size and validation data are specified, with the network validated every ValidationFrequency iterations during training.

In the last, the finalized model was evaluated on the test data set, which only included data not used in training or validation processes. The output parameter on the test set calibrates the model generalization capability and real-world performance.

3. Results Obtained

In the fruit classification experiment, the validation accuracy reached 91.7%, indicating that all validation images were correctly classified by the model. Consequently, the final accuracy of the model is 1, denoting perfect classification performance. However, it's noteworthy that the training process consumed a considerable amount of time, totaling 134 minutes and 40 seconds. In this experiment, the maximum number of epochs for GoogLeNet was set to 8, with a maximum of 3928 iterations. Where the iteration per epoch is 471. The learning rate is scheduled constant and it is 0.0003. The detailed results of the experiment, including accuracy curves and confusion matrices, are depicted in Figure 4 and 5. These outcomes demonstrate the effectiveness of GoogLeNet in accurately classifying fruit images, albeit at the expense of significant computational resources and time.

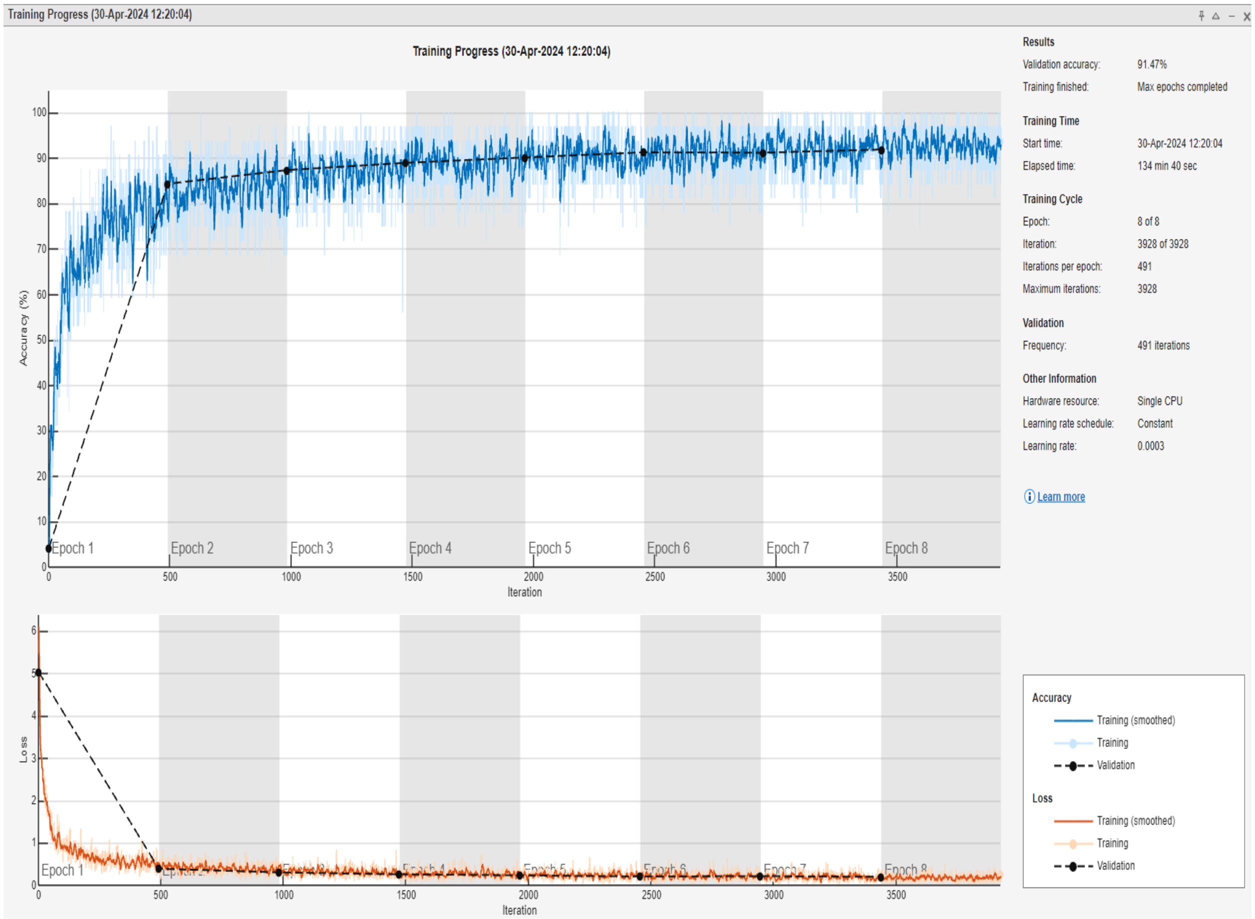

Figure 4: The detail results of GoogLeNet

Figure 5 shows the confusion matrix of fruit classification. It gives a detailed categorization of the model's performance on the test set, which includes the number of true and false predictions for each class. While the columns represent the predicted labeling. In which horizontal line provides the predicted class and vertical line provides the true class. Along the diagonal of the confusion matrix, which represents the true positives and true negatives, higher values indicate accurate classification within each class. Conversely, off-diagonal elements represent misclassifications, where the model wrongly predicts one class as another.

