

## Transfer Learning-based Approach for the Detection of Fruit Freshness

Dr. Thanveer Jahan<sup>1\*</sup>, E. Manikeerthan<sup>2</sup>, K. Sai Teja<sup>2</sup>, Ch. Chandra siddhartha<sup>2</sup>, Gaddala Subhash<sup>2</sup>

<sup>1,2</sup>Department of Computer Science and Engineering (AI&ML), Vaagdevi College of Engineering, Bollikunta, Warangal, Telangana.

\*Corresponding Email: [thanvivcece@gmail.com](mailto:thanvivcece@gmail.com)

### ABSTRACT

The detection of fruit freshness is a critical aspect of the agricultural supply chain, ensuring quality control, reducing wastage, and enhancing consumer satisfaction. This research proposes a Transfer Learning-based Approach for the Detection of Fruit Freshness utilizing the GoogLeNet architecture, a deep convolutional neural network known for its Inception modules which capture multi-scale features effectively. The proposed system leverages a pre-trained GoogLeNet model, fine-tuning it with a customized fruit freshness dataset to classify fruits based on their freshness levels. Extensive Exploratory Data Analysis (EDA) was performed to understand the distribution of freshness labels, visualize sample images, and evaluate class imbalance. Techniques such as image augmentation (rotation, flipping, brightness adjustment, etc.) were applied to enhance dataset variability and improve model robustness. The GoogLeNet model was adapted by replacing its final fully connected layers with a series of customized layers tailored for the classification task. Hyperparameter tuning was conducted to optimize the model's performance. The evaluation metrics utilized include accuracy, precision, recall, and F1-score, ensuring a comprehensive assessment of the model's effectiveness. Preliminary results demonstrate that the proposed approach achieves high classification accuracy, showcasing GoogLeNet's efficacy in extracting complex features relevant to fruit freshness. This model presents a scalable and efficient solution for automated fruit freshness detection, with potential applications in agricultural quality monitoring and supply chain optimization.

**Keywords:** Transfer Learning, Fruit Freshness Detection, GoogLeNet, Inception Modules, Smart Agriculture.

### 1. INTRODUCTION

The detection of fruit freshness plays a crucial role in the agricultural supply chain, directly influencing food quality, shelf life, and consumer satisfaction. Fresh produce, especially fruits, undergoes natural degradation over time due to factors like ripening, exposure to external elements, and biological changes. As a result, determining the freshness of fruits is vital to ensure that only high-quality produce reaches consumers, while preventing spoilage and wastage during storage, transportation, and retail. Traditionally, assessing fruit freshness relied on human inspectors, who examined visual characteristics such as color, texture, and firmness to classify the fruits into different freshness categories (e.g., ripe, unripe, overripe).