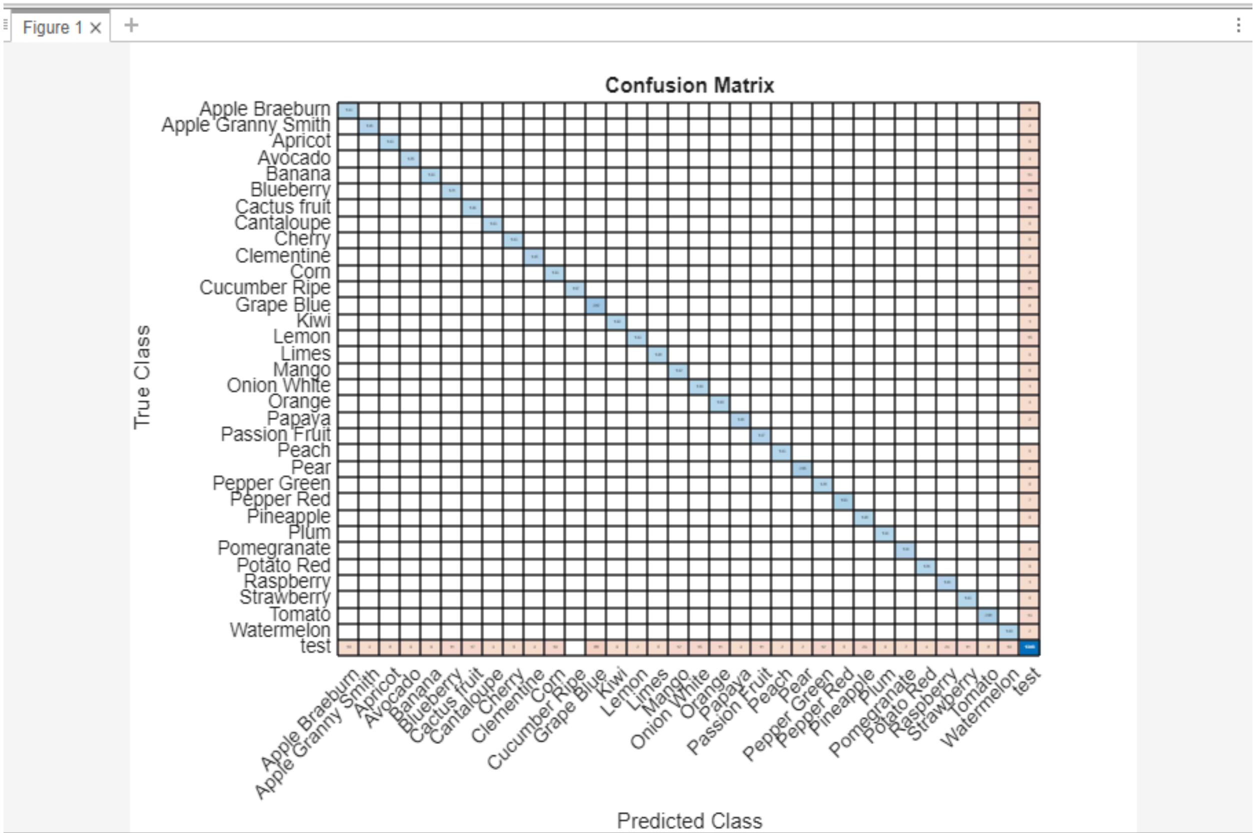

Figure: 5 Confusion Matrix

4. Critical Analysis of Results

4.1 Information about the results by making changes to the simulations

Obtaining the reported result, perform the certain changes and adjustment to the simulations as well as the neural network architecture. Firstly, I tried the AlexNet to train this model. It has about 62 layers. In this training network the validation accuracy I get is 88.4% and maximum number of epochs is 6 and the total number of iterations is 9954, per each epoch it has 571 number of iterations and it takes nearly 10 hours to get the validation accuracy of the test data. The learning rate set for this network is 0.0001. For better accuracy I use the GoogLeNet for this dataset and in this experiment I get 91.7% of validation accuracy. And it takes a total 134 minutes and 40 seconds to perform 8 numbers of epochs with maximum 3928 iterations. The learning rate for this network is set to 0.0003.

Also, after this I tried restnet for this dataset but it takes too much time to train the data for validation. These underscores need the high-powered computing environment to expedite the execution of such complex tasks.

4.2 Detailed analysis and discussion of the results obtained

The results of the fruit classification experiment reveal several noteworthy insights. Achieving 91.7% validation accuracy indicates a remarkable level of precision in classifying fruit images, suggesting that the model has effectively learned to distinguish between different fruit types. However, the substantial training time of nearly 8 hours underscores the computational demands associated with training complex neural network architectures like GoogLeNet. This highlights the need for high-performance computing resources to expedite the training process and mitigate the associated time and cost burdens. Despite the impressive validation accuracy, it's important to critically assess the model's generalization performance and potential limitations, such as overfitting to the training dataset or challenges in classifying certain fruit types. Future research efforts could focus on refining the model architecture, optimizing training procedures, and exploring techniques to enhance generalization and scalability for real-world deployment scenarios.

5. Conclusion