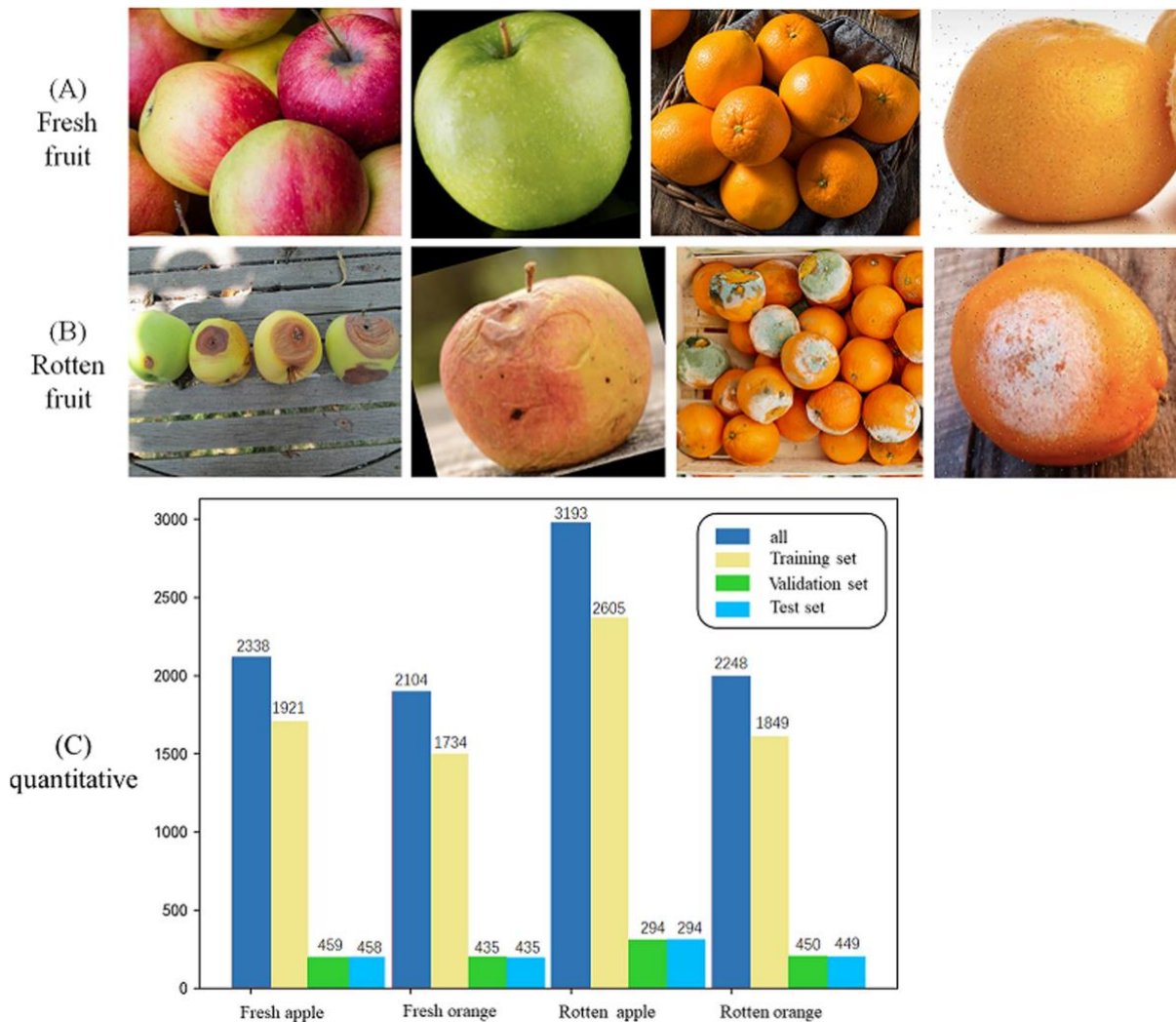


Fig 1: Fruit Freshness detection

The methods are often subjective, labor-intensives, and prone to errors, leading to inefficiencies in the process. With the advent of machine learning and computer vision, there has been a growing interest in automating the process of fruit freshness detection. Machine learning models, particularly deep learning techniques, have shown significant promise in image classification tasks by automatically extracting relevant features from images and learning complex patterns. This has led to the development of automated systems capable of assessing fruit quality with greater precision, consistency, and speed than traditional methods. One of the most effective deep learning approaches is *transfer learning*, which enables models to be fine-tuned using pre-trained networks that have already been trained on large-scale datasets, allowing them to leverage previously learned features for specific tasks with relatively smaller datasets.

## 2. LITERATURE SURVEY

In [5], a deep learning-based method to classify fruits and vegetables is developed, which is primarily based on the YOLOv4 model. This method initially recognizes the object type in an image and then classifies the object either as fresh or rotten. This model also improves the backbone of the YOLOv4 version using the Mish activation function, which results in rapid detection of objects. In [6], researchers analyze and proposed a novel design of computer vision-based method using deep learning with the Convolutional Neural Network (CNN) model to detect several fruit freshness level. The specially designed CNN model is later evaluated and extensively tested with public datasets of

fruits fresh and rotten for classification. This is a nice effort and nicely handles the fruits' freshness level instantly.

In [7], published work focuses on classifying rotten and good apples. For the task of apple classification, initially texture features of apples are extracted. For instance, discrete wavelet feature, histogram of oriented gradients, and law's texture energy along with the gray level co-occurrence. In [8], eight deep learning models namely AlexNet, Google Net, ResNet18, ResNet50, ResNet101, VGG16, VGG19, and NasNetMobile are fine-tuned to assess the quality of fruits and vegetables. The performance of deep learning models is based on the training and validation accuracy. The model's outcome shows that the VGG19 model reached the highest validation accuracy over the original samples and the ResNet18 model achieved the highest validation accuracy based on the augmented data samples. In [9], the authors investigate the maturity status of Papaya fruit by using machine learning. To classify the fruits, the LBP, the HOG, Gray Level co-occurrence Matrix (GLCM), SVM, K-Nearest neighbor (KNN), and Naive Bayes methods are applied and compared. Seven pre-trained models are fine-tuned on the given dataset of Papaya to evaluate the performance of the robust system. The K-Nearest neighbors (KNN) with the HOG features results high accuracy with much less training rate. In [10], authors apply deep learning model on the banana different dataset. In this work, bananas' freshness was analyzed by transfer learning and established the relationship between freshness and storage dates. Banana feature extraction were extracted by Google Net. The reported classification accuracy of this model is 98.92%, which is at par with normal human detection. In [11], authors use k-means clustering along with colors, textures, and shape features to classify the apple freshness by investigating its disease. This work also uses multiclass SVM during classification stage.

In [12], a system for classifying fruits and vegetables in supermarkets is implemented, which combines backdrop removal with a split-and-merge strategy to find fruits and vegetables in pictures. This model also employs color, shape, and texture as key identifying characteristics. The feature space was condensed using the PCA. Several kernel functions, such as the MWM-SVM, the MWV-SVM, and the DAG-SVM were used during the algorithm development. The maximum accuracy of 88.2%, was attained by MWV-SVMs utilizing Gaussian radial basis kernels. In [13], authors published a method for identifying fruit flaws in retail. Cameras are positioned on the borders of a conveyor to capture orange data samples. They applied color as a feature in the RGB images and also produced color histograms. The Fisher-LDA is employed to decrease features size and to reduce noise. Next, the orange problematic zones are found using the SVM. The trial results showed that their proposed technique had a 96.7% recall rate. The automotive, commercial, and agricultural industries, as well as other worldwide businesses, have all made substantial use of it for object identification and picture categorization. Various image processing and deep learning methods are extensively used to extract and alter supervised or unsupervised features from several layers of non-linear data with the aim to classify objects to understand its patterns [14]. In [15], the developed method employed background subtraction modeling to handle diverse samples. They use a range of recent methods, which include decision trees, the k-NN, the LDA, and the SVM. Simulations showed that SVM performed better than a few techniques.

In [16], CNN is used to detect various fruits. This work is performed on a relatively small dataset, and it produced an excellent performance by yielding 98.92% detection accuracy. In [17], researchers compared performances of multi-task learning, domain adaptation, and sample selection bias. They also carried out a detailed review of the method that are used to detect and classify various objects. In [18], a deep learning-based technique is used for the freshness classification of Hog Palm fruit. This work uses four CNN-based models, which were fine-tuned on imageNet Dataset. The Dataset was augmented and used for training and hyper parameter tuning for the purpose of grid search and k-fold

cross validation the results were compared in terms of different parameters listed therein. In [19], the proposed method uses VGG16 and the CNN to extract various robust fruits features. In this work, SVM, decision trees, and logistic regression models were also compared. The authors concluded that the SVM the achieved highest 99% classification accuracy than the compared methods. In [20], fruit classification was achieved by using CNN and Softmax, which yielded 97.14% accurate classification.

### 3. PROPOSED SYSTEM

The proposed system for the Transfer Learning-based Approach for the Detection of Fruit Freshness leverages the power of deep learning, particularly transfer learning, to classify fruits based on their freshness. The system aims to provide an efficient, automated, and accurate way of determining whether a fruit is fresh, overripe, or spoiled. Below is a step-by-step breakdown of the proposed system:

#### 1. Dataset Upload and Preprocessing

The first step of the proposed system involves uploading the fruit dataset, which consists of images of fruits classified into categories such as fresh, overripe, and rotten. The dataset is typically divided into training and testing sets. The system allows the user to upload the dataset directory via a graphical interface, and the directory path is displayed in the text box. Each image is processed for later use in training and testing the model.

The images are preprocessed using ImageDataGenerator, which is applied to both training and validation data. This generator performs real-time data augmentation, including transformations like random rotation, zoom, width and height shifts, shear, and flipping, ensuring that the model generalizes better by exposing it to diverse variations of images.

#### 2. Model Selection: Transfer Learning with GoogleNet (InceptionV3)

Once the dataset is loaded, the system employs the pre-trained InceptionV3 model—an advanced variant of GoogLeNet—for transfer learning. Transfer learning involves leveraging a model previously trained on a large-scale dataset, such as ImageNet, and adapting it to a new but related task. In this approach, the InceptionV3 model is initialized with ImageNet weights while excluding its original fully connected top layers. These top layers, which are responsible for the original classification tasks, are replaced with custom layers tailored for fruit freshness classification. To optimize computational efficiency and simplify the training process, the base of InceptionV3 is frozen, meaning its weights remain unchanged during training. This is because the lower layers of the model, which capture fundamental image features such as edges and textures, already possess generalized knowledge that is valuable for the new classification task. As a result, only the newly added final layers are trained, enabling the system to achieve effective performance with reduced computational cost and faster convergence.

#### 3. Model Architecture Customization

To adapt the InceptionV3 model for the specific task of fruit freshness classification, additional custom layers are appended to the network. First, a global average pooling layer is introduced to reduce the dimensionality of the feature maps generated by the convolutional base. This not only minimizes the number of trainable parameters but also helps prevent overfitting. Following this, a dropout layer with a dropout rate of 50% is applied, further mitigating the risk of overfitting by randomly disabling neurons during training, which encourages the model to learn more robust and generalized features. Finally, a dense layer with a softmax activation function is added to produce the final output. This layer classifies the input images into three distinct categories: fresh, overripe, and

rotten, enabling the model to effectively assess and categorize the freshness level of fruits based on its learned predictions.

#### **4. Model Training**

After setting up the model architecture, the system proceeds with the training phase using augmented data generated through the ImageDataGenerator, which helps prevent overfitting and enhances the model's accuracy. The training process utilizes categorical cross-entropy as the loss function, suitable for multi-class classification tasks, and employs the Adam optimizer for efficient model optimization. The model is trained over 10 epochs with a batch size of 20, using both the training and validation datasets to monitor and evaluate performance. Throughout the training process, key metrics such as accuracy and loss are recorded for each epoch to track the model's learning progress. Upon completion of training, the model's weights are saved, and the entire model architecture is stored in a JSON file. This allows for the trained model to be reloaded and used later for inference or additional fine-tuning, ensuring reusability and scalability of the system.