Fruit classification may be an exceptionally imperative assignment in numerous areas such as industrial or agriculture. In this study, I proposed an approach that uses deep learning-based learning of 22,495 images of fruit and vegetables from Kaggle site. I utilize a pre-trained CNN Model GoogLeNet. In this report, I trained and validated the proposed demonstration and tested its execution with in-seen dataset for testing. The accuracy rate I achieved was 91.7%. This indicates that the proposed model can effectively predicate and classify different fruits without error and full performance. As for future work, I will generalize the evaluation of the proposed framework for more classes (using extra fruit and vegetables). I will examine the impact of distinctive parameters such as activation function, pooling function optimization method, and a loss function. The proposed framework can also be deployed into a cloud-based framework.

6. MATLAB Certificates

6.1 Course 1: MATLAB Onramp


MathWorks | Training Services

Course Completion Certificate

Dhruvi Vekariya

has successfully completed **100%** of the self-paced training course

MATLAB Onramp

DIRECTOR, TRAINING SERVICES

6 February 2024

6.2 Course 2: Machine Learning Onramp


MathWorks | Training Services

Course Completion Certificate

Dhruvi Vekariya

has successfully completed **100%** of the self-paced training course

Machine Learning Onramp

DIRECTOR, TRAINING SERVICES

2 May 2024

6.3 Course 3: Deep Learning Onramp

MathWorks | Training Services

Course Completion Certificate

Dhruvi Vekariya

has successfully completed **100%** of the self-paced training course

Deep Learning Onramp

DIRECTOR, TRAINING SERVICES

12 March 2024

6.4 Course 4: Image Processing Onramp

MathWorks® | Training Services

Course Completion Certificate

Dhruvi Vekariya

has successfully completed **100%** of the self-paced training course

Image Processing Onramp

DIRECTOR, TRAINING SERVICES

2 May 2024

7. References

1) M. Shamim Hossain and Ghulam Muhammad, "Automatic Fruit Classification Using Deep Learning for Industrial Applications," IEEE transaction on industrial informatics, vol. 15, no. 2, February 2019.
2) Fruit Classification - 22,495 images of Fruit and Vegetables https://www.kaggle.com/datasets/sshikamaru/fruit-recognition
3) Mohammed A. Alkahlout, Samy S. Abu-Naser, Azmi H. alsaqqa, Tanseem N. Abu-Jamie, "Classification of Fruits Using Deep Learning," IJAER, vol. 5 Issue 12, December - 2021, Pages: 56 - 63.
4) Mehenag Khatun and Pritom Sarker, "Fruits Classification Using Convolutional Neural Network," ResearchGate.net, July 2020.