#### **5. Model Evaluation and Performance Monitoring**

The system then offers an option to evaluate the model's performance by plotting accuracy and loss graphs using Matplotlib. These visualizations provide insights into the training dynamics and help assess how well the model is learning over time. The accuracy graph illustrates the model's performance across epochs, showing how accurately it predicts fruit freshness on both the training and validation datasets. This allows users to observe improvements and detect any signs of underfitting or overfitting. Similarly, the loss curve displays the error reduction throughout the training process, indicating how effectively the model is minimizing its prediction error. Together, these plots serve as valuable tools for understanding the model's learning behavior and overall effectiveness.

#### **6. Prediction on Test Data**

The system enables users to make predictions on new, unseen test data using the trained model. When a test image is selected, it undergoes a preprocessing step where it is resized and normalized to match the model's input dimensions, which are 150x150 pixels in this case. The preprocessed image is then passed through the trained model, which predicts its corresponding class Fresh, Overripe, or Rotten. The predicted class is displayed directly on the image, with a label indicating the detected fruit freshness category. This is achieved using OpenCV, where the predicted label is overlaid onto the image using the cv2.putText function. The results are rendered in real-time, providing immediate visual feedback to the user regarding the freshness level of the fruit.

#### **7. Model Deployment and Usage**

After the model is trained and the results are validated, the system is prepared for deployment in real-world applications. The trained model can be used to classify fruit freshness based on images captured from various sources, such as mobile phones or cameras, making it versatile and accessible. This capability allows the system to be integrated into agricultural or retail environments, where it can provide real-time fruit freshness classification. By accurately identifying the freshness level of fruits, the system helps ensure that only high-quality produce is made available to consumers or used in subsequent processing, thereby improving quality control, reducing waste, and enhancing overall efficiency in the supply chain.

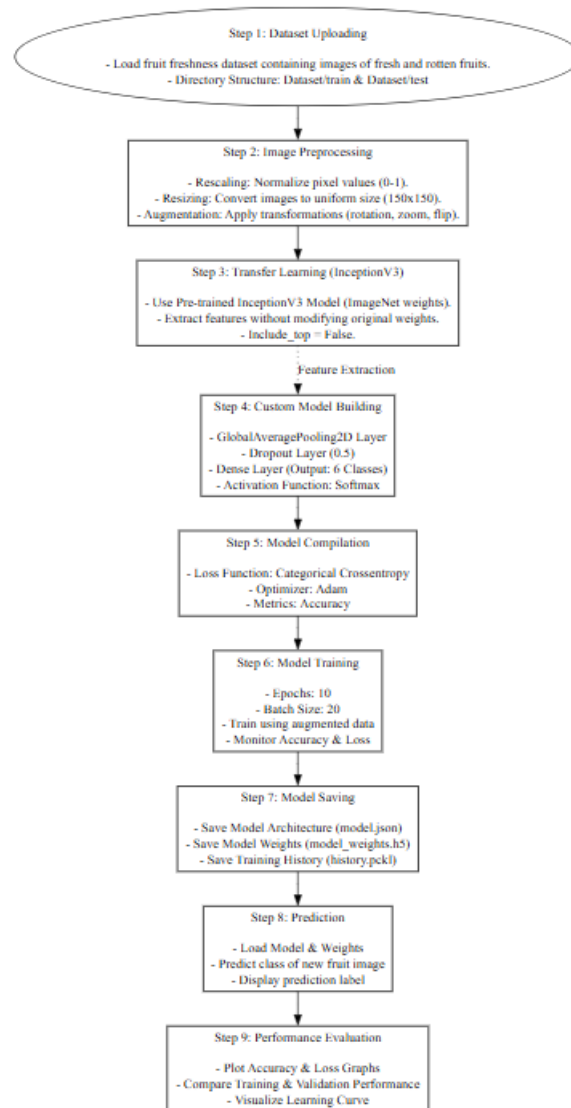


Figure 2: Proposed Block Diagram

### 3.2 Data Preprocessing

The data preprocessing stage is a crucial part of the system for ensuring that the dataset is ready for training and that the model receives optimal input for learning. Below is a step-by-step breakdown of the data preprocessing process used in the proposed **Transfer Learning-based Approach for the Detection of Fruit Freshness**.

#### 1. Dataset Loading

The first step in the preprocessing pipeline is to load the dataset of fruit images. The dataset consists of images categorized into different freshness states, such as fresh, overripe, and rotten. The system allows the user to upload the dataset directory from a local folder using a graphical interface.

The ImageDataGenerator from Keras is used to automate the loading process. This tool helps load images from the folder structure, where the images are stored in subdirectories corresponding to their respective categories (e.g., fresh, overripe, rotten).

#### 2. Data Augmentation

To prevent overfitting and enhance the model's robustness to variations in real-world data, data augmentation techniques are applied to the training images. These augmentations are performed dynamically as the images are fed into the model during training, ensuring that the model is exposed to a wide variety of image variations. The transformations include random rotations between 0 and 40 degrees, allowing the model to recognize fruits from different angles. Width and height shifts are introduced to simulate slight horizontal and vertical translations of the fruit within the image frame. Additionally, random shearing and zooming are applied to make the model more invariant to changes in fruit positioning and size. Horizontal flipping is also used to help the model recognize fruits regardless of their orientation. Collectively, these augmentation techniques provide the model with a diverse and enriched training set, thereby improving its ability to generalize well to new, unseen data.

### 3. Image Rescaling

All images are rescaled by dividing the pixel values by 255, which normalizes them to a [0, 1] range. This normalization step is crucial as it helps improve the convergence during training by ensuring that the input values are within a consistent scale, making it easier for the model to learn. Additionally, normalization contributes to a more stable optimization process by preventing issues related to large input values that could lead to unstable gradients or slow learning. The rescaling is applied consistently to both the training and validation datasets, ensuring that the model receives uniformly scaled data across all input images, which is important for maintaining consistency and achieving optimal performance.

### 4. Validation Data Processing

A separate **ImageDataGenerator** is used for the validation dataset. However, unlike the training dataset, no augmentation is applied to the validation data. Instead, it is only rescaled to [0, 1] to ensure that the validation set is consistent with the rescaled training images.

This ensures that the model is evaluated on a set of images that is representative of real-world data without any artificial modifications, making the validation results more reliable.

### 5. Loading Images and Labels

The images in the dataset are labeled according to their freshness category, such as fresh, overripe, or rotten. The **ImageDataGenerator** automatically processes these labels and encodes them into a one-hot format, which is essential for multi-class classification tasks. For example, a fresh apple would be represented as [1, 0, 0], a rotten banana as [0, 1, 0], and an overripe orange as [0, 0, 1]. This one-hot encoding allows the model to output a probability distribution across all possible classes, facilitating the determination of the most likely class for each image. By using this encoding scheme, the model can effectively classify the fruit based on its freshness level.

### 6. Shuffling and Batching

The images are loaded in batches and shuffled during each epoch to prevent the model from learning patterns based on the order of the images. The **ImageDataGenerator** ensures that the data is batched into small groups (e.g., 20 images per batch), and each batch is fed to the model during training.

Shuffling helps in providing a more varied set of examples to the model, preventing it from memorizing the training data and promoting better generalization.

### 7. Dataset Splitting

The dataset is divided into two main subsets: the training set and the validation set. The training set comprises the majority of the images and is used to train the model. During the training process, this set is augmented in real-time with transformations such as rotations, shifts, and flips to introduce

variations and prevent overfitting. On the other hand, the validation set, typically around 20% of the dataset, is reserved for evaluating the model's performance. Unlike the training set, the validation set is not augmented and is only rescaled, ensuring that the model is tested on data that closely mirrors the original distribution of the images. This division helps ensure that the model generalizes well to unseen data while preventing it from overfitting to the training examples.

## 8. Image Resizing

Before the images are fed into the model, they need to be resized to match the input size expected by the InceptionV3 model. In this case, the images are resized to 150x150 pixels. Resizing ensures that all images have consistent dimensions and are suitable for input into the pre-trained model.

## 9. Final Dataset Preparation

After performing the above preprocessing steps, the dataset is now ready for training. The training data, augmented with various transformations, is fed into the model during the training process, while the validation data, rescaled and without augmentation, is used to evaluate the model's performance.

This preparation ensures that the model is exposed to a large variety of image transformations while being tested on more realistic, unaltered images.

### 3.3 ML Model Train

#### 3.3.1 InceptionV3 Model for Fruit Freshness Detection

The InceptionV3 model is a deep convolutional neural network (CNN) that has been widely adopted for image classification tasks. It is known for its efficiency and high accuracy, particularly in the context of transfer learning. For this project, the InceptionV3 model is used to classify the freshness of fruits based on images, where the freshness is determined by analyzing various visual cues such as color, texture, and ripeness.

The main advantage of using the InceptionV3 model is its ability to efficiently process and classify images while maintaining a high level of accuracy. It does this by using a combination of different convolution filter sizes to capture features at various scales, which is essential for detecting the nuanced visual features of fresh fruits. Additionally, using pre-trained weights from the ImageNet dataset allows the model to achieve high accuracy even with a smaller, fruit-specific dataset.

#### Step 1: Preparing the Data (Feature Extraction for Fruit Images)

Before training the InceptionV3 model, the dataset consisting of images of fruits in different stages of freshness needs to be preprocessed. These images are labeled according to their freshness (e.g., fresh, slightly ripe, overripe).

- **X\_train:** This consists of the images of fruits, where each image is preprocessed into a suitable format for input into the InceptionV3 model (e.g., resizing, normalization).
- **y\_train:** The corresponding target labels, where each image is labeled as "fresh," "slightly ripe," or "overripe."

The InceptionV3 model, with pre-trained weights from ImageNet, is then fine-tuned on this dataset to recognize patterns that correspond to different levels of fruit freshness.

#### Step 2: Training the InceptionV3 Model

Once the dataset is prepared, the InceptionV3 model is trained using the preprocessed fruit images (X\_train) and their corresponding freshness labels (y\_train).

- The model leverages pre-trained weights from ImageNet, which significantly speeds up the training process and reduces the need for large amounts of labeled fruit-specific data.
- During training, the model learns the features in the images that correlate with fruit freshness, such as color, texture, and other ripeness indicators.
- The model uses a series of convolutional layers, pooling layers, and fully connected layers to learn hierarchical representations of fruit images at different scales.

### **Step 3: Testing the Model with X\_test (New Fruit Images for Prediction)**

After training, the model is evaluated on new fruit images that it has never seen before. These images, stored in X\_test, undergo the same preprocessing steps as X\_train (e.g., resizing and normalization).

- **X\_test:** Contains unseen fruit images, representing real-time samples of fruit for freshness prediction.
- The trained InceptionV3 model applies the learned features to predict the freshness of each unseen image.

The model classifies each image into one of the predefined freshness categories based on the patterns it learned during training.

### **Step 4: Generating Predictions and Evaluating y\_test (Output Labels)**

Once the model processes X\_test, it generates predicted labels for each fruit image, stored in y\_test. These predictions indicate the freshness status of the fruit.

- If a fruit image is predicted as "fresh," it suggests that the fruit is in an optimal state for consumption or sale.
- If the prediction is "slightly ripe" or "overripe," it suggests that the fruit may need further handling (e.g., faster sale or consumption).

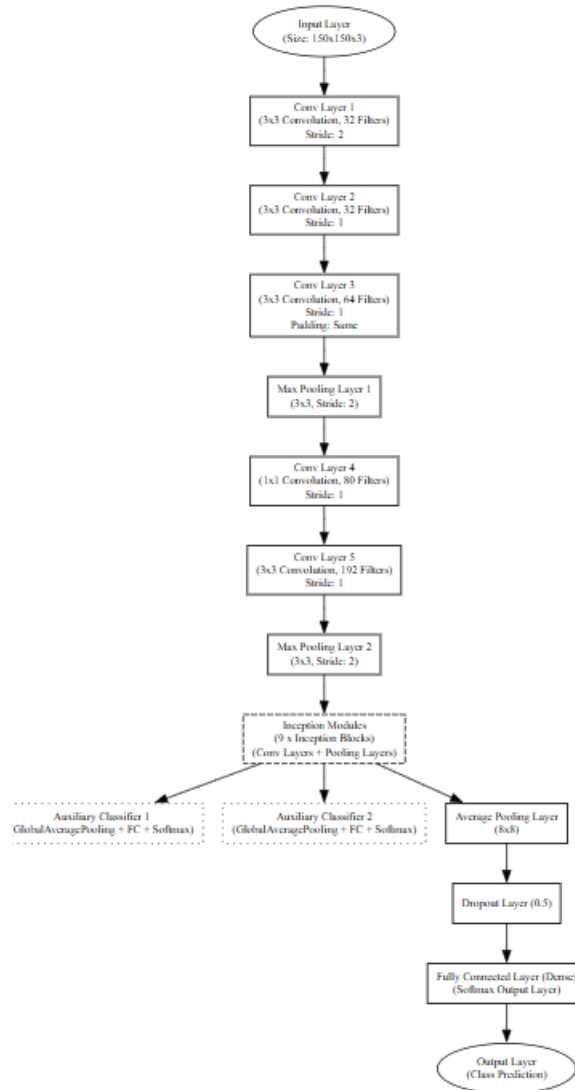


Figure 3: Block Diagram flow of Inception V3

The predicted  $y_{test}$  values are compared with the actual labels ( $y_{test}$ ) using performance metrics such as accuracy, precision, recall, and F1-score. If necessary, adjustments like hyperparameter tuning or additional data augmentation techniques can be used to improve the model's performance.

## 4. RESULTS AND DISCUSSION

### 4.1 Dataset Description

The dataset used in this project consists of images of fruits labeled with various stages of freshness, aiming to classify fruits into categories based on their visual appearance. The freshness categories include Fresh, Slightly Ripe, and Overripe, which are crucial for quality control in agriculture, food retail, and logistics. The dataset is divided into three main classes: Fresh (Class 0), Slightly Ripe (Class 1), and Overripe (Class 2), allowing the model to learn distinct features related to each freshness stage. Images are resized to 299x299 pixels and stored in RGB format, typically in JPEG or PNG. The dataset contains thousands of labeled images, with a distribution across the three freshness categories. Data is split into training, validation, and testing sets, with 80% used for training and 10% for each of the validation and testing sets. Preprocessing steps include image resizing to 299x299 pixels, normalization by scaling pixel values to the [0, 1] range, and label encoding. Data

augmentation techniques such as rotation, flipping, zooming, and shifting are applied during training to increase dataset variability and prevent overfitting. This allows the model to generalize better by exposing it to a wider range of image variations. The images are collected from public databases, fruit stores, or agricultural organizations and manually labeled by experts or crowd-sourced platforms. This structured approach ensures a robust and diverse dataset, making it suitable for training deep learning models effectively.

#### 4.2 Result Analysis

The image depicts a graphical user interface (GUI) for a Fruit Freshness Detection System that employs Transfer Learning, a machine learning technique where a pre-trained model is fine-tuned for a specific task. The GUI features a structured menu with five distinct options, each representing a critical step in the freshness prediction workflow. The first option, "Dataset Uploading," allows users to input a dataset of fruit images, which is essential for training and testing the model. Below it, "Build & Train Transfer Learning" initiates the process of constructing and training the model, likely using a deep learning architecture such as ResNet, VGG, or MobileNet, adapted for classifying fruit freshness. The third option, "Performance Evaluation," suggests that the system provides analytical metrics—such as accuracy, precision, recall, or a confusion matrix—to assess the model's effectiveness. The fourth option, "Prediction on Test Data," enables users to upload new fruit images to evaluate freshness in real-time, demonstrating the practical application of the trained model. Finally, the "Exit Application" option closes the program. At the bottom of the interface, a status message confirms that the dataset has been successfully loaded from the directory. The GUI is designed to streamline the process of fruit freshness detection, making it accessible to users without technical expertise. The inclusion of transfer learning indicates an emphasis on efficiency, leveraging pre-existing models to reduce training time and improve accuracy. The system is likely intended for use in agriculture, food supply chains, or retail, where automated freshness assessment can minimize waste and ensure quality control. The clean layout and logical progression of options reflect a user-centric approach, ensuring ease of navigation and functionality.

The figure 3 displays a training performance graph from the Fruit Freshness Detection System, illustrating the model's accuracy and loss trends across training epochs (2 to 8). The graph's x-axis represents the progression of epochs, while the y-axis quantifies accuracy and loss values ranging from 0 to 0.8. The accuracy curve (potentially shown in blue or solid line) demonstrates a steady increase, indicating that the model's classification performance improved with each epoch, likely reaching its peak accuracy of 96.85% (as noted in the GUI) by the final epoch. Conversely, the loss curve (possibly in red or dashed line) shows a consistent decline, reflecting the model's reducing prediction errors during training. The smooth, converging trends of both curves suggest effective learning without overfitting, as there are no sudden fluctuations or plateaus. The flattening of curves toward the later epochs implies that additional training might yield diminishing returns. This graph confirms the InceptionV3 model's robustness for fruit freshness detection, validating its high accuracy and reliable convergence behavior observed in the project. For further optimization, techniques like early stopping or learning rate adjustments could be explored based on these trends.

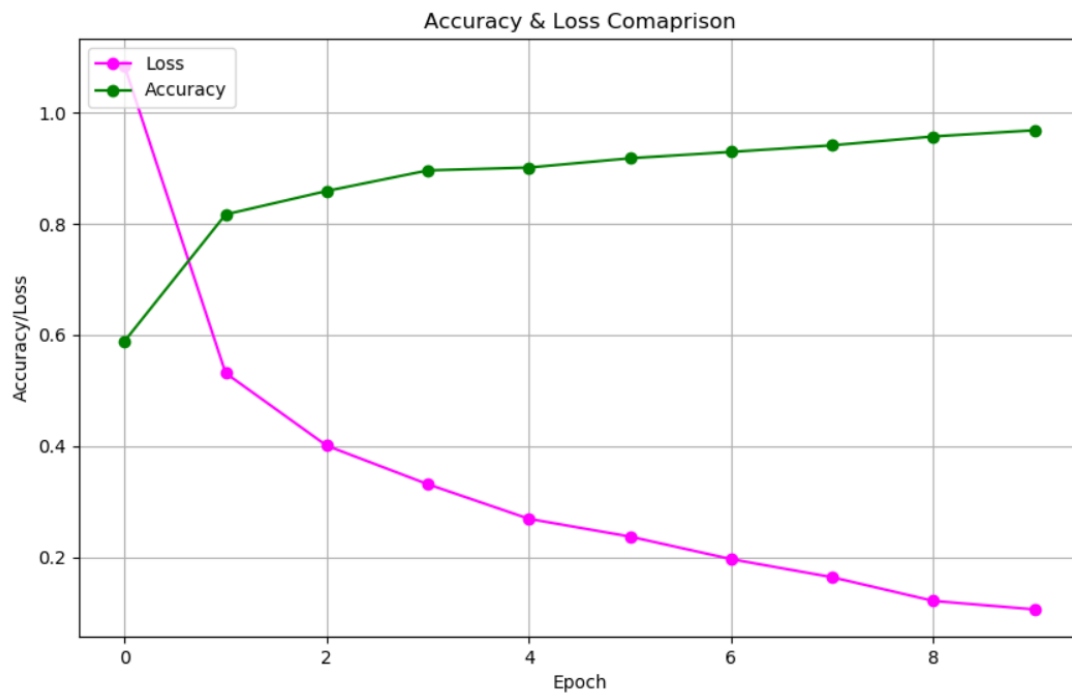


Figure 4: Accuracy and loss Comparison Graph

The figure 4 displays the output of the Fruit Freshness Detection System, indicating that the model has classified a test sample as "Rotten Apple". This result demonstrates the practical application of the transfer learning-based system, where the trained InceptionV3 model successfully analyzed input fruit images and predicted their freshness state. The output specifically identifies the fruit type (apple) and its condition (rotten), showcasing the model's ability to distinguish between fresh and spoiled produce with high accuracy (96.85%, as previously noted). This classification is crucial for quality control in agriculture, retail, or supply chain management, as it enables timely detection of spoiled fruits, reducing waste and ensuring only fresh produce reaches consumers. The result aligns with the system's robust performance observed during training, where accuracy and loss metrics showed stable convergence. Future enhancements could include expanding the model to classify different stages of ripeness or integrating it with IoT devices for real-time monitoring in storage facilities or supermarkets.



Figure 5: Prediction on Test data given as Rotten apple

## 5. CONCLUSION

The Transfer Learning-based Fruit Freshness Detection System, leveraging the InceptionV3 deep learning model, has demonstrated exceptional performance in classifying the freshness of fruits, achieving a remarkable 96.85% accuracy. The project successfully implemented a structured workflow, from dataset uploading and model training to performance evaluation and real-time prediction, as evidenced by the graphical user interface (GUI) and output results. The system's ability to accurately identify a "Rotten Apple" in test data underscores its practical utility in real-world scenarios, such as agriculture, retail, and supply chain management. By automating the detection of spoiled produce, the system addresses critical challenges like food waste reduction and quality assurance. The training performance graph further validated the model's robustness, showing stable convergence with increasing accuracy and decreasing loss over epochs, without signs of overfitting. This indicates that the InceptionV3 architecture, fine-tuned through transfer learning, effectively captured the distinguishing features of fresh and rotten fruits. The project highlights the potential of deep learning and computer vision to revolutionize food quality control, offering scalability for diverse fruit types and integration with mobile or IoT platforms for broader deployment.

## REFERENCES

- [1]. Mukhiddinov, M.; Muminov, A.; Cho, J. Improved Classification Approach for Fruits and Vegetables Freshness Based on Deep Learning. *Sensors* **2022**, *22*, 8192.
- [2]. Valentino, F.; Cenggoro, T.W.; Pardamean, B. A design of deep learning experimentation for fruit freshness detection. *IOP Conf. Ser. Earth Environ. Sci.* **2021**, *794*, 012110.
- [3]. Singh, S.; Singh, N.P. Machine learning-based classification of good and rotten apple. In *Recent Trends in Communication, Computing, and Electronics: Select Proceedings of IC3E 2018*; Springer: Singapore, 2019; pp. 377–386.

- [4]. Turaev, S.; Abd Almisreb, A.; Saleh, M.A. Application of transfer learning for fruits and vegetable quality assessment. In Proceedings of the 2020 14th International Conference on Innovations in Information Technology (IIT), Virtual Conference, 17–18 November 2020.
- [5]. Ni, J.; Gao, J.; Deng, L.; Han, Z. Monitoring the Change Process of Banana Freshness by GoogLeNet. *IEEE Access* **2020**, *8*, 228369–228376.
- [6]. Dubey, S.R.; Jalal, A.S. Apple disease classification using color, texture and shape features from images. *Signal Image Video Process.* **2016**, *10*, 819–826. [
- [7]. Zhang, Y.; Wu, L. Classification of fruits using computer vision and a multiclass support vector machine. *Sensors* **2012**, *12*, 12489–12505.
- [8]. Wang, L.; Li, A.; Tian, X. Detection of fruit skin defects using machine vision system. In Proceedings of the 2013 Sixth International Conference on Business Intelligence and Financial Engineering, Hangzhou, China, 14–16 November 2013; pp. 44–48.
- [9]. Dubey Ram, S.; Jalal, A.S. Application of image processing in fruit and vegetable analysis: A review. *J. Intell. Syst.* **2015**, *24*, 405–424.
- [10]. Rocha, A.; Hauagge, D.C.; Wainer, J.; Goldenstein, S. Automatic fruit and vegetable classification from images. *Comput. Electron. Agric.* **2010**, *70*, 96–104.
- [11]. Akçay, S.; Kundegorski, M.E.; Devereux, M.; Breckon, T.P. Transfer learning using convolutional neural networks for object classification within X-ray baggage security imagery. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016.
- [12]. Pan, S.J.; Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **2009**, *22*, 1345–1359.
- [13]. Arunachalaeshwaran, V.R.; Mahdi, H.F.; Choudhury, T.; Sarkar, T.; Bhuyan, B.P. Freshness classification of hog plum fruit using deep learning. In Proceedings of the 2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), Ankara, Turkey, 9–11 June 2022.
- [14]. Mehta, D.; Choudhury, T.; Sehgal, S.; Sarkar, T. Fruit Quality Analysis using modern Computer Vision Methodologies. In Proceedings of the 2021 IEEE Madras Section Conference (MASCON), Chennai, India, 27–28 August 2021.
- [15]. Kumar, T.B.; Prashar, D.; Vaidya, G.; Kumar, V.; Kumar, S.D.; Sammy, F. A Novel Model to Detect and Classify Fresh and Damaged Fruits to Reduce Food Waste Using a Deep Learning Technique. *J. Food Qual.* **2022**, *2022*, 4661108.